KEDBYTE

SOFTWARE · AI · AUTOMATION

HOW MONEY MOVES

From a Tally Stick to Instant Settlement

The complete plain-English guide to payments and banking, followed by the engineer's version of the same thing.

WRITTEN BY

Shikhar Singh

Founder & Chairman

KedByte Technologies Private Limited

Five Volumes • Fifty-Four Chapters • First Edition

HOW MONEY MOVES

From a Tally Stick to Instant Settlement

Author	Shikhar Singh
Designation	Founder & Chairman, KedByte Technologies Private Limited
Published by	KedByte Technologies Private Limited
Imprint	KedByte Press
Edition	First Edition
Subject	Payments, banking systems, financial technology
Extent	Five volumes, fifty-four chapters

Copyright © KedByte Technologies Private Limited. All rights reserved.

No part of this publication may be reproduced, distributed or transmitted in any form or by any means, including photocopying, recording or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other non-commercial uses permitted by copyright law.

Trademarks. All product names, company names, service marks and trademarks referred to in this book are the property of their respective owners. They are used in this book in an editorial and educational capacity only, with no intention of infringement of the trademark. No such use, or the use of any trade name, is intended to convey endorsement or any other affiliation with this book or its publisher.

Accuracy and currency. This book describes technology that changes. Specifications, prices, version numbers, product names and industry figures were checked at the time of writing and are dated in the text wherever they are liable to change. The chapters on artificial intelligence in particular describe a field that moves quickly; readers are told, in the text, where to re-check a figure before relying on it. Where experts disagree, the book says so rather than choosing a side.

Disclaimer. The information in this book is provided on an "as is" basis, for education. The security chapter is written for defence: it explains how classes of attack work so that systems can be built to resist them, and it deliberately contains no working exploit code. Neither the author nor KedByte Technologies Private Limited shall have any liability to any person or entity with respect to any loss or damage caused, or alleged to have been caused, directly or indirectly by the information contained in this book.

Worked examples. Parts F and G are built around one real network fault and one real version-control session, reproduced exactly as they were observed, including the addresses, the traceroute output and the error messages. Nothing in those traces has been altered or invented.

FOREWORD

From the Founder & Chairman

You tap a card against a machine and a shop lets you leave with the coffee. Somewhere behind that gesture, money you cannot see is supposed to move from a place you have never visited to another place you have never visited, and almost nobody involved can tell you how. Not the person who served you. Often not the person who installed the terminal. Frequently not the people who work at the bank.

That collective not-knowing is expensive. It produces businesses that cannot reconcile their own takings, engineers who store money in floating point, support teams who ask customers for the one reference number that cannot possibly help, and founders who discover the difference between a payment being authorised and a payment being settled on the day it costs them something. None of these are stupidity. They are the predictable result of an industry that documents itself for people who are already inside it.

This book is the outsider's way in, written to be useful to the insider. It begins with what money actually is, which is a record of obligation rather than a substance, and it ends with regulation, dispute and the ugly arithmetic of reconciliation. In between: the card in your pocket and the small computer embedded in it, the machine on the counter, the message formats that have carried the world's card traffic for forty years, the difference between a payment clearing and a payment settling, the rails that move money without cards at all, and what happens when any of it fails.

The method is the same one used in *How Computers Work*, and it matters more than the coverage. Every idea is explained twice. First in plain language, with an everyday comparison and a worked example, such that a twelve-year-old could follow it. Then again in the exact professional vocabulary, with the real field numbers, the real timings and the real institution names that a practitioner would use. Between the two sits a section that most writing on this subject omits entirely: an explicit statement of where the simple version stops being true. A useful simplification told without warning is how people end up confidently wrong, and in payments confidently wrong is a category of loss.

One thing about this book differs from its predecessor, and the reader should know it. How a transistor works will not change. Payments will. Scheme rulebooks are revised annually, regulatory caps move, and migrations between message standards run on live deadlines. The volumes are therefore arranged so that decay is contained: the mechanics that will outlive us sit in the early volumes, and the material that is true as of writing is gathered deliberately at the end, marked as

such. Where a figure is time-bound, this book says so rather than pretending to a permanence it does not have.

At KedByte Technologies Private Limited we build systems that move other people's money, which is a privilege that ought to come with an obligation to explain the machinery. If this book leaves one reader able to follow their own money from a card terminal to a bank ledger, and to say honestly which part of that journey they have verified and which part they have merely assumed, it has done its job.

Start at Chapter 1. It begins long before cards, and for good reason.

Shikhar Singh

Founder & Chairman

KedByte Technologies Private Limited

BEFORE YOU START

How to read this book

The shape of every section

Every section of every chapter is built from the same six blocks, always in the same order. Once you know the shape, you can navigate two thousand pages without ever being lost.

Block	What it is for
PLAIN in simple words	The idea for a complete beginner. No jargon at all.
PLAIN a picture in your head	One everyday comparison, and then a line telling you exactly where that comparison stops being true.
PLAIN a worked example	Concrete numbers, worked step by step.
PLAIN what is really happening inside	The mechanism, still in easy words, but with nothing hand-waved.
TECHNICAL the engineer's version	The correct professional vocabulary, exact units, real specifications, real figures, standard numbers and the commands that observe it.
WORDS remember these	Each term given twice: the plain meaning, then the technical meaning.

Two ways to read it

1. **The single pass.** Read everything, in order, from Chapter 1. This is the intended route and it works.
2. **The double pass.** Read only the blocks marked PLAIN, front to back. You will understand the whole of computing. Then return to the start and read the blocks marked TECHNICAL to be able to hold your own in an engineering conversation.

Conventions used throughout

1. Almost everything is written as short numbered lines, one idea per line, so nothing hides inside a paragraph.
2. *The honest version* marks the places where a simple explanation is not strictly true. The accurate model follows immediately.

3. The book states whether a thing is a **standard** (written down in a specification), a **convention** (everybody just does it this way), or an **implementation detail** (true of one product, not of the idea).
4. In the artificial intelligence chapters, claims are separated into established fact, active research, and marketing claim.
5. Anything that goes stale is dated. Check it again before you rely on it.
6. Every chapter ends with *Common wrong ideas* and a twenty-line summary. If you are short of time, read those two first.

Where to find things

1. **Chapter 1** is the map of the whole book.
2. **Chapter 52** is the reference desk: the numbers, the command cheat sheets and the two diagnostic decision trees.
3. **Chapter 53** is the master timeline and a glossary of more than seven hundred terms, each given in plain words and then technically.

Contents

Volume I — What Money Is 1

1 The Oldest Question 2

1.0 What this chapter gives you 2
The plain version 3
Where the plain version stops being true 6
The technical version 8
1.98 Common wrong ideas 13
1.99 Chapter summary in 20 lines 14

2 A Balance Is Not a Thing 16

2.0 What this chapter gives you 16
The plain version 17
Where the plain version stops being true 19
The technical version 21
2.98 Common wrong ideas 27
2.99 Chapter summary in 20 lines 28

3 Double Entry 30

3.0 What this chapter gives you 30
The plain version 31
Where the plain version stops being true 34
The technical version 36
3.98 Common wrong ideas 43
3.99 Chapter summary in 20 lines 44

4 Ledgers, Journals and Accounts 46

4.0 What this chapter gives you 46
The plain version 47
Where the plain version stops being true 49
The technical version 51
4.98 Common wrong ideas 58
4.99 Chapter summary in 20 lines 58

5 What a Bank Is 60

5.0 What this chapter gives you 60
The plain version 61
Where the plain version stops being true 64
The technical version 66
5.98 Common wrong ideas 73
5.99 Chapter summary in 20 lines 74

6 Two Kinds of Money 76

6.0 What this chapter gives you 76
The plain version 77
Where the plain version stops being true 79

The technical version 81
6.98 Common wrong ideas 87
6.99 Chapter summary in 20 lines 87

7 Clearing Versus Settlement 89

7.0 What this chapter gives you 89
The plain version 90
Where the plain version stops being true 92
The technical version 94
7.98 Common wrong ideas 101
7.99 Chapter summary in 20 lines 102

8 Netting 104

8.0 What this chapter gives you 104
The plain version 105
Where the plain version stops being true 107
The technical version 109
8.98 Common wrong ideas 118
8.99 Chapter summary in 20 lines 119

9 Currency, Decimals and the Money Type 121

9.0 What this chapter gives you 121
The plain version 122
Where the plain version stops being true 123
The technical version 126
The point of all this 133
9.98 Common wrong ideas 133
9.99 Chapter summary in 20 lines 134

10 Reading a Statement 136

10.0 What this chapter gives you 136
The plain version 137
Where the plain version stops being true 139
The technical version 141
Where Volume I leaves you 149
10.98 Common wrong ideas 150
10.99 Chapter summary in 20 lines 150

Volume II — The Card 153

11 What Is Physically on a Card 154

11.0 What this chapter gives you 154
The plain version 155
Where the plain version stops being true 156
The technical version 157
11.98 Common wrong ideas 168
11.99 Chapter summary in 20 lines 169

12 The Number 171

12.0 What this chapter gives you 171
The plain version 172
Where the plain version stops being true 174
The technical version 176
12.98 Common wrong ideas 184
12.99 Chapter summary in 20 lines 184

13 The Four Things Called the Security Code 187

13.0 What this chapter gives you 187
The plain version 188
Where the plain version stops being true 190
The technical version 192
13.98 Common wrong ideas 199
13.99 Chapter summary in 20 lines 200

14 The Chip Is a Computer 202

14.0 What this chapter gives you 202
The plain version 203
Where the plain version stops being true 204
The technical version 205
14.98 Common wrong ideas 217
14.99 Chapter summary in 20 lines 218

15 The Cryptogram 220

15.0 What this chapter gives you 220
The plain version 221
Where the plain version stops being true 223
The technical version 224
15.98 Common wrong ideas 233
15.99 Chapter summary in 20 lines 234

16 Online and Offline 236

16.0 What this chapter gives you 236
The plain version 237
Where the plain version stops being true 238
The technical version 240
16.98 Common wrong ideas 249
16.99 Chapter summary in 20 lines 250

17 Proving It Is You 252

17.0 What this chapter gives you 252
The plain version 253
Where the plain version stops being true 255
The technical version 256
17.98 Common wrong ideas 266
17.99 Chapter summary in 20 lines 267

18 Contactless 269

18.0 What this chapter gives you 269
The plain version 270
Where the plain version stops being true 272
The technical version 273
18.98 Common wrong ideas 281
18.99 Chapter summary in 20 lines 282

19 The Token in Your Phone 284

19.0 What this chapter gives you 284
The plain version 285
Where the plain version stops being true 286
The technical version 288
19.98 Common wrong ideas 297
19.99 Chapter summary in 20 lines 298

20 The Terminal **300**

20.0 What this chapter gives you 300
The plain version 301
Where the plain version stops being true 303
The technical version 304
20.98 Common wrong ideas 312
20.99 Chapter summary in 20 lines 313

21 Point-to-Point Encryption **315**

21.0 What this chapter gives you 315
The plain version 316
Where the plain version stops being true 317
The technical version 318
21.98 Common wrong ideas 327
21.99 Chapter summary in 20 lines 328

22 What the Terminal Actually Sends **330**

22.0 What this chapter gives you 330
The plain version 331
Where the plain version stops being true 332
The technical version 333
22.98 Common wrong ideas 342
22.99 Chapter summary in 20 lines 343

Volume III — The Networks **345**

23 The Four-Party Model **346**

23.0 What this chapter gives you 346
The plain version 346
Where the plain version stops being true 348
The technical version 349
23.98 Common wrong ideas 357
23.99 Chapter summary in 20 lines 358

24 ISO 8583 **360**

24.0 What this chapter gives you 360
The plain version 361
Where the plain version stops being true 363
The technical version 363
24.98 Common wrong ideas 373
24.99 Chapter summary in 20 lines 374

25 The Bitmap **376**

25.0 What this chapter gives you 376
The plain version 377
Where the plain version stops being true 379
The technical version 380
25.98 Common wrong ideas 387
25.99 Chapter summary in 20 lines 388

26 The Fields That Matter **390**

26.0 What this chapter gives you 390
The plain version 391
Where the plain version stops being true 392
The technical version 393

26.98 Common wrong ideas 403
26.99 Chapter summary in 20 lines 404

27 A Transaction, Narrated 406

27.0 What this chapter gives you 406
The plain version 407
Where the plain version stops being true 408
The technical version 409
27.98 Common wrong ideas 419
27.99 Chapter summary in 20 lines 420

28 The Number You Were Given 422

28.0 What this chapter gives you 422
The plain version 422
Where the plain version stops being true 424
The technical version 424
28.98 Common wrong ideas 432
28.99 Chapter summary in 20 lines 433

29 Response Codes 435

29.0 What this chapter gives you 435
The plain version 435
Where the plain version stops being true 437
The technical version 438
29.98 Common wrong ideas 448
29.99 Chapter summary in 20 lines 449

30 Authorisation, Clearing, Settlement 451

30.0 What this chapter gives you 451
The plain version 452
Where the plain version stops being true 454
The technical version 454
30.98 Common wrong ideas 463
30.99 Chapter summary in 20 lines 464

31 Where the Money Goes 466

31.0 What this chapter gives you 466
The plain version 466
Where the plain version stops being true 468
The technical version 469
31.98 Common wrong ideas 477
31.99 Chapter summary in 20 lines 478

32 ISO 20022 480

32.0 What this chapter gives you 480
The plain version 481
Where the plain version stops being true 483
The technical version 484
32.98 Common wrong ideas 493
32.99 Chapter summary in 20 lines 494

33 The Physical Network 496

33.0 What this chapter gives you 496
The plain version 497
Where the plain version stops being true 499
The technical version 499
33.98 Common wrong ideas 509

33.99 Chapter summary in 20 lines 509

Volume IV — Moving Money Without Cards 512

34 Batch Against Real Time 513

34.0 What this chapter gives you 513
The plain version 514
Where the plain version stops being true 515
The technical version 517
34.98 Common wrong ideas 525
34.99 Chapter summary in 20 lines 526

35 Bacs and the Three-Day Cycle 528

35.0 What this chapter gives you 528
The plain version 529
Where the plain version stops being true 530
The technical version 531
35.98 Common wrong ideas 541
35.99 Chapter summary in 20 lines 541

36 The Mandate 543

36.0 What this chapter gives you 543
The plain version 544
Where the plain version stops being true 546
The technical version 547
36.98 Common wrong ideas 555
36.99 Chapter summary in 20 lines 556

37 Faster Payments 558

37.0 What this chapter gives you 558
The plain version 559
Where the plain version stops being true 561
The technical version 562
37.98 Common wrong ideas 569
37.99 Chapter summary in 20 lines 570

38 CHAPS and RTGS 572

38.0 What this chapter gives you 572
The plain version 573
Where the plain version stops being true 575
The technical version 577
38.98 Common wrong ideas 583
38.99 Chapter summary in 20 lines 584

39 Addressing: Sort Codes, IBANs and Confirmation of Payee 586

39.0 What this chapter gives you 586
The plain version 587
Where the plain version stops being true 589
The technical version 590
39.98 Common wrong ideas 599
39.99 Chapter summary in 20 lines 600

40 SEPA 602

40.0 What this chapter gives you 602
The plain version 603

Where the plain version stops being true 604
The technical version 606
40.98 Common wrong ideas 616
40.99 Chapter summary in 20 lines 616

41 SWIFT and Correspondent Banking 618

41.0 What this chapter gives you 618
The plain version 619
Where the plain version stops being true 621
The technical version 623
41.98 Common wrong ideas 631
41.99 Chapter summary in 20 lines 631

42 UPI 633

42.0 What this chapter gives you 633
The plain version 634
Where the plain version stops being true 635
The technical version 636
42.98 Common wrong ideas 644
42.99 Chapter summary in 20 lines 645

43 Instant Rails Elsewhere 647

43.0 What this chapter gives you 647
The plain version 648
Where the plain version stops being true 650
The technical version 652
43.98 Common wrong ideas 658
43.99 Chapter summary in 20 lines 659

44 Open Banking 661

44.0 What this chapter gives you 661
The plain version 662
Where the plain version stops being true 664
The technical version 666
44.98 Common wrong ideas 674
44.99 Chapter summary in 20 lines 675

Volume V — Trust, Failure and the Law 677

45 Where Cards Are Stolen 678

45.0 What this chapter gives you 678
The plain version 679
Where the plain version stops being true 680
The technical version 681
45.98 Common wrong ideas 690
45.99 Chapter summary in 20 lines 691

46 Card Not Present 693

46.0 What this chapter gives you 693
The plain version 693
Where the plain version stops being true 695
The technical version 696
46.98 Common wrong ideas 705
46.99 Chapter summary in 20 lines 706

47 Three-D Secure and Strong Customer Authentication 708

47.0 What this chapter gives you 708
The plain version 708
Where the plain version stops being true 710
The technical version 712
47.98 Common wrong ideas 721
47.99 Chapter summary in 20 lines 721

48 PCI DSS and Its Relatives 724

48.0 What this chapter gives you 724
The plain version 724
Where the plain version stops being true 726
The technical version 727
48.98 Common wrong ideas 737
48.99 Chapter summary in 20 lines 738

49 Chargebacks 740

49.0 What this chapter gives you 740
The plain version 740
Where the plain version stops being true 742
The technical version 743
49.98 Common wrong ideas 753
49.99 Chapter summary in 20 lines 753

50 Reconciliation 755

50.0 What this chapter gives you 755
The plain version 755
Where the plain version stops being true 758
The technical version 759
50.98 Common wrong ideas 766
50.99 Chapter summary in 20 lines 767

51 Fraud Detection 769

51.0 What this chapter gives you 769
The plain version 769
Where the plain version stops being true 772
The technical version 773
51.98 Common wrong ideas 781
51.99 Chapter summary in 20 lines 781

52 Anti-Money-Laundering 783

52.0 What this chapter gives you 783
The plain version 783
Where the plain version stops being true 785
The technical version 786
52.98 Common wrong ideas 794
52.99 Chapter summary in 20 lines 795

53 Regulation 797

53.0 What this chapter gives you 797
The plain version 797
Where the plain version stops being true 799
The technical version 800
53.98 Common wrong ideas 809
53.99 Chapter summary in 20 lines 810

54 What Breaks and What It Costs **812**

54.0 What this chapter gives you . . . 812
The plain version . . . 813
Where the plain version stops being true . . . 814
The technical version . . . 816
54.98 Common wrong ideas . . . 824
54.99 Chapter summary in 20 lines . . . 825

VOLUME I

What Money Is

Obligation, records and double entry. What a balance actually is, what a bank actually is, and the difference between clearing and settlement. Everything here is permanent.

CHAPTER 1

The Oldest Question

1.0 What this chapter gives you

1. You will be able to explain why a payment is an edit to a record rather than the movement of an object, using Ada's notebook as the worked example.
2. You will be able to state Caroline Humphrey's claim precisely — that no barter economy, pure and simple, has ever been described — and say what it does not claim, so that you are not caught out by anyone who has read the source.
3. You will be able to describe how a tally stick worked, why splitting a notched stick lengthways made it hard to forge, and how the words *stock* and *foil* reached modern finance from a piece of hazel.
4. You will be able to place coinage correctly in time: administrative writing at Uruk from about 3100 BC, the first coins some two and a half thousand years later beneath the temple at Ephesus.
5. You will be able to explain why the earliest coins undermine the commodity theory of money rather than supporting it, given that Lydian electrum of roughly 54 per cent gold was tariffed as though it were 73 per cent.
6. You will be able to distinguish the unit in which an obligation is denominated from the asset in which it is settled, and recognise that distinction as four thousand years old.
7. You will be able to say what actually makes a record money — that an identifiable party is obliged to honour it, and that something makes the obligation stick — rather than repeating the slogan that money is information.
8. You will be able to name the counterparty behind any unit of money you hold: your bank for a deposit, the Bank of England for a banknote, the retailer for a gift card.
9. You will be able to quote the British proportions that matter: about £91.5 billion of notes across roughly 4.98 billion of them, against bank deposits making up 97 per cent of the money in circulation.
10. You will be able to read a hundred-character Bacs Standard 18 record and explain why three of its fields exist purely so that human beings on both sides can later agree what the record meant.

Ask a hundred people what happened when they last paid for something and ninety-nine will describe a movement. Money left my account and went to theirs. Money came out of the machine. Money changed hands.

Almost none of that is true, and the exceptions are rarer than you think. When you tap a card in a shop in Leeds, nothing physical travels between you and the shopkeeper. When your salary arrives on the twenty-fifth, no lorry of banknotes crosses the country. In every one of these cases, what changed was a record. Somebody's obligation to somebody else was written down, cancelled, netted off, or

handed on. The physical object, when there is one, is a token that stands for the record. The record is the money.

This is not a modern discovery about a modern system. It is the oldest fact about money there is, and the evidence for it is older than coins and older than writing. This chapter is about that evidence: the clay of Mesopotamia, the split hazel sticks of the English Exchequer, and the awkward, well-documented fact that the barter economy every schoolchild is taught about has never actually been found.

The reason this matters for a book about payments is simple. If money were a substance, moving it would be a logistics problem, and the interesting questions would be about vaults and armoured vans. Because money is a record, moving it is an information problem, and the interesting questions are about who keeps the record, how two records are reconciled, what happens when they disagree, who bears the loss when one is wrong, and how quickly the thing becomes final. That is what the rest of this book is about. This chapter is about why.

The plain version

Imagine a village with one shop.

The shop is run by a woman called Ada. She sells bread, milk, nails, batteries and the sort of things you need at short notice. Nobody in the village has much cash, because the nearest bank is forty minutes away, so Ada keeps a notebook under the counter. When Tom the carpenter comes in for a loaf, Ada writes: *Tom, 1 loaf, £1.20.* When Tom finishes a set of shelves for Ada's back room, she writes: *Ada owes Tom, £150.*

Nothing has moved. There are no coins in this story yet. But something real has happened. Tom is owed £150 by Ada. Tom owes Ada £1.20. If Tom keeps buying bread every day for four months, the two numbers will meet in the middle and cancel each other out, and the whole relationship will settle without a single coin ever appearing.

Now watch what happens when the village gets a little bigger.

Sam the mechanic fixes Tom's van. The bill is £80. Tom has no cash. But Tom knows that Ada owes him £150, so he says to Sam: "Go to Ada. Tell her to take £80 off what she owes me and put it against your account instead." Sam agrees, because Sam trusts Ada's notebook, and because Sam owes Ada money anyway for a year of milk and nails. Ada opens the book and writes three lines. Tom's credit falls from £150 to £70. Sam's debt to Ada falls by £80. Tom's debt to Sam is gone.

Look at what just occurred. A debt was paid. Nobody handed over anything. No object changed owners. A record was edited, and everyone accepted the edit because everyone accepted the record-keeper. This is a payment. It is the whole of payment, in miniature, and every payment system in the world is a more elaborate version of Ada's notebook with more checks, more speed and more lawyers.

Now here is the part that surprises people. This is not a clever modern shortcut that people invented once coins became inconvenient. As far as anyone can tell from the archaeological and anthropological record, this is the *original* arrangement. Ada's notebook came first. Coins came very much later.

The story you were probably told

The version most of us learn goes like this. Long ago, people bartered. If you had shoes and wanted bread, you had to find a baker who happened to want shoes. This was maddening, so somebody hit on the idea of a thing everyone would accept — cattle, salt, shells, eventually lumps of silver — and money was born to fix the problem of barter.

It is a beautiful story. Adam Smith told a version of it in 1776 in *The Wealth of Nations*, using the example of a butcher with more meat than he can eat and a brewer and a baker who want some of it but have nothing the butcher happens to need. Smith's conclusion was that every prudent person would start keeping a stock of "some one commodity or other, such as he imagined few people would be likely to refuse in exchange", and that metals eventually won because they do not rot and can be cut into precise amounts.

The trouble with the story is that when anthropologists went looking for a society that actually worked this way, they could not find one. Not a single one. What they found instead, over and over, in every corner of the world, was some version of Ada's notebook: people who knew each other, kept mental or written track of who owed what, and settled up occasionally or never. Where they found barter, they usually found it happening between strangers, or between people who had no shared record-keeper and no intention of building a relationship, or in places where a money system had recently fallen apart.

The reason is not hard to see once you look at Ada's village. Barter is what you do when you cannot trust a record. If you and I are neighbours, a running tab is easier for both of us than lugging goods around and haggling over exchange rates. It is only when I do not expect to see you again, or cannot enforce anything against you, that we both insist on a simultaneous swap where neither of us has to trust the other for a single day.

Two copies of one fact

The obvious weakness of Ada's notebook is Ada. What if she writes the wrong number, or rubs a line out, or dies and leaves the book to a nephew who reads Tom's £150 as £15?

Human beings solved this a very long time ago, and one of the neatest solutions was a stick.

Take a piece of hazel or willow, about the length of your hand, and cut notches across it. A wide notch means a large amount, a narrow notch a small amount. Now — and this is the clever part — split the stick lengthways, right through every notch, so that you end up with two halves that each carry the same notches, along the same grain, with the same knots and flaws.

Give one half to the person who is owed the money. Keep the other half yourself. Neither half can be altered without the alteration being obvious, because if either of you adds or removes a notch the two halves will no longer match. To settle the debt, the two halves are brought back together. If they fit, the record is proved. If they do not, somebody is lying, and the wood says so.

That is a tally stick. Two copies of one fact, made from a single piece of wood so that no forger can produce a matching partner. The English Exchequer — the government department that collected the king's revenue — used them for something like seven hundred years, and the vocabulary has stuck to English so firmly that we no longer notice it. The longer half of the split stick, held by the person owed the money, was called the *stock*. That is why an investor holds *stock*, and why a person who lends to a company is a *stockholder*. The shorter half was the *foil*, held by the debtor, which is why getting a raw deal is getting the *short end of the stick*.

And here is the point that ought to stop you. A tally stick was not a receipt in the modern sense, a dead scrap of paper filed away. It circulated. If the Exchequer owed you £100 and gave you a tally saying so, you could pay your own creditor by handing over the tally. He would accept it because he, in turn, could hand it to the taxman and have it counted against his own bill. The stick moved from hand to hand carrying an obligation with it, and it did this for centuries. It was, in every practical sense, money. It was also, unmistakably, a record. Not gold, not silver, not a substance of any kind. A

notched stick, whose entire value came from the fact that a specific institution had written down that it owed you something.

The clay before the wood

Go back much further and you find the same idea in a different material.

In the cities of southern Mesopotamia — modern-day Iraq — around five thousand years ago, people were running large organised economies centred on temples and palaces. Grain came in from the fields. Rations went out to workers. Sheep were counted, beer was issued, labour was allocated. All of this had to be tracked, and the tracking was done in clay, because clay was free and it lasted.

The earliest of these records are not sentences. They are not literature or prayer or law. They are accounts. They say things like: *this many measures of barley, from this store, to this person.* One clay tablet in the British Museum, made in the very last centuries of the fourth millennium BC, is a record of beer, marked with five different kinds of number sign. It is about the size of a postcard. It is not a poem. It is a ledger entry.

This is worth sitting with, because it turns the usual story of civilisation upside down. Writing was not invented so that people could tell stories and later, as an afterthought, applied to business. On the best current evidence, it went the other way. The pressure that produced writing was the pressure of keeping track of who owed what to whom. The alphabet is a side effect of accountancy.

And how long after this did coins arrive? Roughly two and a half thousand years. The first objects most people would recognise as coins — small, standardised, stamped lumps of precious metal — appear in what is now western Turkey around the end of the seventh century BC. Ninety-three of them were dug up by British Museum archaeologists beneath the great temple of Artemis at Ephesus, left there as offerings.

Put that gap in terms you can feel. If the clay ledgers of Uruk had been written on the day of the Battle of Hastings, coins would not have been invented until about four hundred years from now. Writing down debts is not a step on the road to coins. Coins are a late refinement on a system of written obligation that was already ancient when they turned up.

What money is, in one sentence

Every form of money you have ever handled is a record of an obligation before it is an object.

The five-pound note in your pocket says so on its face, in words most people have stopped reading: *I promise to pay the bearer on demand the sum of five pounds.* That is not decoration. It is a description of what the note is. The note is the Bank of England's IOU. The bank owes you; the note is the evidence.

The number in your banking app is the same thing one step further along. It is not a store of your pounds sitting in a box with your name on it. It is your bank's written admission that it owes you that amount, payable on demand. If you have ever wondered why bank failures are frightening in a way that losing a wallet is not, this is why. A wallet is a container. A bank balance is a promise.

And the proportions are not close. At the end of February 2026, all the Bank of England banknotes in existence — every fiver, tenner, twenty and fifty in every till, pocket and cash centre in the country — came to about £91.5 billion, spread across roughly 4.98 billion notes. Set that against the money that exists only as entries in bank ledgers and it is a rounding error. The Bank of England's own analysis puts bank deposits at 97 per cent of the money in circulation. Ninety-seven pounds in every hundred is a line in a database. Three are paper.

Now look at what that money does in a year. In 2025, the Bacs system — the one that pays most British salaries and collects most British household bills — processed 6.86 billion payments worth £6.05 trillion. Faster Payments, the system behind the instant transfer you make from your phone, handled 5.55 billion payments worth £4.84 trillion. That is well over twelve billion instructions in a single year, moving nearly eleven trillion pounds, and at no point did a physical object representing value travel from a payer to a payee. What travelled was information: a short, structured message saying *reduce this account by this much, increase that account by this much, and here is the reference so both sides can agree it happened.*

Ada's notebook, scaled to sixty-eight million people.

Where the plain version stops being true

The plain version above is a good map. Like all good maps it lies by simplification, and four of those lies matter enough to correct before we build anything on top of them.

Barter did exist; what has never been found is a barter economy

It is fashionable now to say flatly that barter is a myth. That overshoots, and a reader who repeats it at dinner will be caught out by anyone who has actually read the source.

The foundational statement is Caroline Humphrey's, in the journal *Man* in 1985, and it is more careful than its popular version: "No example of a barter economy, pure and simple, has ever been described, let alone the emergence from it of money; all available ethnography suggests that there never has been such a thing." Read the clauses. She is not saying that people never swap goods directly. Of course they do; it happens in playgrounds, at borders, between prisoners, and across the front lines of wars. She is saying two narrower and much stronger things: that no society has ever been observed running its whole economy on direct exchange without money, and that money has never been observed emerging out of such a society.

Humphrey's own fieldwork makes the point sharply. She studied the Lhomi of north-eastern Nepal in 1979 and 1980, a community that did a great deal of barter. They did not do it because they had never heard of money. They did it because currency was scarce and their links to monetary markets had broken down. Her argument is that barter in the observed world is overwhelmingly a *post*-monetary phenomenon: it appears where a money economy has decayed, been cut off, or run short of the stuff. It coexists with money rather than preceding it.

So the correction is this. Barter is real. What is unsupported is the sequence — barter first, then money to fix barter — and the idea that money is fundamentally a solution to the inconvenience of swapping shoes for bread. The evidence points instead to credit and reckoning coming first, with barter as one of several things that happen at the edges: between strangers, across hostile boundaries, and in the wreckage of monetary systems.

There is a second, subtler correction hiding here. Societies without general-purpose money often do not have one undifferentiated market at all. Paul Bohannan's study of the Tiv of central Nigeria, published in *American Anthropologist* in 1955, described distinct spheres of exchange, with goods in one sphere not freely convertible into goods in another, because the culture did not treat them as commensurable. General-purpose money did not merely lubricate exchange in such societies; it dissolved walls that people had wanted kept up. Money is not a neutral efficiency improvement laid over an existing market. It changes what counts as a market.

"Credit came first" does not mean neighbours writing IOUs

The Mesopotamian evidence is real, but it is easy to over-read it in a modern direction.

The tablets from Uruk are overwhelmingly *institutional* accounts. They are the internal records of large temple and palace estates disbursing rations and tracking obligations owed to and by the institution. They are much closer to the payroll and stock-control ledgers of a very large organisation than to two neighbours agreeing a private loan. The obligations they record are frequently administrative — quotas, allocations, deliveries due — rather than freely negotiated credit between equals. Calling this "credit" is fair, but you should picture a state accounting department, not a friendly overdraft.

The second over-read concerns the tokens. Denise Schmandt-Besserat's thesis — that small clay tokens used for counting from around the eighth millennium BC were sealed inside clay envelopes called *bullae*, that the shapes were then impressed on the outside of the envelope so you could see the contents without breaking it, and that the impressions eventually made the tokens redundant and became writing — is enormously influential and genuinely elegant. It is also contested in the detail. Not every specialist accepts that the token system maps cleanly onto the earliest sign inventory, or that a single evolutionary line runs from token to envelope to tablet. Treat the token-to-writing story as the leading hypothesis rather than a settled fact, and note that the broader claim — that the earliest writing was administrative and economic — rests on the tablets themselves and does not depend on the token thesis being right.

The tally stick was a receipt that *became* negotiable, and the sources disagree about the details

Two cautions. First, the tally began life as proof of payment into the Exchequer, not as a freely circulating currency. Its negotiability was a consequence of what it was — a hard-to-forge, institution-backed record of a sum owed by the Crown — rather than its design purpose. A tally in a strongbox was a receipt. A tally passed to a creditor was money. Same object, different use. Money is a role played by a record, not a class of object, and the tally is the cleanest illustration in European history.

Second, be careful about the notch conventions, because sources genuinely conflict. The *Dialogus de Scaccario*, the treatise on Exchequer procedure written around 1179, gives them in body parts: a cut the thickness of the palm for a thousand pounds, the breadth of a thumb for a hundred, the breadth of the little finger for twenty, the width of a swollen barleycorn for a single pound, and a narrower cut for a shilling. The 1911 *Encyclopædia Britannica* gives them in inches, and does not match: one and a half inches for a thousand pounds, one inch for a hundred, three-eighths of an inch for ten, a half-notch of that size for a pound, three-sixteenths for a shilling, the smallest notch for a penny, and small drilled holes for halfpennies. Note that the two schemes do not even agree on which denominations existed — the Dialogus has a twenty-pound notch and the Britannica has a ten-pound notch. The honest position is that the conventions varied across seven centuries and that no single table describes them all.

Anyone who tells you the tally system was uniform for seven hundred years is telling you something the sources do not support.

Information is only money when someone is obliged to act on it

This is the correction that matters most for everything that follows, and it is where the slogan "money is information" starts to mislead.

A record is not money because it is a record. It is money because a specific, identifiable party is obliged to honour it, and because there is a mechanism that makes that obligation stick. Ada's notebook worked because Ada was solvent, present, honest and answerable to a village that would make her life difficult if she cheated. Take away any one of those and the notebook becomes a list of numbers.

Three consequences follow, and they run through the rest of this book.

The first is that every unit of money in a modern economy has a counterparty. Your bank balance is your bank's liability. A banknote is the central bank's liability. A gift card is the retailer's liability. When people say a payment "settles", what they mean precisely is that the obligation has been discharged and the record can no longer be reversed. Settlement is the moment a promise stops being a promise, and a great deal of the machinery in this book exists to define that moment exactly, because until it arrives somebody is carrying risk.

The second is that cash is genuinely different, and the plain version glosses over it. When I hand you a twenty-pound note, the transaction is complete at the instant of handover: no clearing, no settlement window, no counterparty left owing anything, no possibility of recall. That is a real and unusual property, and it is why cash remains stubbornly useful in exactly the situations where records are unreliable — power cuts, disasters, distrust and crime. Every electronic payment is an attempt to reproduce the finality of cash across distance and time, and none fully manages it.

The third is that Alfred Mitchell-Innes's famous formulation — writing in the *Banking Law Journal* in 1913, "Money, then, is credit and nothing but credit. A's money is B's debt to him, and when B pays his debt, A's money disappears" — is a position in a long-running argument, not a neutral summary of the facts. Innes was demolishing the commodity theory of money, and he was right about the Exchequer in ways that mainstream histories of his day had got badly wrong. He was also making a maximalist claim. Economists still disagree about whether commodity-money origins matter, about how far the state's power to demand taxes explains a currency's acceptance, and about whether "credit" is the right frame for physical cash. This book leans towards the credit view because the historical record supports it and because it is by far the more useful frame for understanding payment systems. You should know that it is a lean and not a law.

The technical version

The anthropological critique, stated precisely

The claim under examination is what economists call the *double coincidence of wants* narrative, given its canonical modern statement in Adam Smith, *An Inquiry into the Nature and Causes of the Wealth of Nations* (1776), Book I, Chapter IV, "Of the Origin and Use of Money". Smith's argument is deductive rather than evidential: he reasons from the division of labour to the necessity of a commonly accepted medium, then to the selection of metals on the grounds of durability and divisibility. He offers no ethnographic case, because in 1776 there was none to offer.

The empirical counter-position is set out in Caroline Humphrey, "Barter and Economic Disintegration", *Man* (New Series), Volume 20, Issue 1 (1985), pages 48 to 72, whose ethnographic base is fieldwork among the Lhomi of the Upper Arun valley in north-eastern Nepal in 1979 and 1980. Her thesis is that observed barter is characteristically a phenomenon of economies "de-coupled from monetary markets" — that it coexists with money rather than preceding it, and that under severe currency scarcity money itself may become one of the items bartered rather than the index of value for the rest.

The argument was popularised, and considerably extended, in David Graeber, *Debt: The First 5,000 Years* (Melville House, 2011), which treats the barter narrative as an origin myth internal to economics rather than a historical finding. Graeber's synthesis has been criticised by economists and by some historians on points of detail; his central negative claim — that no one has produced the barter economy the textbooks require — has held up.

A related strand concerns the non-fungibility of value categories. Paul Bohannan, "Some Principles of Exchange and Investment among the Tiv", *American Anthropologist*, Volume 57, Number 1 (1955), pages 60 to 70, described goods segregated into distinct spheres, with movement within a sphere ordinary and movement between spheres exceptional and morally freighted. Marcel Mauss's *Essai sur le don* (1925) had earlier established that obligation-creating gift exchange, rather than spot barter, dominates in many non-market societies. The cumulative picture is not that early societies had a primitive form of our market, but that they had systems of obligation whose logic differs from market exchange at the root.

The Mesopotamian record

Proto-cuneiform, the earliest attested phase of the cuneiform writing system, appears at Uruk in southern Mesopotamia and is conventionally dated to approximately 3100 to 2900 BC. The Metropolitan Museum of Art's account of the origins of writing notes that the site "was dominated by large temple estates whose need for accounting and disbursing of revenues led to the recording of economic data on clay tablets", that roughly 6,000 proto-cuneiform tablets survive with more than 38,000 lines of text, and that among the earliest signs are pictures of "rations that needed to be counted, such as grain, fish, and various types of animals".

A representative object is British Museum registration number 1989,0130.4, the so-called British Museum Society tablet, acquired at Christie's in 1989 from the Erlenmeyer collection and displayed in Room 56. It is dated 3100 to 3000 BC, measures 9.4 by 6.87 by 2.3 centimetres, and is catalogued as a record of beer impressed with five different types of numerical symbol. The presence of multiple distinct numerical notations on a single tablet is itself informative: proto-cuneiform used different sign systems for different commodity classes, so that counting sheep and measuring grain did not share a notation.

The prehistory of these tablets is the token thesis associated with Denise Schmandt-Besserat, set out in *Before Writing, Volume I: From Counting to Cuneiform* (University of Texas Press, 1992). The proposed sequence runs: small geometric clay tokens used as counters from the Neolithic; tokens sealed inside hollow clay envelopes (*bullae*) to fix a transaction's content; tokens impressed on the envelope's exterior before sealing so the contents could be read without breaking it; and finally the impressions alone, on a flat tablet, the tokens having become redundant. The thesis remains influential and remains disputed in its details.

On units and rates, the Old Babylonian and earlier systems were sexagesimal. The mina was divided into 60 shekels; the shekel was a weight of silver of roughly 8 grams. Michael Hudson's work on palatial credit records the equivalence of the shekel-weight of silver with a *gur* of grain of 300 *sila*, an equation set administratively by the palace rather than discovered in a market. Hudson's argument on interest is that the canonical Mesopotamian rate was a product of the number system rather than of economic conditions: one shekel per mina per month, that is one sixtieth per month, giving 12 shekels per mina per year, or 20 per cent per annum in decimal terms, and doubling the principal in 60 months under simple accrual. He notes, importantly for our purposes, that a debt denominated in silver did not require silver in settlement — "What was called a 'silver' debt thus did not mean that actual silver had to be paid, but simply that the interest rate was 20 percent" — with barley an accepted means of discharge.

That distinction between the unit in which an obligation is denominated and the asset in which it is settled is not an antiquarian curiosity. It is the same distinction that separates a currency from a settlement asset in a modern payment system, and it is four thousand years old.

The legal architecture around these obligations is visible in the laws of Hammurabi, of the eighteenth century BC. The provisions on merchants and agents are unambiguous about documentation: an agent must give a receipt for money received, and one who "is careless, and does not take a receipt for the money which he gave" bears the consequence. On deposits, the code requires a party leaving goods for safekeeping to "show everything to some witness, draw up a contract, and then hand it over for safe keeping", and denies remedy to one who deposits "without witness or contract". A society that legislates in this much detail about receipts and witnesses is a society whose economy runs on records.

The English Exchequer and the tally

The Exchequer tally is documented in the *Dialogus de Scaccario*, written about 1179 and attributed to Richard FitzNeal, Treasurer of England. The physical description in the 1911 *Encyclopædia Britannica* is of "a willow or hazel stick about one inch in depth and thickness, and roughly shaped like a thick knife-blade", notched across its width and then split lengthways through the notches.

The two halves had names. The longer piece, retained by the party to whom the sum was owed, was the *stock*; the shorter, retained by the Exchequer as the debtor's record, was the *foil*. Details of the sum were also written on the wood in ink before splitting, so that each half carried both a notched and a written record. Verification was by physical fit: if genuine, the halves "fitted so exactly that they appeared evidently to be parts the one of the other". This is a mechanical integrity check with no cryptographic component and no central register, and it worked for centuries because the grain and flaw pattern of a split stick was effectively unforgeable by the technology of the period.

On notch values, as noted above, the sources conflict. Set them side by side.

Denomination	Dialogus de Scaccario (c. 1179)	1911 Encyclopædia Britannica
£1,000	cut the thickness of the palm	notch 1.5 inches wide
£100	breadth of a thumb	notch 1 inch wide
£20	breadth of the little finger	not listed
£10	not listed	notch 0.375 inch wide
£1	width of a swollen barleycorn	half-notch of the £10 size
1 shilling	a single cut, narrower	notch 0.1875 inch wide
1 penny	not specified in this form	smallest notch
halfpenny	not specified in this form	a small drilled hole

The correct reading of that table is that the convention changed over time and that neither source describes the whole period.

The tally's monetary character rests on its transferability. Mitchell-Innes, in his 1913 *Banking Law Journal* article "What is Money?", put it directly: "Tallies were transferable, negotiable instruments, just like bills of exchange, bank-notes or coins." His accompanying claim about the Exchequer has aged best: "The general belief that the Exchequer was a place where gold or silver was received, stored and paid out is wholly false. Practically the entire business of the English Exchequer consisted in the issuing and receiving of tallies." From this he drew the conclusion that gives the credit theory its name: "Money, then, is credit and nothing but credit. A's money is B's debt to him, and when B pays his debt, A's money disappears."

The system's end is precisely datable and unusually consequential. An Act of 1782 provided for the discontinuance of wooden tallies, but the change took effect only on the death of the last chamberlain, in 1826. The accumulated stock of obsolete tallies — two cart-loads of them — remained in the Palace of Westminster until 16 October 1834, when Richard Weobley, the Clerk of Works, acting on Treasury

instruction, had them burned in the two underfloor stoves in the basement of the House of Lords. Two labourers, Joshua Cross and Patrick Furlong, fed the stoves through the day. The first flames were seen at six o'clock that evening. The fire destroyed the House of Lords, the House of Commons, St Stephen's Chapel, the Painted Chamber, the libraries and committee rooms of both Houses, the Speaker's House, most of the procedural records of the Commons going back to the late fifteenth century, and the British standard yard and pound of 1496. Westminster Hall and the Jewel Tower survived. Charles Dickens, addressing the Administrative Reform Association two decades later, remarked that in the matter of the tallies "all the red tape in the country grew redder at the bare mention of this bold and original conception".

The Palace of Westminster you see today exists because a government disposed carelessly of an obsolete accounting medium. It is the most expensive records-management incident in British history.

Coinage as a late and specialised development

The chronology matters, so state it plainly. Administrative writing in Mesopotamia is attested from approximately 3100 BC. Coinage — standardised, stamped, portable metal of certified weight — appears roughly two and a half millennia later.

The evidential anchor is the deposit beneath the Temple of Artemis at Ephesus, where British Museum excavators recovered 93 electrum coins deposited as offerings during the latter part of the seventh century BC. The Lydian royal issues bearing a lion's head run from about 4.7 grams, one third of a Lydian weight stater, down through fractions to pieces of 0.29 grams (one forty-eighth) and 0.15 grams (one ninety-sixth) bearing a lion's paw. Analysis of the metal is instructive: the alloy is around 54 per cent gold, 44 per cent silver and 1 to 2 per cent copper, but the coins were tariffed as if they were natural Lydian electrum of roughly 73 per cent gold. The issuer was, in other words, extracting a seigniorage from the gap between the metal's intrinsic content and its official rating.

That fact deserves emphasis, because it undermines the commodity story at its supposed strongest point. Even the very first coins were not simply weighed metal. They were metal whose accepted value depended on an institution's stamp rather than on its assay. The stamp is a record. Coinage did not replace institutional trust with intrinsic value; it packaged institutional trust into a portable, hard-to-counterfeit form.

Obligation on the modern balance sheet

The Bank of England's *Quarterly Bulletin* for 2014 Q1 carried the article "Money creation in the modern economy" by Michael McLeay, Amar Radia and Ryland Thomas. It is the single most useful primary source for a British reader on what money now is, and its central statements are worth quoting exactly.

On composition: "Broad money is made up of bank deposits — which are essentially IOUs from commercial banks to households and companies — and currency — mostly IOUs from the central bank."

On proportion: "Of the two types of broad money, bank deposits make up the vast majority — 97% of the amount currently in circulation."

On creation: "Whenever a bank makes a loan, it simultaneously creates a matching deposit in the borrower's bank account, thereby creating new money." And, correcting the textbook account: "Banks do not act simply as intermediaries, lending out deposits that savers place with them."

Set the physical stock against that. Bank of England banknote statistics record over 4.98 billion notes in circulation with a total value of about £91.5 billion as at 28 February 2026 — three per cent of the money in the economy, held in the form everyone pictures when the word "money" is said.

The consequence for a payments practitioner is structural. If 97 per cent of money is a liability recorded in commercial bank ledgers, then a payment between customers of two different banks is a message that causes two banks to adjust two sets of liabilities, plus a settlement between those banks in a third asset — central bank reserves — that neither customer ever sees. Every layer of the payments stack in this book exists to make that three-way adjustment happen correctly, in the right order, with an unambiguous point of finality.

What actually moves: the United Kingdom in numbers

Pay.UK's annual statistics for calendar year 2025 give the retail picture.

System	Volume, 2025	Value, 2025
Bacs (total)	6.86 billion items	£6.05 trillion
of which Direct Debit	5.03 billion items	£1.53 trillion
of which Direct Credit	1.83 billion items	balance of Bacs total
Faster Payments	5.55 billion items	£4.84 trillion
Cheque and credit (image clearing)	78.7 million items	£112.7 billion

The wholesale picture sits above it. The Bank of England reports that in 2024 CHAPS, the United Kingdom's high-value real-time gross settlement system, settled over £87 trillion, averaging over £344 billion each working day across around 208,000 payments a day. Roughly 208,000 instructions a day carry more value than 27 million Bacs items, which tells you most of what you need to know about the difference between retail and wholesale payment.

Now look at the shape of a single Bacs instruction, because it is the moment where the argument of this chapter becomes concrete. The Bacs Standard 18 format, the submission format used across the United Kingdom for Direct Debits and Direct Credits, is a fixed-width record of 100 characters, extended to 106 characters for multi-processing files.

Positions	Length	Field
1–6	6	Destination sorting code
7–14	8	Destination account number
15	1	Destination account type (zero filled)
16–17	2	Transaction code
18–23	6	Originating sorting code
24–31	8	Originating account number
32–35	4	Free format / RTI or PAYE indicator
36–46	11	Amount in pence
47–64	18	User's name
65–82	18	User's reference
83–100	18	Destination account name
101–106	6	Bacs processing day (multi-processing files only)

One hundred characters. Two account identifiers, a two-character verb, an amount in pence, and three eighteen-character strings that exist purely so that human beings on both sides can later agree about what the record meant. That is a payment. That is the whole of it.

The record then moves through the three-day cycle that Pay.UK's system principles set out: day one, input, when payment instructions and Direct Debit Instructions are submitted to the Bacs system; day

two, processing, when payment service providers process the items; day three, entry and settlement, when providers apply the payments to customer accounts and interbank settlement occurs. Input, processing and entry occur only on processing days; Saturdays, Sundays and English bank holidays are excluded. That three-day rhythm is not a technical limitation of any modern computer. It is a convention about when obligations become final, and conventions about finality are what payment systems fundamentally are.

The industry is now migrating that structure to a richer one. ISO 20022 replaces fixed-width, position-defined records with structured XML messages carrying a far larger and more explicitly typed set of fields. For cross-border payments, the coexistence period during which Swift's legacy MT messages and ISO 20022 both ran ended on 22 November 2025, after which MT is no longer supported for payment instructions in the live environment, and in-flow translation became a chargeable contingency from 1 January 2026. The direction of travel is towards more information per payment, not less — remittance data, structured parties, purpose codes, legal entity identifiers. If money were a substance, none of this would matter. Because money is a record, the richness of the record is the product.

Why this framing organises the whole book

Every question that follows in this book is a question about records.

Clearing is the process of working out, from two sets of records, what each party owes the other. Settlement is the moment at which an obligation is extinguished and the record becomes final. Reconciliation is the discipline of proving that two independently kept records of the same events agree — which is what the two halves of a tally stick did mechanically, in wood, in the twelfth century. Double-entry bookkeeping is the same insight applied inside a single organisation. A chargeback is a rule about when a settled record may be unwound and who bears the loss. Anti-money-laundering controls are rules about what the record must contain and who must be able to read it. A cryptographic ledger is a mechanism for making a shared record tamper-evident without a trusted keeper, which is to say a tally stick for people who cannot meet to fit the halves together.

The units and institutions change. The problem does not. Somebody must write down who owes what to whom, somebody must be trusted to keep that writing honest, and everybody must agree when a promise has stopped being a promise.

Volume I of this book asks what money is: obligation, denomination, settlement asset, legal tender, currency, and the several different things people mean when they say "cash". The volumes that follow ask how the information moves: through schemes and rails, files and messages, banks and processors, borders and clearing houses, and the specific British machinery — Bacs, Faster Payments, CHAPS, the Image Clearing System, Open Banking — that most readers will actually be building against.

Start from the wrong premise and every one of those chapters becomes a list of arbitrary rules to memorise. Start from the right one and they become variations on a single, very old idea, already fully formed when somebody in Uruk pressed a number into wet clay to record how much beer was owed to a workman.

Money is information. This book is about how that information moves.

1.98 Common wrong ideas

1. **Wrong:** money was invented to solve the inconvenience of barter. **Right:** no society has ever been observed running its whole economy on direct exchange, and money has never been observed emerging from one; credit and reckoning come first in the evidence we actually have.

2. **Wrong:** barter is a myth. **Right:** barter is real and well documented, but the observed cases are overwhelmingly post-monetary, appearing where currency is scarce or a money economy has broken down, as among the Lhomi of north-eastern Nepal.
3. **Wrong:** the Mesopotamian tablets show neighbours writing IOUs to one another. **Right:** they are overwhelmingly institutional accounts of temple and palace estates, far closer to payroll and stock control than to a friendly overdraft.
4. **Wrong:** the token-to-envelope-to-tablet account of the origin of writing is settled fact. **Right:** it is the leading hypothesis and is contested in its details, and the broader claim that the earliest writing was administrative rests on the tablets themselves rather than on the token thesis.
5. **Wrong:** the tally stick was designed as a circulating currency. **Right:** it began as proof of payment into the Exchequer, and its negotiability was a consequence of being a hard-to-forge, institution-backed record of a Crown debt — a receipt in a strongbox, money when passed to a creditor.
6. **Wrong:** the notch conventions were uniform across the seven centuries the tally was in use. **Right:** the *Dialogus de Scaccario* and the 1911 *Encyclopædia Britannica* conflict, and do not even agree on which denominations existed, so no single table describes the whole period.
7. **Wrong:** money is information. **Right:** a record becomes money only when a specific, identifiable party is obliged to honour it and a mechanism makes that obligation stick; remove Ada's solvency, presence, honesty or answerability and the notebook is a list of numbers.
8. **Wrong:** cash is simply the physical version of a bank balance. **Right:** handing over a note completes the transaction at the instant of handover, with no clearing, no settlement window, no counterparty left owing anything and no possibility of recall, which is a property no electronic payment fully reproduces.
9. **Wrong:** "money is credit and nothing but credit" is the neutral summary of the historical evidence. **Right:** it is Mitchell-Innes's position in a long-running argument, and this book leans towards it because the record supports it and it is the more useful frame, but a lean is not a law.
10. **Wrong:** the three-day Bacs cycle is a technical limitation of the computers involved. **Right:** it is a convention about when obligations become final, and conventions about finality are what payment systems fundamentally are.

1.99 Chapter summary in 20 lines

1. Almost everyone describes a payment as a movement of money, and almost nobody is right.
2. What changes in a payment is a record: somebody's obligation to somebody else is written down, cancelled, netted off or handed on.
3. Ada's village notebook contains the whole of payment in miniature, because Tom's debt to Sam is discharged by three edited lines and nothing changes hands.
4. The textbook story that money was invented to fix the awkwardness of barter comes from Adam Smith in 1776 and is deductive rather than evidential.
5. Caroline Humphrey's careful 1985 statement is that no barter economy, pure and simple, has ever been described, let alone the emergence of money from one.
6. Barter itself is real, but observed barter is overwhelmingly a post-monetary phenomenon that co-exists with money rather than preceding it.
7. Societies without general-purpose money often kept goods in separate spheres of exchange, so money did not merely lubricate a market but changed what counted as one.
8. The earliest writing at Uruk, around 3100 BC, is administrative: rations, grain, sheep and beer, recorded with several distinct numerical notations.
9. Those tablets are the internal accounts of large temple and palace estates, and should be read as a state accounting department rather than as private credit.

10. Coins appear roughly two and a half thousand years after those tablets, in what is now western Turkey at the end of the seventh century BC.
11. Even the first coins were valued by the issuer's stamp rather than by assay, since electrum of about 54 per cent gold was tariffed as though it held 73 per cent.
12. The English Exchequer tally makes the same point in wood: one stick split lengthways through every notch, verified by physical fit, unforgeable by the technology of the day.
13. Tallies circulated from hand to hand because the taxman would accept them, which made a notched stick money without making it a substance.
14. The longer half was the stock and the shorter the foil, and the careless disposal of two cart-loads of obsolete tallies burned down the Palace of Westminster in 1834.
15. A record is money only when an identifiable party is obliged to honour it and there is a mechanism that makes the obligation stick.
16. Every unit of money in a modern economy therefore has a counterparty: your bank for a deposit, the central bank for a note, the retailer for a gift card.
17. Cash is the genuine exception, because handover is final and leaves nobody owing anything, which is why it survives where records are unreliable.
18. In Britain the proportions are stark: about £91.5 billion of banknotes exist, against bank deposits making up 97 per cent of the money in circulation.
19. In 2025 Bacs and Faster Payments together carried over twelve billion instructions worth nearly eleven trillion pounds, and not one of them moved an object.
20. Every question in the rest of the book — clearing, settlement, reconciliation, chargebacks, finality — is a question about who keeps the record and when a promise stops being a promise.

Sources used: Caroline Humphrey, "Barter and Economic Disintegration", Man (NS) 20:1 (1985), 48–72; Adam Smith, The Wealth of Nations (1776), Book I Ch. IV; Paul Bohannan, American Anthropologist 57:1 (1955), 60–70; A. Mitchell Innes, "What is Money?", Banking Law Journal (May 1913); The Metropolitan Museum of Art, "The Origins of Writing"; The British Museum, object 1989,0130.4; Denise Schmandt-Besserat, Before Writing (1992); Michael Hudson on palatial credit and Mesopotamian interest; the laws of Hammurabi (Harper translation); Dialogus de Scaccario (c. 1179) and the 1911 Encyclopædia Britannica entry "Tally"; UK Parliament, "The Great Fire of 1834" and "Tally Sticks"; John H. Kroll, "The Coins of Sardis" (Sardis Expedition); Bank of England, "Money creation in the modern economy", Quarterly Bulletin 2014 Q1, and Banknote Statistics (28 February 2026); Bank of England, RTGS and CHAPS introduction (2024 figures); Pay.UK, Annual Summary of Payment Statistics 2025 and Bacs System Principles v18 (January 2026); Bacs Standard 18 Message Implementation Guide; Swift ISO 20022 implementation guidance.

2

CHAPTER

A Balance Is Not a Thing

2.0 What this chapter gives you

1. You will be able to explain why the number in your banking app is a promise rather than a location, and why the envelope-in-a-drawer picture of a bank is the wrong one.
2. You will be able to state, in the Bank of England's own words, why bank lending creates deposits rather than deposits funding loans.
3. You will be able to say what genuinely constrains a bank's lending — capital, liquidity, funding cost and the price of money — and why no reserve multiplier appears on that list in the United Kingdom.
4. You will be able to distinguish a ledger balance from an available balance, and diagnose the customer who says a payment "took the money twice".
5. You will be able to cite *Foley v Hill* and explain why a depositor is an unsecured creditor of the bank rather than the beneficiary of a trust.
6. You will be able to state the FSCS limit precisely — £120,000 per eligible depositor per authorised firm for failures from 1 December 2025 — and explain why it attaches to the banking licence and not to the brand on the card.
7. You will be able to explain the difference between FSCS deposit protection and safeguarding under the Payment Services Regulations and Electronic Money Regulations, including the narrow case in which the FSCS looks through a safeguarding account.
8. You will be able to describe a bank run in terms of maturity transformation, sequential service and multiple equilibria, and say why deposit insurance works mainly by preventing runs rather than by paying for them.
9. You will be able to explain why the queue outside Northern Rock was an aftershock and why about $42 billion left Silicon Valley Bank in a single day without anyone queuing at all.
10. You will be able to say what legal tender actually means, why it rarely comes up, and why Scottish and Northern Irish banknotes are not legal tender anywhere.

Open your banking app. There is a number at the top of the screen. Most people, if pressed, would say that number represents money that is somewhere — in a vault, in a safe, in a very large drawer with their name on it. Some part of the mind pictures a stack of notes, slightly dusty, waiting.

There is no stack. There is no drawer. There is no vault with your name on it, and there never was.

What there is instead is a record. Your bank has written down that it owes you a certain amount, and the number on your screen is a rendering of that record. It is not a description of where anything is. It is a description of a promise.

This is not a cynical observation and it is not a warning. It is simply the mechanical truth of how deposit banking works, and almost everything else in this book depends on the reader having absorbed it properly. Payments do not move substances around. They rewrite records. Until that is clear, the rest of it — clearing, settlement, netting, why a refund takes five days — will feel arbitrary. Once it is clear, most of it becomes obvious.

The plain version

Imagine you are going away for a fortnight and you ask your neighbour to look after £500 in cash.

There are two entirely different arrangements she might be agreeing to, and it matters enormously which one it is.

In the first, she takes your five hundred pounds, puts it in an envelope, writes your name on the envelope, and puts the envelope in a drawer. She does not touch it. When you come home she gives you back the same envelope with the same notes inside. If you asked to see it at three in the morning, it would be there. This is safekeeping. The money never stopped being yours; she was just holding it.

In the second, she takes your five hundred pounds and puts it in her purse. She spends some of it on shopping. She lends a bit to her son. She uses some to pay a plumber. When you come home, she gives you five hundred pounds — but not *your* five hundred pounds, because those particular notes are long gone, scattered across a supermarket, a plumber's till and her son's petrol tank. What she gives you is five hundred pounds of somebody else's money that she happened to have. What you had for two weeks was not an envelope. It was a promise.

Nearly everyone believes a bank is the first arrangement. Every bank in the world is the second one.

Following £500 into a branch

Walk into a high street branch on a Tuesday morning and hand over £500 in twenty-pound notes. Watch what actually happens.

The cashier counts the notes. She puts them in her till. At the end of the day the till is emptied, the notes are bundled, and eventually they go off to a cash centre in a van, where they are mixed in with the notes from every other branch. Nobody writes your name on them. Nobody could. There is no mechanism by which a particular note stays associated with a particular person, and there is no reason anyone would want one.

Two things get written down. The bank's own stock of banknotes goes up by £500 — those notes are now the bank's notes, in exactly the way the shopping money was your neighbour's. And, separately, the number beside your name in the bank's records goes up by £500.

That second number is not a location. It is an obligation. It says: *this institution owes this person five hundred pounds, payable on demand.*

Now the bank does what banks do. It has thousands of customers doing the same thing, and it knows from long experience that on any ordinary day most of them will leave most of their money where it is. So it keeps enough notes and enough instantly available funds to handle the ordinary comings and goings, and it puts the rest to work — lending it to somebody buying a house, buying government bonds, financing a business. That is the entire commercial logic of a bank. It is not a cloakroom. It is an intermediary that stands between people who want their money back at any moment and borrowers who want to keep it for twenty-five years.

The arithmetic that shows there is no vault

Here is the number that makes the point unarguable.

Every Bank of England note that exists anywhere in the world — in tills, in pockets, under mattresses, in cash centres, in shoeboxes in Spain — adds up to about **£91.5 billion**, spread across roughly **4.98 billion individual notes**. That is the total. Those are all of them.

And yet the money held in British bank accounts runs to trillions of pounds. The Bank of England's own explanation puts it plainly: about **96% of the money in the UK is bank deposits** and only about **4% is physical cash**.

So if every account holder in the country walked into a branch tomorrow and asked for notes, the notes would not exist. Not because anyone stole them. Not because of any wrongdoing at all. Because notes were never the plan. The overwhelming majority of money in Britain has never had a physical form and never will. It is entries in the records of banks.

So who makes sure you get your money back?

If a bank is a promise, the sensible next question is what happens when a bank cannot keep its promise.

Britain's answer is the **Financial Services Compensation Scheme**, the FSCS. It is a statutory scheme, funded by levies on the industry, and it does something very simple: if an authorised UK bank, building society or credit union fails, the FSCS pays your money back up to a limit, without you having to sue anybody or wait for the wreckage to be sorted out.

Since **1 December 2025** that limit has been **£120,000 per eligible depositor per authorised firm**. Before that date it was £85,000, a figure that had stood since 2017. The Prudential Regulation Authority — the arm of the Bank of England that supervises banks — consulted on raising it to £110,000 in March 2025 and then settled on £120,000 in its final rules, having recalculated against inflation up to September 2025.

The FSCS aims to pay most depositors **within seven days** of the failure, automatically, without a claim form. For certain large one-off sums — the proceeds of selling your home, an inheritance, a redundancy payment, an insurance payout — there is extra cover called a **temporary high balance**, currently up to **£1.4 million for six months** from the date the money reaches you.

If you have more than £120,000, the answer is not to worry about it but to spread it. Two banks, two limits. That is the whole strategy, and it works.

What a bank run actually is

Now put the two halves together, and the mechanics of a bank run fall out on their own.

A bank has promised everyone their money on demand. It cannot possibly have everyone's money available at once, because most of it is out on loan. This is not a scandal; it is the design, and it works perfectly well as long as people withdraw at the ordinary, boring rate.

A run is what happens when they stop. If enough customers try to take their money out at the same time, the bank has to sell things quickly to raise cash. Selling quickly means selling cheaply. Selling cheaply produces losses. Losses make the bank look weaker. Looking weaker frightens more customers, who withdraw, which forces more selling.

The vicious part is that running is rational. If you think there is a serious chance the bank will run out, the sensible thing to do is get in the queue early, because the money goes to whoever asks first.

Everybody knows that everybody knows this. A bank can therefore be destroyed by a belief, and the belief does not have to be true when it starts.

This is precisely why deposit protection exists. If you are certain the state-backed scheme will make you whole within a week, you have no reason to queue. Deposit insurance does not mainly work by paying people after a failure. It works by making the failure less likely in the first place, because it removes the incentive to run.

The exception that proves the rule

There is one form of money that really is a thing you hold: the note in your wallet.

A £20 note is not a record of what anyone owes you that has to be looked up. It is the claim itself, made physical. Hand it over and the payment is complete — instantly, finally, with no institution consulted, no message sent, no possibility of reversal, and no record that you were ever involved. There is nothing to reconcile because nothing was written down.

Everything else in this book is the elaborate machinery required to reproduce, over a network, what a banknote does for free when it changes hands. Seen that way, cash is not the primitive version of digital money. Digital money is a very complicated attempt to catch up with cash on the one dimension where cash is unbeatable: handing it over settles it.

Where the plain version stops being true

That account is good enough to explain at dinner and it is wrong in four specific places. The corrections matter, because each one is a place where confident, articulate people say things that are simply not so.

First: the bank does not lend out your deposit. That story is backwards.

The plain version said the bank keeps some of your £500 and lends the rest. Almost every school textbook says this, and the Bank of England has publicly said it is not how modern banking works. In *Money creation in the modern economy*, published in the Bank's Quarterly Bulletin in 2014, McLeay, Radia and Thomas put it flatly: "Rather than banks receiving deposits when households save and then lending them out, bank lending creates deposits." And: "Whenever a bank makes a loan, it simultaneously creates a matching deposit in the borrower's bank account, thereby creating new money."

Read that carefully, because it inverts the intuition. When a bank grants a £250,000 mortgage, it does not go looking for £250,000 of other people's savings to hand over. It writes two entries: a new asset (the borrower owes it £250,000) and a new liability (the borrower's account now shows £250,000). The deposit did not exist a moment earlier. Commercial banks create the great majority of the money in the economy, in the act of lending.

Which means the accompanying story — that banks are limited by a "reserve ratio", multiplying up a fixed base of central bank money — is also not how the constraint works. The United Kingdom has no mandatory minimum reserve ratio. What used to look like one, the Cash Ratio Deposit scheme under which larger institutions placed non-interest-bearing deposits at the Bank of England, was a funding mechanism for the Bank's own operations, not a lending constraint, and it was replaced by the Bank of England Levy on **1 March 2024**. Banks are constrained by capital requirements, by liquidity requirements, by what they can lend profitably, and by monetary policy acting on the price of money. Not by a multiplier.

Your £500 is still a real liability and the bank still has to be able to pay it. But the sentence "the bank lends out your money" should be retired.

Second: "a row in a database" is itself a simplification, and in a revealing direction.

The chapter's own framing needs correcting. In a well-built banking system, your balance is usually not a stored number that gets edited. It is a *derived* figure, computed from an append-only sequence of postings — credits and debits, each immutable once written. You do not update a balance; you add an entry and the balance follows. This is not a stylistic preference. It is what makes a ledger auditable, and it is why the accounting principle in the next chapter is the single most load-bearing idea in the book.

There is a second, more practical consequence. There is rarely only one balance. Your **ledger balance** reflects postings that have been made. Your **available balance** reflects the ledger balance adjusted for holds — card authorisations that have been approved but not yet cleared, cheques not yet through, funds earmarked for a standing order due tomorrow. These two numbers are routinely different, and the difference is not an error. When a customer says a payment "took the money twice", the overwhelmingly common explanation is that they are comparing an available balance against a ledger balance and seeing an authorisation hold that has not yet dropped off. Anyone building a payments product who models "balance" as a single mutable integer will discover this the expensive way.

Third: cash is not outside the system. It is a claim too — just a bearer one.

The plain version implied that a banknote is the one form of money that is not somebody's promise. Look at the note. It says, in English, *I promise to pay the bearer on demand the sum of twenty pounds*, above the signature of the Bank's Chief Cashier. That is not decorative. A Bank of England note is a liability of the Bank of England, and it appears as one on the Bank's balance sheet, alongside the reserve accounts that commercial banks hold there.

What distinguishes a note is not that it is not a claim. It is that it is a claim which transfers by physical possession, against an issuer nobody expects to fail. Nobody has to be told, nobody has to agree, nothing has to be looked up. That is a different property from "not being a promise", and confusing the two leads people to strange conclusions about what digital cash would have to be.

Scottish and Northern Irish notes make this sharper still. Those are the liabilities of commercial banks, not of the Bank of England, and the issuing banks are required to hold ring-fenced backing assets covering their entire note issue at all times.

Fourth: the run you can photograph is usually not the run that kills the bank.

The image everyone carries is the queue outside the branch. It is a nineteenth-century image and it has become, for the most part, theatre.

Northern Rock is the case in point. The queues that formed in September 2007 are the most reproduced photographs in modern British banking. But by the summer of that year, retail deposits made up only about **23% of Northern Rock's liabilities**; the other roughly 77% was wholesale funding and securitised notes. The Bank for International Settlements published Hyun Song Shin's analysis of exactly this, and his conclusion is worth quoting: "The Northern Rock depositor run, although dramatic on television, was an event in the aftermath of the liquidity crisis at Northern Rock, rather than the event that triggered its liquidity crisis." The real run had already happened in the wholesale markets in August, when short-term lenders declined to roll their funding over.

Silicon Valley Bank in 2023 makes the modern version explicit. There were no queues. There was no branch network to queue at. Depositors withdrew about **$42 billion in a single day, on Thursday 9 March 2023**, by wire transfer, largely because they had heard the same thing at the same time on the same channels. The bank was closed the following morning. A twenty-first-century run is an API call

executed by a lot of people at once, and it is faster than anything the queue-outside-the-branch mental model prepares you for.

So: the run is real, the mechanism in the plain version is right, and the picture is wrong. Runs happen at the speed of the fastest funding channel, and retail depositors are rarely the fastest channel any more.

The technical version

The deposit as a liability

A commercial bank's balance sheet is the cleanest way to see what your balance is.

On the asset side sit the things the bank owns or is owed: loans and mortgages advanced to customers, holdings of government and corporate securities, reserve balances held at the central bank, and physical cash. On the liability side sit the things the bank owes: customer deposits, wholesale borrowing, issued debt securities. The residual — assets minus liabilities — is equity.

Your current account balance is a liability of the bank. Specifically it is an unsecured claim against the bank, generally repayable on demand, ranking in an insolvency according to the statutory creditor hierarchy. It is not property held on trust for you, and it is not segregated.

English law has been unambiguous about this since **Foley v Hill (1848) 2 HLC 28**, decided in the House of Lords, which established that the relationship between banker and customer is one of debtor and creditor rather than trustee and beneficiary. Lord Cottenham LC:

> Money, when paid into a bank, ceases altogether to be the money of the principal; it is then the money of the banker, who is bound to return an equivalent by paying a similar sum to that deposited with him when he is asked for it.

That is the legal statement of everything in this chapter, written 178 years ago and never seriously disturbed since. The practical consequence is that when a bank fails, depositors are creditors in an insolvency, which is precisely why a compensation scheme exists.

The same logic applies one tier up. The Bank of England's own consolidated balance sheet at **31 March 2025** showed, on the sterling liabilities side, **reserves balances of £694,512 million** and **notes in circulation of £91,230 million**, against total liabilities of £861,868 million. A banknote is a liability of the issuer, recorded as such.

What the record physically is

In a core banking system, an account is an entity with attributes — an identifier, a product type, a currency, a status, an owning customer, an interest and fee configuration — and a stream of postings against it.

A posting carries at minimum a value date, a booking date, an amount with an explicit currency, a direction (debit or credit), a reference to the transaction that produced it, and a link to the contra entry required by double entry. Postings are written once. Corrections are made by writing further postings, not by editing history: a mistaken credit is reversed with a compensating debit, and both remain visible. This is the same discipline that makes any well-built audit trail work, and it is the reason a bank statement is a list of movements rather than a series of snapshots.

The balance is a function of the postings. Real systems typically maintain running totals for performance — recomputing from the beginning of time on every enquiry would be absurd — but those

totals are caches, and reconciling them back to the posting stream is a routine control. If the derived total and the sum of postings ever disagree, the postings are right by definition.

Several distinct balances are maintained or derived, and the vocabulary is not perfectly standardised across the industry:

Balance	Roughly means	Typically differs because of
Ledger or book balance	Sum of all postings made to the account	Nothing; it is the accounting truth
Cleared balance	Postings that have completed clearing	Items in a clearing cycle not yet complete
Available balance	What the customer may actually spend	Authorisation holds, uncleared items, agreed overdraft, earmarks
Value-dated balance	Balance as at a future or past value date	Forward-dated instructions and back-value adjustments

Different institutions use these terms with slightly different boundaries, and international groups frequently use them inconsistently between systems. Where a specification matters — a reconciliation file, an API contract — the definition should be stated rather than assumed.

Two further properties of the record are worth naming now because they recur throughout the book. Amounts are held in the currency's minor units as exact integers, never as binary floating point, for reasons taken up in detail in Volume I's chapter on the money type. And every instruction that moves money needs an idempotency discipline, because a network that can deliver a message twice will eventually deliver a payment instruction twice, and the ledger must be able to recognise the second one as a duplicate rather than post it.

Two tiers of money

The phrase "your money" conceals a distinction that governs almost every settlement rule later in this book.

Commercial bank money is a deposit liability of a commercial bank. It is what households and businesses hold and what the number in your app represents. Its value depends on the solvency of that particular bank, which is why deposit protection exists and why the identity of the bank matters.

Central bank money is a liability of the central bank. In the United Kingdom it takes two forms: reserve balances held by eligible institutions in accounts at the Bank of England, and banknotes. Reserves are the settlement asset for the UK's high-value systems: they sit in the Bank's Real-Time Gross Settlement infrastructure, and a CHAPS payment is settled by moving reserves between accounts there. Settlement in central bank money is final in a way that settlement in commercial bank money is not, because the Bank of England does not fail.

That two-tier structure — customers holding claims on banks, banks holding claims on the central bank — is the skeleton of the whole payments system. When you pay someone at another bank, your bank's liability to you falls, their bank's liability to them rises, and the imbalance between the two banks is squared up in central bank money. Chapter 6 takes this apart properly; it is introduced here because "a balance is a liability of a specific institution" is the fact that makes it necessary.

Why the bank can lend, and what actually constrains it

Since bank lending creates deposits, the interesting question is not where the money comes from but what stops banks creating an unlimited amount of it. Four things, in practice.

Capital. Under the Basel III framework as implemented in the UK by the PRA, banks must hold regulatory capital against risk-weighted assets, plus buffers. Every new loan consumes capital. Capital is finite and expensive.

Liquidity. The Liquidity Coverage Ratio requires a bank to hold sufficient high-quality liquid assets to survive a 30-calendar-day stress scenario, with a minimum of 100%. The standard was phased in from 60% on 1 January 2015, rising ten percentage points a year to 100% on 1 January 2019. The LCR is the direct regulatory answer to the run mechanism described above: it is a rule about surviving a month of people wanting their money back.

Profitability and competition. A bank creates a deposit when it lends, but it cannot control where that deposit goes. The borrower spends it, and it very often ends up at another bank, which the lending bank must settle in central bank money. Lending therefore has a funding cost, and imprudent lending has a credit cost.

Monetary policy. Bank Rate sets the price at which reserves are remunerated and thereby the reference price of money throughout the system. The central bank influences the quantity of broad money by influencing its price, not by rationing a reserve base.

Structurally, the largest UK banking groups have also been required since **1 January 2019** to place their retail deposit-taking business inside a legally separate ring-fenced bank, insulated from investment banking activity.

Deposit protection: what the FSCS actually covers

Precision matters here more than anywhere else in the chapter, because this is the topic on which confident misinformation is most common.

The limit. £120,000 per eligible depositor, per authorised firm, for firm failures occurring on or after **1 December 2025**. This was set by the PRA in policy statement **PS24/25** (November 2025), following consultation paper **CP4/25** (March 2025), which had proposed £110,000. The previous limit of £85,000 had been in place since 2017.

Per licence, not per brand and not per account. The limit attaches to the authorised firm — the banking licence — not to the trading name on the card. The FSCS states the position directly: protection is "up to £120,000 per person per banking licence". Two of its own worked examples: HSBC trades as HSBC and as first direct on a single licence, so the £120,000 covers the total across both; Nationwide Building Society's licence also carries the Derbyshire, Cheshire and Dunfermline brands. A depositor holding £120,000 with each of two brands that share a licence is protected for £120,000 in total, not £240,000. The authoritative check is the Financial Services Register maintained by the FCA, not the marketing.

Joint accounts. Each account holder has their own entitlement, so a joint account is generally treated as split equally between the holders for the purposes of the limit.

Temporary high balances. Up to **£1.4 million** for **six months**, on top of the standard limit, where the money arises from a defined qualifying event. The FSCS lists these as including the sale of a main residence, property purchase or equity release, inheritance, insurance payouts, retirement benefits, redundancy, divorce or dissolution of a civil partnership, and compensation for unfair dismissal or

wrongful conviction. Protection runs from the point the money becomes legally transferable to the depositor or is first credited. Compensation for personal injury, disability or incapacity is protected without limit.

Timing. The FSCS works to a seven-day payout for the bulk of depositors, achieved by requiring failed firms to produce a **Single Customer View** file — a standardised extract of depositor identity, contact details, account details and aggregate balances — within 24 hours of the failure event. Note a genuine inconsistency in the published wording: FSCS industry materials refer to compensation "within seven working days of a deposit taker failing", while consumer-facing materials say customers "will typically get their money back within seven days". The operating requirement placed on firms is what matters in practice: the SCV must be ready before anything goes wrong, which is why SCV testing is a standing supervisory expectation and not a crisis activity.

Disclosure lag. Firms were given until **31 May 2026** to update their customer-facing information sheets and exclusion lists to the new limit. Documents published before that date, including some still in circulation, may quote £85,000 while the rule in force is £120,000. If you are checking a bank's own PDF rather than the FSCS site, check its date.

What is not covered. The FSCS deposit protection scheme covers deposits with PRA-authorised deposit takers. It does not cover funds held with electronic money institutions or payment institutions. The FSCS is unambiguous: "We can't protect the money you have with e-money institutions and payment providers."

This is the single most consequential misunderstanding in UK consumer fintech, because e-money accounts look exactly like bank accounts. They have sort codes and account numbers, they receive salaries, they issue cards. What protects the customer's money there is a different regime: **safeguarding** under the Payment Services Regulations 2017 and the Electronic Money Regulations 2011, which requires the firm either to segregate relevant funds in a designated safeguarding account with a credit institution or to cover them with an insurance policy or comparable guarantee. Safeguarding is a real protection and it is not nothing. But it is a claim on segregated assets resolved through an insolvency process, not a seven-day payout from a statutory scheme, and the FSCS warns that money held this way "could be tied up for a while during the insolvency process".

There is one important bridge between the two regimes. Following PRA policy statement **PS2/23**, with rule amendments in force from **12 March 2023**, the FSCS can "look through" a safeguarding account and compensate the underlying customers of an e-money or payment institution if **the bank holding the safeguarded funds** fails. This does not extend protection to the case where the e-money institution itself fails and its bank does not. The distinction is narrow and it is exactly the sort of thing that gets flattened in marketing copy.

Covered deposits and bail-in. Deposits protected by the FSCS are legally excluded from the bail-in tool under the UK's special resolution regime. When the Bank of England writes down or converts a failing bank's liabilities, protected deposits are not in scope. This is the resolution-side counterpart of deposit insurance and it is worth knowing, because it means the answer to "could they take my savings to save the bank?" is no, within the protected limit.

The mechanics of a run, stated properly

The formal treatment is Diamond and Dybvig, "Bank Runs, Deposit Insurance, and Liquidity", *Journal of Political Economy* volume 91, number 3 (1983), pages 401–419 — work for which Douglas Diamond and Philip Dybvig shared the 2022 Sveriges Riksbank Prize in Economic Sciences with Ben Bernanke.

The model isolates three features that together generate the run.

Maturity transformation. The bank issues liabilities that are short and liquid — demand deposits — against assets that are long and illiquid — mortgages, loans, held-to-maturity securities. That transformation is the economic service the bank provides, and it is inherently fragile.

The sequential service constraint. Depositors are served in the order they arrive, and the bank pays in full until it cannot pay at all. Withdrawing early is therefore strictly better than withdrawing late if you believe others will withdraw.

Multiple equilibria. Given the first two, the model has more than one self-consistent outcome. If nobody expects a run, nobody runs and the bank is fine. If everybody expects a run, running is individually rational and the bank fails — even if its assets, held to maturity, would have been worth more than its liabilities. Deposit insurance works by eliminating the bad equilibrium: if you will be paid regardless, you have no reason to be first.

The critical distinction the model forces is between **illiquidity** and **insolvency**. A bank is insolvent when its assets are worth less than its liabilities. A bank is illiquid when it cannot convert good assets into cash fast enough to meet demands. A run turns the second into the first, because forced sales into a falling market crystallise losses that a patient holder would never have taken.

The three cases already cited illustrate the range.

Case	What happened	Figures
Northern Rock, 2007	Wholesale funding withdrawn first; retail run followed the announcement of support	BoE liquidity support announced 14 September 2007; retail deposits about 23% of liabilities by summer 2007; wholesale liabilities fell from £26.7bn in June 2007 to £11.5bn in December 2007; HM Treasury guarantee announced 17 September, extended 20 September and again 9 October 2007
Silicon Valley Bank, 2023	Digital run at unprecedented speed; uninsured deposit base	About $175bn total deposits at 31 December 2022, of which roughly $151bn (about 95%) uninsured; about $42bn withdrawn on 9 March 2023; closed 10 March 2023
Silicon Valley Bank UK, 2023	Resolved rather than liquidated	About £8.8bn total assets and £6.7bn deposits; sold to HSBC on 13 March 2023 using the Bank of England's private sector purchaser stabilisation power under the Banking Act 2009, with all depositors' money confirmed safe

The SVB UK case is the one practitioners should study, because it shows the modern default. The preferred outcome is not a payout at all. It is a resolution — a transfer of the failing bank's business to a purchaser over a weekend, so that on Monday morning the deposits still exist, still work, and simply have a different owner. Compensation schemes are the backstop for when that cannot be arranged, not the first line.

Note also the SVB deposit composition: about 95% uninsured. Deposit insurance calms retail depositors. A bank funded overwhelmingly by corporate treasurers holding tens of millions each has almost no insured base to calm, which is why concentration and insured share are supervisory concerns and not just balance-sheet trivia.

Cash, precisely

Physical currency is the exception, and being precise about it clarifies the rule.

Bank of England notes in circulation, from the Bank's own banknote statistics measured at the end of February each year:

Denomination	Value, £ millions (2026)	Volume, millions of notes (2026)
£5	1,993	399
£10	12,948	1,295
£20	58,633	2,932
£50	17,932	357
Total	91,505	4,984

A separate line in the same statistics, £4,704 million of "other notes", covers higher-denomination notes held as backing for the note issues of banks in Scotland and Northern Ireland — the internal high-value notes long known in the trade as giants and titans.

Legal tender is where public understanding is worst, and the Bank of England's own explainer is blunt that the term "has a narrow technical meaning that will rarely come up in everyday life". It concerns the discharge of a debt in court proceedings. In England and Wales, Royal Mint coins and Bank of England notes are legal tender. In Scotland and Northern Ireland, only Royal Mint coins are — banknotes issued by Scottish and Northern Irish banks are not legal tender anywhere, including Scotland and Northern Ireland, and circulate on acceptance rather than statute. Even the coins carry limits: 1p and 2p coins are legal tender only up to 20p, and 5p and 10p coins up to £5. No retailer is obliged to accept any particular means of payment, and debit cards, credit cards, cheques and contactless payments are not legal tender anywhere in the UK.

The Scottish and Northern Irish issuers operate under the Scottish and Northern Ireland Banknote Regulations 2009, supervised by the Bank of England. All such notes must be fully backed by ring-fenced assets at all times, with a minimum of 60% of those backing assets held in Bank of England notes or UK coin. A Scottish banknote is therefore a commercial bank liability with a statutory asset backing behind it — a genuinely different instrument from a Bank of England note, despite looking like the same kind of object in the same wallet.

What makes cash distinctive as a payment instrument is a short list of properties that the rest of this book spends thousands of words trying to reconstruct electronically. It is a bearer instrument, so possession is title. It settles finally at the moment of handover, with no clearing cycle and no settlement lag. It carries no counterparty risk beyond the issuer. It works with no connectivity, no power and no third party. It is irreversible: there is no chargeback on a twenty-pound note. And it leaves no record, which is simultaneously its greatest privacy virtue and the reason it attracts the attention of anti-money-laundering regimes.

Counterfeiting, incidentally, is not the systemic problem folklore suggests. In 2025 the Bank of England received about 200,000 counterfeit notes with a notional face value of £3.97 million, which it reports as fewer than one in 24,390 notes in circulation.

What this means if you are building something

Four practical consequences follow from this chapter, and they show up in real systems constantly.

Model balances as derived from an immutable posting stream, not as a mutable field. Any design in which a balance is updated in place will eventually produce a number that cannot be explained, and an unexplainable balance in a financial system is a regulatory problem as well as an engineering one.

Be explicit about which balance you mean, everywhere — in APIs, in reports, in support scripts, in the screen label the customer reads. "Balance" alone is not a specification.

Know exactly what protects your customers' money and never let the language drift. If you are an e-money institution, your customers' funds are safeguarded, not FSCS-protected, and saying otherwise in a marketing email is a regulatory failure and not a wording preference. If you are a bank, know your licence, know which brands share it, and know that the limit is £120,000 per depositor per licence.

And carry the central idea into everything that follows. A payment is never the transportation of a substance. It is the coordinated amendment of at least two records, held by at least two institutions, each of which must end up agreeing about what happened. The next chapter is about the discipline that makes that agreement possible, and it is seven hundred years old.

2.98 Common wrong ideas

1. **Wrong:** the bank keeps some of your deposit and lends the rest of it out. **Right:** bank lending creates deposits, because a new loan writes a new asset and a matching new liability, and the deposit did not exist a moment earlier.
2. **Wrong:** banks are limited by a reserve ratio that multiplies up a fixed base of central bank money. **Right:** the United Kingdom has no mandatory minimum reserve ratio, and the real constraints are capital, liquidity, profitability and monetary policy acting on the price of money.
3. **Wrong:** a shop has to take cash because cash is legal tender. **Right:** legal tender has a narrow technical meaning about discharging a debt in court proceedings, and no retailer is obliged to accept any particular means of payment.
4. **Wrong:** a balance is a number stored in a field and edited when money moves. **Right:** in a well-built system it is derived from an append-only sequence of immutable postings, which is what makes a ledger auditable and a statement a list of movements.
5. **Wrong:** the app showed two different figures, so the payment was taken twice. **Right:** the overwhelmingly common explanation is an authorisation hold that has not yet dropped off, seen by comparing an available balance against a ledger balance.
6. **Wrong:** a banknote is the one form of money that is nobody's promise. **Right:** a Bank of England note is a liability of the Bank of England and is recorded as one; what distinguishes it is that the claim transfers by physical possession against an issuer nobody expects to fail.
7. **Wrong:** FSCS cover is per account, or per brand, so two accounts mean two limits. **Right:** it is £120,000 per eligible depositor per authorised firm, so brands sharing a licence — HSBC and first direct, for instance — share a single limit.
8. **Wrong:** money in an e-money or payment institution account is FSCS-protected, because it has a sort code, receives salaries and issues cards. **Right:** those funds are safeguarded under the Payment Services Regulations 2017 and the Electronic Money Regulations 2011, which is a claim on segregated assets resolved through insolvency rather than a seven-day statutory payout.
9. **Wrong:** a bank run looks like a queue outside a branch. **Right:** runs happen at the speed of the fastest funding channel, which at Northern Rock was the wholesale market in August 2007 and at Silicon Valley Bank was about $42 billion of wire transfers on 9 March 2023.

10. **Wrong:** when a bank fails, depositors get compensated. **Right:** the preferred outcome is resolution — Silicon Valley Bank UK was sold to HSBC on 13 March 2023 — and a compensation payout is the backstop for when a transfer cannot be arranged.

2.99 Chapter summary in 20 lines

1. The number at the top of your banking app is not a description of where anything is; it is a description of a promise.
2. Handing £500 to a neighbour who seals it in a labelled envelope is safekeeping, and handing it to one who spends it and repays you later is deposit banking.
3. Every bank in the world is the second arrangement, and English law has said so since *Foley v Hill* in 1848.
4. Notes paid into a branch become the bank's notes, and the only thing written beside your name is an obligation payable on demand.
5. Bank deposits are about 96 to 97 per cent of the money in the United Kingdom, against roughly £91.5 billion of Bank of England notes in existence anywhere in the world.
6. If every account holder asked for notes tomorrow the notes would not exist, not because of any wrongdoing but because notes were never the plan.
7. The textbook claim that a bank lends out your deposit is backwards, because the Bank of England's own bulletin states that bank lending creates deposits.
8. Nothing multiplies up a fixed reserve base, and the binding constraints are capital, liquidity, funding cost and Bank Rate.
9. A balance is best modelled as a figure derived from an immutable posting stream rather than as a mutable field.
10. There is rarely only one balance: ledger, cleared, available and value-dated balances differ routinely, and the differences are not errors.
11. A banknote is a claim too, and what makes it special is that it transfers by possession rather than that it is not a promise.
12. Scottish and Northern Irish notes are commercial bank liabilities with statutory ring-fenced backing, and are not legal tender anywhere.
13. Legal tender is a narrow point about discharging debts in court and obliges nobody to accept anything in a shop.
14. A run follows from three features: liabilities that are short against assets that are long, service in order of arrival, and the rationality of going early.
15. Diamond and Dybvig showed that this produces more than one self-consistent outcome, and deposit insurance works by eliminating the bad one.
16. Northern Rock's real run happened in the wholesale market in August 2007, and the photographed queues came after the announcement of support.
17. Silicon Valley Bank lost about $42 billion by wire in a day, with roughly 95 per cent of its deposits uninsured, which is what a run looks like when the branch is an API.
18. The FSCS covers £120,000 per eligible depositor per authorised firm from 1 December 2025, per licence rather than per brand, with temporary high balance cover of up to £1.4 million for six months.
19. E-money and payment institutions are outside the deposit scheme entirely, and their customers rely on safeguarding, which is a real protection but a slower and different one.
20. A payment is never the transportation of a substance; it is the coordinated amendment of at least two records held by at least two institutions that must end up agreeing.

Chapter sources: Bank of England — Quarterly Bulletin 2014 Q1, McLeay, Radia and Thomas, "Money creation

in the modern economy"; "What is money" and "What is legal tender" explainers; banknote statistics (page updated 23 April 2026); consolidated balance sheet at 31 March 2025; Bank of England Levy pages; statement on Silicon Valley Bank, 13 March 2023; Scottish and Northern Ireland banknote regime documentation; Freedom of Information response on bail-in and customer deposits. Prudential Regulation Authority — CP4/25 and PS24/25, Depositor protection; PS2/23. Financial Services Compensation Scheme — deposit limit increase, banking licences, temporary high balances, Single Customer View, and e-money protection pages. Financial Conduct Authority — safeguarding requirements for payment and e-money institutions. Bank for International Settlements — Hyun Song Shin, "Reflections on Northern Rock". Basel Committee on Banking Supervision — Liquidity Coverage Ratio executive summary. Foley v Hill (1848) 2 HLC 28. Diamond and Dybvig, Journal of Political Economy 91(3), 1983. US House Financial Services Committee background materials on the failure of Silicon Valley Bank. UK Parliament research briefing SN04478 on Northern Rock.

3

CHAPTER

Double Entry

3.0 What this chapter gives you

1. You will be able to state the rule of double entry in one sentence — never write down one end of a journey — and explain why a record-keeping method that checks itself survived seven hundred years without replacement.
2. You will be able to define debit and credit correctly as left and right, and explain why the same £3.20 is a debit in Lloyds' ledger and a credit in the cardholder's.
3. You will be able to post a VAT-inclusive £3.20 coffee sale as a three-legged entry, and say why the shop's cash rose by £3.20 while its equity rose by only £2.26.
4. You will be able to explain why "double" refers to the two sides rather than to a count of lines, and why a model hard-coded to one debit and one credit cannot express a partial refund.
5. You will be able to name the errors a trial balance cannot catch — omission, commission, principle, original entry, reversal and compensation — and explain why fraud generally balances beautifully.
6. You will be able to trace one card payment through the issuer's, acquirer's, merchant's and cardholder's books, and show the same interchange appearing as income in one and cost in another.
7. You will be able to explain why a card-accepting merchant books a receivable rather than cash, and why that gap has to be funded.
8. You will be able to explain why an authorisation posts nothing at all, why "pending" exists on the app, and why a refund is a fresh transaction rather than a cancellation.
9. You will be able to describe what happens at the apex of the system, where an interbank payment merely relabels which reserves account the Bank of England owes.
10. You will be able to list the invariants a ledger implementation must enforce: balance at write time, immutable append-only postings, signed integers in minor units, idempotency keys, multi-leg entries and named accounts for in-flight money.

There is a sentence that people say without noticing what it commits them to. They say: *I spent three pounds twenty on a coffee.*

Notice how many facts that sentence contains. It sounds like one. It is at least two, and if you are the coffee shop it is four or five. Three pounds twenty left something. Three pounds twenty arrived somewhere. The shop is three pounds twenty richer in one respect and slightly poorer in another, because a cup of coffee that cost it forty-one pence to make no longer exists. Some of that three pounds twenty was never the shop's money at all; fifty-three pence of it belongs to His Majesty's Revenue and Customs and the shop is merely holding it. None of this is philosophy. It is all recorded, in books, by people whose job it is, and the discipline they use to record it is seven hundred years old and has not needed replacing.

The discipline is called double entry, and it is the single most load-bearing idea in this book.

Everything that follows — clearing, settlement, netting, interchange, chargebacks, the reason a refund takes five days, the reason your available balance is not your ledger balance, the reason a bank can fail on a Friday and reopen on a Monday with every customer's number intact — is double entry operating at a scale and speed that would have astonished the Venetians who invented it. Not an analogy for double entry. Not something like double entry. It is double entry, running on faster machinery.

The previous chapter established that your balance is not a substance but a record. This chapter is about the rule that governs how such records may legally and sensibly be changed. Get this chapter and the rest of the book becomes a set of variations. Skip it and every later chapter will feel like a list of arbitrary industry customs.

The plain version

Every fact about money is a journey, and a journey has two ends.

If your friend hands you five pounds, that is not one thing that happened. It is two things that happened at the same instant: you have five pounds more, and your friend has five pounds less. If you write down only your half, your notebook is claiming that five pounds appeared out of nowhere. Notebooks that claim money appears out of nowhere are, sooner or later, wrong about everything.

So the rule is this: **never write down one end of a journey. Always write down both.**

That is the whole idea. Everything else is bookkeeping for how to do it tidily.

The two columns

Draw a line down the middle of a page. On the left you write where the money went *to*. On the right you write where it came *from*.

Your friend hands you five pounds:

Left: where it went to	Right: where it came from
My pocket, £5	My friend, £5

Two ends. Same amount on both sides. If the two sides do not match, you have made a mistake, and you know it immediately, without anybody having to tell you.

That last part is the reason the idea survived. It is not merely a tidy way to keep records. It is a record-keeping method that **checks itself**. If you add up everything in the left column and everything in the right column, the two totals must be identical. If they are not, something is wrong and you go and find it. A shopkeeper in 1350 with no calculator, no computer and no auditor could catch his own errors before they compounded. That is an enormous thing to be able to do.

Aisha's cake stall

Aisha is thirteen. She is running a cake stall at the school summer fair and she has decided to keep proper books, because her uncle is an accountant and has been unbearable on the subject.

She starts by putting in £20 of her own money, so there is a float. Then she spends £12 on flour, eggs, butter and icing sugar. She spends £3 on greaseproof paper and little labels. The school charges £5 to hire a table. On the day, she sells £34 of cakes.

Here is her book. Every row has a left and a right, and every row balances by itself.

What happened	Left (where it went to)	Right (where it came from)
Aisha puts in her own £20	Cash tin, £20	Owed back to Aisha, £20
Buys ingredients	Ingredients, £12	Cash tin, £12
Buys paper and labels	Packaging, £3	Cash tin, £3
Pays for the table	Table hire, £5	Cash tin, £5
Sells cakes	Cash tin, £34	Sales, £34

At the end of the fair, how much is in the tin? Add up everything that went into it and take away everything that came out: £20 in, then £12 out, £3 out, £5 out, then £34 in. That is £34.

Now check it from the other direction, which is the point of the whole exercise. The stall took £34 of sales and spent £20 getting there (£12 plus £3 plus £5), so it made £14 of profit. It also still owes Aisha the £20 float she put in. So the total claim on the stall is £20 plus £14, which is £34 — exactly the cash in the tin.

Two completely separate routes to the same number. That is not a coincidence and it is not luck. It is a consequence of never writing down half a journey. Aisha now knows, without checking anything else, that her book is internally consistent. If she had counted £31 in the tin, she would know instantly that either £3 had gone missing or she had forgotten to write something down, and she would go looking. Before double entry, she would simply have had £31 and a vague feeling.

The two words that confuse everybody

Accountants do not say "left" and "right". They say **debit** and **credit**. Debit means the left-hand column. Credit means the right-hand column. That is all they mean.

Read that again, because almost everyone in Britain believes something else. Almost everyone believes that "debit" means money going out and "credit" means money coming in. That belief comes from bank statements, and bank statements are not lying to you, but they are written from a point of view you have probably never considered.

Look at Aisha's book. When she sells £34 of cakes and the money goes *into* the tin, that is a **debit** to the cash tin. Money coming in is a debit. When she pays £5 for the table and money goes *out* of the tin, that is a **credit** to the cash tin. Money going out is a credit.

That is the exact opposite of what your bank statement says, and both are correct. We will resolve this properly in a moment. For now, hold on to the only definition that never changes: **debit is the left side, credit is the right side, and every entry needs both.**

The coffee, over the counter

Ama buys a flat white from Kalypso Coffee for £3.20 and pays with a £5 note.

In the shop's book, that single sale is two rows, not one, because two separate journeys took place.

The first journey is the money. Three pounds twenty arrived in the till, and it arrived from a customer buying something. Left: cash, £3.20. Right: sales, £3.20.

The second journey is the coffee. A cup of coffee left the shop's stock and went into a customer's hand. The beans, milk, cup, lid and sleeve cost the shop 41 pence. Left: cost of what was sold, £0.41. Right: stock, £0.41.

Both rows balance. Together they say something a single row could never say: the shop is £3.20 richer in cash, £0.41 poorer in stock, and therefore £2.79 better off overall. A shop that only recorded "sales £3.20" would think it was doing brilliantly right up until the moment it noticed it had no beans.

The same coffee, paid by card

Now Ama taps her debit card instead.

No money moves. Nothing physical happens at all. What happens is that four different organisations each write down two things.

Ama's bank writes down that it owes Ama £3.20 less than it did a moment ago, and that it now owes £3.20 to somebody else in the payments chain. Two facts. The shop's card company writes down that it is owed £3.20 by that chain, and that it owes £3.20 to the shop. Two facts. The shop writes down that it has made a £3.20 sale, and that it is owed £3.20 by its card company. Two facts. And if Ama keeps a household budget, she writes down that she spent £3.20 on coffee and that her bank balance is £3.20 lower. Two facts.

Nobody moved anything. Everybody wrote two things down. That is what a payment is.

And here is the part that makes the whole system hang together: those separate books have to agree with each other. Ama's bank thinks it now owes £3.20 into the system. Somebody else in the system thinks they are owed £3.20. Those two numbers must match, and at the end of the day somebody checks that they do. When they do not match, that is not a rounding annoyance. That is how fraud is caught, how outages are detected, and how a payments company discovers it has lost real money. There is a whole chapter about it in Volume V and it is titled, without affection, *Reconciliation*.

Where this came from

Nobody invented double entry in a single stroke. It grew up among Italian merchants in the thirteenth and fourteenth centuries, in exactly the places you would expect: Florence, Genoa, Venice, and the trading colonies those cities planted around the Mediterranean. The oldest surviving books that unambiguously use the full method belong to a Florentine firm, Giovanni Farolfi & Company, and were written by a partner named Amatino Manucci at the firm's branch in Salon, in Provence, covering 1299 and 1300. They were identified and analysed by the accounting historian Geoffrey Lee in a 1977 paper in the *Accounting Historians Journal*. Manucci did not invent the technique. He was simply using it, competently, the way a clerk uses a spreadsheet.

The method spread by apprenticeship for two hundred years before anyone thought to write it down properly for strangers. The man who did was **Luca Pacioli**, a Franciscan friar and mathematician born around 1447 in Sansepolcro in Tuscany, who died on 19 June 1517. In November 1494 the Venetian printer Paganino de Paganini published Pacioli's enormous *Summa de arithmetica, geometria, proportioni et proportionalità*, a general survey of everything then known about arithmetic, algebra and geometry, running to over six hundred pages of dense type.

Buried inside it, as the eleventh treatise of the ninth part, is a section called *Particularis de computis et scripturis* — "particulars of reckonings and writings". In it Pacioli sets out what he calls the method of Venice: keep three books, a rough memorandum for jotting things down as they happen, a journal for writing them up in order, and a ledger where each account has its own page with debits on the left and credits on the right. Check the ledger periodically by listing every balance and proving that the two columns agree. He is widely quoted as advising that a merchant should not go to sleep at night until his debits equal his credits.

Pacioli did not invent double entry and never claimed to. What he did was put it on a printing press, at the moment printing presses existed, in the commercial capital of Europe. Three hundred years later Goethe had a character in *Wilhelm Meister's Apprenticeship* say of book-keeping by double entry that "it is among the finest inventions of the human mind". The method has been continuously in use ever since, which makes it older than the Bank of England, older than the pound sterling in its modern form, and older than every institution named anywhere in this book.

Where the plain version stops being true

The plain version above is a good foundation and it is not a lie. But four things in it will mislead you if you carry them forward unamended, and the fourth one is where fintechs lose money.

Correction one: "debit" does not mean money out

This is the most common misunderstanding in personal finance and it is entirely the fault of the banking industry, which chose to print the words on statements without explaining whose books they refer to.

Your bank statement says "debit" next to the £3.20 coffee. Aisha's cake stall book says a debit is money *arriving*. Both are correct because they are two different books written from two different points of view.

Recall the previous chapter: your deposit is not your money sitting in the bank's vault. It is the bank's **promise to pay you**. In the bank's own ledger, your current account is a *liability* — a debt the bank owes you. When you spend £3.20, the bank's debt to you gets smaller. Reducing a liability is, by the universal convention, a debit. So the bank debits your account, and prints the word on your statement.

Meanwhile, in your own books, your bank account is an *asset* — something you own. Spending reduces an asset. Reducing an asset is a credit. So in your books, the very same event is a credit.

The same £3.20, at the same instant, is a debit in Lloyds' ledger and a credit in yours. Neither is wrong. They are mirror images, and they must be mirror images, because the thing that grew smaller for one party is the thing that grew smaller for the other in the opposite direction.

The practical consequence is that **"debit" and "credit" are meaningless without two further pieces of information: whose book, and what kind of account.** Any sentence of the form "we'll credit your account" is ambiguous until you know who "we" are. When a merchant says "we will credit your card", they mean your card balance will move in your favour. When a bank's core system says "credit the customer account", it means increase the bank's liability, which is the same thing. When a payments engineer says "credit the merchant's payable", they mean the acquirer now owes the merchant more. Same word, three books, three referents. Systems that blur this produce reversed postings, which are the single most tedious class of bug in financial software.

Correction two: "double" does not mean two lines

The name is a historical accident and it misleads people who take it literally. "Double" refers to the two *sides*, not to a count of lines. A single journal entry may have three legs, or twelve, or two hundred. The rule is not "one debit and one credit". The rule is:

The sum of the debits equals the sum of the credits.

Return to the flat white. The shop is registered for VAT, and hot drinks in the United Kingdom are standard-rated at 20 per cent. Of the £3.20 the customer paid, £2.67 is the shop's revenue and 53 pence

is output tax that the shop is collecting on behalf of HMRC and will hand over later. That is one economic event and it needs three legs:

Account	Debit	Credit
Cash	£3.20	
Sales		£2.67
VAT payable		£0.53

Debits £3.20, credits £3.20. Balanced, with three lines.

Correspondingly, one real-world event may require several *separate* journal entries. The coffee sale generates one entry for the money and a second for the movement of stock into cost of sales. The card payment in the technical section below generates one entry when the sale is recorded, another when the acquirer settles, and a third when the monthly fee invoice is raised. Do not expect a one-to-one relationship between things that happen in the world and entries in a ledger. There isn't one, and building software on the assumption that there is will produce a system that cannot represent a partial refund.

Correction three: balancing proves consistency, not truth

This is the correction that matters most and that is stated least often.

A trial balance that agrees tells you exactly one thing: that for every debit posted, an equal credit was posted. It tells you nothing whatsoever about whether the entries were *correct*.

Post a £4,000 laptop purchase to "office refreshments" instead of "computer equipment", and the trial balance agrees. Post an entire month's rent to nowhere at all — leave it out of the books completely — and the trial balance agrees. Post a sale as £810 when the invoice said £108, on both sides, and the trial balance agrees. Swap the debit and the credit so a payment out looks like a payment in, and the trial balance agrees. Make two independent mistakes of £20 that happen to fall on opposite sides, and the trial balance agrees.

Every one of those has a name in British accountancy training, and the whole class of them — errors of principle, omission, commission, original entry, reversal and compensation — is invisible to the balancing check. The technical section tabulates them.

This has a direct implication for anyone building a payment system. A balanced ledger is a *necessary* condition of correctness and nowhere near a sufficient one. Balance is cheap: any halfway competent implementation enforces it at write time and it will never fail again. The expensive question is whether the entries describe what actually happened, and that question is answered not by the ledger but by comparing the ledger against an independent record produced by somebody else. That comparison is reconciliation, and it is where the real work is.

Fraud, it should be said plainly, generally balances beautifully. A dishonest employee does not create unbalanced entries. They create balanced entries to the wrong account.

Correction four: there is no world ledger, and the two halves are not simultaneous

The plain version quietly implies a single book in which both ends of a journey get written. For a cake stall, that is true. For payments, it is not true and never has been.

There is no global ledger of money. There are millions of separate ledgers — one per bank, one per payment institution, one per scheme, one per merchant, one at the central bank — each of which

balances *within itself* and each of which is entirely capable of disagreeing with its neighbours. When Ama's bank debits her account and credits an amount payable into the card scheme, that is complete and balanced double entry *inside Ama's bank*. It says nothing about whether the acquirer at the other end has recorded the matching amount receivable, or recorded it for the right value, or recorded it at all.

The two halves are also frequently separated in time, sometimes by days. Ama's card was approved in about a second at 07:42 on a Tuesday. No posting was made at 07:42. Her bank placed a *hold*, which reduced her available balance but did not touch her ledger balance, and the actual double entry was made when the transaction cleared — typically the next business day for a UK card payment. That gap is not a defect. It is a designed consequence of authorisation and clearing being separate events, and it is the reason "pending" exists on your app.

Between the two halves, the money lives in accounts that exist purely to hold incomplete journeys: suspense accounts, clearing accounts, in-transit accounts. A large payment institution's suspense accounts are where its operational health can be read most honestly, because anything sitting in suspense is, by definition, a journey with only one end written down so far. A suspense balance that is growing rather than clearing is an incident in progress.

The technical version

The accounting equation

Everything in double entry descends from one identity. In the form used in the United Kingdom and internationally:

Assets = Liabilities + Equity

An entity's resources must equal the claims on those resources. The claims come from two sources: outsiders (liabilities) and owners (equity). This is not an empirical finding about businesses. It is true by construction, because equity is *defined* as the residual.

The IASB's *Conceptual Framework for Financial Reporting*, revised in March 2018, gives the definitions that UK and international reporting rests on. An asset is "a present economic resource controlled by the entity as a result of past events". A liability is "a present obligation of the entity to transfer an economic resource as a result of past events". Equity is "the residual interest in the assets of the entity after deducting all its liabilities". Income is "increases in assets, or decreases in liabilities, that result in increases in equity, other than those relating to contributions from holders of equity claims", and expenses are the mirror.

Because income and expenses are themselves changes in equity, the equation expands to cover trading:

Assets + Expenses = Liabilities + Equity + Income

Left-hand-side accounts have natural debit balances. Right-hand-side accounts have natural credit balances. Every rule you will ever be asked to memorise about debits and credits is a restatement of which side of that equation an account sits on.

Debits and credits, defined properly

A debit is an entry on the left of an account. A credit is an entry on the right. The words carry no directional meaning about cash. Their effect depends entirely on the account type:

Account type	Natural balance	A debit	A credit
Asset	Debit	Increases	Decreases
Expense	Debit	Increases	Decreases
Liability	Credit	Decreases	Increases
Equity / capital	Credit	Decreases	Increases
Income	Credit	Decreases	Increases

British bookkeeping students learn this as **DEAD CLIC**: Debits increase Expenses, Assets and Drawings; Credits increase Liabilities, Income and Capital.

The etymology is worth knowing because it explains the counter-intuitive part. "Debit" comes through Old French from Latin *debitum*, the past participle of *debere*, "to owe" — recorded in English from the mid-fifteenth century, and in its bookkeeping sense of an entry in an account by 1776. "Credit" arrives in the 1540s from Italian *credito*, from Latin *creditum*, "a loan, a thing entrusted", from *credere*, "to trust". The Italian merchant's ledger was originally a list of people: those who owed him (*debitori*, on the left) and those he owed or who had entrusted him (*creditori*, on the right). Every other account type was later fitted into that same two-column frame. That is why the words feel wrong when applied to a cash account: they were never about cash in the first place.

Books of prime entry, ledgers and posting

The structure Pacioli described in 1494 is, with different names and vastly better tooling, the structure in use today.

A transaction is first captured in a **book of prime entry** — the day book, the sales day book, the cash book, or in a modern system the transaction log of the till, the gateway or the API. Nothing is classified yet; it is a chronological record that something happened.

It is then written as a **journal entry**: a dated, balanced set of debits and credits with a narrative and a reference. The journal entry is the atomic unit of accounting. It either exists in full or not at all.

The journal entry is **posted** to the **general ledger**, which is organised by account rather than by date. The list of available accounts is the **chart of accounts**, and its design is one of the more consequential decisions in a finance system: too coarse and you cannot answer questions, too fine and nobody classifies consistently.

High-volume accounts are broken out into **subsidiary ledgers** — one account per customer, per merchant, per cardholder — summarised into a single **control account** in the general ledger. The subledger must sum to the control account. That check, run daily, is the first line of defence in every core banking system on earth, and its failure is a severity-one incident.

Finally, the **trial balance** lists every ledger account balance in two columns and proves the totals agree.

The trial balance and what it does not catch

Error	Description	Trial balance detects?
Single-sided entry	Only the debit or only the credit posted	Yes
Same-sided entry	Two debits or two credits posted	Yes
Unequal amounts	Debit and credit posted at different figures	Yes

Error	Description	Trial balance detects?
Casting error	An account balance added up wrongly	Yes
Extraction error	A balance transcribed wrongly into the trial balance	Yes
Omitted balance	An account left off the trial balance	Yes
Error of omission	Transaction never entered at all	No
Error of commission	Right class of account, wrong account within it	No
Error of principle	Wrong class of account entirely, e.g. capital expenditure posted as revenue expenditure	No
Error of original entry	Both sides posted at the same incorrect figure	No
Error of reversal	Debit and credit entered the wrong way round	No
Compensating error	Two independent errors that cancel out	No

Where a difference is found and cannot be resolved immediately, the difference is parked in a **suspense account** so that reporting can proceed, and cleared as the errors are identified. A suspense account is an admission written into the books; a suspense balance that persists across period ends is a finding, not a feature.

Worked example one: £3.20 over the counter

Kalypso Coffee, VAT-registered, sells a flat white for £3.20 including VAT at the UK standard rate of 20 per cent, paid in cash. Ingredient and packaging cost is 41 pence.

The VAT-exclusive value is £3.20 ÷ 1.20 = £2.6667, rounded to £2.67, leaving output tax of £0.53.

Journal 1 — the sale:

Account	Dr	Cr
Cash in hand (asset)	£3.20	
Sales (income)		£2.67
VAT payable (liability)		£0.53

Journal 2 — the cost of what was sold:

Account	Dr	Cr
Cost of sales (expense)	£0.41	
Inventory (asset)		£0.41

Gross margin on the cup is £2.26. Note that the shop's cash rose by £3.20 but its equity rose by only £2.26, because 53 pence was never its money and 41 pence of stock was consumed. A single-entry record showing "+£3.20" would overstate the shop's performance by 42 per cent.

Worked example two: the same £3.20 across two banks

Ama taps a contactless consumer debit card issued by Lloyds Bank. Kalypso Coffee's acquirer is Worldpay; the merchant's bank account is with NatWest; the card scheme is Visa. Ama's bank and the acquirer's settlement bank both hold accounts at the Bank of England.

For clarity the fee figures below are illustrative of a small UK merchant's contract rather than quoted from a published rate card. The one fee that is not illustrative is the interchange cap: under the Interchange Fee Regulation as applied in the United Kingdom, interchange on a consumer debit card transaction is capped at **0.2 per cent** of transaction value, and on a consumer credit card at **0.3 per cent**, for transactions where the merchant, the acquirer and the issuer are all in the UK. Those caps have applied since 9 December 2015 and remain in force as of August 2026. Suppose Worldpay charges Kalypso a blended merchant service charge of 1.5 per cent.

Interchange at 0.2 per cent of £3.20 is £0.0064 — sixty-four hundredths of a penny. Merchant service charge at 1.5 per cent is £0.048. Both are computed to sub-penny precision and rounded once, at the level of the daily batch, not per transaction. This is not fussiness; it is the reason Volume I closes with a chapter insisting that money must never be represented in floating point.

Issuer (Lloyds), on clearing:

Account	Dr	Cr
Ama — current account (liability to customer)	£3.2000	
Interchange income		£0.0064
Scheme settlement payable		£3.1936

Lloyds' liability to Ama falls by £3.20 — a debit, because reducing a liability is a debit, and this is the entry that prints the word "debit" on her statement. Lloyds retains interchange and owes the remainder into the scheme.

Acquirer (Worldpay), on clearing:

Account	Dr	Cr
Scheme settlement receivable	£3.1936	
Interchange expense	£0.0064	
Merchant payable — Kalypso Coffee		£3.2000

and then, for its own charge:

Account	Dr	Cr
Merchant payable — Kalypso Coffee	£0.0480	
Merchant service charge income		£0.0480

leaving £3.152 owed to the merchant. Note that interchange is *revenue* to the issuer and *cost* to the acquirer: the same 0.64 pence appears as a credit in one institution's books and a debit in another's. That is the four-party model in miniature, and Volume III takes it apart properly.

Merchant (Kalypso Coffee), at the till:

Account	Dr	Cr
Card receivable — Worldpay (asset)	£3.20	
Sales (income)		£2.67
VAT payable (liability)		£0.53

The shop has made a sale but holds no cash. It holds a receivable. This is the single most under-appreciated fact about card acceptance and the reason card-heavy businesses can be profitable and still fail: revenue is recognised at the till, cash arrives days later, and the gap must be funded.

Merchant, on funding the following business day:

Account	Dr	Cr
Bank — NatWest current account (asset)	£3.15	
Merchant service charge (expense)	£0.05	
Card receivable — Worldpay		£3.20

Cardholder (Ama), if she keeps personal books:

Account	Dr	Cr
Coffee and eating out (expense)	£3.20	
Lloyds current account (asset)		£3.20

Set Ama's entry beside Lloyds' entry and the whole chapter is visible in two rows. The identical event is a **debit** to the account in the bank's ledger and a **credit** to the account in hers. Neither party is confused. They are keeping mirror books, and mirrors reverse.

Where the central bank's books balance too

The entries above leave one obligation outstanding: Lloyds owes money into the scheme, and the acquirer's side is owed money out of it. That obligation is not discharged by the card network. Card networks calculate; they do not settle. Settlement happens when the participating banks move balances at a central bank.

In sterling, that means the Bank of England's Real-Time Gross Settlement system. Institutions hold either a **reserves account** — available to banks, building societies and certain investment firms, remunerated at Bank Rate as set by the Monetary Policy Committee — or a **settlement account**, available to institutions that do not qualify for reserves but meet the Bank's settlement criteria, including payment system operators and non-bank payment service providers.

The Bank supports several settlement models. CHAPS uses real-time gross settlement: each payment settles individually and irrevocably, debiting one participant's account and crediting another's, throughout the operating day. CHAPS is usually open from 06:00 to 18:00 Monday to Friday excluding England and Wales bank holidays; participants must be open to receive by 08:00 and to send by 10:00, customer payments must be submitted by 17:40 and the system closes to bank-to-bank payments at 18:00. There are over 35 direct participants, with several thousand further institutions reaching CHAPS indirectly. Retail systems, including card schemes, generally use deferred or prefunded net settlement instead, in which a day's obligations are netted down to a handful of positions and settled in cycles — the subject of the netting chapter later in this volume.

Whichever model is used, the entry at the Bank of England has the same shape:

Account	Dr	Cr
Reserves account — paying participant	net amount	
Reserves account — receiving participant		net amount

And here is the observation that ties the two-tier system together, which the chapter on central bank money will develop. Both of those accounts are **liabilities of the Bank of England**. When one commercial bank pays another, the Bank of England's total liabilities do not change by a penny. Nothing is created and nothing is destroyed. The Bank simply records that it owes a different set of people. An interbank payment, at the apex of the system, is a relabelling exercise — which is precisely what the previous chapter said money was.

The gap between authorisation and posting

Ama's card was approved in roughly a second. Nothing was posted to her account in that second.

Authorisation is a request for, and a promise of, funds. In double-entry terms it is not a transaction at all: no debit, no credit, no change to the ledger balance. What the issuer records is a **hold** or **shadow** entry, which reduces the *available* balance — the number the app shows — while the *ledger* balance, the accounting figure, is untouched. Only at clearing, when the acquirer submits the transaction for settlement, does a balanced journal entry come into existence.

This produces effects that customers experience as inexplicable and that are in fact perfectly determinate. A pending transaction can disappear without ever posting, because a hold expired and no clearing record arrived. A refund appears slowly because it is a fresh transaction travelling the same route in reverse, requiring its own clearing and settlement, rather than a cancellation of the original. Two identical taps a second apart can both be approved because authorisation is checked against available balance, not against a queue of intentions.

Every one of these is the same underlying fact: authorisation and posting are separate events, and only one of them is double entry.

Mirror accounts, and why reconciliation exists at all

Because each institution keeps its own books, the linkage between institutions is always a **pair of accounts that are supposed to be equal and opposite**. Lloyds' "scheme settlement payable" should correspond to a scheme receivable; Worldpay's "merchant payable — Kalypso" should correspond, to the penny, to Kalypso's "card receivable — Worldpay".

In correspondent banking the pattern has its own vocabulary. An account a bank holds *with* another bank, in that bank's books, is a **nostro** — from the Italian for "ours", as in "our account with you". The same account viewed from the other side is a **vostro** — "yours", as in "your account with us". Banks additionally maintain an internal **mirror nostro**: their own running record of what the correspondent's statement ought to say. The daily job of proving that the mirror matches the correspondent's statement is nostro reconciliation, and Volume IV returns to it.

The general principle holds everywhere. Two balanced ledgers can be individually perfect and jointly wrong. Only comparison detects that, and comparison is a separate discipline with a separate cost. A payment institution that has invested in a beautiful ledger and not in reconciliation has built half a system.

The balancing invariant in software

Several practical rules follow for anyone implementing this, and they are not stylistic preferences.

Enforce the invariant at write time. A ledger service should refuse to persist a journal entry whose debits do not equal its credits. The check belongs in the same transaction as the write, not in a nightly job. Once enforced, it never fails again, which is exactly the point: correctness that costs nothing to maintain.

Make postings immutable and append-only. Never update a posted entry. To correct an error, post a reversing entry and then post the correct one, leaving all three visible. This preserves the audit trail, makes any historical balance reproducible by replaying entries up to a point in time, and is the reason regulators can ask what a balance was on a given afternoon eighteen months ago and receive an answer.

Use signed integers in minor units. Amounts are held in pence, cents or paise as integers, with the currency and its exponent carried alongside per ISO 4217. Floating point must never touch a monetary amount. The final chapter of this volume is about nothing else.

Make posting idempotent. Every entry carries a client-supplied idempotency key. A retried API call after a timeout must not create a second, equally balanced, entirely spurious journal entry. Double-posting is the most expensive bug in payments precisely because the ledger will accept it without complaint: it balances.

Support multi-leg entries natively. A model that hard-codes exactly one debit account and one credit account cannot express VAT, fees, foreign exchange spreads or partial refunds without contortion. Purpose-built ledger databases — TigerBeetle is the best-known open-source example — take the opposite approach and expose only strict two-account transfers, composing complex movements from several of them. Either design is defensible; what is not defensible is a design in which balance is optional.

Model the incomplete journeys explicitly. Give in-flight money real accounts: authorisation holds, clearing accounts, unsettled scheme positions, unallocated receipts. Every penny in the business should be in a named account at all times, including the pennies whose ownership is currently undetermined. Systems that leave in-flight money implicit discover it during an incident, which is the worst possible moment.

What the law requires

Double entry is not merely good practice in the United Kingdom; the outcome it produces is a statutory duty.

Section 386 of the Companies Act 2006 provides that "every company must keep adequate accounting records", and defines adequate as sufficient "to show and explain the company's transactions" and "to disclose with reasonable accuracy, at any time, the financial position of the company at that time". Subsection (3) requires "entries from day to day of all sums of money received and expended by the company" together with "a record of the assets and liabilities of the company". Companies dealing in goods must additionally keep year-end stock statements and stocktaking records.

Section 388 requires those records to be kept at the registered office or such other place as the directors think fit, open to inspection by the company's officers, and preserved for three years from the date they were made in the case of a private company and six years in the case of a public company.

The statute does not mandate double entry by name. It does not need to. A record that shows and explains transactions and discloses the financial position at any time is a double-entry record; no other method produces it. For regulated firms the requirements go considerably further — the FCA's safeguarding rules for payment and e-money institutions impose specific internal and external reconciliation duties on top of the Companies Act baseline — and Volume V takes those up.

Why this is the load-bearing chapter

Every mechanism in the rest of this book is a special case of what is above.

Clearing is the exchange of the information needed to make matching entries in two institutions' books. Settlement is the pair of entries at the central bank that discharges the resulting obligation. Netting is the arithmetic that reduces a million pairs of entries to one pair. Interchange is a credit in the issuer's ledger and a debit in the acquirer's. A chargeback is a reversing entry travelling the original route backwards. Reconciliation is the comparison of one institution's balanced books against another's. Safeguarding is the requirement that a specific pool of assets equal a specific pool of liabilities at all times. An outage is a period during which one side of a set of journeys was written and the other was not; the incident is the outage, and the aftermath is finding the unmatched halves.

There is no point in the entire apparatus at which anything is transported. There is only a very large number of books, each of which must balance internally, all of which must agree with each other, and none of which is allowed to record half a journey.

Pacioli's advice was that a merchant should not sleep until the debits equal the credits. Modern payments run continuously and nobody sleeps at all, which is why the check has been moved into the machinery. But it is the same check, and it is asking the same question it asked in Venice in 1494: is there anything here that came from nowhere?

3.98 Common wrong ideas

1. **Wrong:** debit means money going out and credit means money coming in. **Right:** debit is the left-hand side and credit is the right-hand side, so money arriving in a cash tin is a debit to it and money leaving is a credit.
2. **Wrong:** your bank statement is written from your point of view. **Right:** it is written from the bank's, where your account is a liability, so the bank reducing what it owes you is a debit and that is the word it prints.
3. **Wrong:** "we will credit your account" is an unambiguous instruction. **Right:** it means nothing until you know whose book and what class of account, which is why reversed postings are the most tedious class of bug in financial software.
4. **Wrong:** double entry means every transaction has exactly two lines. **Right:** it means the sum of the debits equals the sum of the credits, and one entry may have three legs, or twelve, or two hundred.
5. **Wrong:** one event in the world produces one entry in the ledger. **Right:** a single coffee sale produces one entry for the money and another for the stock, with further entries at settlement and at invoicing, and software built on a one-to-one assumption cannot represent a partial refund.
6. **Wrong:** a trial balance that agrees means the books are correct. **Right:** it means only that for every debit posted an equal credit was posted, and a whole month's rent left out entirely leaves the totals untouched.
7. **Wrong:** fraud shows up as an imbalance. **Right:** a dishonest employee does not create unbalanced entries but balanced entries to the wrong account, which is why reconciliation exists as a discipline separate from bookkeeping.

8. **Wrong:** there is a single global ledger into which both halves of a payment are written. **Right:** there are millions of separate ledgers, each balancing within itself and each entirely capable of disagreeing with its neighbours.
9. **Wrong:** the two halves of a payment happen at the same moment. **Right:** the card was approved in about a second and the double entry was made at clearing, typically the next business day, with the money sitting in suspense and clearing accounts in between.
10. **Wrong:** a ledger that enforces balance is a correct ledger. **Right:** balance is cheap and never fails again once it is enforced at write time; the expensive question is whether the entries describe what actually happened, and only comparison against somebody else's record answers it.

3.99 Chapter summary in 20 lines

1. "I spent three pounds twenty on a coffee" sounds like one fact and is at least two, and for the shop it is four or five.
2. The rule of double entry is that you never write down one end of a journey, only both ends together.
3. Debit means the left-hand column and credit means the right-hand column, and the words mean nothing more than that.
4. Because the two column totals must agree, the method checks itself, which let a shopkeeper in 1350 catch his own errors before they compounded.
5. Aisha's cake stall reaches £34 by two entirely separate routes, and that is a consequence of the rule rather than luck.
6. Almost everyone in Britain believes debit means money out, because bank statements are written from the bank's point of view and never explain whose books they are.
7. In the bank's ledger your account is a liability, so spending reduces it, and reducing a liability is a debit.
8. In your own books that account is an asset, so the identical event is a credit: the two parties keep mirror books, and mirrors reverse.
9. "Double" refers to the two sides rather than to two lines, and the real rule is that the sum of the debits equals the sum of the credits.
10. A VAT-registered shop's £3.20 flat white needs three legs, because 53 pence of it was never the shop's money.
11. Some real-world events need several separate journal entries, so there is no one-to-one relationship between happenings and entries.
12. A trial balance that agrees proves internal consistency and says nothing whatever about whether the entries are true.
13. Errors of omission, commission, principle, original entry, reversal and compensation are all invisible to the balancing check.
14. Fraud generally balances beautifully, because the dishonest entry is balanced and merely posted to the wrong account.
15. There is no world ledger: millions of separate books each balance internally and can still disagree with one another.
16. Following one £3.20 card payment through issuer, acquirer, merchant and cardholder shows the same interchange as a credit in one institution's books and a debit in another's.
17. The merchant records a receivable rather than cash, which is why card-heavy businesses can be profitable and still fail on the funding gap.
18. Authorisation posts nothing and touches only the available balance, so pending items can vanish, refunds travel the whole route again, and two identical taps can both be approved.
19. At the apex, an interbank payment moves balances between two accounts that are both liabilities

of the Bank of England, so nothing is created and nothing is destroyed.

20. Clearing, settlement, netting, interchange, chargebacks, reconciliation and safeguarding are all special cases of the same refusal to record half a journey.

Chapter sources: Luca Pacioli, Summa de arithmetica, geometria, proportioni et proportionalità (Venice: Paganino de Paganini, November 1494), tractatus XI of distinctio IX, "Particularis de computis et scripturis"; Mathematical Association of America, Convergence, "Mathematical Treasure: Luca Pacioli's Summa"; Institute of Chartered Accountants in England and Wales, historical accounting literature collection, entries on Pacioli and the Summa; G. A. Lee, "The Coming of Age of Double Entry: The Giovanni Farolfi Ledger of 1299–1300", Accounting Historians Journal 4(2), 1977; Johann Wolfgang von Goethe, Wilhelm Meister's Apprenticeship, Book I, Carlyle translation. IFRS Foundation, Conceptual Framework for Financial Reporting, revised March 2018, project summary. Companies Act 2006, sections 386 and 388 (legislation.gov.uk). Bank of England, "A brief introduction to the Real-Time Gross Settlement system and CHAPS", and the CHAPS operating-hours pages. Payment Systems Regulator, guidance on the Interchange Fee Regulation. Association of Accounting Technicians, study guidance on identifying and correcting errors. Online Etymology Dictionary, entries for "debit" and "credit".

4

CHAPTER

Ledgers, Journals and Accounts

4.0 What this chapter gives you

1. You will be able to explain why a bank's general ledger contains no account with your name on it, and where your own balance actually lives.
2. You will be able to describe the difference between the journal and the ledger — what happened against where we stand — and say why a business of any size needs both shapes of the same information.
3. You will be able to say what a chart of accounts is for, why it is designed backwards from the returns it must produce, and what its segments typically carry.
4. You will be able to separate recording, posting, settlement and valuation as four events with four timings, and use booking date and value date correctly.
5. You will be able to explain a card memo post: why the available balance drops, why the ledger balance does not, and why the amount can change before the real entry lands.
6. You will be able to answer "we missed the cut-off" with the only useful follow-up question, given that CHAPS, Bacs, the card schemes and the bank's own close are four different moments.
7. You will be able to give four independent reasons a bank must have an end of day, and describe the close-of-business sequence from stopping postings to rolling the business date.
8. You will be able to explain why immutability is a set of controls rather than a property of the technology, and what to ask a vendor who sells you an "immutable ledger".
9. You will be able to distinguish a reversal from an adjustment, say when each is appropriate, and explain why amendment in place is never legitimate.
10. You will be able to explain why a payment often cannot be cancelled, using camt.056, camt.029 and pacs.004, or the ISO 8583 reversal messages, to show that not one of them is an edit.

The previous chapter established that every payment is two facts rather than one, and that the two facts must agree. This chapter is about where those facts are written down, in what order, under what names, and according to what rules — because a principle that is never implemented is just a slogan, and double entry is implemented with almost fanatical specificity.

The distinction matters more than it sounds. It is possible to understand double entry perfectly and still have no idea what a bank's books actually look like, in the same way that it is possible to understand arithmetic and have no idea how a payroll system is organised. The discipline is the easy part. The filing is the hard part, and the filing is where the interesting failures live.

There is also a piece of misinformation to clear away first. Most people, if they think about it at all, imagine that somewhere inside their bank there is a very large table with one row per customer, and that their balance is the number in that row. There is no such table. There is a system that holds customer balances, certainly, but it is not the bank's ledger — it is one of a dozen feeder systems

whose totals are summarised into the ledger. The ledger itself has no idea who you are. It knows only that the bank owes something in the order of fourteen billion pounds to a category of people called personal current account holders, and it knows that figure to the penny.

The rest of this chapter explains how that arrangement works, why it is arranged that way, what "end of day" means to an institution that never actually closes, and why the single most important rule in the whole apparatus is that you never rub anything out.

The plain version

Go back to Ada's village shop from the first chapter, and imagine that it has done rather well.

Ada now has four shops, a delivery van, eleven staff, an accountant who visits on Thursdays, and about six hundred customers who run accounts with her. The notebook under the counter stopped working around the time she opened the second shop, for a reason that is entirely practical: a single notebook, written in the order things happen, is wonderful for answering the question *what happened today* and hopeless for answering the question *how much does Tom owe me*. To answer the second question from a notebook you would have to read every page since the beginning of time and add up all the Toms.

So Ada does what every business in history has eventually done. She splits her record-keeping into two shapes of the same information.

The diary and the folders

The first shape is a diary. Every time anything happens, someone writes it down in the order it happened, with the date, the amount, and what it was for. Nothing is ever skipped and nothing is ever written out of order. This is the record of events, and it is complete. In bookkeeping this is called the **journal**, and older British businesses called it the day book.

The second shape is a set of folders. There is a folder for Tom, a folder for Sam, a folder for the till, a folder for the van, a folder for wages, a folder for rent, a folder for everything Ada cares about separately. Whenever something goes into the diary, it also gets copied into the folders it affects — and because of double entry, it always affects exactly two of them at minimum, one on each side. This set of folders is called the **ledger**, each folder is called an **account**, and the act of copying an event from the diary into the folders is called **posting**.

The diary answers *what happened*. The folders answer *where do we stand*. They contain identical information arranged for two different questions, and a business needs both.

There is a third document, less glamorous and quietly essential: the list of which folders exist. Ada cannot have her Tuesday assistant inventing a folder called "bits and bobs" whenever something puzzling arrives, because in a year nobody will know what is in it. So there is a fixed, numbered, agreed list of every folder that may be used, and if something does not fit any of them, it goes into a holding folder and somebody senior sorts it out. That list is the **chart of accounts**, and the holding folder is the **suspense account**.

A day at Ada's Bank

Now make Ada a banker, because the arithmetic is the same and the example is more useful.

Ada's Bank opens on Monday morning with the following position. It has £40,000 of banknotes in its vault, £45,000 sitting in its account at the central bank, and £430,000 of loans it has made to people. Against that, it owes £500,000 to its depositors, and the £15,000 left over belongs to Ada. Notice that

the two sides already agree: £40,000 plus £45,000 plus £430,000 is £515,000, and £500,000 plus £15,000 is £515,000.

Four things happen during the day, and one thing happens because the day happened at all.

Tom brings in £300 in cash. The vault goes up by £300 and what the bank owes Tom goes up by £300. Two facts.

Priya pays Sam £120, both of them being Ada's customers. What the bank owes Priya falls by £120; what it owes Sam rises by £120. Two facts, and notice that from the bank's point of view nothing at all has changed in total. It owed £500,000 before and it owes £500,000 after. The money did not go anywhere. The label on it changed.

Sam takes £200 out of the cash machine. The vault falls by £200 and what the bank owes Sam falls by £200.

Ada charges Priya a £5 monthly account fee. What the bank owes Priya falls by £5, and the bank's income rises by £5. This is the first entry in the day that makes Ada better off rather than simply rearranging her obligations.

And then the day itself does something. Ada's £430,000 of loans earn interest at 5% a year, and one day has passed. Five per cent of £430,000 is £21,500 a year, which is £58.90 for the day. Nobody has paid it yet, and nobody will until the borrowers' payment dates, but Ada has earned it, so it goes in the books: the bank is owed another £58.90, and the bank's income rises by £58.90.

That last one is worth pausing on, because it is the entry that people find strangest and it explains a great deal about why banks have an "end of day" at all. Nothing happened. Nobody did anything. No customer touched anything. But time passed, and time passing is itself an economic event when you are in the business of lending money. Somebody or something has to notice that the day has ended and write down what the ending of the day earned. That is a job, it happens once per day, and it is one of the reasons the concept of a banking day exists.

Proving it adds up

At the end of Monday, Ada's accounts stand like this. On the left-hand side are the things the bank has and is owed. On the right-hand side are the things the bank owes and has earned.

Account	Left (debit)	Right (credit)
Cash in vault	40,100.00	
Balance at central bank	45,000.00	
Loans to customers	430,000.00	
Interest earned but not yet received	58.90	
Owed to depositors		500,095.00
Ada's own capital		15,000.00
Fee income		5.00
Interest income		58.90
Total	**515,158.90**	**515,158.90**

The two columns agree, to the penny. This list is called the **trial balance**, and producing it is the oldest sanity check in commerce. If the columns do not agree, something has gone wrong and you know it immediately, before it turns into a mystery.

Work through the arithmetic once and it stops feeling like magic. The vault is £40,000 plus Tom's £300 minus Sam's £200. Deposits are £500,000 plus Tom's £300 minus Sam's £200 minus Priya's £5 fee — Priya's payment to Sam nets to nothing because both are Ada's customers. Every single line came from an event that was written down twice, once on each side, and so the totals cannot help but match.

The rule about the eraser

Now the most important part of this chapter, and the part that governs everything that comes later in this book.

On Tuesday, Ada's assistant discovers that £250 which arrived on Monday for a customer called Priya Shah was credited to a different customer, also called P. Shah. It is an honest mistake and it needs fixing.

The wrong way to fix it — the way that feels natural, and the way every spreadsheet in the world invites you to — is to find Monday's entry, cross out the wrong name, and write the right one. Two seconds, problem solved, books still balance.

Ada's books do not permit this, and no proper set of books anywhere has permitted it for five hundred years. What actually happens is that three separate things are written down, all dated Tuesday, none of them touching Monday's page.

First, an entry that is the exact mirror image of Monday's mistake: take £250 back off the wrong P. Shah and put it back where it came from. This is called a **reversing entry**, and it does not make Monday disappear. Monday's entry is still there, still visible, still wrong. It has simply been cancelled out by an equal and opposite entry the following day.

Second, the correct entry: take the £250 out of the holding account and credit it to Priya Shah, where it should have gone in the first place.

Third — and this is the part that separates a real ledger from a tidy one — a note, attached to both of the above, saying who found the error, who authorised the correction, when, and why.

Six lines of writing where an eraser would have taken none. This looks like bureaucratic self-punishment until you consider what the eraser costs you. Once erasing is permitted, no statement of the books is ever evidence of anything. If Monday's page can be quietly changed on Tuesday, then Monday's page proves nothing about Monday, and neither does any other page. The auditor cannot check the books, because the books are whatever the last person to touch them says they are. The customer cannot dispute a figure, because there is no fixed figure to dispute. And someone stealing money can hide it perfectly by simply amending history.

The reversing entry is therefore not an accounting formality. It is the mechanism by which a set of books becomes evidence rather than opinion. And once you have seen it, you will see it everywhere in payments, because the same logic applies at every level of the system. A refund is not the deletion of a purchase; it is a second, opposite transaction. A cancelled bank transfer is not an unsent message; it is a new message asking the other bank to send the money back. Nothing is ever undone. Things are only ever answered.

Where the plain version stops being true

The shop is a good model and it will get you through a dinner party. Four things about it are wrong in ways that matter.

First: there is no folder with your name on it in the general ledger. There never was.

The plain version implied a single ledger containing an account for each customer. Real institutions do not work this way and could not. A bank with eight million current accounts does not maintain eight million accounts in its general ledger; the general ledger might contain a few thousand accounts in total, and one of them will be something like "personal current accounts, sterling, retail division" holding a single aggregate balance of several billion pounds.

Your individual balance lives in a **sub-ledger** — in this case the deposit system, which is part of what the industry calls the core banking platform. The sub-ledger holds the detail: your account, your balance, your transactions, your overdraft limit. The general ledger holds the summary, in a **control account**, and the two are reconciled, usually daily. If the deposit sub-ledger says the bank owes £14,208,331,402.71 across all personal current accounts and the general ledger control account says £14,208,331,398.44, there is a problem, it is a serious one, and somebody's afternoon is now about finding £4.27.

This is not merely a technical arrangement; it changes what questions each system can answer. The general ledger cannot tell you anything about you. It can tell you what the bank owes in total, by product, by currency, by legal entity, by branch. That is what it is for, because that is what regulatory returns, statutory accounts and capital calculations are built from. Confusing the two leads people to make wrong assumptions about what a bank can look up and how fast.

Second: recording, posting, settling and valuing are four different events, and they routinely happen on different days.

The plain version used "write it down" and "post it" almost interchangeably. In a real system they are distinct stages with distinct timestamps and distinct meanings, and the gaps between them explain a large fraction of the questions customers ask.

An entry is **recorded** when it is captured into the system. At that stage it may exist in an unposted state — validated or not, awaiting authorisation, awaiting a matching leg. It is **posted** when it is committed to the account and actually changes the balance. It carries a **booking date**, being the day the bank's books recognise it, and a **value date**, being the day from which it counts for interest and availability. Those two dates are frequently different. A cheque paid in on Friday may be booked on Friday and valued on the following Wednesday. A payment made on a Saturday may be booked with Monday's date because Saturday is not a business day in the bank's accounting calendar.

And **settlement** — the moment the two banks are actually square with each other, usually across accounts at the central bank — is a fourth event with its own timing, which frequently happens after the customer has seen the money. Chapter seven is entirely about the difference between clearing and settlement, and the reason it needs a chapter is that the two are conflated constantly, including inside banks.

The most visible consequence of this is the card authorisation. When you tap for £43.20 in a supermarket, no accounting entry is posted anywhere. What happens is a **memo post**: your available balance drops by £43.20 while your ledger balance is untouched, because no double entry has occurred. The real posting happens when the transaction clears, typically one to three business days later, at which point the memo hold is released and a genuine entry lands. This is why the amount can change between the two — a restaurant tip, a fuel pump, a hotel — and why the date on your statement is sometimes not the date you were in the shop.

Third: "end of day" is not a moment when the bank stops, and there is more than one of them.

The shop closes at six. A bank does not close at all — Faster Payments has run around the clock since 2008, cash machines never sleep, and card authorisations arrive at four in the morning from customers in other time zones.

What a bank has instead is a **business date**, which is a label rather than a wall clock. Postings are stamped with a business date, and at some point the system performs a set of end-of-day procedures and rolls the business date forward. Transactions arriving after the roll get tomorrow's date even though it is still today outside. The batch is a logical boundary, not a physical shutdown, and for a bank operating in several countries there may be several business dates alive at once across different books.

Worse, there is no single cut-off, because the bank sits inside other people's calendars as well as its own. In the United Kingdom, CHAPS opens at 6am and closes at 6pm on working days, with customer payments required by 5.40pm. Bacs input files must reach the scheme between 07.00 and 22.30 on input day. Card schemes have their own daily cut-offs which do not align with either. Each of those is somebody's end of day, none of them is the same moment, and the bank's own accounting close is yet another. When somebody in payments says "we missed the cut-off", the useful follow-up question is always *whose*.

Fourth: immutability is a set of controls, not a property of the technology.

The plain version said you never rub anything out. That is the rule. It is not, however, a law of physics.

Underneath almost every general ledger in the world sits a perfectly ordinary relational database in which any sufficiently privileged person can execute an update statement against a posted journal line. What prevents it is not that the operation is impossible but that it is prohibited, monitored, and detectable — application designs that only ever append, database permissions that deny direct write access to production financial tables, audit logging that records every access, hash-chaining or sequence numbering that makes gaps visible, and separation of duties so that the person who creates a journal entry is never the person who approves it.

This matters practically, because software is sold on the claim of being an "immutable ledger" when what is meant is that the vendor's own screens have no delete button. That is a real and useful control, but it is a control, not a guarantee, and anyone assessing a system should ask what stops a database administrator, what stops a support engineer with an emergency fix script, and what would make either visible afterwards. The answer is usually a combination of restricted access, mandatory approval workflow, and independent reconciliation — and if the answer is "nobody would do that", the system does not have immutability, it has trust.

There is also a narrower correction. Reversal is not the only permitted repair. A **reversing entry** restores the position exactly and then the correct entry is made afresh, which is preferred because it preserves the gross record of what was originally done. An **adjusting entry** posts only the net difference, which is shorter but leaves the original figure standing uncorrected in the account. Both are legitimate, both are used, and the choice depends on materiality, on whether the accounting period has closed, and on the institution's own policy. What is never legitimate is amendment in place.

The technical version

Accounts and their classes

An account is a classification bucket with a running balance. Every account belongs to one of five classes, and the class determines which side of the entry increases it.

Class	Increased by	Example in a bank
Asset	Debit	Loans and advances to customers; balances with central banks; nostro accounts
Liability	Credit	Customer deposits; debt securities issued; accrued charges
Equity	Credit	Called-up share capital; retained earnings; revaluation reserves
Income	Credit	Interest income; fee and commission income; interchange income
Expense	Debit	Interest expense; staff costs; scheme fees; impairment charges

The perpetual confusion for non-accountants is that a customer deposit is a liability, so the bank *credits* your account when it owes you more. Your statement is written from the bank's point of view, which is why a credit is money in and a debit is money out, and why the words appear reversed relative to the bank's own internal usage in ordinary conversation. Chapter ten returns to this, at length, because it is the single most common source of misreading a statement.

The fundamental identity that all of this enforces is that assets equal liabilities plus equity, extended within a period to include income and expense, which are closed to retained earnings at period end.

The chart of accounts

The chart of accounts is the controlled list of every account that may be posted to, together with its class, its posting rules and its reporting mappings. In an institution of any size it is not a flat list of numbers but a structured code, typically composed of independent segments.

A realistic account code in a bank might carry a legal entity identifier, a natural account, a currency, a product or portfolio code, a cost or profit centre, and an intercompany counterparty segment. The natural account alone might be six or seven digits; the full posting string might be thirty characters or more. This structure exists because the same economic fact needs to be sliced several ways: interest income needs to be reportable by product for management, by entity for statutory accounts, by currency for treasury, and by counterparty type for regulatory returns, and it is far cheaper to encode all of that on the entry than to reconstruct it later.

The chart is not designed in isolation. It is designed backwards from the reports it must produce. In the United Kingdom those include the company's statutory accounts under the Companies Act 2006 and the applicable accounting framework, the prudential and financial reporting returns required by the Prudential Regulation Authority, and the statistical returns required by the Bank of England. In the European Union the equivalent frameworks are the European Banking Authority's COREP, covering own funds and prudential metrics, and FINREP, covering financial information anchored to the accounting figures. A chart of accounts that cannot be mapped cleanly to the required returns is not a stylistic problem; it is an annual manual reconciliation exercise for the rest of the institution's life.

Multi-currency handling adds a further layer. Accounts are held in a transaction currency and translated into the entity's functional currency for reporting, with periodic revaluation of monetary balances and the resulting gains or losses posted to defined revaluation accounts. The revaluation run is another of the jobs that must happen at a defined point in the day, using a defined rate, from a defined source.

Journals, journal entries and journal lines

A journal is the chronological record. A journal entry is the atomic unit of posting: a header describing the event and two or more lines describing its effects, whose debits and credits must sum to zero before the entry is accepted. The system rejects an unbalanced entry outright, which is the single most valuable structural property of double entry as implemented in software.

A minimally serious journal entry carries, at header level, a unique identifier, the source system, the journal category or type, the business date, the currency, the description or narrative, the creating user or process, the approving user, the approval timestamp, and a reference to the originating business object such as a payment identifier or a card transaction identifier. At line level it carries the full account code string, the debit or credit indicator, the transaction amount, the functional-currency amount and the rate used, the value date if it differs from the business date, and a line-level narrative.

Journals divide broadly into automatic and manual. Automatic journals are generated by feeder systems — deposits, lending, cards, treasury, payments — from business events, according to fixed accounting rules held in an accounting rules engine that maps event types to debit and credit account pairs. Manual journals are typed in by the finance function to record accruals, provisions, reclassifications and corrections. Manual journals are the ones auditors care about, because they are the ones a human chose. International Standard on Auditing 240, which governs the auditor's responsibilities relating to fraud, requires auditors to test the appropriateness of journal entries recorded in the general ledger — with particular attention to entries made outside the normal course of business, entries posted at period end, and entries made by people who do not usually post journals.

Recurring journals — accruals, prepayments, depreciation — are templated and generated on a schedule. Reversing journals are a special case worth naming: an accrual posted at month end is very often flagged to reverse automatically on the first day of the following month, so that when the actual invoice arrives it can be posted gross without double counting. This is an entirely routine use of reversal that has nothing to do with error correction.

Sub-ledgers and control accounts

The general ledger is fed by sub-ledgers, each of which holds transaction-level detail for one domain and reports a summary into a control account.

Sub-ledger	Holds	Typical control account
Deposits (core banking)	Individual customer current and savings accounts	Customer deposits, by product and currency
Lending	Individual loan accounts, balances, accrued interest, arrears	Loans and advances to customers
Cards	Cardholder balances, authorisations, presentments, fees	Card receivables; merchant payables
Nostro	The bank's own accounts held at correspondent banks	Balances with other banks

Sub-ledger	Holds	Typical control account
Fixed assets	Individual assets, cost, depreciation, disposals	Property, plant and equipment
Payables and receivables	Supplier invoices and customer invoices	Trade creditors; trade debtors

British practice historically named these the sales ledger and the purchase ledger, with the general ledger itself called the nominal ledger — terminology still common in UK accounting software and worth recognising when it appears.

The **sub-ledger to general ledger reconciliation** is the control that binds the two together, and in a bank it is performed daily rather than monthly. Its logic is simple: the sum of all detail balances in the sub-ledger must equal the balance of the corresponding control account. Any difference is a break, and breaks are aged, categorised and escalated. Where an item cannot be resolved immediately it is parked in a **suspense account** — a temporary general ledger account for items that are known to have occurred but are not yet correctly classified. A suspense account that is cleared daily is housekeeping. A suspense account with a growing balance and items aged over ninety days is one of the clearest early indicators that an institution's operations are failing, and it is precisely what supervisors and auditors look for.

Posting versus recording, precisely

The lifecycle of an entry runs: capture, validate, authorise, post, report. Only the fourth step changes a balance.

The distinction is visible on the wire in ISO 20022 cash management reporting. The message camt.053, formally BankToCustomerStatement, is the end-of-day statement covering a completed business day. Each entry within it carries a status; the value BOOK indicates that the entry has been booked to the account, as opposed to being merely pending. Each entry also carries `BookgDt`, the booking date, and `ValDt`, the value date, as separate elements — the standard treats them as two different facts about the same entry precisely because they routinely differ. Statement-level balances are typed by code, with OPBD for the opening booked balance and CLBD for the closing booked balance, and separate codes exist for available balances, which is the standard's way of making explicit the ledger-versus-available distinction introduced in chapter two.

The companion messages complete the picture. Camt.052, BankToCustomerAccountReport, carries intraday reporting, including items that are not yet booked. Camt.054, BankToCustomerDebitCreditNotification, notifies the account owner of individual debit or credit entries as they occur. In the older SWIFT FIN world the equivalents are MT940 for the end-of-day customer statement and MT942 for the interim transaction report, in which the same information is encoded positionally inside fixed-format tags — a Tag 61 statement line packs the value date and entry date into the first ten characters, which is compact, unforgiving, and the reason the industry is migrating.

The trial balance, and what it does not prove

The trial balance is the list of every account's closing balance with debits in one column and credits in the other. It is produced as a matter of routine, at least monthly and in banks daily, and its agreement is a necessary condition for the books being correct.

It is emphatically not a sufficient one. A trial balance will balance perfectly in the presence of at least four distinct classes of error. An entry omitted entirely, both legs missing, leaves the totals untouched.

An entry posted to the wrong account of the same class — Priya's £250 credited to the wrong customer — leaves the totals untouched, which is exactly the error the plain version used. An entry posted twice in full leaves the totals untouched. And compensating errors, where two mistakes happen to offset, leave the totals untouched. A balanced trial balance means the arithmetic is consistent. It says nothing whatever about whether the entries describe reality.

This is why reconciliation exists as a separate discipline from bookkeeping, and why chapter six of volume five is devoted to it. Balancing proves internal consistency. Reconciliation proves agreement with an external record — the counterparty bank's statement, the scheme's settlement file, the central bank's account. The two controls catch different things and neither substitutes for the other.

End of day, and why batch processing exists at all

Modern engineers, raised on event streams and continuous deployment, tend to treat batch processing as a relic. It is not. There are at least four independent reasons a bank must have a defined daily boundary, and none of them is about the age of the software.

The first is that some financial facts are defined per day and cannot exist without a day. Interest accrual is the obvious case: interest accrues on a balance for a period, and the period has to end for the accrual to be computed. So do daily fee applications, daily limit resets, daily average balance calculations, and the ageing of arrears. A system with no end of day cannot compute these because the input to the calculation does not exist until the day is closed.

The second is that the bank sits inside external cycles that are batch by design. The Bacs three-day cycle, run by Pay.UK, is the clearest example in Britain: files are submitted on input day, with transmission to Bacs permitted between 07.00 and 22.30; they are processed and distributed to the receiving payment service providers on processing day; and on entry day, day three, the payments are simultaneously credited to the recipients' accounts and debited from the originator's. That structure is not negotiable by an individual bank, and a bank's own processing has to align to it.

The third is settlement. Obligations between banks are settled at the Bank of England across settlement accounts in the RTGS service, and the settlement arrangements are themselves periodic. CHAPS settles individually and in real time on a gross basis throughout the day, between 6am and 6pm. Bacs settles on a net basis once each business day. The Faster Payments Service, despite running around the clock for customers, settles net three times each business day. The Image Clearing System for cheques runs a two-day cycle and settles net once each business day. LINK, Visa Europe and Mastercard Europe settle 24-hour cycles, with cycles falling on weekends and public holidays settling on the following business day. Every one of those is a boundary the bank's books must respect.

The fourth is that reporting requires a frozen point. A regulatory return, a statutory account, a customer statement and a management report are all statements about a defined moment. If the underlying figures never stop moving, no two reports of the same period will ever agree, and the institution loses the ability to say anything definite about itself.

In practice this is implemented as a sequence of scheduled jobs, known in core banking platforms as close of business or COB. Temenos, whose Transact platform is one of the most widely deployed core banking systems in the world, structures this as pre-COB, COB and post-COB stages. The typical sequence within it is: stop accepting new postings for the current business date; complete or fail any in-flight transactions; execute standing orders and scheduled payments due today; calculate and post interest accruals; apply fees and charges; revalue foreign currency positions; run limit and arrears checks; generate customer statements; extract summarised postings into the general ledger; produce

the trial balance; run the sub-ledger reconciliations; produce regulatory and statistical extracts; take the backup; and finally roll the business date.

The reason this sequence is treated with something close to reverence in banking operations is that it is both mandatory and time-boxed. It must complete before the next business day's processing begins, which in a 24/7 institution means it must complete inside a window measured in hours while customers are still transacting. If COB does not finish, statements do not generate, interest does not post, files do not go to the schemes, and the bank enters the next day without knowing its own position. Delayed or failed end-of-day runs are among the most serious incidents a bank can have, and they are the reason batch scheduling and workload automation remain specialist and well-paid disciplines.

The consequences of getting this wrong are not theoretical. On 20 December 2022 the Financial Conduct Authority and the Prudential Regulation Authority jointly fined TSB Bank plc a total of £48,650,000 — £29,750,000 by the FCA and £18,900,000 by the PRA — for operational risk management and governance failures relating to the bank's 2018 IT migration programme, in which customers lost access to accounts and the bank lost the ability to present them with reliable information about their own money. The fine was for the management of the risk, not for the technology. That is the correct place to put it.

Immutability, reversal, and why payment systems inherit the rule

The governing principle is that a posted entry is never amended and never deleted. Corrections are made by further entries.

The legal underpinning in the United Kingdom starts with the Companies Act 2006. Section 386 requires every company to keep adequate accounting records, defined as records sufficient to show and explain the company's transactions, to disclose with reasonable accuracy at any time the financial position of the company at that time, and to enable the directors to ensure the accounts comply with the Act. Section 386(3)(a) specifies that those records must contain "entries from day to day of all sums of money received and expended by the company and the matters in respect of which the receipt and expenditure takes place". Section 388 requires records to be preserved for three years from the date they are made for a private company, and six years for a public company.

For regulated firms the requirement is stricter. The FCA's Senior Management Arrangements, Systems and Controls sourcebook, at SYSC 9.1.1R, requires a firm to "arrange for orderly records to be kept of its business and internal organisation, including all services and transactions undertaken by it, which must be sufficient to enable the FCA to monitor the firm's compliance with the requirements under the regulatory system, and in particular to ascertain that the firm has complied with all obligations with respect to clients". SYSC 9.1.2R requires a common platform firm to retain records relating to its MiFID business for at least five years. A record that can be silently altered is not a record for these purposes.

The mechanics of correction have names. A **reversal**, known on the continent and in a good deal of banking software by the German term *storno*, posts the exact contra of the original entry: same accounts, same amount, opposite sides, with an explicit reference linking it to the entry being reversed. The original and the reversal both remain visible. An **adjustment** posts only the net correction. Reversal is generally preferred for erroneous postings because it preserves the gross history and makes the sequence of events legible; adjustment is used where the original entry was correct in kind but wrong in amount, and where the period is still open.

Around both sits the audit trail: who created the entry, who approved it, when, from what terminal, under what authority, and with what stated reason. Segregation of duties — maker and checker as

different people, with the system enforcing it — is the control that makes the audit trail meaningful, since an audit trail showing that one person did everything is a record of a risk rather than a mitigation of it.

Now the important part for this book. The same principle governs payment networks, and for the same reason: a message that has been sent to another institution cannot be unsent, because the other institution has already acted on it.

In the card world, ISO 8583 provides message type indicators 0400 for an acquirer reversal request and 0420 for an acquirer reversal advice. A reversal does not delete the original authorisation; it is a separate message referring to it, and it must be matched to it by the identifiers described in volume three. A refund is not a reversal at all — it is a fresh transaction in the opposite direction, submitted through the normal clearing process, which is precisely why a refund takes days when the original purchase appeared to take seconds. Nothing is being undone. A second thing is being done.

In the ISO 20022 world the vocabulary is explicit about this. Camt.056, the FIToFIPaymentCancellationRequest, is a *request* to cancel a payment — it can be refused, because the receiving institution may already have paid the money away. Camt.029, ResolutionOfInvestigation, carries the answer. And pacs.004, PaymentReturn, is the message that actually sends the funds back: a new payment, in the opposite direction, referencing the original. Three messages, and not one of them is an edit.

In Bacs the same logic appears as an operational rule rather than a message. Once a submission has passed its cut-off, the individual payments within it cannot be removed; a payment can be recalled only before a specified deadline through the originator's payment service provider, and after entry day the remedy lies outside the clearing — through the Direct Debit Guarantee for a collection, or through an indemnity claim for a credit sent in error.

This is why "can you cancel that payment?" is so often answered with an apologetic no. It is not obstruction and it is not usually a technical limitation. It is the same rule as the one that stops Ada rubbing out Monday, applied across institutional boundaries, where the stakes are higher and the parties do not report to the same management.

What this means if you are building something

Five consequences follow directly, and each of them is a mistake that is expensive to discover late.

Model postings as an append-only sequence and derive balances from them. If your data model has an `UPDATE balances SET amount = ?` in it anywhere, you have built a system that cannot explain itself, and an unexplainable balance is a regulatory problem before it is an engineering one.

Give every entry at least three dates and never conflate them: the timestamp at which the event occurred, the business date under which it is booked, and the value date from which it counts. Store the timezone explicitly. A great many reconciliation breaks are timezone bugs wearing a costume.

Make reversal a first-class operation with a mandatory link to the entry being reversed, and make it the only way an entry can be neutralised. If a support tool needs to fix something, it should issue a reversal through the same path as everything else, under the same approval controls, and appear in the same audit trail.

Reconcile sub-ledger to general ledger on a fixed cycle, automatically, with breaks aged and escalated. Do not wait for month end. In payments, a break that is one day old is a bug report and a break that is thirty days old is an investigation.

And define your business date explicitly, in one place, as a value the system knows about, rather than letting it be implied by whatever the server clock says. Every scheduled job, every report, every statement and every extract should agree on which day it is, and they will not do so by accident.

The next chapter takes the aggregate view that the general ledger produces and asks what it adds up to. A bank's balance sheet is not a summary of a bank; it is a description of what a bank fundamentally is, and once the ledger machinery is understood, the balance sheet stops being an accounting artefact and becomes the clearest available answer to the question of why your asset is somebody else's debt.

4.98 Common wrong ideas

1. **Wrong:** somewhere inside your bank there is a very large table with one row per customer, and that table is the ledger. **Right:** your balance lives in a sub-ledger, and the general ledger holds only an aggregate control account for a whole product, currency and division.
2. **Wrong:** the general ledger can look you up. **Right:** it can tell you what the bank owes in total by product, currency, legal entity and branch, because that is what regulatory returns, statutory accounts and capital calculations are built from.
3. **Wrong:** recording an entry and posting it are the same thing. **Right:** the lifecycle runs capture, validate, authorise, post, report, and only the fourth step changes a balance.
4. **Wrong:** the booking date and the value date are the same date. **Right:** they are two different facts about the same entry, which is why ISO 20022 carries `BookgDt` and `ValDt` as separate elements.
5. **Wrong:** tapping your card posts an accounting entry. **Right:** it creates a memo post that reduces the available balance while the ledger balance is untouched, with the genuine posting arriving one to three business days later and sometimes at a different amount.
6. **Wrong:** a bank has an end of day, singular. **Right:** CHAPS, Bacs, the card schemes and the bank's own accounting close are separate cut-offs, so the useful follow-up to "we missed the cut-off" is always *whose*.
7. **Wrong:** batch processing is a relic that event streams have made unnecessary. **Right:** interest accrual, external scheme cycles, periodic settlement and the need for a frozen reporting point each independently require a defined daily boundary.
8. **Wrong:** an immutable ledger is a property of the software you bought. **Right:** underneath sits an ordinary database in which a sufficiently privileged person can run an update, so immutability is restricted access, approval workflow, audit logging, sequence numbering and independent reconciliation.
9. **Wrong:** a mistake is fixed by finding the original entry and correcting it. **Right:** a reversal posts the exact contra with an explicit link, the correct entry is then made afresh, and both remain visible along with a note of who authorised the repair.
10. **Wrong:** a refund reverses the original payment. **Right:** a refund is a fresh transaction in the opposite direction through the normal clearing process, which is exactly why it takes days when the purchase appeared to take seconds.

4.99 Chapter summary in 20 lines

1. A principle that is never implemented is a slogan, and double entry is implemented with almost fanatical specificity.
2. Records take two shapes: a journal in the order things happened, and a ledger of accounts arranged by where things stand.
3. The chart of accounts is the controlled list of which accounts may be used, and the suspense account is where anything that fits none of them waits.

4. A day at Ada's Bank shows the pattern: a cash deposit, an internal transfer, a withdrawal and a fee, each written twice and each balancing.
5. Time passing is itself an economic event for a lender, which is why a day's interest accrual has to be computed and posted by somebody.
6. The trial balance lists every account in two columns and proves the totals agree, which is the oldest sanity check in commerce.
7. The rule about the eraser is absolute: a mistake is answered by a reversing entry, a fresh correct entry, and a note of who found and authorised it.
8. Once erasing is permitted no page proves anything about its own day, the auditor cannot check, the customer cannot dispute, and theft can be hidden by amending history.
9. There is no folder with your name on it in the general ledger, because your balance lives in a sub-ledger summarised into a control account.
10. That sub-ledger to general ledger reconciliation runs daily in a bank, and a £4.27 difference is somebody's afternoon.
11. Recording, posting, settlement and valuation are four events on four timings, and the gaps between them explain a large fraction of what customers ask about.
12. A card authorisation is a memo post against the available balance only, with the real entry landing at clearing days later.
13. A bank has a business date rather than a closing time, and a group operating in several countries may have several alive at once.
14. There is no single cut-off, because CHAPS, Bacs and the card schemes each impose their own and none of them is the bank's accounting close.
15. Batch exists because some facts are defined per day, because external cycles are batch by design, because settlement is periodic, and because reporting needs a frozen point.
16. Close of business is a mandatory, time-boxed sequence that ends by rolling the business date, and a failed run leaves a bank entering the next day without knowing its own position.
17. TSB's £48.65 million fine in December 2022 was for the management of operational risk rather than for the technology, which is the correct place to put it.
18. Immutability is a set of controls — restricted access, maker and checker, audit logging, reconciliation — rather than a property of the database underneath.
19. Reversal preserves the gross record and adjustment posts only the net difference; both are legitimate, and amendment in place is not.
20. Payment networks inherit the same rule, so camt.056 is only a request, pacs.004 is a new payment in the opposite direction, and nothing anywhere is an edit.

Chapter sources: Companies Act 2006, sections 386 and 388 (legislation.gov.uk). FCA Handbook, SYSC 9.1.1R and SYSC 9.1.2R. Bank of England — Payment and settlement pages covering CHAPS, Bacs, Faster Payments, the Image Clearing System, LINK and card scheme settlement; policy statement "Extending RTGS and CHAPS settlement hours – early morning extension" (2026). Bacs / Pay.UK — Bacs Direct Credit getting-started guidance on the three-day processing cycle and input window. ISO 20022 message definitions for camt.052, camt.053, camt.054, camt.056, camt.029 and pacs.004, together with the SWIFT camt.053.001.02 message definition report; SWIFT MT940 and MT942. ISO 8583 message type indicators 0400 and 0420. International Standard on Auditing 240 on journal entry testing. European Banking Authority reporting frameworks (COREP and FINREP). Temenos documentation on close-of-business processing. Bank of England and Financial Conduct Authority joint news release, "TSB fined £48.65m for operational resilience failings", 20 December 2022.

What a Bank Is

5.0 What this chapter gives you

1. You will be able to explain why your bank balance is a debt the bank owes you rather than a pile of notes held somewhere with your name on it.
2. You will be able to draw a bank's two lists and say which side a customer deposit sits on and which side a mortgage sits on, without hesitating.
3. You will be able to show, in two entries, how a bank makes a loan by typing a number into an account, and why that leaves its equity unchanged.
4. You will be able to say why the goldsmith-and-vault story and the money multiplier are wrong, and point to what the Bank of England published in 2014 instead.
5. You will be able to separate the constraints on the banking system as a whole from the constraints on one bank, and explain why a treasurer who says "we lend out deposits" is describing their job accurately.
6. You will be able to explain why a bank can report a capital ratio of fourteen per cent while the owners' money is about five per cent of its balance sheet, and why both figures are true.
7. You will be able to say why capital did not save Silicon Valley Bank, and keep the defence against losses distinct from the defence against withdrawals.
8. You will be able to describe what happens to depositors, shareholders and AT1 holders when a British bank is resolved over a weekend, using SVB UK as the worked case.
9. You will be able to state what the FSCS protects, why "per banking licence" is not the same as "per brand", and where an uncovered corporate depositor ranks in an insolvency.
10. You will be able to explain why a payment institution is not a bank, has neither the capital nor the liquidity regime, and must therefore safeguard its customers' funds rather than lend them.

Ask a hundred people in a British high street what a bank is and roughly ninety-five will describe a building where money is kept. The remaining five will say something about apps. Both answers describe the furniture.

A bank is not a place where money is kept. Very nearly the opposite is true: a bank is an institution that has spent the money you gave it, is legally obliged to give it back the moment you ask, and stays alive by being extremely careful about the gap between those two facts. Everything difficult about banking lives in that gap. Every rule, every regulator, every capital ratio, every crisis of the last four hundred years.

This is not a criticism. If banks did nothing but hold money they would be warehouses, they would charge you rent, and no house in Britain would ever be bought with a mortgage. The whole point of the institution is that it takes money people want back at a moment's notice and turns it into money other

people can borrow for twenty-five years. That transformation is useful, dangerous, and impossible to perform without the possibility of failure. Banking regulation is an attempt to make failure rare, survivable and orderly, not to abolish it, because abolishing it would mean abolishing the bank.

Two earlier chapters carry the weight here. Chapter two established that your balance is not a substance but a record. Chapter three established that no record of money may be changed by writing down one fact; every change is two entries that balance. This chapter puts them together: if your balance is a record in somebody else's book, whose book is it, what else is in it, and what happens when it stops adding up?

Along the way we will demolish the most widely taught and most thoroughly wrong story about banking: the goldsmith, the vault, and the fraction kept in reserve. It appears in school textbooks, in economics degrees, in newspaper explainers and in a great many bank employees' understanding of their own employer. The Bank of England published a paper in 2014 saying, in effect, that it is not how anything works. We will get to that paper. First, the plain version.

The plain version

Your money is somebody else's promise

Start with the thing in your hand. Open your banking app. It says, let us say, £1,842.60.

That number is not a picture of money sitting somewhere with your name on it. There is no drawer. There is no bundle of notes in a vault in Edinburgh labelled *Ms A. Okonkwo*. The number is a **promise**: the bank saying we owe you one thousand eight hundred and forty-two pounds and sixty pence, and we will hand it over, or send it wherever you tell us, whenever you ask.

Your bank balance is a debt. It is a debt the bank owes to you.

This feels alarming the first time you hear it, then stops feeling alarming when you notice it is exactly what you already believed. You have never once thought a specific pile of notes was yours. You have never asked which notes. What you care about is that when you tap your card for £3.20, the coffee shop gets £3.20 and your app shows £3.20 less. A promise that is always kept is indistinguishable, in daily life, from a thing you own.

The coat-check that lends the coats

Here is the analogy, and it is a good one for about four paragraphs.

Imagine a cloakroom at a theatre. You hand over your coat, you get a ticket, and at the end of the evening you swap the ticket back for the coat. It would be outrageous if, halfway through the second act, the attendant lent your coat to somebody walking past outside. Your coat is a particular coat. You want *that* one back.

Money is not like a coat. Any ten-pound note is as good as any other. So a cloakroom that took in money instead of coats could lend it out during the evening and nobody would mind, provided it could hand *a* ten-pound note back to each ticket-holder at the end. Nobody checks the serial numbers. That is roughly the story people tell about the goldsmiths of seventeenth-century London: they took in gold for safekeeping, issued receipts, noticed that most of the gold sat there untouched, and started lending it.

Hold that picture for one more minute, because we are going to break it. It gets you to the first important idea: a bank has two lists, not one.

The two lists

Every bank keeps two lists, and understanding a bank means understanding both at once.

The first is **what the bank owes**. Your £1,842.60 is on it. So is every other customer's balance, plus money the bank has borrowed from other banks, plus bonds it has issued to investors.

The second is **what the bank owns**. Mortgages it has lent out and expects to be repaid. Business loans. Credit card balances people owe it. Government bonds it has bought. Actual cash. A pile of computers and some rather expensive branch leases.

In a healthy bank, the second list is bigger than the first. It has to be. The difference between them is the owners' money — the shareholders' stake, built from what they originally put in and from profits kept rather than paid out. It is the cushion. If the bank owns £100 of things and owes £95, the cushion is £5.

That cushion is not a fund sitting in an account somewhere. It is not money set aside. It is simply the gap: what would be left over for the owners if every asset were turned into cash and every debt paid. Its entire job is to absorb bad news. When a borrower does not pay, the first list stays exactly where it is — the bank still owes you every penny of your £1,842.60 — and the second list shrinks. **The loss comes out of the owners' cushion, not out of the depositors' promises. The cushion is there precisely so that depositors never notice.** A bank fails when the losses are big enough to eat the whole cushion and start eating the promises.

Building a small bank with real numbers

Meet Priya, who is starting a very small, very simplified bank in a town called Little Sutton. The numbers are tiny and invented so they can be followed; everything structural about them is true of a real bank.

Priya puts in £10,000 of her own money. On day one the bank owns £10,000 of cash, owes nothing, and the owners' cushion is £10,000. Then two hundred residents open accounts and pay in £90,000 between them.

What the bank owns		What the bank owes	
Cash	£100,000	Owed to depositors	£90,000
		Owners' cushion	£10,000
Total	**£100,000**	**Total**	**£100,000**

Notice that the totals still match, and why. Priya received £90,000 of cash and simultaneously took on £90,000 of promises. Two entries, as chapter three insisted. The cushion did not change, because taking a deposit does not make a bank richer or poorer. It makes it bigger.

Now Priya lends. Mr Hollis wants £60,000 to expand his bakery. Here the coat-check analogy has to be retired. Priya does not hand him sixty thousand pounds in notes from the drawer. She opens him an account and types **£60,000** into it. That is the loan.

Watch the lists. The bank now owns something new: Mr Hollis's promise to repay £60,000, an asset, because it is money owed *to* the bank. And it owes something new: £60,000 sitting in Mr Hollis's account, which he can spend, because it is money owed *by* the bank.

What the bank owns		What the bank owes	
Cash	£100,000	Owed to depositors	£150,000
Loan to Mr Hollis	£60,000	Owners' cushion	£10,000
Total	**£160,000**	**Total**	**£160,000**

Read that table twice. The bank is now £60,000 bigger on both sides. Nothing came in the door. Nobody deposited anything. Priya created £60,000 of spendable money by typing it, and created £60,000 of debt owed to her at the same instant, and the two exactly cancel, so her cushion is still £10,000 and her books still balance.

This is not a trick and it is not fraud. It is what banking is. When the Bank of England explains this to the public it says, in plain words, that most money in the economy is created not by printing presses but by banks when they make loans, and that when a loan is repaid the money is deleted and ceases to exist. Around eight in every ten pounds in the British economy is money of this kind. Notes and coins are around three per cent of the total.

So the goldsmith story is backwards. Banks do not take in money and lend it out. They lend, and the lending is what puts the money there.

So what stops Priya typing numbers forever?

Four things, and each is a whole industry.

First, the money walks out. Mr Hollis did not borrow £60,000 to look at it. He spends it — on ovens from a supplier who banks with Barclays, on a builder who banks with Santander. Real money must then move from Priya's bank to those banks. She cannot type that; she has to have it, or borrow it. So while the banking system as a whole creates deposits by lending, any individual bank constantly leaks. Managing that leak is called funding, and it is why banks compete so hard for your savings account even though they do not need your savings in order to lend.

Second, some borrowers do not pay. If Mr Hollis's bakery fails and Priya recovers only £35,000, she has lost £25,000. Her cushion was £10,000. It is gone, and then some, and her bank is bust — not because it ran out of cash, but because what it owns is worth less than what it owes. This is why regulators insist the cushion be large relative to how risky the lending is.

Third, everyone might want their money at once. Priya has £100,000 of cash and £150,000 of promises. If more than £100,000 of promises are called in on the same afternoon she cannot pay, even though she is perfectly solvent and Mr Hollis is making excellent sourdough. Being unable to pay today is fatal regardless of being rich tomorrow. This is why regulators separately insist that banks hold a stock of things that can be turned into cash immediately.

Fourth, somebody is watching. In the United Kingdom that is the Prudential Regulation Authority, part of the Bank of England. It sets how big the cushion must be, how much instantly-sellable stuff must be held, and what happens if either falls short. It can and does stop banks paying dividends, and in the last resort it can close them.

What a run looks like

The third danger is the one that actually kills banks, and nobody believes it will happen until it is happening.

On Friday 14 September 2007, queues formed outside branches of Northern Rock across Britain. The bank had asked the Bank of England for emergency liquidity support two days earlier and the news

had leaked. Those queues are commonly described as the first run on a British bank since Overend, Gurney and Company collapsed in 1866. Northern Rock was not obviously insolvent that morning. It was illiquid: it had funded long-dated mortgages with short-dated wholesale borrowing that had suddenly stopped being available. The Chancellor guaranteed the deposits the same day to stop the queues. The bank was taken into state ownership on 22 February 2008.

Sixteen years later the same thing happened at hundreds of times the speed, because the queue was now an app. Silicon Valley Bank's customers attempted to withdraw more than forty billion dollars on 9 March 2023 and its management expected over a hundred billion more the following day. It was closed on 10 March. Its British subsidiary lost about a third of its deposits in two days. Nobody needed to stand in a queue.

The two words

Accountants and bankers do not say "what it owns" and "what it owes". They say **assets** and **liabilities**. The cushion is called **equity**, or **capital**.

And now the sentence beginners find upside down, which chapter three prepared you for: **your deposit is the bank's liability, and the bank's loans are its assets.** Your asset — the money you think of as yours — is the bank's debt. The mortgage you think of as your debt is the bank's asset. Every payment in the rest of this book is a movement between somebody's assets and somebody's liabilities, and getting the direction right is the difference between a payments system that reconciles and one that does not.

Where the plain version stops being true

The plain version above is honest as far as it goes. Four things in it will actively mislead you if you carry them forward unamended.

Correction one: there is no fraction, and there is no reserve

The version taught in most schools says banks must keep a fixed fraction of deposits — the classic figure is ten per cent — and may lend the rest, and that this fraction, applied repeatedly as money is redeposited, "multiplies" an injection of central bank money into a larger stock of deposits. The ratio is the reserve requirement; the mechanism is the money multiplier.

Two separate things are wrong with this.

The first is empirical. **The United Kingdom has had no mandatory reserve ratio since 1981.** The number is not ten per cent; there is no number. The Bank of England does require banks above a size threshold to place non-interest-bearing cash ratio deposits with it, but that is a levy funding the Bank's policy functions, not a liquidity rule. The United States abolished its requirements outright: the Federal Reserve Board reduced all reserve requirement ratios to zero per cent effective 26 March 2020. The euro area has kept a requirement, but it is one per cent of a defined deposit base, unchanged since 18 January 2012, and since the maintenance period beginning 20 September 2023 the European Central Bank has remunerated those reserves at zero per cent. A one per cent ratio is not a binding constraint on lending; it is a small tax.

The second is causal, and matters more. The multiplier story has the arrow pointing the wrong way. It says central bank money comes first and deposits are built on top. In practice the central bank supplies reserves on demand to keep the policy rate where it has decided it should be. If banks collectively need more reserves, they get more reserves, at the policy rate. The quantity of reserves is an outcome, not a constraint.

Correction two: "banks lend out your savings" is wrong, but so is "banks can lend without limit"

Having demolished the vault, it is easy to overcorrect into an equally false position: that because lending creates deposits, a bank is unconstrained and money is conjured at will. The distinction that resolves it is between the system and the firm.

For the **banking system as a whole**, lending creates deposits and no prior pot of savings is required. That is the Bank of England's point, and it is correct.

For an **individual bank**, the picture is much tighter, because the deposits it creates walk straight out to other banks. Priya's £60,000 becomes Barclays' problem within a week, and when it leaves she must settle in central bank reserves, which she cannot create. If she persistently lends more than she attracts, she must buy funding in the wholesale market, and that funding has a price and, crucially, an availability that can vanish overnight. Northern Rock died of exactly this. So a bank treasurer who tells you "we lend out deposits" is describing their daily job accurately even though the sentence is wrong as monetary economics. Practitioners and economists talk past each other about this constantly.

A further constraint the plain version understates: **capital is scarce and expensive.** Every new loan consumes capital, because the regulator requires the cushion to scale with the risk taken. A bank wanting to lend another billion pounds must find the capital to support it, and capital comes from retained profits or shareholders who want a return. This, not any reserve ratio, is the operative brake on lending in a modern bank.

Correction three: capital is not cash, and capital ratios do not measure survival

The plain version described the cushion as "what would be left over". That is right, but three things about it routinely mislead.

Capital is **not a pot of money**. You cannot spend it, hold it or point at it. It is an accounting difference between two other numbers. When a regulator says a bank must hold more capital, it is not saying "set aside more cash"; it is saying "fund yourself with more shareholder money and less borrowed money", which changes nothing on the asset side.

Capital ratios are measured against **risk-weighted assets**, not actual assets. A mortgage at fifty per cent loan-to-value and an unsecured loan to a start-up both sit on the balance sheet at face value, but the second absorbs far more capital. This is why a bank can report a capital ratio of fourteen per cent while its equity is about five per cent of its actual balance sheet. Both numbers are true and answer different questions. Confusing them is the commonest error in commentary about banks.

And most importantly: **capital does not stop a run.** Capital protects against losses; liquidity protects against withdrawals. They are separate defences against separate deaths, and the second kills faster. Silicon Valley Bank's capital ratios were above the required minimums when it failed. It failed because its depositors — approximately ninety-four per cent of whom were uninsured, and who knew each other — asked for their money at once, and selling assets fast enough to meet them would have crystallised losses that then *did* exceed the capital. A bank can be killed by either.

Correction four: the balance sheet is not the whole bank

The two-list picture is complete only in the sense that a photograph is complete. It omits three things that matter.

It omits **time**. Both lists have maturities. A bank's assets are typically long — a twenty-five-year mortgage — and its liabilities short, or instant. That mismatch is maturity transformation, and it is

not a flaw in the business model, it is the business model. It is also why banks are fragile in a way a manufacturer is not.

It omits **what is not on it**. Undrawn credit facilities, loan commitments, guarantees and letters of credit produce no balance-sheet entry until somebody draws on them — which they do precisely when things are going badly. Regulators handle this with credit conversion factors, a technical way of saying "assume some of it becomes real".

And it omits **valuation**. "What the bank owns" is an estimate, governed by accounting standards, involving forecasts of who will default. Under IFRS 9 a bank must book expected credit losses before anything has gone wrong. Two honest banks with identical loan books can report different asset values. The balance sheet is a considered opinion presented in the grammar of fact.

The technical version

The identity, and a real balance sheet

Everything above is a restatement of the accounting identity introduced in chapter three:

Assets = Liabilities + Equity

For a bank, the composition of those three terms is unusual in a way no other industry shares: its principal liability is money, and its principal asset is credit. Here is a real one — Lloyds Banking Group plc, statutory consolidated position at 31 December 2025, as reported in the group's 2025 results announcement of 28 January 2026.

Item	31 Dec 2025	31 Dec 2024
Total assets	£944.1bn	£906.7bn
Total liabilities	£896.2bn	£860.8bn
Ordinary shareholders' equity	£41.8bn	£39.5bn
Other equity instruments	£5.9bn	£6.2bn
Non-controlling interests	£0.2bn	£0.2bn
Total equity	**£47.9bn**	**£45.9bn**

Total equity of £47.9 billion against total assets of £944.1 billion is approximately 5.1 per cent — the honest, unweighted answer to "how much of this bank is the owners' money".

On the **liability** side, customer deposits were £496.5 billion — retail current accounts £102.8 billion, retail savings £212.5 billion, Wealth £9.9 billion, Commercial Banking £171.1 billion. Wholesale funding (money markets, debt securities in issue, covered bonds, securitisations, subordinated debt) was £99.4 billion, of which £37.0 billion had a residual maturity under one year. Also here sit derivative liabilities, liabilities arising from insurance and investment contracts — Lloyds owns Scottish Widows — and repurchase agreements.

On the **asset** side, underlying loans and advances to customers were £481.1 billion: UK mortgages £323.1 billion, credit cards £17.3 billion, UK retail unsecured loans £10.5 billion, UK Motor Finance £16.4 billion, Business and Commercial Banking £28.3 billion, Corporate and Institutional Banking £62.0 billion. Financial assets at amortised cost totalled £554 billion; at fair value through profit or loss £240 billion; at fair value through other comprehensive income £36 billion; derivative financial assets £20 billion.

The loan-to-deposit ratio was 97 per cent — the clearest single statement of what this institution is. It has deliberately turned very nearly every pound the public entrusted to it into somebody else's long-dated debt. The expected credit loss allowance of £3,353 million, 0.7 per cent of loans and advances, is the accounting recognition of correction four: the asset side is net of a forecast.

Money creation: the 2014 Quarterly Bulletin

The definitive modern statement is *Money creation in the modern economy*, by Michael McLeay, Amar Radia and Ryland Thomas of the Bank of England's Monetary Analysis Directorate, in the Bank of England Quarterly Bulletin 2014 Q1, published 14 March 2014. Anyone working in payments should read it once. Its central claims, in its own words:

"In the modern economy, most money takes the form of bank deposits. But how those bank deposits are created is often misunderstood: the principal way is through commercial banks making loans."

"Whenever a bank makes a loan, it simultaneously creates a matching deposit in the borrower's bank account, thereby creating new money."

"Rather than banks lending out deposits that are placed with them, the act of lending creates deposits — the reverse of the sequence typically described in textbooks."

And, explicitly refuting both traditional models: banks "do not act simply as intermediaries, lending out deposits that savers place with them, and nor do they 'multiply up' central bank money to create new loans and deposits."

Mechanically, at the moment of origination, the lending bank posts:

	Debit	Credit
Loan asset (customer receivable)	Amount	
Customer deposit (liability)		Amount

Two entries. Balance sheet grows on both sides. Equity unchanged. Nothing has been taken from anyone. On repayment the entries reverse and the deposit ceases to exist — money destruction, which is why aggregate deposits fall when credit contracts even though nobody has withdrawn anything.

The Bank's public explainer gives the composition of UK money as approximately 3 per cent notes and coin, 18 per cent central bank reserves and 79 per cent commercial bank deposits. Reserves and deposits are different instruments circulating in different closed circuits, which is the subject of the next chapter.

The article is equally clear about what does constrain creation. Banks must lend profitably in a competitive market. They must manage the credit and liquidity risk new lending creates. Prudential regulation constrains their activities in the interest of system resilience. And "monetary policy acts as the ultimate limit on money creation" — in normal times through the price of reserves, in unconventional times through asset purchases.

The Basel framework in outline

The rules implementing those constraints are, in outline, international. The Basel Committee on Banking Supervision, hosted at the Bank for International Settlements, comprises 45 members from 28 jurisdictions, consisting of central banks and authorities with formal responsibility for banking supervision. Its standards have **no legal force whatsoever**. They are agreements between supervisors, which member jurisdictions then implement — or do not, or implement late, or differently — through

their own law. In the United Kingdom the binding instrument is the PRA Rulebook, made under the Financial Services and Markets Act 2000 as amended. Anyone who says a British bank is "required by Basel" to do something is speaking loosely; it is required by the PRA, which has chosen to implement Basel.

Capital: the stack

Regulatory capital is a hierarchy ordered by how readily it absorbs losses.

Common Equity Tier 1 (CET1) is ordinary shares, share premium and retained earnings, less regulatory deductions — goodwill and other intangibles, deferred tax assets relying on future profitability, prudent valuation adjustments, the excess of expected losses over accounting provisions, defined benefit pension surpluses and significant investments in financial sector entities. This is the loss-absorbing core.

Additional Tier 1 (AT1) is perpetual, deeply subordinated instruments with no maturity date and fully discretionary coupons, which convert to equity or are written down when CET1 falls through a contractual trigger — the instruments the press calls "cocos". **Tier 2** is dated subordinated debt, absorbing losses only in resolution or liquidation.

Lloyds' regulatory capital position at 31 December 2025 makes the structure concrete:

	31 Dec 2025
CET1 capital	£32,930m
Total Tier 1 capital	£38,053m
Total capital resources	£44,579m
Total MREL resources	£75,732m
Risk-weighted assets	£235,513m
CET1 ratio	14.0%
Tier 1 ratio	16.2%
Total capital ratio	18.9%
MREL ratio	32.2%
UK leverage ratio	5.4%

Risk weighting, and why £944 billion becomes £235 billion

Note the gap between total assets of £944.1 billion and risk-weighted assets of £235.5 billion. Two effects produce it.

First, **risk weighting proper**. Each exposure is multiplied by a weight reflecting its assessed riskiness, either from the standardised approach's look-up tables or from the bank's own internal ratings-based models where the PRA has granted permission. A low loan-to-value residential mortgage attracts a small weight; an unsecured corporate exposure a large one. Cash and central bank reserves attract zero.

Second, **scope of consolidation**. The regulatory consolidation is not the accounting consolidation. Lloyds' insurance business is not consolidated for capital purposes; instead the group deducts £4,708 million of significant investments from CET1. The same effect explains why the leverage exposure measure is smaller than accounting total assets.

Because two banks with identical books could report very different RWAs, the Basel III finalisation package of December 2017 imposed an **output floor**: RWAs must be at least 72.5 per cent of what the

standardised approaches would produce, phasing in from 50 per cent over five years. It also barred the advanced IRB approach for large corporates, banks and equities, imposed input floors (probability of default at five basis points for corporates, loss-given-default at 25 per cent for unsecured corporate exposures), and removed internal models from operational risk and CVA entirely. In the United Kingdom this package is called Basel 3.1; the PRA announced on 17 January 2025 a one-year delay to 1 January 2027, with full implementation still set for 1 January 2030.

The required ratios

Basel III's Pillar 1 minimums are 4.5 per cent CET1, 6 per cent Tier 1 and 8 per cent total capital, all as percentages of risk-weighted assets. On top sit buffers, which are not minimums in the same sense: breaching a buffer does not make a bank unlawful, it triggers automatic restrictions on dividends, buybacks and discretionary bonuses through the maximum distributable amount mechanism.

Layer	Calibration
CET1 minimum	4.5% of RWAs
Tier 1 minimum	6.0% of RWAs
Total capital minimum	8.0% of RWAs
Capital conservation buffer	2.5% of RWAs, CET1 only
Countercyclical capital buffer	0–2.5% (UK rate currently 2%)
Systemic buffers	Firm-specific, for G-SIIs and O-SIIs
Pillar 2A	Firm-specific, set by the PRA for risks not captured in Pillar 1
PRA buffer (Pillar 2B)	Firm-specific, confidential, stress-test driven

The countercyclical buffer is the interesting instrument: the Financial Policy Committee raises it when it judges risk to be building and cuts it in stress, so that capital accumulated in good years can be released rather than forcing banks to shrink lending exactly when the economy needs it.

The leverage ratio

Because risk weighting can be gamed and models can be wrong, Basel III added a deliberately crude backstop: Tier 1 capital divided by a total exposure measure that includes on-balance-sheet assets, derivative exposures, securities financing transactions and off-balance-sheet items converted at prescribed factors, with no risk weighting at all.

The UK framework, set by Financial Policy Committee Direction and implemented in the Leverage Ratio (Capital Requirements and Buffers) Part of the PRA Rulebook, imposes a **minimum of 3.25 per cent** of Tier 1 capital, plus a countercyclical leverage ratio buffer and, for systemic firms, an additional leverage ratio buffer. Following PRA Policy Statement 22/25 of 12 November 2025, the retail deposits threshold for application of the requirement rose from £50 billion to £75 billion with effect from 1 January 2026, calculated on a three-year moving average rather than a point-in-time value; the separate threshold of £10 billion of non-UK assets was left unchanged. Firms below the thresholds face a supervisory expectation, in Supervisory Statement 45/15, that their leverage ratio should not ordinarily fall below 3.25 per cent. Lloyds reported a UK leverage ratio of 5.4 per cent at 31 December 2025.

Liquidity

Capital answers "can this bank absorb losses". Liquidity answers "can this bank pay today". Basel III introduced two standards.

The **Liquidity Coverage Ratio** requires the stock of high-quality liquid assets to be at least 100 per cent of total net cash outflows over a 30-calendar-day stress scenario. HQLA are assets convertible into cash quickly, by sale or by being pledged as collateral, without significant loss of value. Level 1 assets — cash, central bank reserves, certain sovereign debt — count without limit or haircut. Level 2A assets take a 15 per cent haircut. Level 2 assets in aggregate may not exceed 40 per cent of the stock, and Level 2B may not exceed 15 per cent. The standard took effect on 1 January 2015 at 60 per cent, rising ten percentage points a year to 100 per cent on 1 January 2019.

The **Net Stable Funding Ratio** works over a one-year horizon: available stable funding divided by required stable funding, minimum 100 per cent. ASF weights liabilities and capital by how likely they are to stay put — long-dated funding and stable retail deposits score highly, overnight wholesale funding zero. RSF weights assets by illiquidity and by how long they must be funded for. It became a minimum standard for internationally active banks on 1 January 2018.

Lloyds reported an LCR of 145 per cent at 31 December 2025 on LCR-eligible assets of £131.4 billion, and an NSFR of 124 per cent. From those figures the modelled 30-day net outflow can be inferred at roughly £91 billion — about 18 per cent of its customer deposit book. Hold that against Silicon Valley Bank UK, which lost about 32 per cent of its deposits in two days. That is the honest limit of the LCR: it is calibrated to a severe scenario, not the worst imaginable one, and a modern digital run can outrun it.

Why a bank failing is not like a company failing

A car parts manufacturer that becomes insolvent enters administration. An officeholder is appointed, trading may continue, assets are realised, creditors queue in a statutory order, and the process takes months or years. Nobody outside the supply chain much notices on the day.

A bank cannot be treated this way, for one reason: **its liabilities are the money supply and the payments system.** Freezing a bank's balance sheet does not merely disappoint creditors, it deletes the current accounts from which salaries are paid, direct debits collected and card transactions settled. It destroys money and severs access to the payment rails simultaneously, for people who never chose to be creditors of anything.

Britain's answer is the **Special Resolution Regime** in the Banking Act 2009, administered by the Bank of England as resolution authority, which sits alongside and largely displaces ordinary insolvency law for banks, building societies and certain investment firms. It provides five stabilisation options:

1. **Bail-in** — write-down of the claims of the bank's unsecured creditors, including holders of capital instruments, with conversion to equity, recapitalising the firm in place.
2. **Transfer to a private sector purchaser**, executed without the consent of shareholders or counterparties.
3. **Transfer to a bridge bank** controlled by the Bank of England pending an onward sale.
4. **Transfer to an asset management vehicle** to run down a problem portfolio.
5. **Temporary public ownership**, exercised by HM Treasury as a last resort.

Where resolution is not justified — typically a small firm whose failure poses no systemic threat — the **bank insolvency procedure** applies instead: modified insolvency, with a liquidator holding two objectives in strict priority. First, work with the Financial Services Compensation Scheme so that pro-

tected depositors are paid out within seven days or their accounts transferred elsewhere. Only then, wind the firm up in the general creditors' interest.

Depositor preference

Insolvency law itself has been rewritten to put depositors near the front of the queue. Under Schedule 6 of the Insolvency Act 1986 as amended by the Banks and Building Societies (Depositor Preference and Priorities) Order 2014, paragraph 15B makes the portion of an eligible deposit not exceeding FSCS compensation an **ordinary preferential debt** — in practice the FSCS pays the depositor and stands in their place, which is why it is sometimes described as super-preferred. Paragraph 15BA makes the portion of an eligible deposit *exceeding* the compensation limit, owed to eligible persons — broadly individuals and micro, small and medium-sized enterprises — a **secondary preferential debt**. Paragraph 15BB extends secondary preference to deposits made through non-UK branches which would have been eligible had they been made in the UK.

Secondary preferential debts rank after ordinary preferential debts, but both rank ahead of floating charge holders and ordinary unsecured creditors. A large corporate depositor, by contrast, is an ordinary unsecured creditor and ranks with the bondholders.

FSCS cover

The Financial Services Compensation Scheme protects eligible deposits up to **£120,000 per eligible person per banking licence**, a limit that took effect on 1 December 2025, having previously been £85,000. Joint accounts are covered to £240,000. Qualifying temporary high balances — house sale proceeds, redundancy payments, insurance settlements — are protected up to £1.4 million for six months from the date first deposited.

The phrase "per banking licence" is doing real work and is routinely misread. Several familiar high-street brands may share a single authorisation, in which case the limit applies once across all of them, not once per brand.

MREL and loss-absorbing capacity

For resolution to work without public money, there must be enough liabilities available to be written down. That is the **minimum requirement for own funds and eligible liabilities**: own funds plus eligible liabilities, positioned appropriately in the creditor hierarchy, sufficient to absorb losses and recapitalise the firm. Lloyds reported total MREL resources of £75,732 million at 31 December 2025, an MREL ratio of 32.2 per cent of risk-weighted assets — well over twice its total capital ratio, because MREL includes £31,232 million of other eligible liabilities issued by the holding company.

Resolution carries a statutory safeguard: **no creditor worse off**. Following a bail-in or partial transfer, no pre-resolution shareholder or creditor may be left worse off than in an ordinary insolvency. An independent valuation determines this after the event, and compensation is payable if the test is failed.

The regime in action: SVB UK

The clearest worked example in British experience is the resolution of Silicon Valley Bank UK Limited, documented by the Bank of England in its report under section 79A of the Banking Act 2009.

SVB UK operated as a branch from 2012 and converted to a UK subsidiary in July 2022. Its parent was closed by the California Department of Financial Protection and Innovation on 10 March 2023. At 11pm that night the Bank of England announced its intention to place SVB UK into the bank insolvency

procedure. Over the weekend it changed course, determining that a private sector purchaser transfer better served the special resolution objectives. At 7am on Monday 13 March 2023 the Bank exercised its transfer powers and moved the whole of SVB UK's share capital to HSBC UK Bank plc.

SVB UK	Figure
Total assets at 10 March 2023	£8,756m
Total liabilities at 10 March 2023	£7,343m
Net assets	£1,413m
Customer deposits, 8 March 2023	£9,779m
Customer deposits, 10 March 2023	£6,688m
Demand deposits, 8 March 2023	£5,215m
Demand deposits, 10 March 2023	£2,140m
AT1 written down under section 6	£322m
Tier 2 written down under section 6	£33m
Consideration paid by HSBC	£1

The AT1 and Tier 2 instruments were mandatorily written down as required by section 6 of the Act, wiping out £355 million of investors' claims. The equity was transferred for one pound. Every depositor — covered and uncovered alike — retained uninterrupted access to their accounts. No UK public funds were used. Both independent valuations concluded that FSCS-covered depositors would have recovered 100 per cent of their claim in a counterfactual insolvency, so the no-creditor-worse-off test was comfortably satisfied for them.

Read that table as a summary of the chapter. A bank with £8.8 billion of assets and £1.4 billion of net assets — an owners' cushion of over 16 per cent of its balance sheet, far above any regulatory minimum — was resolved over a weekend because roughly a third of its deposits left in two days. Solvency did not save it. What saved its depositors was a statutory regime built on the assumption that this would happen to somebody, eventually.

Ring-fencing

One further piece of British structural regulation belongs here. The Financial Services (Banking Reform) Act 2013, implementing the recommendations of the Independent Commission on Banking, requires the largest UK banking groups to separate core retail banking — deposit-taking from individuals and small businesses, and the payment services attached to it — from investment banking activity, into a legally distinct, separately capitalised, separately governed **ring-fenced body**. The regime came into force on 1 January 2019, by which point approximately £1.2 trillion of core deposits had been placed inside ring-fenced entities. The threshold, set by the Financial Services and Markets Act 2000 (Ring-fenced Bodies and Core Activities) Order 2014, is now **£35 billion of core deposits**, above which a group conducting material investment banking activity must ring-fence, subject to a trading assets condition allowing exit where trading assets are less than 10 per cent of Tier 1 capital. Reforms described by the government as "smarter ring-fencing" came into force on 4 February 2025.

The practical consequence for anyone building payment systems is that "Barclays" and "HSBC" are not single legal entities. A group may contain a ring-fenced bank, a non-ring-fenced bank, an overseas subsidiary and a service company, each with its own authorisation, balance sheet and, in some cases, scheme participation. Which entity a payment is with has a real answer and real consequences.

What to carry forward

Your balance is the bank's liability and the bank's loans are its assets. Every payment message in Volumes III and IV is an instruction to move a liability from one institution's book to another's, and the direction of every posting depends on remembering whose book you are in.

Lending creates deposits; deposits are not lent out. This is why the money supply expands and contracts with credit, and why "where did the money come from" is usually the wrong question to ask about a payment.

Capital protects against loss and liquidity against withdrawal, and they are entirely different defences. A payments business — an authorised payment institution, an e-money firm, a fintech — is not a bank and has neither regime, which is precisely why its customers' funds must be safeguarded in a segregated account rather than lent.

And commercial bank money is not the only kind. When Priya's £60,000 walks out of Little Sutton to Barclays, something else has to move — something no commercial bank can create. That something is central bank money, and it is the subject of the next chapter.

5.98 Common wrong ideas

Wrong: A bank keeps your money in a vault and lends out a fraction of it. **Right:** A bank has already spent what you gave it; your balance is a promise, and lending creates new deposits rather than passing on old ones.

Wrong: British banks must hold ten per cent of deposits in reserve. **Right:** The United Kingdom has had no mandatory reserve ratio since 1981, and the Federal Reserve reduced all its ratios to zero per cent on 26 March 2020.

Wrong: Because lending creates deposits, a bank can lend without limit. **Right:** The deposits an individual bank creates walk out to other banks and must be settled in reserves it cannot create, and capital, funding and the regulator bind long before anything else does.

Wrong: Capital is a pot of money the bank sets aside for a rainy day. **Right:** Capital is the accounting difference between two other numbers, and requiring more of it means funding yourself with more shareholder money and less borrowed money.

Wrong: A well-capitalised bank cannot fail. **Right:** Capital absorbs losses and liquidity meets withdrawals; Silicon Valley Bank's capital ratios were above the required minimums on the day it was closed.

Wrong: A capital ratio of fourteen per cent means fourteen per cent of the balance sheet is the owners' money. **Right:** Ratios are struck against risk-weighted assets, which is why Lloyds could report 14.0 per cent while £47.9 billion of equity against £944.1 billion of assets is about 5.1 per cent.

Wrong: Basel rules bind British banks directly. **Right:** Basel standards have no legal force whatsoever; what binds a UK firm is the PRA Rulebook, made under the Financial Services and Markets Act 2000.

Wrong: FSCS cover of £120,000 applies to each of your bank's brands. **Right:** It applies per eligible person per banking licence, and several familiar high-street brands may share a single authorisation.

Wrong: A failing bank goes into administration like any other company. **Right:** Its liabilities are the money supply and the payments system, so the Special Resolution Regime in the Banking Act 2009 largely displaces ordinary insolvency law.

Wrong: The balance sheet tells you what a bank owns. **Right:** It is a considered opinion presented in the grammar of fact, net of IFRS 9 expected credit losses, silent on undrawn commitments and silent on the maturity mismatch that makes the business fragile.

5.99 Chapter summary in 20 lines

1. A bank is not a place where money is kept but an institution that has spent what you gave it and is obliged to return it the moment you ask.
2. Your balance is a record of a debt the bank owes you, not a claim on any particular notes.
3. Every bank keeps two lists: what it owns, called assets, and what it owes, called liabilities.
4. The difference between them is equity, the owners' cushion, and its whole job is to absorb losses so that depositors never notice.
5. Taking a deposit makes a bank bigger rather than richer, because the cash coming in and the promise going out are two entries of the same size.
6. When a bank lends it types a number into the borrower's account, creating an asset and a liability at the same instant and leaving its equity untouched.
7. Lending therefore creates deposits and repayment destroys them, which is the reverse of the sequence taught in most textbooks.
8. The Bank of England said exactly that in *Money creation in the modern economy* in the 2014 Q1 Quarterly Bulletin.
9. Around seventy-nine per cent of UK money is commercial bank deposits, eighteen per cent central bank reserves and about three per cent notes and coin.
10. There is no reserve ratio in the United Kingdom and no money multiplier, because reserves are supplied on demand to keep the policy rate where the central bank wants it.
11. An individual bank is nonetheless tightly constrained, since the deposits it creates leave for other banks and must be settled in reserves it cannot create.
12. Capital is scarce and expensive and must scale with the risk taken, and that, not any reserve ratio, is the operative brake on lending.
13. Capital is not cash but an accounting difference, and capital ratios are measured against risk-weighted assets rather than actual ones.
14. That is why Lloyds reported a CET1 ratio of 14.0 per cent against £235.5 billion of risk-weighted assets while its equity was about 5.1 per cent of £944.1 billion of actual assets.
15. Capital protects against loss and liquidity protects against withdrawal, and they are separate defences against separate deaths.
16. Northern Rock in 2007 and Silicon Valley Bank in 2023 both died of the second, the latter's British subsidiary losing about a third of its deposits in two days.
17. A bank cannot be wound up like a car parts manufacturer, because its liabilities are the money supply and the payment rails at the same time.
18. Britain's answer is the Special Resolution Regime of the Banking Act 2009, offering bail-in, private sector transfer, a bridge bank, an asset management vehicle and temporary public ownership.
19. SVB UK showed the regime working: £355 million of AT1 and Tier 2 written down, the share capital transferred to HSBC for one pound, and every depositor's access unbroken with no public funds used.
20. Carry forward that your deposit is the bank's liability and its loans are its assets, and that when money leaves one bank for another something no commercial bank can create has to move.

Chapter sources: McLeay, Radia and Thomas, "Money creation in the modern economy", Bank of England Quarterly Bulletin 2014 Q1; Bank of England explainer "How is money created?"; Bank of England, "The Bank

of England's approach to resolution" (2023), the section 79A report on the transfer of SVB UK to HSBC UK, the ring-fencing pages and CP10/26, and the countercyclical capital buffer pages; PRA CP2/25 and PS22/25 on the leverage ratio retail deposits threshold, PS1/26 on Basel 3.1 final rules, and the news release of 17 January 2025 on the Basel 3.1 delay; Basel Committee on Banking Supervision membership pages and the December 2017 "Basel III: Finalising post-crisis reforms" high-level summary; Financial Stability Institute executive summaries of the LCR and NSFR; Lloyds Banking Group plc 2025 results announcement, 28 January 2026; Federal Reserve Board reserve requirements page and its 28 April 2023 review of the supervision and regulation of Silicon Valley Bank; Deutsche Bundesbank and ECB on minimum reserves, including the Governing Council decision of 27 July 2023; Insolvency Act 1986 Schedule 6 as amended and the Banks and Building Societies (Depositor Preference and Priorities) Order 2014, legislation.gov.uk; Financial Services Compensation Scheme cover pages; Banking Act 2009; Financial Services (Banking Reform) Act 2013.

Two Kinds of Money

6.0 What this chapter gives you

1. You will be able to explain why a payment between two customers of the same bank needs no cooperation from anybody, while an identical-looking payment to another bank eventually requires the Bank of England to change a number in a ledger of its own.
2. You will be able to name the two kinds of sterling — central bank money and commercial bank money — and say exactly who may hold each of them.
3. You will be able to work a day's multilateral net position out of a list of customer payments, and explain why £2,245 of payments moved £450 in the upper ledger.
4. You will be able to explain why drawing £20 from a cash machine converts one kind of money into the other, and what the singleness of money means.
5. You will be able to say why reserves are not depositors' money in storage, and why the total fell by £63 billion in a year without anybody being paid a penny.
6. You will be able to decide whether a payment is genuinely on-us by asking which legal entity holds each account rather than which brand is on the app.
7. You will be able to distinguish the four account types at the Bank of England and say which are remunerated at Bank Rate and which may not hold funds overnight.
8. You will be able to explain why a Faster Payment landing at two on a Sunday morning leaves one bank exposed to another, and what prefunding does about it.
9. You will be able to describe tiering, and say why an indirect participant's settlement asset is commercial bank money rather than central bank money.
10. You will be able to answer "why is this payment behaving perversely" by asking which tier the obligation is sitting in and what has to happen before it moves.

The previous chapter established what a bank is: an institution whose fundamental product is a promise, and whose balance sheet records your deposit as its debt. This chapter takes that one step further and asks an awkward question. If your bank's promise is money to you, whose promise is money to your bank?

The question is not philosophical. It has a specific, mechanical, unromantic answer, and that answer is the hinge on which the rest of this book turns. Almost every rule about settlement that arrives later — why a CHAPS payment cannot be recalled, why Bacs takes three working days, why a Faster Payment reaches the recipient in seconds but the banks square up hours later, why a card refund is slow, why an international transfer travels through institutions nobody asked for — is a consequence of the fact that there are two distinct kinds of money in circulation at the same time, in the same currency, denominated in the same pounds, and that they are not interchangeable in the way the word "pounds" implies.

Consider two payments that look identical from the outside. Priya banks with Lloyds and pays £120 to Sam, who also banks with Lloyds. Priya's neighbour banks with Lloyds and pays £120 to a friend at Barclays. Both payments are for £120, both are sent from a phone, both arrive in seconds, both produce identical-looking lines on identical-looking statements. But the first payment never leaves Lloyds and requires no cooperation from any other institution in Britain. The second cannot be completed by Lloyds alone at any price, and eventually requires the Bank of England to change a number in a ledger of its own.

That asymmetry is invisible to customers, is largely invisible to bank staff, and governs everything. This chapter makes it visible.

The plain version

Imagine a town with two banks. One is run by Ada, the other by Mercer. Between them they hold the accounts of everybody in town.

Ada's customers include Priya, Sam and Tom. Mercer's customers include Ravi and Nadia. Each bank keeps a book with one page per customer, showing how much the bank owes that person. That is what a bank balance is: a written record of a debt owed to you. Nobody has a personal pile of coins in a personal box. There is a book, and there is a line in it with your name on.

The easy payment

On Monday, Priya pays Sam £120. Both of them are Ada's customers.

Ada does not need to consult anybody. She opens her book, reduces Priya's line by £120 and increases Sam's line by £120. That is the whole event. Nothing has left Ada's building. No cash has moved, no van has driven anywhere, no message has been sent to any other institution. Ada owed £120 to Priya before, and she owes it to Sam now. Her total debt to her customers is exactly the same as it was five minutes ago. What changed is the label on it.

This is worth stopping on because it is genuinely surprising the first time you see it clearly. From Ada's point of view, a payment between two of her own customers is not a transfer of anything. It is a relabelling exercise. She has rewritten who she owes, and that is all.

The hard payment

On Tuesday, Tom pays £800 to Ravi. Tom is Ada's customer. Ravi is Mercer's.

Ada can do her half easily. She reduces Tom's line by £800, and her total debt to her customers falls by £800. She is £800 better off.

Mercer can do his half easily too. He increases Ravi's line by £800, and his total debt to his customers rises by £800. He is £800 worse off.

And that is the problem. Mercer has just taken on a new obligation to Ravi in exchange for nothing at all. Ada has just been released from an obligation, also in exchange for nothing at all. If the story stops here, Ada has been enriched by £800 at Mercer's expense, and Mercer would be an idiot to let it happen again.

So Ada has to give Mercer something worth £800. The obvious candidate is a promise: Ada writes Mercer an IOU. But Mercer does not want Ada's IOU. Ada might be perfectly sound today, but if she is unsound in six months Mercer is left holding a piece of paper and a real debt to Ravi. If the town's banks paid each other in IOUs, every bank would end up quietly lending to every other bank all day long, without ever deciding to, and a single failure would take the lot down.

What Mercer wants is money that does not depend on Ada being solvent.

The vault in the market square

So the town builds one. In the market square there is an institution — call it the Old Vault — which is not a bank for people. It is a bank for banks. Ada has an account there. Mercer has an account there. So does every other bank in town, and nobody else.

Now Tuesday's payment can complete. Ada tells the Old Vault to move £800 from her account to Mercer's. The Vault reduces Ada's line by £800 and increases Mercer's by £800. Mercer has been paid in something he trusts, and he is content to owe Ravi the £800, because he has received £800 of something better than an IOU from Ada.

That is the two-tier system, in one picture. There are two ledgers, not one. Down at street level, banks keep books recording what they owe their customers. Up above, the Vault keeps a book recording what it owes the banks. Ordinary payments between customers of different banks cause changes in both books at once, and the payment is not really finished until the upper book has been changed.

The two kinds, named

There are therefore two kinds of money in this town, and they are not the same substance even though they are counted in the same pounds.

The first kind is what you and I have. A balance at Ada's Bank is a promise from Ada. It is only as good as Ada. If Ada fails, you are a creditor of a failed business.

The second kind is what banks have. A balance at the Old Vault is a promise from the Vault. Banks accept it from one another without hesitation for three reasons: the Vault is not trying to make money out of them, the Vault cannot run out of the town's currency because the Vault is the thing that issues it, and the Vault is not going to go bust.

In the real United Kingdom the Old Vault is the Bank of England, and the accounts banks hold there are called reserves accounts. The money in them is called central bank money. The money in your account is called commercial bank money. Once you have those two phrases, an enormous amount of otherwise baffling behaviour becomes obvious.

One thing ordinary people can hold

There is exactly one form of central bank money that a member of the public can hold, and it is in your wallet. A £20 note is a direct promise from the Bank of England. It is not a promise from Barclays or Nationwide or anybody else. As of the Bank's most recent published figures there are over 4.98 billion Bank of England notes in circulation, together worth about £91.5 billion.

So when you draw £20 from a cash machine, something genuinely interesting happens: you convert £20 of commercial bank money into £20 of central bank money. Your claim on your bank shrinks by £20 and you acquire a claim on the central bank instead. It does not feel like a change of substance because the two spend identically, and the fact that they spend identically — that nobody in any shop in Britain asks whether your £20 is a Lloyds £20 or a paper £20 — is the single greatest achievement of the whole arrangement. It is called the singleness of money, and it is not an accident. It is engineered, and the thing that engineers it is the shared ledger in the middle.

A day, with numbers

Take one full day in the town, so the arithmetic is concrete.

At the start of Wednesday, Ada holds £45,000 at the Old Vault and Mercer holds £60,000. Five payments happen.

Payer	Payer's bank	Payee	Payee's bank	Amount
Priya	Ada	Sam	Ada	£120
Tom	Ada	Ravi	Mercer	£800
Ravi	Mercer	Priya	Ada	£250
Nadia	Mercer	Tom	Ada	£1,000
Sam	Ada	Priya	Ada	£75

Five customers have had their balances changed. In total, £2,245 of payments have been made. Every payer's line went down and every payee's line went up by the full amount, exactly as they would expect.

Now look at the Old Vault's book. Two of those five payments — Priya to Sam, and Sam to Priya — were between customers of the same bank. The Old Vault never heard about them. They do not exist as far as the upper ledger is concerned. That leaves three payments that cross between banks: £800 from Ada's side to Mercer's, and £250 plus £1,000 coming back the other way.

Ada owes Mercer £800. Mercer owes Ada £1,250. There is no reason to move both amounts, so the Vault moves the difference: £450 from Mercer to Ada. Ada finishes the day with £45,450 and Mercer with £59,550. The Vault's total is unchanged at £105,000, because nothing was created or destroyed up there — one bank's pile grew and another's shrank by the same amount.

Look at what that means. Five payments, £2,245 of customer activity, and the number that actually changed hands in the upper ledger was £450, once. The rest either cancelled out or never left the bank it started in.

That single observation explains more about how payment systems are built than anything else in this book. It is why banks care so much about which of their customers pay each other. It is why the machinery that works out the £450 exists at all. And it is why, when things go wrong, the question that matters is never "did the customer's balance change" but "did the upper ledger change, and when".

Where the plain version stops being true

The vault-in-the-square picture is the right mental model, and practitioners carry a version of it in their heads. But four things about it are wrong in ways that will bite.

The reserves are not your money in storage

The picture invites you to imagine that Ada's £45,000 at the Vault is the town's real money, kept safe on behalf of Ada's depositors, and that Ada's deposits are somehow backed by it. That is not what reserves are, and the error matters because it produces confident nonsense about banking.

Reserves are a liability of the central bank, created by the central bank, in whatever quantity the central bank chooses. They are not a warehouse receipt for anything. There is no ratio that must hold between a bank's reserves and its deposits — the United Kingdom has had no mandatory reserve ratio for decades. The two figures are not even the same order of magnitude. At end-February 2026, total balances held in reserves accounts at the Bank of England stood at £643,547 million, while the

UK's broad money measure M4, which is overwhelmingly commercial bank deposits, was running at around £3.3 trillion. Reserves cover something in the region of a fifth of deposits, and that fraction is not a design parameter. It is a residue of monetary policy.

You can see this directly. Reserves balances fell from £706,745 million at end-February 2025 to £643,547 million a year later. No bank paid that £63 billion to anybody. It disappeared because the Bank of England was shrinking its own balance sheet through the sale and maturity of assets held in the Asset Purchase Facility. The upper ledger's total is not a fact about the banks. It is a fact about the central bank.

The other half of the correction: reserves cannot leave the upper tier. A bank cannot lend reserves to you. Reserves circulate only among institutions that hold accounts at the central bank, plus a leak into banknotes when the public wants cash. When a bank makes a loan it creates a deposit — commercial bank money — out of nothing, as the Bank of England's own *Money creation in the modern economy* set out in its 2014 Quarterly Bulletin. It does not hand over reserves. Reserves are what it needs afterwards, if and when the borrower spends the money at a customer of a different bank.

"Same bank" is a legal question, not a branding question

The plain version says a payment between two customers of the same bank never leaves that bank. That is true, and it is the most useful sentence in the chapter. But "the same bank" means the same legal entity, and British banking brands routinely span several.

Since ring-fencing, the large UK groups are split. Barclays operates both Barclays Bank UK PLC and Barclays Bank PLC as separately authorised entities. HSBC operates HSBC UK Bank plc alongside HSBC Bank plc. Lloyds Banking Group contains Lloyds Bank plc, Bank of Scotland plc and Lloyds Bank Corporate Markets plc. A payment from a retail customer of one entity to a corporate customer of another entity in the same group is not an on-us payment. It has an interbank obligation in it, and it settles like any other.

The second half of this correction is about risk rather than plumbing. Even a genuine on-us payment does something real: it moves the exposure. Before Priya pays Sam, Priya is the bank's creditor for £120. Afterwards, Sam is. If the bank fails, it matters enormously which of them is holding the claim, because deposit protection is per depositor, per authorised firm. In the United Kingdom that limit rose from £85,000 — the level set in 2017 — to £120,000 on 1 December 2025, with the temporary high balance limit for qualifying life events rising from £1 million to £1.4 million on the same date. An on-us payment that pushes a customer from £90,000 to £150,000 has quietly moved £30,000 out of protection. Nothing settled, nothing cleared, and the risk profile of the bank's deposit book changed.

Nothing moves at the Old Vault when your payment arrives

The picture implies that a cross-bank payment causes an immediate transfer in the upper ledger. For one UK payment system that is true. For the systems that carry almost all the volume, it is false, and the gap is where a whole category of risk lives.

CHAPS settles gross and in real time: each payment moves central bank money individually, when it is made. Everything else settles net and later. Faster Payments net settlement takes place three times each business day. Bacs settles once a day. The Image Clearing System for cheques settles once a day. Visa Europe and Mastercard Europe settle on twenty-four-hour cycles, with weekend and bank holiday cycles settling the following business day. PEXA settles six times a business day. LINK settles daily.

So when a Faster Payment lands in a stranger's account at two in the morning on a Sunday, the recipient's bank has credited a real customer with real, spendable money while holding nothing but an obligation from the sending bank, and it will hold that obligation until the next settlement cycle. That interval is a credit exposure of one bank to another, created by a customer, without either bank's credit committee being consulted. The Bank of England's answer is prefunding: it offers a service that lets a payment system operator cap each participant's maximum possible net debit position at the amount that participant has locked in a segregated prefunding account in RTGS. If the participant fails, the cash is already there and settlement completes anyway.

The general rule to carry forward is this. A customer payment and its settlement are two different events with two different timestamps. Assuming they are the same event is the single most common error in payments engineering, and it is the reason Chapter 7 exists.

There is not one account per bank, and most institutions have none

The picture has a tidy one-to-one mapping: one bank, one account at the Vault. Reality has four account types with different rights, and most regulated firms have none of them.

The Bank of England distinguishes reserves accounts, reserves-and-settlement accounts, settlement accounts, and omnibus accounts. They differ on who may hold them, whether they are remunerated, and whether balances may sit overnight. A settlement account held by a non-bank payment service provider is unremunerated and may not hold funds beyond what settlement requires. A reserves account is remunerated at Bank Rate and may hold balances overnight. That distinction is not administrative trivia; it decides whether an institution earns anything on its central bank balances, which is a first-order fact about its business model.

And the population is small. The Prudential Regulation Authority regulates around 1,300 firms. At end-February 2026, 221 participants had access to Sterling Monetary Framework facilities. More than 70 organisations use RTGS to settle directly in one or more payment systems. Just over 35 organisations access CHAPS directly. Several thousand financial institutions reach CHAPS indirectly, by sending their payments through one of those direct participants.

For all of those indirect institutions, the two-tier picture repeats one level down, and the top tier is not the central bank. A small firm that clears through Barclays settles in a Barclays deposit. Its settlement asset is commercial bank money. It has the same exposure to Barclays that Priya has, and it has it in size. This is called tiering, it is a standing concern of financial stability regulators, and it is the mechanism behind correspondent banking and the nostro and vostro accounts that Volume IV takes apart.

The technical version

Definitions

Central bank money is a liability of the central bank. In sterling it takes two forms: Bank of England banknotes, which are bearer liabilities available to the general public, and balances held in accounts at the Bank of England, which are book-entry liabilities available only to institutions that meet the Bank's access criteria. Central bank money is the settlement asset for sterling: it is what an institution receives when a sterling obligation to it is finally discharged.

Commercial bank money is a deposit liability of a commercial bank, building society or credit union — an unsecured, unsubordinated claim on that institution, ranking as a depositor claim in insolvency and covered, up to the applicable limit, by the Financial Services Compensation Scheme. It is the

money the public holds, the money businesses hold, and by value it is the overwhelming majority of the money in existence.

The two-tier system is the arrangement in which the central bank provides accounts to a restricted set of institutions, those institutions provide accounts to the public, and the public's money is therefore a claim on a private firm while the private firms' money is a claim on the state. The Bank for International Settlements described this in its September 2025 report on wholesale central bank money as a "time-tested two-tier structure", and identified the central bank settlement asset as what preserves the *singleness* of money: the property that payments denominated in the sovereign unit of account settle at par regardless of which form of money is used. The foundational treatment remains the Committee on Payment and Settlement Systems report *The role of central bank money in payment systems*, published on 12 August 2003.

	Central bank money	Commercial bank money
Issuer	Bank of England	Authorised deposit-taker
Forms	Banknotes; reserves and settlement balances	Current and deposit account balances
Who may hold it	Public (notes only); eligible institutions (balances)	Anyone the bank will onboard
Credit risk	Effectively none in sterling	Full unsecured exposure to the issuer
Protection on failure	Not applicable	FSCS, £120,000 per depositor per firm since 1 December 2025
Approximate sterling stock	£91.5bn notes; £643.5bn reserves at end-February 2026	M4 around £3.3 trillion
Remuneration	Reserves at Bank Rate; notes zero	Set by the bank

Accounts at the Bank of England

Access is governed by the Bank's published access policy for RTGS settlement accounts and services. Four account types exist.

A **reserves account** is available to PRA-authorised UK-incorporated banks, building societies and investment firms designated by the PRA for prudential supervision, to UK subsidiaries and branches of their non-UK incorporated equivalents, and to central counterparties and international central securities depositories. Balances are remunerated at Bank Rate — 3.75% at end-February 2026, having fallen from 4.5% over the preceding twelve months — and may be held overnight. Holding a reserves account requires participation in the Sterling Monetary Framework.

A **reserves and settlement account** is the same account used for both purposes: holding funds and settling obligations arising from direct participation in a payment system that settles in RTGS.

A **settlement account** is for firms not eligible for reserves: FCA-authorised non-bank payment service providers, systemically important financial market infrastructures, and entities within a wider banking group that already holds a reserves account. Settlement accounts are unremunerated and may not hold funds beyond what is needed to support settlement. The Bank opened this route to non-bank PSPs in 2017, making it the first G7 central bank to do so.

An **omnibus account**, offered since 2021, allows the operator of a payment system recognised by HM Treasury under the Banking Act 2009, and designated under the Financial Markets and Insolvency (Settlement Finality) Regulations 1999, to co-mingle reserves belonging to multiple participants in

a single account, so that balances in the operator's own ledger are backed by central bank money. Omnibus accounts are remunerated at Bank Rate. The first user was the Sterling Fnality Payment System, which processed its first live payments on 14 December 2023.

RTGS: the ledger that holds the upper tier

The Bank of England's Real-Time Gross Settlement infrastructure was introduced in 1996. It is, in the Bank's own words, "an accounting system" — the ledger that underpins settlement of sterling payments, rather than a payment system in its own right. Responsibility for operating CHAPS, the UK's high-value payment system, transferred to the Bank in November 2017, so that the Bank now acts as both settlement agent and system operator. On 28 April 2025 the Bank completed a multi-year renewal programme, moving to a new core ledger and settlement engine known as RT2; the participant-facing interface Enquiry Link was replaced by BERTI at the same time.

RTGS provides settlement for CHAPS, for the payment system embedded within CREST, for seven retail payment systems — Bacs, the Image Clearing System, Faster Payments, LINK, Mastercard Europe, PEXA and Visa Europe — and for the Bank's Note Circulation Scheme. It supports four settlement models: real-time gross settlement, delivery versus payment, prefunded deferred net settlement and standard deferred net settlement.

Average daily settlement in RTGS, Q2 2026:

Settlement service	Daily average value (£mn)
CHAPS	411,562
CREST DvP	477,841
Bacs (net)	6,205
Visa Europe (net)	2,356
Faster Payments (net)	2,298
Mastercard Europe (net)	1,480
LINK (net)	238
Cheque imaging (net)	37
PEXA (net)	1
Total	**902,018**

Two things in that table deserve attention. The first is the gulf between CHAPS at £411.6 billion a day and Faster Payments at £2.3 billion a day, when Faster Payments carries vastly more transactions. That is netting, and Chapter 8 is about it. The second is that the entire retail payments experience of sixty-odd million people — every Faster Payment, every direct debit, every card transaction settling through Visa Europe or Mastercard Europe, every cash machine withdrawal — reduces to about £12.6 billion of central bank money changing hands each day, roughly three per cent of the RTGS total.

The on-us case, stated exactly

An intrabank payment — variously called an on-us payment, an internalised payment, or a book transfer — is a payment in which the payer's account and the payee's account are held by the same legal entity.

The accounting is a single pair of postings within that institution's own ledger. The payer's account is debited and the payee's account is credited. Both are liability accounts. The bank's balance sheet total is unchanged; only the composition of its deposit liabilities changes. There is no interbank obligation, and therefore nothing to clear, nothing to net and nothing to settle. No reserves move. No message

reaches the Bank of England. The payment is complete at the moment the bank's ledger says it is complete, and its finality is a matter of that bank's own terms and systems rather than of any scheme rulebook or settlement finality designation.

Whether the bank routes such a payment through the relevant scheme's central infrastructure anyway — for addressing, for Confirmation of Payee, for tracking, or simply because it is easier to have one code path — is an implementation choice rather than a settlement necessity, and the Payment Systems Regulator says as much in its reporting guidance on authorised push payment scams. That guidance defines an on-us transaction as a push payment between two account holders within the same provider group, and records that, depending on a provider's commercial arrangements, such payments may pass either externally via Faster Payments or internally as a book transfer. Firms are required to report both, and to itemise the internally booked ones separately so that the regulator can check its totals against Pay.UK's own Faster Payments figures. The settlement consequence, however, is not in doubt: an on-us payment generates no interbank obligation, and no quantity of central bank money moves as a result of it.

The best evidence that this is a material phenomenon rather than a curiosity is that the Bank of England's own published statistics have to carve it out. The Bank states that CHAPS represents 0.4% of UK total payment volumes but 91% of total sterling payment values, and appends the qualification "excluding internalised flows within payment service providers". The official numbers exclude on-us traffic because it never touches the settlement layer and therefore cannot be observed there. A payment that does not settle is, from the settlement system's point of view, a payment that did not happen.

For a practitioner this has consequences beyond accounting. On-us payments do not attract interbank settlement obligations or, in card schemes, interchange between different institutions. They are not visible in scheme-level reconciliation files. They may be subject to different internal cut-offs, different limits and different fraud controls from their interbank equivalents. And in a large bank with a high market share, they can be a very large fraction of volume — which is exactly why a payments platform must be designed so that the on-us and interbank paths cannot diverge in behaviour that customers can see.

The interbank case, stated exactly

When payer and payee bank with different institutions, the payment creates an obligation between those institutions that must be discharged in the settlement asset. There are two ways to do it.

Real-time gross settlement. Each payment is settled individually, in central bank money, at the moment it is processed. In sterling this model is used only by CHAPS. Settlement is final and irrevocable between direct participants as it occurs, which is why CHAPS is the instrument of choice for house purchases and for the sterling leg of transactions that must not unwind: CLS, the foreign exchange settlement system, completes its final sterling settlement through direct participation in CHAPS, and LCH and Euroclear Bank are also direct participants.

The CHAPS numbers for calendar year 2025: volumes grew 1.1% to a record 53.3 million payments, an average of 210,482 each working day; total value settled rose 7.3% to £93.9 trillion, an average of £371.3 billion daily. The average payment value was £1.8 million and the median was £4,586 — a spread that tells you a great deal, since 94% of CHAPS payments were for £1 million or less while financial institution payments, carried in the pacs.009 message, were 24% of volume but 73% of value. The Bank observes that CHAPS settles the annual UK GDP every nine working days. The all-time peak volume day was 2 April 2024 with 344,099 payments; the all-time peak value day was 3 October 2022 with £642.7 billion.

The CHAPS settlement day currently runs 06:00 to 18:00, Monday to Friday. Direct participants must be able to receive by 08:00 and to send by 10:00; the cut-off for customer payments is 17:40. Participants choose whether to submit a payment as urgent, settling immediately subject to available liquidity, or non-urgent, accepting some delay in exchange for better liquidity efficiency through the Liquidity Saving Mechanism; non-urgent payments settle around eight minutes after submission on average, and are auto-promoted to urgent after 17:30. From September 2027 the Bank will open CHAPS settlement from 01:30 rather than 06:00, and it has consulted on weekend settlement not before 2029 and a 22-hours-by-6-days pattern not before 2031, with 22x7 and 23.5x7 identified as possible end states.

Deferred net settlement. The system operator collects the individual payment instructions, calculates each participant's multilateral net position at the end of a cycle, and instructs the Bank of England to settle a single figure per participant. This is dramatically cheaper in liquidity than gross settlement, and it is what makes high-volume, low-value payments economic. The cost is that customers are credited and debited before interbank settlement occurs, so participants carry an intraday exposure to one another. Prefunding removes that exposure by capping each participant's maximum net debit at cash it has already segregated in RTGS.

The legal layer

Two statutory mechanisms turn these arrangements from private contracts into something that survives an insolvency.

Recognition under the Banking Act 2009 brings a payment system within the Bank of England's supervisory remit, on a recognition order made by HM Treasury. Designation under the Financial Markets and Insolvency (Settlement Finality) Regulations 1999 protects transfer orders and netting arrangements within a designated system from being unwound by ordinary insolvency law — so that a settlement which has occurred stays occurred even if a participant enters administration later the same day. Both are prerequisites for an omnibus account, and both are the reason the phrase "settled in central bank money" carries legal weight rather than merely describing where the entry was made.

The same structure elsewhere

The two-tier arrangement is not a British peculiarity. Every major currency is organised this way, with local variations in access.

In the United States, the Federal Reserve Banks provide master accounts to legally eligible depository institutions, which are the gateway to Fedwire Funds, Fedwire Securities, the National Settlement Service, FedACH, FedCash and FedNow. Deposits of depository institutions at the Reserve Banks stood at $2,947,630 million on the H.4.1 release of 13 August 2026. In May 2026 the Federal Reserve proposed a new, more limited "payment account" as a route for institutions that do not qualify for a full master account.

In the euro area, the Eurosystem operates TARGET Services. In 2025, T2 settled a daily average of 431,067 euro payments worth €1,932.8 billion, a total of €492,859.0 billion for the year, with an average transaction value of €4.5 million; the ECB notes that T2 settles the equivalent of euro-area GDP in around eight days of operations. The instant payment service TIPS settled a daily average of 2,735,053 euro payments worth €2.2 billion — the same inversion of volume and value that Faster Payments shows against CHAPS.

Central bank digital currency, briefly

A retail central bank digital currency would be central bank money held directly by households and businesses in electronic form. That is the entire proposition, and it explains why the topic generates the heat it does: it would be the first genuinely new form of tier-one money available to the general public since the banknote, and it would change the answer to the question this chapter is built on — who may hold a claim on the central bank.

The Bank of England and HM Treasury are in a design phase for a digital pound, examining technology and policy requirements. No decision to build or issue has been taken. The Bank has said the earliest it would issue a digital pound is the second half of this decade, that a decision on next steps would come in 2026, and that Parliament would have a say before any launch. It has also said that a digital pound would not replace cash, and that the Bank will continue to issue banknotes for as long as people want to use them. A Digital Pound Lab for practical experimentation was announced in October 2025.

The arguments are worth stating without taking sides. Proponents point to declining cash use, to the desirability of a public payment option that does not depend on any commercial firm, and to the anchoring role of a monetary asset available to everyone at par. Opponents and cautious central bankers point to disintermediation — if households can hold central bank money directly, deposits may leave banks, particularly in a stress — and to privacy, which is why most design work assumes holding limits and an intermediated model in which private firms, not the central bank, hold the customer relationship. These are open questions and this book does not pretend to settle them.

On the wholesale side, the framing is different, because wholesale central bank money already exists and is already digital: that is precisely what a reserves balance is. The live work there is about the form it takes rather than its existence — whether reserves can be represented on programmable ledgers so that a securities transfer and its cash leg settle atomically. Omnibus accounts are one answer already in production; the BIS report of 18 September 2025, produced with seven central banks, is the reference for the rest.

Why this chapter governs the rest of the book

Almost every settlement rule that follows is an answer to two questions: in whose money, and when.

CHAPS is irrevocable because it settles in central bank money at the moment of processing, and there is no mechanism by which a completed transfer of the settlement asset can be reversed by one party's regret. Bacs takes three working days because it is a deferred net batch system whose cycle was designed around the exchange of files, and the customer-visible timing is inherited from the settlement design rather than from any technical necessity. Faster Payments has a value cap because the recipient's bank extends credit until the next of the three daily settlement cycles, and that exposure has to be bounded. Card refunds are slow because the refund is a new clearing item that must travel back through the scheme and settle in its own cycle, not a reversal of the original. International transfers acquire a chain of intermediaries because there is no single central bank whose money both ends can hold, so the chain is a sequence of commercial bank claims stitched together — the two-tier problem repeated at each hop. Safeguarding rules exist for e-money and payment institutions precisely because their customers' balances are not deposits and their firms are not banks, so the money must be held somewhere that survives the firm.

None of those rules is arbitrary. Each is a consequence of a decision about which kind of money discharges an obligation and at what moment. Whenever a payment behaves in a way that seems perverse, the productive question is not "why is this system badly designed" but "which tier is this obligation sitting in right now, and what has to happen before it moves".

6.98 Common wrong ideas

Wrong: Reserves are depositors' money kept safe at the Bank of England. **Right:** Reserves are a liability of the central bank, created in whatever quantity it chooses, and no ratio has to hold between them and anybody's deposits.

Wrong: Banks lend out reserves. **Right:** Reserves cannot leave the upper tier; a bank lends by creating a deposit and needs reserves only afterwards, if the borrower pays a customer of a different bank.

Wrong: Two accounts showing the same brand are with the same bank. **Right:** Barclays, HSBC and Lloyds each run several separately authorised entities, and a payment between two of them carries a real interbank obligation.

Wrong: An on-us payment has no consequences because nothing settles. **Right:** It moves the exposure, and it can quietly push a customer past the £120,000 protection limit without anything clearing or settling at all.

Wrong: A cross-bank payment moves central bank money at the moment it arrives. **Right:** Only CHAPS settles that way; Faster Payments settles three times each business day, and Bacs, cheques and the card schemes once.

Wrong: Every bank has an account at the Bank of England. **Right:** The PRA regulates around 1,300 firms, just over 35 organisations access CHAPS directly, and everybody else reaches it through somebody's balance sheet.

Wrong: All Bank of England accounts are the same. **Right:** Reserves accounts are remunerated at Bank Rate and may hold balances overnight; settlement accounts for non-bank payment service providers are unremunerated and may not.

Wrong: Faster Payments matters more than CHAPS because it carries far more traffic. **Right:** In Q2 2026 CHAPS settled £411.6 billion a day against Faster Payments' £2.3 billion, because netting compresses the retail systems before they reach the Bank.

Wrong: A digital pound has been decided on and will replace cash. **Right:** The Bank and HM Treasury are in a design phase, no decision to build or issue has been taken, and the Bank has said it will keep issuing banknotes for as long as people want them.

Wrong: Banknotes are issued by your bank. **Right:** A £20 note is a direct bearer liability of the Bank of England, and it is the only form of central bank money a member of the public may hold.

6.99 Chapter summary in 20 lines

1. Your bank's promise is money to you, but your bank needs money that does not depend on any other bank staying solvent.
2. That requirement produces a two-tier system: banks keep books recording what they owe customers, and the central bank keeps a book recording what it owes banks.
3. A payment between two customers of the same bank is a relabelling exercise inside one ledger and requires nobody else's cooperation.
4. A payment between customers of different banks releases one bank from an obligation and burdens the other, so something of value must pass between them.
5. That something is central bank money, held in reserves and settlement accounts at the Bank of England.

6. Commercial bank money is an unsecured claim on a private firm; central bank money carries effectively no credit risk in sterling.
7. Banknotes are the only form of central bank money the public may hold, and there are over 4.98 billion of them worth about £91.5 billion.
8. That both kinds spend identically is called the singleness of money, and it is engineered by the shared ledger in the middle rather than being an accident.
9. In the worked day, £2,245 of customer payments across two banks moved £450 once in the upper ledger, and two of the five payments never left the bank they started in.
10. Reserves are not deposits in storage: no ratio binds them, and the total fell from £706,745 million to £643,547 million in the year to February 2026 without anybody being paid.
11. Reserves cannot leave the upper tier, so a bank lends by creating a deposit and worries about reserves only when the money walks out of the door.
12. "The same bank" is a legal question rather than a branding question, and ring-fencing has split the large British groups into several authorised entities.
13. Even a genuine on-us payment moves risk, because deposit protection is per depositor per authorised firm, raised to £120,000 on 1 December 2025.
14. Only CHAPS settles gross and in real time; everything else settles net and later, and that interval is a credit exposure created by a customer.
15. The Bank of England's answer is prefunding, which caps a participant's maximum net debit at cash already segregated in RTGS.
16. There are four account types at the Bank — reserves, reserves and settlement, settlement, and omnibus — differing on eligibility, remuneration and overnight balances.
17. Access is narrow, and the several thousand institutions that reach CHAPS indirectly settle in a direct participant's commercial bank money instead.
18. That is tiering: the two-tier picture repeats one level down with a commercial bank at the top of it.
19. RTGS settled about £902 billion a day in Q2 2026, of which the entire retail payments experience of the country was roughly £12.6 billion.
20. Every settlement rule in the rest of the book answers two questions — in whose money, and when — so the productive question about odd behaviour is which tier the obligation is sitting in.

Sources used: Bank of England — A brief introduction to the Real-Time Gross Settlement system and CHAPS; Payment and settlement; Payment and settlement statistics; Access policy for RTGS settlement accounts and services; Bank of England Settlement Accounts; CHAPS; Banknote statistics; Report on the Bank's official market operations March 2025–February 2026; RTGS and CHAPS Annual Report 2024–25; Reviewing access to RTGS accounts for settlement (2024); Extending RTGS and CHAPS settlement hours – next steps (2026); Money creation in the modern economy, Quarterly Bulletin 2014 Q1; The digital pound; PRA news release and PS24/25 on depositor protection (18 November 2025); Which firms does the PRA regulate. Bank for International Settlements — Wholesale central bank money in the context of tokenisation (18 September 2025); CPSS, The role of central bank money in payment systems (12 August 2003). European Central Bank — TARGET Services Annual Report 2025; T2 facts and figures. Federal Reserve — H.4.1 statistical release, 13 August 2026; Federal Reserve Financial Services, Master Account; Federal Register, proposed revisions to the Policy on Payment System Risk and account access guidelines (26 May 2026). Pay.UK — Faster Payment System. Fnality — Sterling Fnality Payment System announcement, 14 December 2023.

CHAPTER 7

Clearing Versus Settlement

7.0 What this chapter gives you

1. You will be able to say precisely what clearing is, what settlement is, and why "the payment went through" carries almost no information.
2. You will be able to name the four separate events that sentence collapses, and say whose record holds each one.
3. You will be able to explain why a Faster Payment credited at two o'clock on a Sunday morning is not settled between the banks until Monday.
4. You will be able to state the trade-off between real-time gross settlement and deferred net settlement as an exchange of liquidity for settlement risk.
5. You will be able to explain what happened to Bankhaus Herstatt's counterparties on 26 June 1974, and why the loss was the whole principal rather than a fraction of it.
6. You will be able to say why the New York clearing process then ran until one in the morning for several nights running, and what that failure mode is called.
7. You will be able to explain why finality is a legally defined moment under the Settlement Finality Regulations 1999 rather than a feeling.
8. You will be able to keep finality between direct participants distinct from consumer reimbursement for authorised push payment fraud.
9. You will be able to explain why a clearing file and a settlement advice will never tie out item by item, and what reconciliation therefore has to prove.
10. You will be able to decide, before releasing goods against an incoming payment, which of the four events has actually occurred and who can tell you.

Ask anybody who works in payments what the hardest thing is about explaining their job, and a good number of them will give you the same answer. It is that the public has one word — "payment" — for what the industry treats as at least four separate events, occurring at four different times, on four different ledgers, with four different sets of consequences if something goes wrong.

The most important of those distinctions is the one this chapter is about. Clearing is the process of working out who owes what to whom. Settlement is the money actually moving. They are not the same thing, they very rarely happen at the same moment, and almost every strange behaviour in the payments world — the three-day wait, the payment that "arrived" on Sunday but was not paid for until Monday, the refund that takes a week, the transaction that shows as pending and then vanishes — is a consequence of the gap between them.

The previous chapter established that there are two kinds of money in circulation: commercial bank money, which is what sits in your account and is a promise from your bank, and central bank money,

which is what banks hold in reserves accounts at the Bank of England. That distinction is the raw material of this one. Clearing happens in the world of obligations. Settlement happens in the world of central bank money. Clearing produces a number. Settlement makes the number true.

The phrase "the payment went through" contains no information about which of these has happened. That is not pedantry. In 1974 the difference between them destroyed a German bank's counterparties, froze the New York clearing process for several nights running, and led directly to the creation of the international committee structure that still governs payment system oversight today. The word for the risk is named after that bank.

The plain version

Imagine two shops facing each other across a small high street. On one side is a bakery run by Nadia. On the other is a hardware shop run by Owen. Their staff buy from each other constantly. The bakery's staff need string, bin bags, light bulbs and a new mop. The hardware shop's staff want coffee, sandwiches and a birthday cake.

At first they paid each other in cash every time. Somebody would walk across the road with a five-pound note, wait for change, and walk back. It worked, but it was tedious, and both tills ended up short of coins at awkward moments.

So they changed the arrangement. Now, when somebody from the bakery buys a mop, Owen writes it in a notebook by the till: "Bakery, mop, £8.50." When somebody from the hardware shop buys a cake, Nadia writes it in hers: "Hardware, cake, £22." Nobody hands over money at the moment of purchase. Goods change hands, and a note gets made.

At six o'clock, one of them walks across the road with their notebook and they sit down together for two minutes.

Agreeing the list

The first thing they do is read out the entries and check that both notebooks agree. Owen has "Bakery, mop, £8.50" and Nadia has the same. Owen has "Bakery, bin bags, £4.20" — Nadia has that too. Then Owen reads out "Bakery, extension lead, £11" and Nadia frowns, because she has no record of anybody buying an extension lead. They talk about it. It turns out one of Nadia's Saturday staff bought it and forgot to write it down. Fine. Both notebooks now agree.

This step — reading out the entries, comparing them, resolving the ones that do not match, and arriving at a list both sides accept — is clearing. Nothing has been paid. Not a penny has moved. What has been produced is agreement: a shared, undisputed statement of who owes what.

That is genuinely all clearing is. It is bookkeeping and argument. It is the part of a payment where two institutions establish, to their mutual satisfaction, the size of the debt between them.

Moving the money

The second thing they do takes ten seconds. Nadia's total for the day is £46.30 owed to Owen. Owen's total is £38.90 owed to Nadia. Rather than one of them handing over £46.30 and the other handing back £38.90, they take the difference. Nadia opens the till, counts out £7.40, and puts it in Owen's hand.

That is settlement. The obligation has been discharged. Whatever happens tomorrow — if the bakery burns down, if Owen is run over by a bus, if either of them stops trading — nobody can come back and

claim that day's trading was not paid for. It was. There is a £7.40 gap in Nadia's till and £7.40 more in Owen's, and that is not an entry in a notebook. It is the money.

Notice how much smaller the settlement is than the trading. Over the day, £85.20 of goods changed hands. The amount of actual cash that moved was £7.40. That compression is the whole reason the notebook system exists, and it is the subject of the next chapter. What matters here is the sequence: agree first, pay after.

The same thing, with banks

Now replace Nadia with Barclays and Owen with Lloyds, and replace mops and cakes with customer payments.

On Monday, four of Barclays' customers send money to Lloyds customers, and three of Lloyds' customers send money to Barclays customers.

Direction	Amounts	Total
Barclays customers to Lloyds customers	£180, £2,400, £65, £1,355	£4,000
Lloyds customers to Barclays customers	£900, £2,750, £150	£3,800

Seven payments. £7,800 of money as the customers experience it — seven people saw a debit, seven people saw a credit, and every one of those amounts was correct and complete from the customer's point of view.

Clearing is the process by which the two banks exchange the records of those seven payments, check that each side has the same list, sort out anything that does not match, and arrive at a single agreed statement: Barclays owes Lloyds £4,000, Lloyds owes Barclays £3,800, and therefore Barclays owes Lloyds £200.

Settlement is the moment somebody moves £200.

And here is the part that surprises people the first time they hear it. In several of the systems that move most of Britain's money, that £200 does not move at the moment the customers see their balances change. It moves later. Sometimes hours later. Sometimes three days later. In the meantime, the receiving bank has given real, spendable money to its customer on the strength of a promise from the sending bank that has not yet been honoured.

Two clocks

There are broadly two ways to organise this, and Britain runs both.

The first way is the notebook. Let everything accumulate, add it all up at a set time, and move one figure. This is cheap, because you move money once instead of thousands of times, and it needs very little cash sitting around waiting. Its cost is the gap. Between the moment a customer's payment is accepted and the moment the banks square up, one bank is exposed to the other. In Britain, Bacs works this way, and so does the Faster Payments Service, and so do the cheque and card systems.

The second way is to settle every payment on its own, one at a time, as it happens. No notebook, no waiting, no gap. Its cost is that each bank must have enough money sitting in its account at the Bank of England to cover each payment at the moment it is made, which is expensive, because money sitting in an account is money not being used for anything else. In Britain, CHAPS works this way — the system

used for house purchases, corporate treasury and the very large payments financial institutions make to each other.

Those two designs have names, and the names are worth learning because they will appear throughout the rest of this book. The notebook way is deferred net settlement. The one-at-a-time way is real-time gross settlement, usually shortened to RTGS.

Why the gap is dangerous

In June 1974 there was a bank in Cologne called Bankhaus Herstatt. It was not large. It had, however, been trading foreign currency very badly, and on 26 June the German banking supervisor took its licence away.

The timing is the whole story. The licence was withdrawn during the banking day in Germany but after the German interbank payment system had closed for the afternoon. Herstatt's counterparties — other banks around the world that had agreed to swap Deutschmarks for dollars with it — had already, that morning, paid their Deutschmarks into Herstatt's German account. That was their side of the bargain, done, irrevocably.

The dollars were supposed to come back to them in New York later the same day, because New York is five or six hours behind Cologne and its business day was still going. At half past ten New York time, Herstatt's New York correspondent bank stopped making dollar payments on its behalf.

So a set of perfectly ordinary banks had handed over the full amount of one currency and received nothing at all in return. Not a reduced amount. Not a delayed amount. Nothing, with a claim against a liquidated German bank as the only consolation.

What happened next was almost worse. Banks in New York, suddenly aware that paying first and hoping was a losing strategy, started holding back their own outgoing payments until they could see the incoming ones arrive. Everybody doing that at once meant nobody could go first. The clearing process in New York seized up and, on two or three consecutive nights, ran until one in the morning before it finished.

That is why the industry has a special word for this. When one party to a currency exchange pays out what it owes and does not receive what it is due, the loss is not a fraction of the trade. It is the entire principal. The industry calls it Herstatt risk, after a bank that had been out of business for a day.

What this means for "the payment went through"

Put the pieces together and you can see why that sentence carries almost no information.

It might mean the sending bank has checked the payment and taken the money off its customer. It might mean the two banks have agreed that one owes the other. It might mean the money has actually moved between them at the Bank of England. It might mean the receiving customer can see it and spend it. Those are four different facts, and depending on which payment system you are using they occur in different orders, minutes or days apart.

If somebody tells you a payment went through and you are about to hand over a car, a house or a shipping container, the useful question is not "has it gone through". It is "has it settled, and where can I see that".

Where the plain version stops being true

The notebook picture is the right shape, and practitioners carry something like it in their heads. Four things about it are wrong in ways that cause real errors.

Clearing is not a meeting at six o'clock

The high-street analogy makes clearing sound like a discrete event at the end of the day: two parties sit down, compare lists, agree. Nothing in a modern payment system works like that.

Clearing is a continuous pipeline, running all day, made of several distinct stages that the analogy compresses into one. A payment instruction is transmitted. It is validated for format and for the existence of the destination account. It is screened against sanctions lists and fraud rules. It is matched against the counterparty's record of the same instruction. It is placed in a queue and given a position in the sequence. Only at the end of that pipeline does it contribute to a figure that anyone would recognise as "what we owe you".

The Committee on Payments and Market Infrastructures, which is the body at the Bank for International Settlements that maintains the industry's shared vocabulary, defines clearing as "the process of transmitting, reconciling and, in some cases, confirming transactions prior to settlement, potentially including the netting of transactions and the establishment of final positions for settlement". Read that carefully. Netting is optional — "in some cases", "potentially including". Clearing is the whole pipeline. Netting is one thing that may happen inside it.

The same glossary then adds a sentence that ought to be printed on the wall of every payments team in the country: "Sometimes this term is also used (imprecisely) to cover settlement." The standards body itself is warning you that the industry misuses its own word. When somebody says a payment has "cleared", you cannot safely infer anything about whether money has moved.

Netting and clearing are not the same idea, and RTGS still clears

The analogy makes it look as though clearing exists in order to produce a net figure, and therefore that a system which settles every payment individually has no clearing at all. Both halves of that are wrong.

CHAPS payments settle one by one, gross, for full value, with no netting anywhere in the process. CHAPS nevertheless has a substantial clearing function: messages are validated, sequenced, checked against liquidity, queued when a participant is short, and released when it is not. What CHAPS does not do is defer. Clearing and settlement in CHAPS are separated by fractions of a second rather than by hours, which makes them easy to conflate, but they remain two distinct operations and the system's rules treat them as such.

Conversely, netting can happen without clearing in any interesting sense — two counterparties can agree bilaterally to offset positions under a contract with no infrastructure involved at all.

The useful axis is not "netted or not". It is when settlement occurs relative to the obligation, and in what asset. Netting is the subject of the next chapter and deserves its own treatment.

Your balance changing is not settlement

This is the correction with the largest practical consequences, and it is the one that catches out software teams building on top of payment rails.

When your bank shows the money as arrived, your bank has updated its own ledger — the record of what it owes you. That is commercial bank money, and it is entirely under your bank's control. It required no cooperation from any other institution and no involvement from the Bank of England. Under Faster Payments, this typically happens within seconds, at any hour, on any day of the year.

Settlement is a different ledger entirely: the settlement accounts held at the Bank of England, in central bank money. Faster Payments net settlement takes place three times each business day. So a payment

received at two o'clock on a Sunday morning has been credited to a customer, spent, perhaps withdrawn as cash, and none of that has been settled between the banks until Monday.

The receiving bank has extended credit to the sending bank in the interval, on the initiative of a customer, without anybody's credit committee being consulted. This is not an oversight; it is the deliberate design of every deferred net system, and it is why those systems need risk controls that RTGS does not.

The practical rule for anyone building or reconciling systems: the timestamp on a customer credit and the timestamp on a settlement are different fields with different meanings, and the interval between them is a real exposure held by a real institution. Treating them as the same event is the single most common error in payments engineering.

"Final" is a legal status, not a feeling

The notebook analogy ends with cash in a hand, which feels final because it is physical. In electronic systems, finality is a legal construct, and it exists because insolvency law would otherwise be able to reach back in time and unwind transfers that everybody had treated as complete.

The CPMI definition is exact: final settlement is "the irrevocable and unconditional transfer of an asset or financial instrument, or the discharge of an obligation by the FMI or its participants in accordance with the terms of the underlying contract", and it adds that "final settlement is a legally defined moment".

In the United Kingdom the relevant instrument is the Financial Markets and Insolvency (Settlement Finality) Regulations 1999, which implemented the EU Settlement Finality Directive. Systems apply to be designated under them, and the Bank of England is the designating authority for everything except systems that are, or whose operator is, a recognised investment exchange, where it is the Financial Conduct Authority. Designation gives a system protection against the ordinary operation of insolvency law, so that transfer orders already entered into the system, and the netting of them, survive the insolvency of a participant instead of being reversed.

Two further corrections follow from this. First, finality attaches to the interbank transfer, not to the customer's experience. The Bank of England states that in CHAPS "the transfer of funds is irrevocable between the direct participants" — between the direct participants. That is not a statement that a defrauded consumer cannot be reimbursed. Since 7 October 2024 the Bank has operated CHAPS reimbursement rules for authorised push payment fraud, with a maximum reimbursement level for UK retail CHAPS payments set at £85,000, aligned with the Payment Systems Regulator's approach for Faster Payments. Settlement finality and consumer redress are different mechanisms operating on different parties.

Second, and less comfortably, finality is not identical across systems. A payment can be final in one legal sense and still be sitting inside a clearing cycle whose settlement has not occurred. If you need to know whether a specific payment is beyond recall, the answer is in the rulebook of the specific system, not in a general principle.

The technical version

The vocabulary, defined

Because the terms are used loosely in commerce and precisely in the rulebooks, it is worth setting out the CPMI definitions verbatim. These are the definitions that scheme rulebooks and central bank documentation are written against.

Term	CPMI definition
Clearing	The process of transmitting, reconciling and, in some cases, confirming transactions prior to settlement, potentially including the netting of transactions and the establishment of final positions for settlement
Settlement	The discharge of an obligation in accordance with the terms of the underlying contract
Final settlement	The irrevocable and unconditional transfer of an asset or financial instrument, or the discharge of an obligation by the FMI or its participants in accordance with the terms of the underlying contract. Final settlement is a legally defined moment
Settlement asset	An asset used for the discharge of obligations as specified by the rules, regulations or customary practice for an FMI
Settlement lag	In a transfer system, the time lag between the acceptance of the transfer order by the system and its final settlement
Real-time gross settlement	The real-time settlement of payments, transfer instructions or other obligations individually on a transaction-by-transaction basis
Deferred net settlement	A net settlement mechanism which settles on a net basis at the end of a predefined settlement cycle
Net settlement system	A funds or securities settlement system in which final settlement of transfer instructions occurs on a net basis at one or more discrete, pre-specified times during the processing day
Settlement risk	The general term used to designate the risk that settlement in a funds or securities transfer system will not take place as expected. This risk may comprise both credit and liquidity risk
Principal risk	The risk that a counterparty will lose the full value involved in a transaction — for example, the risk that a seller of a financial asset will irrevocably deliver the asset, but not receive payment
Payment versus payment	A settlement mechanism that ensures that the final transfer of a payment in one currency occurs if and only if the final transfer of a payment in another currency or currencies takes place

Three of these repay close attention. "Settlement lag" names the interval this chapter is about and makes it a measurable quantity rather than a vague unease. "Principal risk" is the specific, severe form of settlement risk in which the exposure is the whole trade rather than the cost of replacing it. And "payment versus payment" is the engineering answer to principal risk in foreign exchange, which we come to below.

The two models, and the trade-off they encode

The choice between real-time gross settlement and deferred net settlement is a straight exchange of one scarce resource for another. RTGS spends liquidity to buy the elimination of settlement risk. DNS spends settlement risk to buy liquidity efficiency.

The Bank of England states the trade-off in exactly those terms in its description of CHAPS: "Settlement risk is eliminated between CHAPS direct participants, at the cost of an increased need for liquidity, making this model best suited to a high-value payment system with the largest potential systemic risk."

That last clause is the design principle. You put the expensive, risk-free model where the exposures would be catastrophic, and the cheap, exposed model where the exposures are small and manageable — and then you bolt risk controls onto the cheap model until its residual exposure is acceptable.

Britain's arrangement follows this exactly. All of it settles across accounts at the Bank of England; what differs is when, and on what basis.

System	Operator	Settlement basis	Frequency
CHAPS	Bank of England	Real-time gross	Continuous through the settlement day
CREST	Euroclear UK and International	Delivery versus payment	High-frequency cycles through the day
Bacs	Pay.UK	Deferred multilateral net	Once each business day
Faster Payments Service	Pay.UK	Deferred multilateral net, prefunded	Three times each business day
Image Clearing System	Pay.UK	Deferred multilateral net	Once each business day
LINK	LINK Scheme	Deferred multilateral net	Daily cycles; weekend and holiday cycles settle the next business day
Visa Europe	Visa Europe Limited	Deferred multilateral net	Daily cycles; weekend and holiday cycles settle the next business day
Mastercard Europe	Mastercard Europe SA	Deferred multilateral net	Daily cycles; weekend and holiday cycles settle the next business day

CHAPS as the worked example of RTGS

CHAPS is a sterling same-day system for high-value wholesale payments and for time-critical lower-value payments such as property purchases. Responsibility for it transferred from the CHAPS Clearing Company to the Bank of England in November 2017, which means the Bank is simultaneously the system's operator, its settlement agent and its supervisor's colleague — an arrangement it manages through internal separation and an exchange of letters with the Payment Systems Regulator, most recently updated in January 2025.

The mechanics are as follows. Payment instructions are routed over SWIFT to the RTGS system. Settlement occurs across settlement accounts at the Bank of England by debiting the sending participant and crediting the receiving participant, individually, for full value. Only once that has happened is the full payment message forwarded onward to the receiving bank. This is the "Y-copy" arrangement, and its consequence is worth stating plainly: in CHAPS, interbank settlement is complete before the beneficiary's bank has even received the complete instruction. The order of events is the reverse of the retail intuition.

The scale is instructive. In 2024 CHAPS settled over £87 trillion of payments, an average of over £344 billion each working day across around 208,000 payments, giving an average payment value of about £1.7 million. The Bank's own summary of what that means: CHAPS represents around 0.5 per cent of UK total payment volumes but 92 per cent of total sterling payment values.

More recent quarterly figures show the same picture and a rising trend.

Average daily figures	Q3 2025	Q4 2025	Q1 2026	Q2 2026
CHAPS volume	206,786	213,676	205,733	215,118
CHAPS value (£mn)	359,554	387,890	399,457	411,562
CREST DvP value (£mn)	425,351	460,405	485,260	477,841
Bacs net value (£mn)	5,370	5,653	5,800	6,205
FPS net value (£mn)	2,008	2,104	2,287	2,298
Cheque imaging net value (£mn)	41	38	37	37
LINK net value (£mn)	232	235	213	238
Visa net value (£mn)	2,165	2,273	2,173	2,356
Mastercard net value (£mn)	1,379	1,451	1,385	1,480
Total RTGS value (£mn)	796,099	860,050	896,612	902,018

There are over 35 CHAPS direct participants, with several thousand financial institutions reaching the system indirectly through them. The list is more varied than the phrase "high-street banks" suggests: alongside Barclays UK, Lloyds Bank plc, NatWest, HSBC UK Bank plc and Santander UK plc, the direct participants include CLS Bank International, LCH Limited, Euroclear Bank SA/NV, Fnality UK Limited, ClearBank Limited, iFAST Global Bank Limited and Banking Circle S.A., plus a long list of London branches of foreign banks.

The liquidity cost of gross settlement is managed by two devices. The first is the CHAPS Throughput Rules, which oblige direct participants to have settled specified proportions of their daily CHAPS value by fixed points in the day — currently 12:00, 15:00 and 17:00. Without such a rule, every participant's individually rational strategy is to delay outgoing payments and fund them from incoming ones, which collectively produces the New York failure mode of 1974 in slow motion. The second is queueing and central scheduling inside RTGS, so that payments a participant cannot yet fund wait in an ordered queue rather than being rejected.

CHAPS currently opens at 06:00 and closes at 18:00 for bank-to-bank payments, with customer payments to be submitted by 17:40. Direct participants must be open to receive by 08:00 and to send by 10:00. In 2026 the Bank confirmed an early-morning extension under which CHAPS will open at 01:30, with all direct participants required to receive payments from that time whether or not they choose to send; the target is September 2027, and the Throughput Rules will continue to be measured from 06:00.

Bacs as the worked example of DNS

Bacs is the opposite design and demonstrates every property of deferred net settlement.

Payments submitted to Bacs go through a three-day clearing and processing cycle. The deadline for receipt of payment instructions from service users is 22:30 on Day 1. Processing and distribution to the receiving banks happens on Day 2. Debits and credits reach customer accounts on Day 3.

The interbank obligations arising in Bacs are settled at the Bank of England on a multilateral net basis on Day 3 of the cycle. Settlement occurs at 09:30, through the posting of multilateral net settlement

positions directly to settlement accounts in the RTGS processor. One posting, once a day, for the whole system.

The compression this achieves is the point of the design. In 2025 Bacs facilitated 6.86 billion payments with a value of £6.05 trillion, including a record 5.0 billion Direct Debits. Spread over roughly 250 working days, that is on the order of £24 billion of customer payments a day. The corresponding amount of central bank money that actually moved averaged £5,587 million a day across 2025 to date, and £6,205 million a day in the second quarter of 2026. Those two figures come from different publications measuring slightly different things, so the ratio should be treated as indicative rather than exact, but the order of magnitude is not in doubt: something like three-quarters of the value entering Bacs never needs to be settled at all, because it offsets against value going the other way.

Since its inception in 1968, over 183.6 billion transactions have been debited or credited to British bank accounts through Bacs.

The cost of that efficiency is the settlement lag. From 22:30 on Day 1 to 09:30 on Day 3 there exist agreed obligations between banks that have not been discharged. If a participant failed inside that window, the surviving participants would be holding claims rather than money.

The risk controls that make DNS survivable

The industry's answer to that window is not to shorten it but to collateralise it.

The Bank of England's RTGS supports both prefunded and non-prefunded deferred net settlement models. Under the prefunded model, each participant's maximum possible net debit position is capped, and the participant "must hold funds in a special prefunding account in RTGS equal to that cap". If a participant fails, the money required to complete settlement is already segregated at the central bank and settlement proceeds regardless. The exposure between participants is not reduced; it is removed, by being funded in advance.

Faster Payments operates on this basis, with each participant subject to a Net Sender Cap limiting its indebtedness to other participants per settlement cycle. The design has an obvious cost — prefunding is idle liquidity — and that cost is a barrier to entry, particularly for non-bank payment firms. In 2026 Pay.UK introduced a more flexible Net Sender Cap model intended to reduce the amount of prefunding required and widen direct access to Faster Payments.

The 2012 CPSS Red Book chapter on the United Kingdom recorded the three Faster Payments clearing cycles as settling at 07:15, 13:00 and 15:45. The Bank of England's current published description states only that net settlement takes place three times each business day, so the precise times should be taken from the scheme's current documentation rather than from the Red Book.

Herstatt, precisely

The canonical settlement failure deserves stating in exact terms, because it is invoked constantly and described loosely.

On 26 June 1974 the Bundesaufsichtsamt für das Kreditwesen, the German federal banking supervisory office, withdrew the banking licence of Bankhaus Herstatt of Cologne. The BIS account is specific about the timing: the withdrawal occurred during the banking day but after the close of the interbank payments system in Germany. At 10:30 New York time on the same day, Herstatt's New York correspondent suspended dollar payments on its account.

The consequence was that Herstatt's counterparty banks were exposed for the full value of the Deutschmark deliveries they had already made. They had performed their side of foreign exchange

contracts irrevocably and received nothing. The Basel Committee's own case study of the failure, published in April 2004 as Working Paper No. 13, Bank Failures in Mature Economies, records that by June 1974 Herstatt's losses on its foreign exchange operations amounted to DM 470 million, and that a special audit authorised by the supervisor in March of that year had already found open exchange positions of DM 2 billion against an internal limit of DM 25 million.

The 1996 BIS report on settlement risk in foreign exchange transactions — usually called the Allsopp Report — sets out the resulting definition: "One party to an FX trade could pay out the currency it sold but not receive the currency it bought. This principal risk in the settlement of foreign exchange transactions is variously called foreign exchange settlement risk or cross-currency settlement risk. It is also referred to as Herstatt risk." The report itself notes that naming the risk after one institution is in some respects inappropriate, since the circumstances of subsequent failures differed.

The second-order effect is the one the Bank of England's own archives illuminate, in research published on the Bank Underground blog in June 2015. Banks in New York reacted by holding up outgoing payments until their accounts were sufficiently in credit. The Federal Reserve's Charles Coombs recorded that this "had the effect of virtually freezing up the clearing process, which on two or three consecutive days ran until 1am". A payment system in which everybody waits for everybody else does not fail loudly; it congeals. The archives show a chain reaction of payment delays across international financial centres, threatening the stability of the international payment system as a whole.

The institutional response was durable. The G10 members, together with Luxembourg and Spain, established a standing committee under the auspices of the Bank for International Settlements. That lineage runs through the Basel Committee on Banking Supervision and the Committee on Payment and Settlement Systems, now the Committee on Payments and Market Infrastructures, whose glossary supplied the definitions above.

Payment versus payment, and how much risk is left

The engineering fix for principal risk in foreign exchange is payment versus payment: a settlement mechanism that ensures the final transfer of one currency occurs if and only if the final transfer of the other does. Neither leg can complete alone, so neither party can be left having paid without being paid.

The dominant implementation is CLSSettlement, operated by CLS Bank International, which provides PvP settlement for 18 of the world's most traded currencies. CLS reports settling an average daily value of over USD 7 trillion of payment instructions for more than 70 settlement members and over 37,000 indirect participants; its product materials cite a figure of over USD 8.0 trillion of payments each day. CLS Bank International is a CHAPS direct participant and Continuous Linked Settlement is designated under the UK Settlement Finality Regulations, which is a compact illustration of how these arrangements interlock: sterling PvP legs settle in central bank money at the Bank of England.

The 2025 BIS Triennial Survey put numbers on how much of the problem remains. Of roughly USD 14 trillion of gross financial obligations settled daily in April 2025, USD 5.2 trillion — 36 per cent — settled via PvP, eliminating settlement risk. About USD 7.6 trillion, 54 per cent, settled by methods that mitigate but do not eliminate the risk, such as pre-settlement netting, intragroup settlement and timing controls. And more than USD 1.4 trillion, 10 per cent, settled on a gross bilateral basis, fully exposed. More than four fifths of daily settlement — USD 12.2 trillion — involved CLS-eligible currency pairs, of which 40 per cent settled via PvP.

Fifty-one years after Herstatt, on an ordinary day in April 2025, well over a trillion dollars a day of foreign exchange still settled in exactly the manner that destroyed Herstatt's counterparties.

Settlement finality in UK law

The Financial Markets and Insolvency (Settlement Finality) Regulations 1999 allow payment and settlement systems to apply for protection against the normal operation of insolvency law. The designating authority is the Bank of England, except where the system is, or its operator is, a recognised investment exchange for the purposes of the Financial Services and Markets Act 2000, in which case it is the FCA.

Systems designated under UK law include CHAPS, Bacs, the Faster Payments Service, the Image Clearing System, Continuous Linked Settlement, Visa Europe, Euroclear UK and International, LCH Limited, ICE Clear Europe, LME Clear Limited, SIX x-clear and the Sterling Fnality Payment System. A further set of non-UK-law systems is designated for these purposes, including EBA Clearing's STEP2-T, EURO1 and RT1, Euroclear Bank SA/NV, Eurex Clearing AG, LCH SA, Clearstream Banking S.A., the Norges Bank Settlement System and RIX operated by Sveriges Riksbank.

Designation is not a badge. It is the legal machinery that makes settlement stick when a participant fails, and being a participant within the meaning of those Regulations is one of the stated conditions for direct participation in CHAPS.

The four events, and how to talk about them

The chapter's opening claim can now be made precisely. "The payment went through" collapses at least four distinct events.

Event	What has happened	Whose record	Can it be undone?
Authorisation or acceptance	The sending institution has validated the instruction and typically debited its customer	Sending bank's own ledger	Yes, by the sending bank, subject to scheme rules
Clearing	The obligation has been established and, where applicable, netted into a position	Scheme infrastructure and both banks' records	The position can change until the cycle closes
Settlement	Central bank money has moved between settlement accounts at the Bank of England	The Bank of England's RTGS ledger	Not between the direct participants once final
Customer credit	The receiving bank has increased the balance it owes its customer	Receiving bank's own ledger	Yes, in some circumstances, by the receiving bank

The order of those four is not fixed. In CHAPS, settlement precedes the receiving bank getting the full message, which precedes the customer credit. In Faster Payments, the customer credit comes first, often within seconds, and settlement follows in the next of three daily cycles. In Bacs, the customer entries and the settlement both fall on Day 3, near each other but not identical, with settlement at 09:30. In the card systems, authorisation may precede clearing by days and settlement by days more — a sequence Volume III takes apart in detail.

This is why the diagnostic question in an incident is never "did the payment go through". It is a sequence of narrower questions. Did the instruction get accepted, and by whom? Did it enter a clearing cycle, and which one? Has that cycle settled? And has the beneficiary's bank posted to the customer

account? Four questions, four different systems of record, and frequently four different organisations to ask.

What this means for people building on these rails

Two practical consequences follow, and both cost money when ignored.

The first concerns reconciliation, the subject of a later chapter. A clearing file and a settlement advice are not two views of the same fact. The clearing file lists individual items; the settlement advice reports a position. They will not tie out item by item, and they are not supposed to. Reconciliation is the discipline of proving that a set of items produces the position that was settled, which is a different exercise from matching one list against another and considerably more subtle.

The second concerns the meaning of an incoming payment for risk purposes. If your business releases goods, credits an account, or allows a withdrawal on the strength of a payment that has been cleared but not settled, you have taken on a share of a bank's settlement exposure without a bank's balance sheet. For most retail-scale activity that is a reasonable commercial judgement, because prefunding and net sender caps stand behind the retail systems. For unusual amounts, unusual timing, or unusual counterparties, it is worth knowing exactly which of the four events has occurred before acting.

And that, ultimately, is why this distinction is worth a chapter. The public has one word for payment. The industry has a pipeline with several stages, several ledgers and several legally defined moments, and it built that pipeline the way it did because in June 1974 it discovered, expensively, what happens when the gap between agreeing and paying is left unmanaged.

7.98 Common wrong ideas

Wrong: Clearing and settlement are two words for the same thing. **Right:** Clearing establishes who owes what and settlement discharges the obligation, and the CPMI glossary itself warns that the industry uses "clearing" imprecisely to cover both.

Wrong: A payment that has cleared has been paid for. **Right:** Clearing produces a number and only settlement makes the number true, and in Bacs those events are two days apart.

Wrong: Your balance going up means the banks have settled. **Right:** Your balance is your own bank's record of what it owes you, updated without any cooperation from another institution or from the Bank of England.

Wrong: Clearing exists in order to net, so a system that settles gross does not clear. **Right:** CHAPS nets nothing yet validates, sequences, checks liquidity and queues, because clearing is the whole pipeline and netting is only one optional step inside it.

Wrong: Clearing is a discrete event at the end of the day. **Right:** It is a continuous pipeline of transmission, validation, sanctions and fraud screening, matching, queueing and position-building that runs all day.

Wrong: Herstatt risk is a currency risk. **Right:** It is principal risk: paying out the full amount of one currency and receiving nothing of the other, which is why payment versus payment was invented.

Wrong: CLS has solved foreign exchange settlement risk. **Right:** In April 2025 only 36 per cent of roughly USD 14 trillion of daily obligations settled by PvP, and more than USD 1.4 trillion settled on a gross bilateral basis, fully exposed.

Wrong: Settlement finality means a defrauded consumer cannot be reimbursed. **Right:** Finality is irrevocability between the direct participants, and since 7 October 2024 CHAPS has had its own reimbursement rules for authorised push payment fraud, capped at £85,000 for UK retail payments.

Wrong: Finality works the same way in every system. **Right:** A payment can be final in one legal sense and still sit inside a clearing cycle that has not settled, so the answer is in the specific rulebook rather than in a general principle.

Wrong: In CHAPS the beneficiary's bank receives the message and then the banks settle. **Right:** Under the Y-copy arrangement interbank settlement completes first, and only then is the full payment message forwarded onward.

7.99 Chapter summary in 20 lines

1. The public has one word, "payment", for what the industry treats as at least four events on four ledgers at four different times.
2. Clearing is the process of working out who owes what to whom; settlement is the money actually moving.
3. Clearing produces a number and settlement makes the number true.
4. Clearing happens in the world of obligations and settlement in the world of central bank money at the Bank of England.
5. The CPMI defines clearing as transmitting, reconciling and sometimes confirming transactions prior to settlement, potentially including netting, and warns that the term is often used imprecisely to cover settlement.
6. Clearing is a continuous pipeline of validation, screening, matching, queueing and sequencing rather than a meeting at six o'clock.
7. Netting is one optional step inside clearing rather than its purpose, and CHAPS clears substantially while netting nothing at all.
8. There are two settlement models: real-time gross settlement, which spends liquidity to eliminate settlement risk, and deferred net settlement, which spends settlement risk to buy liquidity efficiency.
9. Britain puts the expensive model where the exposures would be catastrophic and the cheap model where they are small, then bolts risk controls onto the cheap one until the residue is acceptable.
10. CHAPS settles each payment individually across accounts at the Bank of England for full value, and forwards the complete message only afterwards, which is the Y-copy arrangement.
11. CHAPS is around 0.5 per cent of UK payment volumes and about 92 per cent of sterling payment values, with Throughput Rules at 12:00, 15:00 and 17:00 to stop every participant waiting for everybody else.
12. Bacs is the opposite design: submission by 22:30 on Day 1, processing on Day 2, and customer entries with settlement at 09:30 on Day 3.
13. Bacs compresses something like £24 billion of daily customer value into roughly £6 billion of central bank money moving once a day.
14. The price of that efficiency is the settlement lag, a window in which agreed obligations exist between banks that have not been discharged.
15. The industry's answer is not to shorten that window but to collateralise it, through prefunding in a segregated RTGS account and through net sender caps.
16. Your balance changing is not settlement, and treating the customer credit timestamp and the settlement timestamp as one event is the commonest error in payments engineering.

17. On 26 June 1974 Bankhaus Herstatt lost its licence after the German interbank payment system had closed, and its counterparties lost the full principal of the Deutschmarks they had already delivered.
18. Banks in New York then held back their own payments until they could see incoming ones arrive, freezing the clearing process until one in the morning for several nights and prompting the committee structure that governs payment oversight to this day.
19. Finality is a legally defined moment, conferred in the United Kingdom by designation under the Financial Markets and Insolvency (Settlement Finality) Regulations 1999, and it is distinct from consumer redress.
20. The diagnostic question in an incident is therefore never "did the payment go through" but which of authorisation, clearing, settlement and customer credit has occurred, and in whose records it can be seen.

Sources used: Bank for International Settlements — CPMI, A glossary of terms used in payments and settlement systems; Settlement risk in foreign exchange transactions (Allsopp Report, 1996); Payment, clearing and settlement systems in the United Kingdom (CPSS Red Book country chapter); Uncovering FX settlement risk: new measures from the 2025 BIS Triennial Survey, BIS Quarterly Review. Bank of England — CHAPS; A brief introduction to the Real-Time Gross Settlement system and CHAPS; Payment and settlement; Payment and settlement statistics; Extending RTGS and CHAPS settlement hours: early morning extension (2026); Financial market infrastructure supervision — Who are we (designated systems lists); Bank Underground, "BoE archives reveal little known lesson from the 1974 failure of Herstatt Bank" (June 2015). Pay.UK — Bacs Payment System; Bacs Payment System statistics; Understanding changes to the Faster Payment System Net Sender Caps. CLS Group — Settlement; Reimagining same-day FX (ShapingFX series). legislation.gov.uk — The Financial Markets and Insolvency (Settlement Finality) Regulations 1999. Payment Systems Regulator — PS24/5, CHAPS APP scams reimbursement requirement.

Netting

8.0 What this chapter gives you

1. You will be able to explain how about £23 billion of daily Bacs payments is discharged by moving about £5.3 billion between settlement accounts, with nobody short-changed.
2. You will be able to compute bilateral and multilateral net positions from a matrix of flows, and say why multilateral almost always wins.
3. You will be able to explain why the sum of all net debit positions must equal the sum of all net credit positions.
4. You will be able to say why Faster Payments nets harder than Bacs, and connect the answer to the direction of the traffic rather than to the technology.
5. You will be able to explain why netting does not reduce risk but concentrates it, turning diversified credit risk into liquidity risk at a single moment.
6. You will be able to work through an unwind and show why a bank with no meaningful exposure to the failed participant can suddenly owe five times more.
7. You will be able to distinguish position netting, netting by novation and close-out netting, and say which of them survives an administrator's cherry-picking.
8. You will be able to state the Lamfalussy "cover one" standard and name the controls that implement it: caps, prefunding and loss sharing.
9. You will be able to describe what a central counterparty does through novation, and the order in which its default waterfall is consumed.
10. You will be able to give the real reasons Bacs takes three days, and explain why the debit and the credit both fall on day three.

On an average working day in 2024, the Bacs system carried about 26.8 million payments worth roughly £23.0 billion between British banks. At the end of that day, the amount of money that actually moved between those banks in the Bank of England's settlement accounts averaged £5,265 million.

Nobody was short-changed. Every salary was paid in full, every direct debit was collected in full, every recipient saw the exact amount they were owed. Twenty-three billion pounds of value was transferred and about five billion pounds of money moved.

That is netting. It is arithmetic so simple that a child can do it, and its consequences run through every payment system on earth. It is the reason interbank settlement is affordable at all. It is also the reason a payment system can fail in a way that a pile of individual payments cannot, and the reason central banks spent the 1980s and 1990s writing rules about it.

The previous chapter separated clearing from settlement. This chapter is about what happens in the gap: the operation that turns a very large number of small obligations into a very small number of large ones.

The plain version

Four friends share a house: Ama, Ben, Cara and Dev. Over the course of a week they keep buying things for each other, because that is what people who live together do. Somebody gets to the front of the queue first and pays for two coffees. Somebody covers a train fare. Somebody buys the bin bags.

By Sunday night the situation looks like this.

Who owes	To whom	Amount
Ama	Ben	£12
Ben	Ama	£9
Ama	Cara	£5
Cara	Ama	£7
Ama	Dev	£8
Dev	Ama	£2
Ben	Cara	£15
Cara	Ben	£4
Ben	Dev	£6
Dev	Ben	£11
Cara	Dev	£20
Dev	Cara	£3

Twelve separate debts. If everybody paid everybody exactly what they owed, twelve payments would be made and £102 in cash would change hands.

Nobody does that. It would take all evening and everybody would need to have the right cash on them.

The first improvement: settle up in pairs

The obvious short cut is to deal with one pair at a time. Ama owes Ben £12 and Ben owes Ama £9, so instead of two payments totalling £21, Ama hands Ben £3 and both debts are gone.

Do this for every pair and you get six payments instead of twelve.

Pair	Owed one way	Owed the other	Who pays	Amount
Ama and Ben	£12	£9	Ama pays Ben	£3
Ama and Cara	£5	£7	Cara pays Ama	£2
Ama and Dev	£8	£2	Ama pays Dev	£6
Ben and Cara	£15	£4	Ben pays Cara	£11
Ben and Dev	£6	£11	Dev pays Ben	£5
Cara and Dev	£20	£3	Cara pays Dev	£17

Twelve payments have become six, and £102 of cash movement has become £44. Every debt has been discharged in full. Nobody has been given a discount and nobody has taken a haircut. The debts cancelled each other out because they pointed in opposite directions.

This is called bilateral netting. Bilateral means two-sided: you look at one pair of people at a time.

The second improvement: one pot

There is a better way, and it takes about a minute with a pencil.

Instead of pairing people up, work out for each person the total they owe everybody and the total everybody owes them. Then take the difference.

Person	Owes in total	Is owed in total	Difference
Ama	£25	£18	Owes £7
Ben	£30	£27	Owes £3
Cara	£31	£23	Owes £8
Dev	£16	£34	Is owed £18

Ama puts £7 on the kitchen table. Ben puts £3 on the table. Cara puts £8 on the table. There is now £18 on the table, and Dev picks it up.

Four movements of cash instead of twelve. Eighteen pounds instead of one hundred and two. And, once again, every single one of those twelve original debts has been paid in full. Nobody is out of pocket by a penny more or less than they should be.

Notice that the amounts owed and the amounts owing balance exactly. Ama, Ben and Cara owe £7, £3 and £8, which comes to £18, and Dev is owed £18. That is not a coincidence and it is not luck. Every debt has two ends. If you add up everything everybody owes and subtract everything everybody is owed, the answer must be zero, because you have counted each debt once as a plus and once as a minus. So the money going into the pot always equals the money coming out of it.

This is called multilateral netting. Multilateral means many-sided: everybody at once.

Now do it with banks

Replace the four friends with four banks, and replace the coffees and train fares with customer payments.

When a Barclays customer pays a Lloyds customer £180, that is not really a payment from a person to a person. Underneath, it is Barclays owing Lloyds £180. The customer's balance goes down at Barclays, the recipient's balance goes up at Lloyds, and now Barclays is £180 short with Lloyds and has to make it good.

There are about 27 million of those a day on Bacs alone, and they run in every direction at once, because Barclays customers pay Lloyds customers all day and Lloyds customers pay Barclays customers all day. The great majority of the value cancels out, exactly the way Ama's £12 and Ben's £9 cancelled out.

So the banks do precisely what the flatmates did. They add everything up, work out one number for each bank, and move that. Twenty-seven million payments become a few dozen movements in the Bank of England's books.

Why this is such a spectacular deal

Money sitting in a bank's settlement account at the Bank of England is money doing nothing else. It is not lending and it is not investing; it is sitting there so that payments can be made out of it. A bank that had to fund every outgoing payment as it happened would need a very large pile of cash standing still all day.

Netting means it needs a very much smaller pile. In the flatmates example the cash needed fell from £102 to £18. If you had asked all four to have the full £102 available on Sunday evening, somebody

would have said no. At £18 nobody even thinks about it. Multiply that by a national payment system and netting stops being convenient and becomes the difference between a payment system a country can afford to run and one it cannot.

And why it is dangerous

Here is the catch, and it is worth understanding before we go any further.

Look at Dev. He is owed £18 and he is expecting to be handed £18 on Sunday night. But he is not being handed £18 by any one person. He is being handed £18 by an arrangement. If Cara turns out to have nothing in her wallet at the crucial moment, the pot has £10 in it, not £18, and Dev is short.

Worse: if Cara's debts have to be taken out of the calculation entirely, everybody's number changes. It is not just that Cara's £8 goes missing. Ama, who owed £7, might now owe more, or less. Ben, who owed £3, might turn out to be owed money. Nobody can predict which way their position moves until the sum is redone.

That is the trade the whole payments industry has made. Netting makes settlement cheap by making everybody depend on one calculation. When the calculation works, it is enormously efficient. When it does not, the failure lands on everybody at once rather than on the person who was unlucky enough to be owed money by the party that failed.

And why waiting is useful

One last thing before we take the analogy apart.

The longer you wait before squaring up, the more cancels out. If the flatmates settled every debt the moment it arose, nothing would ever net, because there would never be two debts alive at the same time to cancel against each other. If they wait a day, some of it cancels. If they wait a week, most of it does.

This is not a small effect. It is the reason a payment system that waits three days can move a fifth of the money that a payment system settling instantly would have to move.

That is the thread that runs to the end of this chapter. When you eventually ask why a Bacs payment takes three days, part of the answer is: because three days is how long you have to wait for enough of it to cancel out.

Where the plain version stops being true

The kitchen table is the right picture, and it is very close to what actually happens. Four things about it are wrong in ways that matter.

Nobody is paid less, and the netting is invisible to customers

This is the misunderstanding that most often survives a first explanation, and it is worth killing immediately.

Netting operates on the obligations between banks. It does not touch the payments between customers. If your employer pays you £2,400 and your landlord takes £1,100 by direct debit on the same morning, you are credited £2,400 and debited £1,100. You are not credited £1,300. Your bank's ledger records both movements in full, because your bank's ledger is a record of what your bank owes you, and it owes you both things separately.

What is netted is the layer underneath: the amount your bank owes the other banks. The customer layer is gross. The interbank layer is net. They are different ledgers with different arithmetic, and the

whole efficiency of the system comes from the fact that one can be compressed without touching the other.

The same is true within a bank. If a Lloyds customer pays another Lloyds customer, that payment never reaches an interbank system at all. It is settled by moving a number from one row of Lloyds' own ledger to another. The Bank of England's published payment figures explicitly exclude payments internalised within a single institution, for exactly this reason.

Netting is a legal act, not just arithmetic

Adding up is easy. Making the answer stick when somebody goes bankrupt is not, and this is where most of the industry's difficulty actually lies.

Suppose Cara is declared insolvent on Sunday afternoon. Her administrator's job is to maximise the return to her creditors. A sufficiently aggressive administrator will look at the twelve debts and say: the £20 that Cara owes Dev is a claim in the insolvency, to be paid at whatever pence in the pound the estate can manage; the £3 that Dev owes Cara is an asset of the estate and must be paid in full, today. That is called cherry-picking, and it converts a tidy £17 net obligation into a £20 unsecured claim and a £3 immediate demand.

If that were allowed, netting would provide no protection whatever, and every bank in the country would have to hold capital and liquidity against its gross exposures rather than its net ones. So it is not allowed, but only because specific law says so. In the United Kingdom the relevant instrument is the Financial Markets and Insolvency (Settlement Finality) Regulations 1999, SI 1999/2979, which implemented Directive 98/26/EC. Systems designated under those regulations get statutory protection: transfer orders entered into a designated system, and the netting of those orders, take precedence over ordinary insolvency law and cannot be unwound by an administrator.

Designation is not automatic and not universal. There is a real difference between netting that is legally robust and netting that is merely a spreadsheet, and practitioners care about which one they are looking at.

There is usually no pot

The kitchen table suggests a fund sitting in the middle holding money. In most net settlement systems there is no such thing.

The system operator calculates the net positions and sends the Bank of England a set of instructions. The Bank debits the accounts of the banks in net debit and credits the accounts of the banks in net credit, in a single simultaneous posting. Nothing is held in the middle at any point. The "pot" is a bookkeeping convenience, not a place.

There is an important exception, and it is exactly where the risk is. Some systems do require participants to put money aside in advance, in a segregated account at the Bank of England, so that there is something to fall back on if a participant cannot pay. That is called prefunding, and the Bank of England introduced it for both Bacs and Faster Payments in September 2015. Where prefunding applies there genuinely is a pot, and it exists because the arithmetic on its own was judged not safe enough.

Netting does not reduce risk; it concentrates it

The flatmates example makes netting look like pure gain. It is not.

Before netting, Dev has three separate exposures: to Ama, to Ben and to Cara. If Cara fails, Dev loses what Cara owed him and keeps what the other two owed him. The damage is proportionate and contained.

After netting, Dev has one exposure: to the arrangement. If any participant fails, the amount Dev receives changes, and by an amount that has nothing to do with his direct dealings with the party that failed. He has traded a set of small, understandable, diversified exposures for a single large exposure to a mechanism.

That is not a criticism of netting. It is the deal. Netting converts credit risk that was spread across many counterparties into liquidity risk concentrated in one moment, and the entire architecture of modern payment system regulation exists to manage that one moment. The technical version works through exactly what happens when it goes wrong.

The technical version

Vocabulary

The Committee on Payments and Market Infrastructures, the standard-setting body at the Bank for International Settlements, defines clearing as "the process of transmitting, reconciling and, in some cases, confirming transactions prior to settlement, potentially including the netting of transactions and the establishment of final positions for settlement". Netting is therefore a sub-process of clearing, not a synonym for it, and the qualifiers matter: "in some cases", "potentially including". A system can clear without netting.

The working definitions used throughout the rest of this book are as follows.

Gross settlement means each payment obligation is settled individually, for its full amount. **Net settlement** means obligations are offset against each other and only the resulting balance is settled.

Bilateral netting offsets obligations between one pair of participants. Each pair produces one net figure and one settlement movement.

Multilateral netting offsets each participant's obligations against all other participants simultaneously. Each participant produces one net figure: a **net debit position** if it owes the system, a **net credit position** if the system owes it. The sum of all net debit positions equals the sum of all net credit positions, always, by construction.

Deferred net settlement (DNS) means the netted figure is settled at a scheduled time after the underlying payments have been exchanged. **Real-time gross settlement** (RTGS) means each payment is settled individually as it arises. The Bank of England currently supports four settlement models in its RTGS infrastructure: real-time gross settlement, delivery versus payment, prefunded deferred net settlement and unfunded deferred net settlement. Only CHAPS uses the first. The compression achieved is expressed as a **netting ratio**: the netted value as a proportion of the gross value, or its complement, the percentage of value eliminated.

Four banks, worked in full

Take four settlement participants and one clearing cycle. The matrix below shows the total value of payment instructions sent by each bank to each other bank during the cycle, in millions of pounds. Rows are payers, columns are payees.

Sending bank	To Northbank	To Eastbank	To Southbank	To Westbank	Total sent
Northbank	—	420	180	260	860
Eastbank	380	—	310	150	840
Southbank	210	290	—	340	840
Westbank	160	120	270	—	550

Sending bank	To Northbank	To Eastbank	To Southbank	To Westbank	Total sent
Total received	750	830	760	750	3,090

Twelve directional flows. Aggregate gross value: **£3,090 million**.

If this system settled gross, twelve settlement movements totalling £3,090 million would be required, and each bank would need enough liquidity in its settlement account to cover its full outgoing total: £860 million for Northbank, £840 million each for Eastbank and Southbank, £550 million for Westbank. Aggregate liquidity requirement: £3,090 million.

Bilateral netting

Offset each pair.

Pair	Owed one way	Owed the other	Net payer	Net amount
Northbank / Eastbank	420	380	Northbank	40
Northbank / Southbank	180	210	Southbank	30
Northbank / Westbank	260	160	Northbank	100
Eastbank / Southbank	310	290	Eastbank	20
Eastbank / Westbank	150	120	Eastbank	30
Southbank / Westbank	340	270	Southbank	70

Six settlement movements totalling **£290 million**. That is 9.4 per cent of the gross value: bilateral netting has eliminated 90.6 per cent of the value that would otherwise have had to move. Note the liquidity consequence. Northbank must fund £40 million to Eastbank and £100 million to Westbank, £140 million in all, while separately receiving £30 million from Southbank, and it cannot use the incoming £30 million to fund the outgoing £140 million, because bilateral netting produces six independent settlements rather than one.

Multilateral netting

Now offset everything against everything.

Bank	Total sent	Total received	Net position
Northbank	860	750	Net debit 110
Eastbank	840	830	Net debit 10
Southbank	840	760	Net debit 80
Westbank	550	750	Net credit 200

Northbank, Eastbank and Southbank are debited £110 million, £10 million and £80 million respectively. Westbank is credited £200 million. Debits sum to £200 million; credits sum to £200 million.

Four postings to settlement accounts, moving **£200 million**. That is 6.5 per cent of the gross value: multilateral netting has eliminated 93.5 per cent.

The three side by side

Settlement basis	Movements	Value settled	Percentage of gross	Largest single obligation
Gross	12	£3,090m	100.0%	£420m
Bilateral net	6	£290m	9.4%	£100m
Multilateral net	4	£200m	6.5%	£110m

Multilateral netting beats bilateral netting here, as it almost always does, because it allows a claim on one participant to fund an obligation to a different participant. Northbank's £30 million claim on Southbank reduces its aggregate obligation from £140 million to £110 million, which bilateral netting cannot achieve. The efficiency grows with the number of participants and with how balanced the flows are, and is weakest when one participant is a persistent net sender to everybody.

What netting buys, in real numbers

The four-bank example is illustrative. The following are not.

Bacs and Faster Payments

Pay.UK, which operates Bacs, the Faster Payment System and the Image Clearing System, reported that in 2024 Bacs processed 6,811 million payments worth £5,838,567 million, an average of 26.8 million payments and £22,986 million per working day. In 2025 the figures rose to 6,864 million payments worth £6,049,248 million, an average of 27.1 million payments and £23,910 million a day.

The Bank of England, which settles those obligations, reported that daily average Bacs net settlement values in RTGS in 2024 were £5,265 million.

So roughly £23 billion of customer payment value per working day was discharged by moving roughly £5.3 billion between settlement accounts. Netting eliminated of the order of three quarters of the value. The two figures come from different publishers and may be computed over slightly different numbers of days, so the ratio should be read as approximately 77 per cent eliminated rather than to three significant figures. The order of magnitude is not in doubt.

The compression in the number of movements is more dramatic still. At the end of 2025 Bacs had 33 direct customers, over 330 indirect customers and around 117,000 service users submitting payment files. Twenty-seven million payment instructions a day, originating from a hundred and seventeen thousand organisations, resolve to at most thirty-three postings in the Bank of England's ledger, once per business day.

Faster Payments compresses harder. Pay.UK reported total Faster Payments value of £4,242,380 million in 2024, while the Bank of England reported daily average Faster Payments net settlement values of £1,929 million. Faster Payments settles three times each business day, so the annual settled value is of the order of £490 billion against gross value of £4,242 billion. Netting is eliminating something in the region of 88 per cent.

The reason Faster Payments nets harder than Bacs is instructive. Bacs traffic is dominated by direct credits — payroll and benefits — which flow overwhelmingly in one direction on particular days of the month. Faster Payments traffic is person-to-person and person-to-business, running in both directions continuously. Balanced flows net well. Directional flows do not.

For comparison, the Bank of England's 2024 daily averages across the whole of RTGS were:

Settlement service	Daily average value 2024
CHAPS (gross)	£344,409m
CREST delivery versus payment	£391,715m
Bacs (net)	£5,265m
Visa Europe (net)	£2,193m
Faster Payments (net)	£1,929m
Mastercard (net)	£1,406m
LINK (net)	£241m
Image Clearing System (net)	£44m
Total	£747,202m

CHAPS settles gross and therefore appears at its full value. Every other retail system in the table has already been compressed before it reaches the Bank of England. That single table is the clearest statement in print of what netting does to a national payment system: seven retail systems carrying the overwhelming majority of the country's payment volume account for about 1.5 per cent of the value settled.

Securities: the DTCC

In the United States, the National Securities Clearing Corporation, part of the Depository Trust and Clearing Corporation, clears the great majority of US equity trades. DTCC states that NSCC netting reduces the value of payments that need to be exchanged by an average of 98 to 99 per cent every day.

That is a higher ratio than any payment system achieves, and the reason is structural: securities trading involves the same instruments changing hands many times in a day between the same set of intermediaries, so the offsetting is extraordinarily dense. NSCC's continuous net settlement service reduces each member's obligations in each security to a single net long or short position. For scale, DTCC reported that on 9 April 2025 NSCC reached a peak value of $5.55 trillion, against a previous peak of $5.22 trillion on 20 December 2024.

Foreign exchange: CLS

CLS Bank settles foreign exchange on a payment-versus-payment basis in eighteen currencies, holding accounts with all eighteen central banks. It settles the gross value of each instruction, but funds the settlement on a multilaterally netted basis: each settlement member transfers only the net amount of its combined obligations in each currency.

CLS reports that this, combined with its in/out swap liquidity tool, produces an average funding requirement of less than 1 per cent of the total value of all trades for participating settlement members. In the first half of 2022 CLS settled an average of over $6.5 trillion every day, with over 70 settlement members and over 30,000 third-party participants.

Less than one per cent. That is the number to remember when somebody claims that netting is a marginal optimisation.

The 2025 BIS Triennial Central Bank Survey put a broader figure on the same phenomenon. More than $14 trillion of gross foreign exchange obligations were settled on an average day in April 2025. More than $2 trillion of that, about 15 per cent, was subject to pre-settlement netting, and that $2.2 trillion was reduced to $337 billion of actual settlement — a reduction of roughly 85 per cent on the netted portion alone.

The legal forms of netting

The distinctions below are not academic. They determine whether a netting arrangement survives an insolvency, and therefore whether a bank may recognise net rather than gross exposures for regulatory capital purposes.

Position netting, sometimes called payment netting or settlement netting, computes a net figure for settlement purposes but does not legally discharge the underlying obligations. The gross obligations continue to exist until they are settled. If a participant fails before settlement, the net calculation has no independent legal force and the gross obligations are what an administrator sees. This is the form that gives rise to unwind risk.

Netting by novation replaces each new obligation between a pair of parties with a single new obligation that discharges and supersedes the previous one. At any moment there is exactly one obligation between the pair, of the net amount. Because the gross obligations have been legally extinguished as they arose, there is nothing for an administrator to cherry-pick.

Close-out netting operates on default rather than on schedule. On a defined termination event, all outstanding contracts between the parties are terminated, valued at market, and reduced to a single net sum payable one way. This is the mechanism in section 6 of the ISDA Master Agreement and it is why the Master Agreement insists that all transactions under it form a single agreement.

Multilateral netting cannot be achieved by novation between pairs alone, because there is no bilateral contract between Northbank and the system. It requires either a central counterparty that becomes the legal counterparty to everybody, or statutory protection of the sort the Settlement Finality Regulations 1999 confer on designated systems, or both.

Where the risk goes: the unwind, worked through

Return to the four banks and suppose Northbank fails during the cycle, before settlement, and cannot meet its net debit of £110 million.

In a system with position netting and no prefunding, the classic remedy is to remove the failed participant's payments and recalculate. This is called an unwind, and it is worth seeing what it does.

Delete every flow to and from Northbank. The remaining matrix is:

Sending bank	To Eastbank	To Southbank	To Westbank	Total sent
Eastbank	—	310	150	460
Southbank	290	—	340	630
Westbank	120	270	—	390
Total received	410	580	490	1,480

Recalculate the net positions and compare.

Bank	Original net position	Position after unwind	Change
Eastbank	Debit 10	Debit 50	Must find £40m more
Southbank	Debit 80	Debit 50	Owes £30m less
Westbank	Credit 200	Credit 100	Receives £100m less

Look carefully at what has happened.

Eastbank had nothing to do with Northbank's failure in any meaningful sense — its bilateral position with Northbank was a net £40 million payable. Yet its settlement obligation has increased fivefold, from £10 million to £50 million, and it must find the difference at the moment of settlement, with no notice. Southbank's obligation has fallen. Westbank, which had planned its day around receiving £200 million, receives £100 million.

Two properties of that table are the reason central banks stopped tolerating unwind as a risk control. First, the direction and size of the change bear no reliable relationship to a participant's dealings with the failed party; Eastbank cannot manage its exposure to an unwind by managing its exposure to Northbank, because the effect propagates through everybody else's positions too. Second, the change is discovered at exactly the worst moment: a bank learns that it needs £40 million more, in central bank money, at the instant a major counterparty has failed and every other bank in the country is scrambling for liquidity. That is the mechanism by which a single failure becomes a system-wide liquidity event.

The rules written in response

In November 1990 the central banks of the Group of Ten published the Report of the Committee on Interbank Netting Schemes, universally known as the Lamfalussy Report after its chairman, Alexandre Lamfalussy. It set six minimum standards for netting schemes, which remain the foundation of every rule written since.

In summary, the six standards require that netting systems have a well-founded legal basis in all relevant jurisdictions; that participants clearly understand the effect of the scheme on each financial risk; that multilateral netting systems have clearly defined procedures for managing credit and liquidity risk, specifying who is responsible for what and placing limits on the maximum credit exposure any participant can create; that multilateral netting systems be capable, at a minimum, of ensuring timely completion of daily settlement in the event of an inability to settle by the participant with the largest single net debit position; that admission criteria be objective, publicly disclosed and permit fair and open access; and that the operational reliability of technical systems and the availability of backup facilities be assured.

The fourth is the one that changed the industry. It is the requirement now universally known as "cover one": the system must be able to complete settlement on time even if its single largest debtor cannot pay. Everything below follows from it.

Net debit caps and net sender caps. A cap limits how large a net debit position a participant may build up between settlements. When a participant reaches its cap it can no longer send payments until incoming payments reduce its position or settlement occurs. In the Faster Payment System these are called Net Sender Caps, and Pay.UK announced in July 2026 a move away from a prescriptive formula, based on a Minimum Net Sender Cap and a Peak Contingency Value, towards participants setting their own caps against quarterly plans reviewed by Pay.UK. The purpose is unchanged: to bound the size of the hole any single participant can leave.

Prefunding. The Bank of England introduced prefunding for Bacs and Faster Payments in September 2015 to address the settlement risk arising from the build-up of obligations in deferred net systems. Under prefunding, each settlement participant holds cash in a segregated prefunding account in RTGS equal to its cap. If it defaults, the cash set aside is used to complete settlement. As the Bank puts it, this eliminates credit risk between settlement participants: the money is already there, in central bank money, before the obligation arises.

Prefunding is the reason the modern British answer to "what if a bank cannot settle its Bacs position" is no longer "unwind the cycle". It is "use the money it already put aside".

Loss-sharing arrangements. Where prefunding is not used, systems typically operate survivors-pay arrangements: a defaulter's shortfall is allocated among the remaining participants according to a pre-agreed formula, usually related to their bilateral exposures to the failed party. These arrangements produce certainty of settlement completion at the cost of mutualised loss, and their existence is one of the reasons access criteria to net settlement systems are strict.

Central counterparties in outline

A central counterparty is the industrial version of the kitchen table. It is worth understanding in outline because it is where netting is taken to its logical conclusion, and because central counterparties are now the largest single concentration of financial risk in the world.

Novation

The defining act of a CCP is novation. When two parties agree a trade and submit it for clearing, the original contract between them is legally extinguished and replaced by two new contracts: one between the buyer and the CCP, and one between the CCP and the seller. The CCP becomes buyer to every seller and seller to every buyer.

That single legal move accomplishes two things at once. It makes multilateral netting legally trivial, because every participant now has exactly one counterparty and can net everything it has against that one counterparty. And it removes the need for any participant to assess the creditworthiness of any other participant, because it deals only with the CCP.

Margin and the default waterfall

Because everybody's credit risk is now concentrated in one place, that place must be made extremely difficult to break. CCPs do this with a layered stack of financial resources, consumed in a fixed order known as the default waterfall.

Initial margin is collateral posted by each clearing member against the potential future loss on its portfolio, sized to cover the likely movement in that portfolio over the period it would take to close it out. LCH's SwapClear service calculates initial margin using its PAIRS methodology — Portfolio Approach to Interest Rate Scenarios — drawing on ten years of historical market data.

Variation margin is collected daily, and intraday where required, to settle the actual mark-to-market change in each portfolio. Variation margin is not collateral; it is payment of losses as they accrue, which prevents losses from accumulating.

The default fund is a mutualised pool contributed by all clearing members.

Skin in the game is a tranche of the CCP's own capital, deliberately placed in the waterfall ahead of the surviving members' contributions so that the CCP shares the loss before its members do.

At LCH the consumption order is the defaulting member's posted margin, then the defaulter's default fund contribution, then LCH's own capital, then the non-defaulting members' default fund contributions. If all of that is exhausted, variation margin gains haircutting is applied as a final measure: members who have made gains have those gains cut to fill the hole.

The international standard governing the size of these resources is the CPMI-IOSCO Principles for Financial Market Infrastructures, published in 2012. Principle 4 requires an FMI to maintain financial resources sufficient to cover the default of the participant to which it has the largest exposure in extreme but plausible market conditions — cover one — and requires CCPs involved in activities with

a more complex risk profile, or that are systemically important in more than one jurisdiction, to cover the two largest participants and their affiliates. That is the origin of "cover two".

The default management process, and the one time it was tested at scale

When a clearing member defaults, a CCP does not simply liquidate. LCH's SwapClear process runs in three stages: porting non-defaulting clients to surviving clearing members; neutralising the residual risk by hedging, with the assistance of a default management group drawn from member firms; and then auctioning the defaulter's portfolio, split by product and currency, to the surviving members.

The definitive test was Lehman Brothers. At default, Lehman's cleared interest rate derivatives portfolio at LCH comprised 66,390 trades with a notional value of $9 trillion, against which it had posted initial margin of the order of $2 billion. LCH completed the default management within about three weeks, using 100 per cent of the variation margin posted by the defaulting member and only 35 per cent of its initial margin. No mutualised default fund resources were consumed and no surviving member lost money.

That outcome is the strongest argument ever made for central clearing, and it is the reason the G20 leaders committed at Pittsburgh in September 2009 to central clearing of standardised over-the-counter derivatives, a commitment implemented in Europe through the European Market Infrastructure Regulation and retained in UK law after withdrawal from the European Union.

The concentration objection

The obvious criticism follows directly from the plain version of this chapter. Netting concentrates risk, and a CCP is netting taken to its limit. A CCP that fails does not fail one counterparty; it fails an entire market simultaneously, and the markets it clears may have no functioning alternative.

This is not a fringe concern. It is why CCPs are supervised directly by central banks — in the United Kingdom, by the Bank of England, which supervises central counterparties alongside central securities depositories and recognised payment systems — why the PFMI impose cover-two requirements, why CCP recovery and resolution regimes exist, and why the question of who bears loss beyond the waterfall is among the most contested in financial regulation.

One institutional fact captures the point. LCH Ltd is a CHAPS Direct Participant: a clearing house sitting directly on the Bank of England's real-time gross settlement system, alongside the high street banks, because when a clearing house needs to move money it cannot afford to depend on anybody else's netting.

Why Bacs takes three days

We can now answer the question properly.

The Bacs cycle has three named days, and the scheme documentation is precise about them.

Day 1, input day. This is the latest day on which a service user or bureau may submit a payment file for a given processing cycle. Files must be transmitted between 07:00 and 22:30. Submissions may be made up to 30 days in advance of the payment date, and a payment can be recalled after submission provided the submitter's payment service provider is notified before a specified cut-off. Through the day, submitted data is validated and sorted into bank order by the central infrastructure.

Day 2, processing day. The sorted files are delivered to the recipient payment service providers, which process each payment into their own systems.

Day 3, entry day. Payments are simultaneously credited to recipients' accounts and debited from the payer's account. Interbank settlement of the multilateral net positions takes place at the Bank of England once that business day.

Now: what does each day buy?

The netting window. Netting only compresses obligations that coexist. A wider window means more offsetting. The Bacs cycle accumulates a full day of national payroll, benefits, utility collections and supplier payments before it computes anybody's position, and the result is the compression already quoted: about £23 billion of gross value discharged with about £5.3 billion of settlement.

Simultaneity of debit and credit. This is the point most commentary misses entirely. Bacs is often described as "three days of the money being in limbo". It is not. Nothing leaves the payer's account on day 1 or day 2. The debit and the credit both occur on day 3, at the beginning of the operating day, and the interbank settlement occurs on the same day. The three-day figure is the length of a pipeline, not the length of an exposure. That is a materially different risk profile from a system where the credit happens instantly and settlement happens later.

The exception window. Direct debits and direct credits can go wrong in specific, catalogued ways: a closed account, an insufficient balance, a cancelled instruction, a changed account number. Bacs provides a set of automated advice services for exactly these cases — ADDACS for direct debit amendments and cancellations, AUDDIS for mandate lodgement, ARUCS and AWACS for returned and amended credits. Those services require a defined window in which the receiving bank can inspect an item, decide whether it can be applied, and generate a report back to the originator. A cycle with no processing day has nowhere to put that work.

Predictability, for both sides. An originator running payroll for 12,000 people needs to know precisely which day the money lands, weeks in advance; a fixed cycle with a published calendar delivers that, and "as fast as possible" does not, because it is not a date. A bank, equally, that knows its Bacs settlement occurs once a day on a known set of items can forecast its central bank money position, arrange collateral and fund accordingly, which is far cheaper than holding a buffer against a continuous unpredictable drain.

So is three days a design choice or a defect?

The honest answer is: both, and the proportions have changed over time. Bacs began in 1968 as the Inter-Bank Computer Bureau, in an era when payment files travelled between banks as reels of magnetic tape moved by van. Three days had a physical meaning then that it does not have now, and it would be dishonest to pretend the current cycle length was derived from first principles in the present decade.

But the evidence for it being load-bearing rather than merely vestigial is strong, and it is this: when the United Kingdom decided it needed same-day retail payments, it did not speed Bacs up. It built the Faster Payment System alongside it in 2008, and left Bacs running exactly as it was. Seventeen years later Bacs is still carrying 27 million payments a day and, in 2025, 5.03 billion direct debits — an all-time annual high for the scheme.

That is what a design choice looks like from the outside. Two systems, two different points on the same trade-off curve, both running at national scale, because the trade-off is real and different users sit at different points on it.

	Bacs	Faster Payments	CHAPS
Settlement basis	Deferred multilateral net	Deferred multilateral net	Real-time gross
Settlement frequency	Once each business day	Three times each business day	Continuously
Customer credit timing	Day 3 of cycle	Within seconds, 24/7	Within the day
Value cap	None	£1,000,000	None
Liquidity cost to banks	Lowest	Low	Highest
Settlement risk between banks	Managed by prefunding	Managed by prefunding and net sender caps	Eliminated
Typical use	Payroll, benefits, direct debits	Person-to-person, bill payment	House purchase, wholesale, treasury

Read that table across and the logic is visible. CHAPS eliminates settlement risk and pays for it in liquidity. Bacs eliminates most of the liquidity requirement and pays for it in time and in the machinery — caps, prefunding, cycle discipline — needed to keep the risk contained. Faster Payments sits in between, buying speed for the customer while keeping the netting benefit for the banks, and paying for it with a value cap and three settlement cycles a day instead of one.

There is no free position on that curve. There never was. Netting is the mechanism by which the industry chooses where on it to sit, and every timing behaviour that irritates the public — the three days, the cut-off, the settlement that happens on Monday for a payment made on Saturday — is a consequence of that choice being made deliberately, by people who understood what they were trading away.

The next chapter leaves the interbank layer entirely and goes down to something much smaller and much more likely to break your code: what a currency amount actually is, and why the type you use to hold it matters more than almost anything else you will write.

8.98 Common wrong ideas

Wrong: Netting means somebody gets paid less. **Right:** Netting operates only on the obligations between banks; the customer layer stays gross and every payment is credited and debited in full.

Wrong: Netting is just arithmetic. **Right:** It is a legal act, and without designation under the Settlement Finality Regulations 1999 an administrator could cherry-pick the gross obligations and ignore the net figure.

Wrong: There is a pot in the middle holding the netted money. **Right:** In most net settlement systems the operator sends positions to the Bank of England, which debits and credits in a single simultaneous posting with nothing held anywhere in between.

Wrong: Netting reduces risk. **Right:** It concentrates it, replacing several small diversified credit exposures with one large exposure to a mechanism and to a single moment.

Wrong: A bank with little exposure to the failed participant is safe in an unwind. **Right:** In the worked example Eastbank's obligation rises fivefold, from £10 million to £50 million, because the effect propagates through everybody else's positions.

Wrong: Bilateral and multilateral netting come to much the same thing. **Right:** Multilateral netting lets a claim on one participant fund an obligation to another, taking the four-bank example from £290 million down to £200 million.

Wrong: Unwinding the cycle is the modern remedy when a participant cannot settle. **Right:** Since September 2015 the British answer for Bacs and Faster Payments has been prefunding: use the money the participant already put aside.

Wrong: Bacs means the money sits in limbo for three days. **Right:** Nothing leaves the payer's account on Day 1 or Day 2; the debit, the credit and the interbank settlement all fall on Day 3.

Wrong: Three days is purely a relic of magnetic tape moved around by van. **Right:** It is both relic and design choice, and the proof is that when Britain wanted same-day retail payments it built Faster Payments alongside Bacs in 2008 rather than speeding Bacs up.

Wrong: A central counterparty takes risk out of the market. **Right:** It removes bilateral credit assessment and makes multilateral netting legally trivial, but it becomes the largest single concentration of risk, which is why cover-two requirements and CCP resolution regimes exist.

8.99 Chapter summary in 20 lines

1. On an average working day in 2024 Bacs carried about 26.8 million payments worth roughly £23.0 billion, and about £5,265 million actually moved between settlement accounts.
2. Nobody was short-changed, because netting operates on the obligations between banks and never on the payments between customers.
3. Bilateral netting offsets obligations one pair at a time; multilateral netting offsets each participant against all the others simultaneously.
4. In the four-flatmate example, twelve debts and £102 of cash became four movements and £18 on the kitchen table, with every debt discharged in full.
5. The sum of all net debit positions always equals the sum of all net credit positions, because every debt is counted once as a plus and once as a minus.
6. In the four-bank matrix, £3,090 million of gross flows became £290 million bilaterally and £200 million multilaterally, or 9.4 and 6.5 per cent of gross.
7. Multilateral netting wins because a claim on one participant can fund an obligation to a different one, which bilateral netting cannot achieve.
8. Netting works best when flows are balanced, which is why Faster Payments eliminates something like 88 per cent of value while directional Bacs payroll traffic eliminates about 77 per cent.
9. The longer the window before squaring up, the more cancels out, so waiting is itself part of the compression.
10. Money in a settlement account is money doing nothing else, so netting is the difference between a payment system a country can afford to run and one it cannot.
11. The cost is that netting concentrates risk: each participant swaps several small exposures for one large exposure to a calculation.
12. Netting is a legal act as well as an arithmetical one, and only statutory protection stops an administrator cherry-picking the gross obligations.
13. Position netting leaves the gross obligations legally alive and gives rise to unwind risk, while netting by novation and close-out netting extinguish them.
14. Multilateral netting cannot be built from bilateral novation alone and needs either a central counterparty or statutory designation, or both.

15. An unwind redistributes obligations unpredictably, and it does so at the worst possible moment, when every bank in the country is already scrambling for liquidity.
16. The Lamfalussy Report of November 1990 set six minimum standards, of which cover one — settling on time despite the largest single debtor failing — changed the industry.
17. That standard is implemented through net debit and net sender caps, through prefunding introduced for Bacs and Faster Payments in September 2015, and through survivors-pay loss sharing where prefunding is absent.
18. A central counterparty is netting taken to its limit: novation makes it buyer to every seller and seller to every buyer, defended by initial margin, variation margin, its own skin in the game and a mutualised default fund.
19. LCH resolved Lehman's 66,390 cleared interest rate trades of $9 trillion notional in about three weeks, using 35 per cent of the posted initial margin and no mutualised default fund resources at all.
20. Bacs takes three days because the netting window, the exception-handling window and a published calendar are worth more to its users than speed, and Britain proved the trade-off was real by building Faster Payments alongside it rather than instead of it.

Sources used: Bank of England, "Payment and settlement" and "A brief introduction to the Real-Time Gross Settlement system and CHAPS" (including Table A, daily average RTGS settlement values and volumes 2024); Bank of England, "Bank of England Settlement Accounts" and "RTGS and CHAPS Service Description", December 2018; Pay.UK, "Annual summary of payment statistics 2025"; Pay.UK, "Bacs Payment System" and "Understanding changes to the Faster Payment System Net Sender Caps"; Bacs, "Bacs Direct Credit — Getting started" (processing cycle); Committee on Payments and Market Infrastructures, Glossary, and Report of the Committee on Interbank Netting Schemes (the Lamfalussy Report), November 1990; CPMI-IOSCO, Principles for Financial Market Infrastructures, 2012; CPMI Red Book, "Payment, clearing and settlement systems in the United Kingdom"; The Financial Markets and Insolvency (Settlement Finality) Regulations 1999, SI 1999/2979, and Directive 98/26/EC; DTCC, UST1 FAQs and press release of 23 April 2025; CLS Group, "What are the necessary ingredients for any PvP arrangement to work?"; BIS Quarterly Review, "Uncovering FX settlement risk: new measures from the 2025 BIS Triennial Survey"; LSEG, LCH SwapClear risk management; CCP Global, "The Lehman Case".

Currency, Decimals and the Money Type

9.0 What this chapter gives you

1. You will be able to explain why ten 10p coins added together as pounds come to 0.9999999999999999, and why a till written to notice exactly £1.00 will never notice.
2. You will be able to say why multiplying a yen amount by 100 charges the cardholder a hundred times too much, and why a Kuwaiti amount loses its last digit in a two-decimal system.
3. You will be able to derive a currency's minor unit from ISO 4217 rather than assuming 100, and name the exceptions: seventeen zero-decimal codes, seven three-decimal, two four-decimal and thirteen with no minor unit at all.
4. You will be able to show that the 42-pence gap across 3,000 merchant fees came from an unwritten decision about where the rounding line falls, and not from a data type.
5. You will be able to split a total across parts by the largest remainder method so that the parts always sum back to the total exactly.
6. You will be able to say why bankers' rounding is not "the correct rounding for money", and cite the euro conversion rule and the VAT Notice 700 treatments that override it.
7. You will be able to explain why a JSON number with a decimal point is the one wire convention that guarantees eventual trouble, and name the two that do not.
8. You will be able to carry one internal representation — an integer count of minor units with an explicit currency — across ISO 8583, ISO 20022, SWIFT MT and Bacs Standard 18.
9. You will be able to record a foreign exchange conversion so that an independent party can re-derive it, and say why a reference rate is not a transaction rate.
10. You will be able to name the eight recurring failure modes of money handling and the single test that catches each one.

There is a category of bug that does not crash anything. The service stays up, the tests pass, and every individual transaction looks correct when you open it. Then, at the end of the month, the settlement report is three pence away from the bank statement and nobody can say why. Three pence is not a material amount of money. Explaining three pence to an auditor, and proving that it is not in fact three pence multiplied by every account you have not checked yet, can consume a fortnight.

This chapter is about the machinery that produces those three pence, and how to build a system in which they cannot occur. It is the most directly practical chapter in this volume for anyone who writes code, and it is also the chapter that finishes the volume's argument. Chapter two established that a balance is a record rather than a substance. Chapter three established that every payment is two facts rather than one. This chapter establishes what kind of thing that record can be made of, and the answer is surprisingly constrained. Money is not a number. Money is a number, plus a currency, plus

a rule about how finely that currency may be cut, plus a written decision about what happens when a calculation lands between two of the permitted cuts.

Leave out any one of those four and you have not built a ledger. You have built something that usually agrees with a ledger.

The plain version

Think about a school tuck shop. Everything is priced in whole pennies: a chocolate bar at 65p, a packet of crisps at 48p, a drink at £1.20. At the end of the day the person running the shop tips the cash box onto the table and counts.

Nobody counts in pounds. They count in coins — four £1 coins, seven 50p pieces, twenty-three 20p pieces — and then convert everything into a single number of pennies, because pennies are the smallest thing the shop can actually receive. Nine hundred and forty-two pennies. Only at the very end, writing the figure on the sheet, does anybody turn 942 into "£9.42".

That is the whole trick, and it is worth saying plainly because almost every serious money system in the world works this way. **Count in the smallest coin. Convert to pounds only when you print something a human will read.**

Why not just count in pounds?

Because computers cannot hold "one tenth" exactly, and this is not a matter of them being sloppy. It is a matter of the ruler they use.

Imagine a ruler with no centimetre marks. Instead, someone has marked the halfway point, then the halfway points of each half, then the halfway points of those, and so on forever. You can mark exactly one half, one quarter, three eighths, eleven sixteenths. You can get extremely close to one tenth — closer than any eye could see. But you can never land exactly on it, because one tenth is not reachable by halving.

The ordinary fractional numbers a computer uses are built on exactly that ruler. And so when a computer stores the number 0.1, it does not store one tenth. It stores the nearest mark on the halving ruler, which is:

0.1000000000000000055511151231257827021181583404541015625

That is a real number, not an exaggeration for effect. It is the exact value your laptop puts in memory when you type `0.1`. Now watch what happens. Ask a computer to add 0.1 and 0.2 and it produces:

0.3000000000000000444089209850062616169452667236328125

which it will politely display to you as `0.30000000000000004`. Ask it whether 0.1 plus 0.2 equals 0.3, and it says no.

Try something even more homely. Put ten 10p coins into a till, one at a time, using pounds:

0.1, then 0.2, then 0.30000000000000004, then 0.4, then 0.5, then 0.6, then 0.7, then 0.7999999999999999, then 0.8999999999999999, and finally **0.9999999999999999**.

Ten ten-pence coins have not made a pound. They have made a hair less than a pound. If the till software is written to notice when the drawer contains exactly £1.00, it will never notice.

Count in pennies instead and the problem does not arise. Ten lots of 10 is 100. Whole numbers on a computer are exact. There is nothing to go wrong.

Not every country cuts money the same way

Here is the second idea, and it catches people out constantly because it is invisible from Britain. We assume every currency has a hundred small units in a big one. A pound has 100 pence, a dollar has 100 cents, a euro has 100 cents. So a programmer writes `pence = pounds * 100` and moves on.

Japan does not work like that. The yen has no subdivision in use. A price of 2,500 yen is 2,500 yen; there is no such thing as 2,500.00. If you take a ¥2,500 payment and helpfully multiply by 100 before sending it, you have just asked for ¥250,000 — a hundred-fold error, and it will go through, because the number is perfectly valid.

Kuwait goes the other way. The Kuwaiti dinar is divided into 1,000 fils, so amounts carry three decimal places: 12.345 dinars is an ordinary price. Bahrain, Jordan, Oman, Iraq, Libya and Tunisia do the same. If your system assumes two decimal places, it will quietly shave the last digit off every Kuwaiti amount it touches.

So the rule is not "multiply by 100". The rule is "look up how many places this particular currency has, and multiply by that". There is an official list. It is called ISO 4217, and it gives every currency a three-letter code — GBP, USD, JPY, KWD — and tells you how many decimal places that currency uses.

A number on its own is not money

Which brings us to the third idea. "£19.99" is money. "19.99" is not; it is a quantity of something unspecified. This sounds like pedantry and it is the single most useful discipline in the chapter. If a value in your system can be 19.99 without also carrying the letters GBP, then sooner or later somebody will add it to a euro amount, or send it to Japan, or store it in a column another team reads as dollars. The currency is not decoration on the amount. It is part of the amount, the way "miles" is part of "sixty miles".

Splitting things up

The last idea is the one that produces the three pence.

Three friends split a £5.00 bill. Five divided by three is £1.6666... Round each share to the nearest penny and each pays £1.67. Three times £1.67 is £5.01: they have paid a penny more than the bill.

Try it the other way. Three people split £100.00. A hundred divided by three is £33.3333... Round each to the nearest penny and each pays £33.33. Three times £33.33 is £99.99. A penny has vanished.

There is no clever arithmetic that avoids this. Five pounds genuinely does not divide into three equal whole numbers of pennies. What a well-built system does is not "avoid the problem" — it is **decide, in advance and in writing, who gets the odd penny, and then make sure the parts always add back up to the whole**. In the £5.00 case: two people pay £1.67 and one pays £1.66. In the £100.00 case: two pay £33.33 and one pays £33.34.

That is it. Count in the smallest coin. Look up how many small coins are in a big one for that particular currency. Never let an amount travel without its currency. And when you divide a total, force the pieces to add back up to it.

Where the plain version stops being true

The plain version is the right instinct and it will keep a small system honest for years. Four things about it are wrong in ways that show up at scale, and the fourth is the one that costs money.

"Count in whole pennies" is only true at the settlement boundary

The tuck-shop picture implies that every number in a financial system is a whole number of minor units. It is not, and insisting that it is will produce worse errors than it prevents.

Plenty of legitimate financial quantities are finer than a penny. Petrol in the United Kingdom is priced at, for example, 149.9 pence per litre. Interest accrues daily at rates like 4.37% per annum. Telecommunications and cloud services bill per second or per gigabyte at rates with four, six or eight decimal places. Foreign exchange rates carry six significant figures by law in the euro area. HM Revenue and Customs explicitly contemplates VAT calculated per article to three decimal places of a pound.

The correct statement is narrower and more useful. **The amount that settles must be a whole number of minor units. Intermediate quantities need not be, and the design question is where the rounding line falls.** A rate is not money. A price per unit is not money. A running accrual is not money until it is capitalised. Money is what leaves one account and arrives in another, and that has to be a whole number of the smallest thing the payment system can carry.

A system that rounds too early loses accuracy for no reason. A system that rounds too late presents a settlement instruction that no rail will accept. Both failures come from not writing down where the line is.

Floating point is not "imprecise" — it is exact in the wrong base, and that is worse

The plain version says computers cannot hold one tenth. True, but the framing invites the wrong conclusion: that the errors are random, tiny, and can be mopped up by rounding at the end.

First, binary floating point is not approximate. Every value it holds is exactly a whole number multiplied by a power of two, and the arithmetic is exactly specified and perfectly repeatable. The problem is that the set of values it can hold does not include the numbers a ledger is made of, so a calculation that ought to be exact acquires a small, deterministic, unpredictable-by-eye residue. That is worse than randomness, because it survives testing. Every one of your test cases will pass and the one combination you did not try will be off by a hair.

Second, and more practically: the damage almost never arrives as a visibly wrong total. It arrives in three other ways.

It arrives as a failed equality test. Debits of £1,234.56, £78.90, £4,321.00, £0.07 and £999.99 against a single credit of £6,634.52 plainly balance. Add the debits in binary floating point and you get 6634.5199999999995, so `total_debits == total_credits` returns false. Your double-entry validator, the one from chapter three that is supposed to be the last line of defence, now rejects a correct journal — or, if somebody has "fixed" it with a tolerance, now accepts an incorrect one.

It arrives as rounding that goes the wrong way. Ask a computer to round 1.15 to one decimal place and it returns 1.1, not 1.2 — not because the rounding rule is wrong but because the value actually stored for 1.15 is 1.1499999999999991118…, genuinely below the midpoint. The function did exactly what it was asked. The input was never what you thought.

And it arrives at serialisation boundaries, because JSON has no decimal type and a JSON number is in practice parsed into a binary double. An amount that was exact in your database becomes inexact the moment it crosses an API.

And here is the part that reframes the whole subject. In a run of 3,000 sales at £19.99, the floating-point error in the total is about 1.6 billionths of a penny — real, but not what will cost you money. Now apply a merchant fee of 1.4% plus 20p to each of those sales. If you round the fee per transaction, each is

27.986p plus 20p, rounded to 48p, and the total is £1,440.00. If you compute the fee on the whole run, it is £1,439.58. **The gap is 42 pence, and floating point had nothing to do with it.** It came entirely from an unwritten decision about where the rounding line falls. That is the shape of nearly every real money bug: a rounding policy nobody specified, not a data type nobody understood.

Minor units are a convention, and the convention is not universally agreed

The plain version implies that "how many decimal places does this currency have" is a fact you can look up once. It is a convention recorded by a standards body, and there are at least three ways it drifts from reality.

First, ISO 4217 itself rounds off awkward cases. The Malagasy ariary is notionally divided into five *iraimbilanja* and the Mauritanian ouguiya into five *khoums* — the only two circulating currencies whose subdivision is not a power of ten. ISO 4217 assigns both a minor unit of 2 anyway, because a field that holds an exponent has nowhere to record "one fifth". The standard is doing something sensible; it is simply not describing the currency.

Second, processors disagree with the standard, and with each other, for backwards-compatibility reasons. Stripe's published list of zero-decimal currencies includes MGA, which ISO 4217 gives a minor unit of 2. In the other direction, ISO 4217 gives both the Icelandic króna and the Ugandan shilling a minor unit of 0, but Stripe's documentation states that both "transitioned to a zero-decimal currency" while "backward compatibility requires you to represent it as a two-decimal value, where the decimal amount is always 00" — so to charge 5 ISK you submit 500. Stripe further treats the Hungarian forint and the New Taiwan dollar as zero-decimal for payouts while accepting two-decimal charges in them, requiring payout amounts to be evenly divisible by 100. None of this is wrong. All of it means "the minor unit" is a property of a currency *within a particular scheme's rules*, not of the currency alone.

Third, the smallest unit you can *account* in and the smallest unit you can *pay* in are different things. Several countries have withdrawn their smallest coin and round cash transactions to the nearest five units at the till while still recording electronic payments to the single unit. A ledger that models one granularity cannot represent the gap between the invoiced amount and the cash tendered.

The practical consequence: derive the exponent from a table you control and can version, validate it against the scheme you are actually talking to, and never from a hard-coded 100 or from whatever your standard library's locale data says this month.

Bankers' rounding is not "the correct rounding for money"

"Use bankers' rounding for money" circulates as settled wisdom. It is not settled, and applying it where the law says otherwise is a compliance problem rather than a style preference.

Round-half-to-even exists for a good reason: rounding halves consistently upwards introduces a systematic upward bias. Take the ten values 0.005, 0.015, 0.025 and so on up to 0.095. Their true sum is exactly 0.50. Round each half upwards and they sum to 0.55; round each to the nearest even last digit and they sum to 0.50. Over a large book of similar figures the half-up rule drifts and the half-even rule does not. That is a genuine and important property.

But it is a statistical argument, and statistics do not override statute. Council Regulation (EC) No 1103/97, which still governs conversion between the euro and the legacy national currencies it replaced, says in Article 5: "If the application of the conversion rate gives a result which is exactly halfway, the sum shall be rounded up." Not to even. Up. In the United Kingdom, VAT Notice 700 permits an invoice trader to round the *total* VAT on an invoice **down** to a whole penny — an explicit concession

in the taxpayer's favour — while stating that the same concession "is not appropriate to retailers", who "must not round the VAT figure down".

So there is no single rounding mode for money. There are modes mandated by specific regimes for specific calculations, and the job is to know which regime you are in. A system that applies half-even everywhere because a blog post recommended it will produce VAT figures HMRC's own guidance does not describe.

The second half of this correction is the one that generates reconciliation breaks. **Rounding each component and then adding is not the same as adding and then rounding.** Three items at £8.99 with VAT at 20%: each line's VAT is £1.798, which rounds to £1.80, so the invoice total from line rounding is £5.40. Compute VAT on the £26.97 order total and you get £5.394, which rounds to £5.39. Both figures are defensible. They are one penny apart. If your invoice engine uses one method and your settlement engine the other, you will be reconciling that penny for the life of the product.

The technical version

ISO 4217: the code, the number and the exponent

ISO 4217 is the international standard for representing currencies and funds; the current edition is ISO 4217:2015. The maintenance agency is SIX Financial Information AG, acting on behalf of the Swiss Association for Standardization, which publishes the authoritative tables and issues amendments as currencies are introduced, redenominated or withdrawn.

The standard defines three tables. **List One** contains the currently active currency and funds codes. **List Two** contains fund codes. **List Three** is the historical list of withdrawn codes — a list that exists precisely because currency codes are not permanent, and a system that stores only a code, with no effective date, cannot always reconstruct what an old record meant.

Each entry in List One carries the country or entity, the currency name, a three-character alphabetic code, a three-digit numeric code, and the minor unit — the number of decimal places conventionally used. The alphabetic code is normally the two-letter ISO 3166-1 country code followed by the initial of the currency name: GB plus P gives GBP, US plus D gives USD, JP plus Y gives JPY. Supranational and non-country codes begin with X, which ISO 3166-1 reserves for exactly this purpose. That is why gold is XAU, silver XAG, platinum XPT and palladium XPD; why the IMF's Special Drawing Right is XDR; why the CFA francs are XOF and XAF; and why XTS is reserved for testing and XXX denotes "no currency involved". The three-digit numeric code exists because a great deal of financial messaging is numeric-only: GBP is 826, USD is 840, EUR is 978, JPY is 392, INR is 356, KWD is 414.

As published on 1 January 2026, List One contains 178 distinct alphabetic codes spread over 280 country-and-currency rows. The overwhelming majority — 139 codes — have a minor unit of 2. The exceptions are the ones your code has to handle.

Minor unit	Count	Codes (with numeric code)
0	17	BIF 108, CLP 152, DJF 262, GNF 324, ISK 352, JPY 392, KMF 174, KRW 410, PYG 600, RWF 646, UGX 800, UYI 940, VND 704, VUV 548, XAF 950, XOF 952, XPF 953
2	139	GBP 826, USD 840, EUR 978, INR 356 and 135 others

Minor unit	Count	Codes (with numeric code)
3	7	BHD 048, IQD 368, JOD 400, KWD 414, LYD 434, OMR 512, TND 788
4	2	CLF 990 (Unidad de Fomento), UYW 927 (Unidad Previsional)
N.A.	13	XAG 961, XAU 959, XBA 955, XBB 956, XBC 957, XBD 958, XDR 960, XPD 964, XPT 962, XSU 994, XTS 963, XUA 965, XXX 999

Three observations. The four-decimal entries are not ordinary currencies: CLF, Chile's Unidad de Fomento, and UYW, Uruguay's Unidad Previsional, are inflation-indexed units of account, and they appear in any complete currency table you load, so a validator assuming nothing exceeds three decimals will reject them. The thirteen "N.A." entries have no minor unit at all — XAU is not a currency but a troy ounce of gold, and if your money type looks up its exponent and defaults to 2 on a miss, you have silently invented centigrams of gold. And XXX exists: it is the correct code for a transaction where no currency is involved. It is not a null and it is not an error.

Representing an amount: the money type

A money value is a triple: an integral quantity, a currency, and the exponent that relates them. Systems usually store the first two and derive the third, which is defensible provided the currency table is versioned and the derivation happens once, at the edge.

The canonical representation is **an integer count of minor units**. GBP 19.99 is stored as 1999 with currency GBP; JPY 2500 as 2500 with currency JPY; KWD 12.345 as 12345 with currency KWD. There is no decimal point anywhere in storage. Formatting for display happens at the last possible moment, and parsing user input at the first.

The range question is usually settled quickly. A signed 64-bit integer holds up to 9,223,372,036,854,775,807, which in pence is £92,233,720,368,547,758.07 — comfortably more than the money supply of any country. It is not unlimited: currencies with large nominal values at institutional scale, hyperinflationary histories, and any field that also stores intermediate accruals at higher precision will get there sooner than you expect.

Where an integer is not enough, use a decimal type — a numeric type whose radix is ten, so that one tenth is exactly representable.

Type	Representation	Practical limit
`java.math.BigDecimal`	Arbitrary-precision unscaled `BigInteger` with a 32-bit scale	Effectively unbounded; scale is per-value and preserved
`decimal.Decimal` (Python)	Arbitrary-precision, context-controlled	Default context precision of 28 significant digits, adjustable
`System.Decimal` (.NET)	128 bits: a 96-bit integer plus sign and a scaling factor of 10 to the power 0 through 28	Maximum 79,228,162,514,264,337,593,543,950,335
IEEE 754 `decimal32` / `decimal64` / `decimal128`	Decimal floating point	7, 16 and 34 significant decimal digits respectively
SQL `NUMERIC` / `DECIMAL`	Implementation-defined exact numeric	PostgreSQL: up to 131,072 digits before and 16,383 digits after the decimal point

The PostgreSQL documentation is unusually direct about which of its numeric types to use: `numeric` "is especially recommended for storing monetary amounts and other quantities where exactness is required", while `real` and `double precision` "are inexact, variable-precision numeric types" where "some values cannot be converted exactly to the internal format and are stored as approximations, so that storing and retrieving a value might show slight discrepancies."

Two traps around that. PostgreSQL also ships a type literally called `money`, whose fractional precision comes from the database's `lc_monetary` setting, so its behaviour depends on server configuration and it carries no currency identifier. And an object-relational mapper that maps a `NUMERIC` column to a language-level `double` undoes the benefit of the column type without producing any error at all. Check the mapping, not the schema.

The serialisation boundary deserves the same suspicion. JSON has no decimal type; JSON numbers are commonly parsed into IEEE 754 binary64, and in JavaScript integers above `Number.MAX_SAFE_INTEGER`, which is 9,007,199,254,740,991, lose exactness. The two safe conventions are to transmit an integer count of minor units alongside a currency code, as most card-processing APIs do, or to transmit the amount as a JSON string and parse it into a decimal type, as many banking APIs do. A JSON number with a decimal point is the one option that guarantees eventual trouble.

Why binary floating point cannot carry money

IEEE 754 binary64 — `double` in C, Java and JavaScript, `float` in Python — is 64 bits: one sign bit, an 11-bit exponent, and 52 stored significand bits giving 53 bits of effective precision. Every finite value it represents is of the form m times 2 to the power e, with m and e integers, so the representable numbers are exactly the dyadic rationals within range: fractions whose denominator is a power of two.

One tenth is not a dyadic rational. Neither is one hundredth. The two quantities a currency system is built out of are precisely the quantities binary floating point cannot hold. The nearest binary64 value to 0.1 is

0.1000000000000000055511151231257827021181583404541015625

and the nearest to 0.2 is

0.2000000000000000111022302462515654042363166809082031250

Their exact sum, correctly rounded to the nearest binary64 value, is

0.3000000000000000444089209850062616169452667236328125

whereas the nearest binary64 value to the literal 0.3 is

0.2999999999999999888977697537484345957636833190917968750

These are different numbers. The arithmetic is not wrong; it is exactly right about a question you did not intend to ask.

The consequences in a ledger are specific and testable:

Equality fails. Adding the debits £1,234.56, £78.90, £4,321.00, £0.07 and £999.99 in binary64 yields 6634.5199999999995 against a credit of 6634.52 — a residue of about minus 9.1 times 10 to the minus 13. Double-entry validation by exact comparison rejects a correct journal.

Accumulation drifts. Adding 0.01 one million times gives 10000.000000171856. Adding 0.07 one thousand times gives 69.99999999999966.

Rounding lands on the wrong side. `round(1.15, 1)` gives 1.1 and `round(2.675, 2)` gives 2.67, because the stored values are 1.1499999999999991118… and 2.6749999999999982236…, both genuinely below the midpoint. The nearest-even rule is not the culprit; the input never was a midpoint.

Truncation compounds. The Vancouver Stock Exchange index is the standard historical illustration, and because it is a truncation error rather than a binary-representation error it is the more general lesson. The index was established at 1,000.000 in January 1982 and recalculated on every trade, with each updated value truncated rather than rounded to three decimal places. Losing a fraction of a thousandth on each of thousands of daily recalculations, it drifted downwards relentlessly, and by the close on Friday 25 November 1983 it stood at 524.811. Over the weekend of 25 to 28 November the calculation was corrected and the index restated at 1,098.892 — more than double the published figure. Nothing had happened in the market. The index had simply been throwing away a sliver, several thousand times a day, for twenty-two months.

The deeper argument against floating point is not accuracy at all. A ledger has to be **reproducible by an independent party**. Your merchant, your bank, your auditor and your regulator must be able to take the same inputs and the same stated rules and arrive at your figure, digit for digit, using different software. Exact decimal arithmetic on integer minor units makes that trivial. Binary floating point makes it a research project.

Rounding: modes, mandates, and the invariant

The named rounding modes a practitioner needs are few.

Mode	Behaviour at a tie	Also called
Round half up	Away from zero	Round half away from zero, commercial rounding, arithmetic rounding
Round half down	Towards zero	
Round half to even	To the nearest even last digit	Bankers' rounding; IEEE `roundTiesToEven`
Round towards zero	Not applicable; always truncates	Truncation, chopping; IEEE `roundTowardZero`
Round towards positive infinity	Always upward	Ceiling; IEEE `roundTowardPositive`
Round towards negative infinity	Always downward	Floor; IEEE `roundTowardNegative`

IEEE 754 specifies `roundTiesToEven` as the default attribute for binary formats and requires `roundTiesToAway` — round half away from zero — to be available for decimal formats, which is the standard acknowledging that commercial practice does not universally use ties-to-even.

The bias argument for ties-to-even is worth being able to demonstrate. The ten values 0.005, 0.015, 0.025, 0.035, 0.045, 0.055, 0.065, 0.075, 0.085 and 0.095 sum to exactly 0.500. Rounded half up to two places they become 0.01 through 0.10 and sum to 0.55, an overstatement of ten per cent of the total. Rounded half to even they become 0.00, 0.02, 0.02, 0.04, 0.04, 0.06, 0.06, 0.08, 0.08, 0.10 and sum to exactly 0.50.

Now the mandates, which override the argument.

Euro conversion. Council Regulation (EC) No 1103/97 fixes the arithmetic in law. Conversion rates "shall be adopted as one euro expressed in terms of each of the national currencies" and "shall be

adopted with six significant figures". They "shall not be rounded or truncated when making conversions". "Inverse rates derived from the conversion rates shall not be used" — you may not divide by the published rate to go the other way. Amounts moving between two national currency units "shall first be converted into a monetary amount expressed in the euro unit, which amount may be rounded to not less than three decimals and shall then be converted into the other national currency unit" — mandatory triangulation through the euro. And on ties, Article 5: "If the application of the conversion rate gives a result which is exactly half-way, the sum shall be rounded up."

United Kingdom VAT. VAT Notice 700 sets out the permitted treatments. At paragraph 17.5, "You may round down the total VAT payable on all goods and services shown on a VAT invoice to a whole penny", with the caveat that this "concession to round down amounts of VAT is designed for invoice traders and applies only where the VAT charged to customers and the VAT paid to HMRC is the same." At 17.5.1, where VAT is worked out per line, the trader must round "down to the nearest 0.1 pence" or "to the nearest 1 pence or 0.5 pence", and "Whatever you decide, you must be consistent." At 17.5.2, VAT per unit or per article must be worked to "4 digits after the decimal point and then round to 3 digits" or to the nearest penny or half penny, with an explicit prohibition on rounding down to nil on any unit liable at the standard or reduced rate. At 17.6, retailers who calculate VAT at line or invoice level "must not round the VAT figure down", though they "may round (up and down) each VAT calculation".

Read those two regimes together and the design conclusion is inescapable: **the rounding mode is a per-calculation configuration item with a legal basis, not a global constant.**

The invariant that must hold regardless of mode is this. For any total that is broken into parts, the parts must sum exactly to the total, in integer minor units, with no residual. Enforce it as an assertion, not as a hope.

Allocation: dividing a total without creating or destroying money

Naive per-part rounding violates that invariant in both directions. Splitting GBP 5.00 three ways gives 166.666... minor units per part; rounding each gives 167, and three parts of 167 total 501 — a penny created from nothing. Splitting GBP 100.00 three ways gives 3,333.33... per part; rounding each gives 3,333, totalling 9,999 — a penny destroyed.

The standard fix is the **largest remainder method**, borrowed from apportionment in electoral mathematics. Compute each part's exact share as a rational number of minor units, take the floor of each, and subtract the sum of the floors from the total to get the residual — always a non-negative integer strictly less than the number of parts. Distribute that residual one minor unit at a time to the parts with the largest fractional remainders, breaking ties by a deterministic rule, such as line sequence, recorded in the specification.

Worked, with a GBP 10.00 order-level discount to be apportioned across three lines of GBP 3.99, GBP 12.50 and GBP 7.51:

Line	Line total (minor units)	Exact share of 1000	Floor	Fractional remainder	Final
A	399	166.2500	166	0.2500	166
B	1250	520.8333	520	0.8333	521
C	751	312.9167	312	0.9167	313
Total	2400	1000.0000	998	residual 2	1000

The floors total 998, leaving a residual of two minor units, which go to lines C and B in descending order of remainder. The allocation is GBP 1.66, GBP 5.21 and GBP 3.13, summing exactly to the GBP 10.00 discount.

The alternative is the **running-remainder method**: process the parts in a fixed order, and for each allocate the difference between the rounded cumulative target and the cumulative amount already allocated. It also guarantees the invariant and is order-dependent by construction, which is fine as long as the order is deterministic and documented.

Whichever you choose, the properties to test are the same. The sum of the parts equals the total exactly. Every part carries the same sign as the total, so an allocation of a positive amount never emits a negative part. The function is deterministic. And the result is stable under reversal: allocating a refund of the same total across the same parts must produce the same pieces, or your refund will not net your original to zero.

The places this bites in production are consistent across the industry: marketplace payouts split among sellers, order-level discounts and shipping apportioned across lines for VAT, subscription proration on mid-cycle upgrades, interchange and scheme fees attributed to individual transactions from a netted invoice, and partial refunds allocated back across multiple tenders.

Multi-currency, in outline

Two rules first, because they eliminate most multi-currency defects before they exist. **An amount never exists without its currency**, enforced in the type system where the language permits it. **Two amounts in different currencies are not addable**, and an attempt should be an error rather than a number.

A ledger account is denominated in exactly one currency. A multi-currency ledger is therefore not a ledger of mixed rows; it is a set of single-currency ledgers, plus explicit foreign exchange gain and loss accounts to absorb the difference when a position is revalued. IAS 21, *The Effects of Changes in Foreign Exchange Rates*, requires foreign currency monetary items to be translated at the closing rate at each reporting date, with the resulting exchange differences recognised in profit or loss in the period in which they arise. That is what those accounts exist to satisfy.

Any conversion must be recorded with five things, or it cannot be re-derived: source amount and currency, target amount and currency, the rate as an exact decimal, the quotation direction, and the timestamp with the identity of the rate source. A stored rate of "1.1642" with no direction is not information. Market convention writes a pair as base and quote — EUR/USD 1.0850 means one euro buys 1.0850 dollars — but conventions are not universal across systems, so store the direction rather than infer it.

Reference rates are not transaction rates, and the institutions that publish them say so plainly. The European Central Bank's euro foreign exchange reference rates are, in the ECB's own words, "usually updated at around 16:00 CET every working day, except on TARGET closing days", based on "the daily concertation procedure between central banks across Europe, which normally takes place around 14:10 CET". The ECB adds that they "are published for information purposes only" and that "Using the rates for transaction purposes is strongly discouraged." A rate you can deal on comes from a counterparty with a bid and an offer around it, and the mid-point between the two is not a price anyone will give you.

Rounding at conversion follows the target currency, not the source: converting into JPY must produce a whole number of yen, converting into KWD a multiple of one fils. Converting an amount, rounding it, and converting it back will not return the original, and any process that assumes it does — a refund path, a reversal, a reconciliation — is broken. So is chained conversion: A to B to C does not in general

equal A to C, which is exactly why Regulation 1103/97 mandates a specific triangulation route with a specified intermediate precision rather than leaving it to the implementer.

How the amount travels on the wire

The four message families a UK payments engineer meets carry amounts four different ways, and the differences are instructive.

ISO 8583, the card authorisation format taken apart in Volume III, carries the amount in data element 4, *Amount, Transaction*, a fixed-length twelve-digit numeric field. There is no decimal point and no exponent in the field itself; the position of the decimal is implied entirely by data element 49, *Currency Code, Transaction*, which carries the three-digit ISO 4217 numeric code. A charge of GBP 19.99 travels as `000000001999` with DE49 set to 826. A charge of JPY 2,500 travels as `000000002500` with DE49 set to 392 — and if the terminal has multiplied by 100 on the way in, the message is still perfectly well-formed and the cardholder is charged a hundred times too much. Nothing in the format can detect that error.

ISO 20022, the XML-based successor, carries the amount as a decimal with the currency as an attribute on the element: an instructed-amount element whose `Ccy` attribute is `GBP` and whose text content is `19.99`. The schema type restricts the value to at most 18 total digits and 5 fractional digits, with a minimum of zero. That schema is deliberately permissive — 5 fractional digits exceeds any List One currency — and the real constraint is a business rule stated in the Message Definition Reports: "The number of fractional digits (or minor unit of currency) must comply with ISO 4217." A validator accepts `10.21`, `10.2` and `10` for EUR and rejects `10.403`. Schema validation alone is not sufficient validation.

SWIFT MT messages carry the amount in fields such as 32A, *Value Date, Currency Code, Amount*, with the format `6!n3!a15d` — six numeric characters of date in YYMMDD, three alphabetic characters of currency code, and an amount of up to fifteen characters. The network validated rules are explicit: "The integer part of Amount must contain at least one digit. The decimal comma ',' is mandatory and is included in the maximum length. The number of digits following the comma must not exceed the maximum number allowed for that specific currency as specified in ISO 4217." The decimal separator is a comma, it is mandatory even for a whole amount — five hundred thousand is written `500000,` — and it consumes one of the fifteen characters.

Bacs Standard 18, the fixed-length format carrying UK Direct Debit and Direct Credit submissions, takes the strictest line of the four. The amount occupies eleven characters at positions 36 to 46 of the record and is expressed in pence: all numeric, not all zeros, right-justified and zero-filled. There is no currency field, because the service is sterling only, and no decimal point anywhere in the file. A payment of £19.99 is written `00000001999`.

One internal representation; four adapters. Hold the amount as an integer count of minor units with an explicit currency, convert at each boundary, and look the exponent up from a versioned table rather than assuming it.

The failure modes, and the test that catches each

Failure mode	Symptom	Test
Binary floating point in a ledger field	Reconciliation off by fractions of a penny; balance checks fail sporadically	Assert exact equality of debits and credits with no tolerance

Failure mode	Symptom	Test
Hard-coded exponent of 2	Yen charged 100 times too much; fils truncated on Kuwaiti amounts	Round-trip a JPY, a KWD and a CLF amount through every boundary
Naive per-part rounding	Allocated parts sum to one minor unit more or less than the total	Property test: for random totals and weights, sum of parts equals total
Unspecified rounding line	Two subsystems disagree by a penny per transaction	Assert invoice total equals the sum of line totals
Amount without currency	Cross-currency addition produces a plausible wrong number	Make currency a required field; forbid addition of unlike currencies
Conversion without stored rate	Historical figures cannot be re-derived	Persist source, target, rate, direction, timestamp and rate source
JSON number for an amount	Value changes across an API round trip	Transmit integer minor units or strings; assert identical round trip
Currency code without effective date	Redenominated historical records misread	Store the effective date; retain List Three codes rather than remapping

The point of all this

This chapter belongs in a volume called *What Money Is*, rather than in an appendix on software engineering, because the money type is where the volume's argument becomes physical.

If money were a substance, its representation would be a detail. You would weigh it, and the scales would either be accurate or not. But money is a record — a claim, recorded twice, denominated in a unit some authority defines. And a record has to be *exact*, not merely accurate, because its whole function is that two parties who do not trust each other can compute the same answer from it and be bound by the result. A ledger uses integer minor units and a documented rounding rule not out of fastidiousness, but because the alternative — a number that is very nearly right, arrived at by a process the counterparty cannot reproduce — is not a record of anything.

Three pence is not a material amount of money. But three pence you cannot explain is a defect in the only property a ledger has.

9.98 Common wrong ideas

Wrong: Every number in a financial system must be a whole number of minor units. **Right:** Only the amount that settles must be; rates, per-unit prices and running accruals are legitimately finer, and the design question is where the rounding line falls.

Wrong: Floating point is merely imprecise, so the small random errors can be mopped up by rounding at the end. **Right:** It is exact in the wrong base, so the residue is deterministic and survives testing — every case you thought to write passes, and the one combination you did not try is off by a hair.

Wrong: The damage from floating point shows up as a visibly wrong total. **Right:** It shows up as a failed equality test in your double-entry validator, as rounding that lands on the wrong side, and at serialisation boundaries where JSON has no decimal type.

Wrong: `round(1.15, 1)` returning 1.1 is a fault in the rounding rule. **Right:** The value actually stored is 1.1499999999999991118..., genuinely below the midpoint; the function did what it was asked and the input was never what you thought.

Wrong: Money bugs come from choosing the wrong data type. **Right:** The 42-pence gap on a run of 3,000 sales came entirely from computing the merchant fee per transaction rather than per run, which is a rounding policy nobody specified.

Wrong: How many decimal places a currency has is a fact you can look up once. **Right:** ISO 4217 rounds off the ariary and the ouguiya, processors disagree with the standard and with each other, and the smallest unit you can pay in is not always the smallest you can account in.

Wrong: Bankers' rounding is the correct rounding for money. **Right:** Council Regulation (EC) No 1103/97 requires an exact half to be rounded up, and VAT Notice 700 lets an invoice trader round total VAT down while forbidding retailers from doing so.

Wrong: Rounding each line and then adding gives the same answer as adding and then rounding. **Right:** Three items at £8.99 with VAT at 20% give £5.40 by line and £5.39 by order total, and if your invoice engine and your settlement engine disagree you will reconcile that penny for the life of the product.

Wrong: Rounding each share to the nearest penny divides a total fairly. **Right:** £5.00 three ways creates a penny from nothing and £100.00 three ways destroys one, so the parts must be forced to sum back to the whole.

Wrong: A currency whose exponent is missing can safely default to 2. **Right:** XAU has no minor unit at all, so defaulting it silently invents centigrams of gold; and XXX, meaning no currency involved, is neither a null nor an error.

9.99 Chapter summary in 20 lines

1. Money is not a number: it is a number, plus a currency, plus a rule about how finely that currency may be cut, plus a written decision about what happens between two permitted cuts.
2. The working discipline is to count in the smallest coin and convert to pounds only when printing something a human will read.
3. Binary floating point cannot hold one tenth or one hundredth, because every value it represents is a whole number multiplied by a power of two.
4. Ten ten-pence coins added as pounds make 0.9999999999999999, so a till checking for exactly £1.00 never fires.
5. The arithmetic is not approximate but exact in the wrong base, which is worse, because a deterministic residue survives every test you thought to write.
6. The damage arrives as failed equality tests, as rounding that lands on the wrong side, and at API boundaries where an exact amount is parsed into a binary double.
7. Not every currency has a hundred minor units: the yen has none in use and the Kuwaiti dinar has a thousand fils, so the exponent must be looked up rather than assumed.
8. ISO 4217 gives every currency a three-letter code, a three-digit numeric code and a minor unit, and its List Three of withdrawn codes exists because codes are not permanent.
9. Of 178 active alphabetic codes, 139 have a minor unit of two, seventeen have none, seven have three, two have four, and thirteen have no minor unit at all.
10. The minor unit is a property of a currency within a particular scheme's rules rather than of the currency alone, so derive it from a table you control and version.

11. The canonical representation is an integer count of minor units with an explicit currency, and there is no decimal point anywhere in storage.
12. Where an integer will not stretch, use a decimal type whose radix is ten, and check that no object-relational mapper quietly turns the column back into a double.
13. The named rounding modes are few, and the bias argument for ties-to-even is genuine, but statistics do not override statute.
14. Euro conversion is fixed in law: six significant figures, no inverse rates, mandatory triangulation through the euro, and exact halves rounded up.
15. United Kingdom VAT permits an invoice trader to round total VAT down and forbids a retailer from doing so, which makes the rounding mode a per-calculation configuration item with a legal basis.
16. Rounding each component and then adding is not the same as adding and then rounding, and one penny of disagreement between two subsystems is permanent.
17. Whatever the mode, the parts of any divided total must sum exactly to the total in integer minor units, enforced as an assertion rather than hoped for.
18. The largest remainder method achieves that: take the floor of each exact share, then hand the residual out one minor unit at a time to the largest fractional remainders, with a deterministic tie-break.
19. An amount never travels without its currency, an account is denominated in exactly one currency, every conversion records source, target, rate, direction, timestamp and rate source, and a published reference rate is not a rate you can deal on.
20. A ledger must be exact rather than merely accurate, because its whole function is that two parties who do not trust each other can compute the same answer from it — and three pence you cannot explain is a defect in the only property a ledger has.

Sources used: SIX Financial Information — ISO 4217 currency code List One, published 1 January 2026; ISO — ISO 4217 and ISO 4217:2015. EUR-Lex — Council Regulation (EC) No 1103/97, Articles 4 and 5. HM Revenue and Customs — VAT Notice 700, paragraphs 17.5, 17.5.1, 17.5.2 and 17.6. European Central Bank — Euro foreign exchange reference rates. IEEE 754 Standard for Floating-Point Arithmetic. PostgreSQL documentation, section 8.1 Numeric Types. Microsoft .NET API documentation, System.Decimal. Stripe — Supported currencies. ISO 20022 — ActiveCurrencyAndAmount schema restriction and Message Definition Report rule on fractional digits, with XMLdation's validation notes. SWIFT field 32A network validated rules, via IBM Financial Transaction Manager documentation. Bacs Standard 18 record layout as reproduced in submitter documentation. The Vancouver Stock Exchange restatement of 25 to 28 November 1983, reported in The Wall Street Journal and catalogued in the numerical-analysis collections at TU Delft and the University of Texas at Austin. Floating-point values here were computed directly in IEEE 754 binary64 and Python's decimal module.

10

CHAPTER

Reading a Statement

10.0 What this chapter gives you

1. You will be able to read a bank statement line by line and say, for each line, which system wrote it, when, on whose authority, and what could still change it.
2. You will be able to explain why Priya's booked balance was £47.30 too high and her available balance £52.70 too low on the same screen at the same moment, with no bug anywhere.
3. You will be able to separate booking date, value date and the underlying transaction's own date, and say why a report that chooses between them implicitly books entries into the adjacent day.
4. You will be able to explain why a pending card item is not an entry at all, why it can vanish leaving no reversal line, and why matching it to a posted entry is inference rather than lookup.
5. You will be able to check the two arithmetic proofs on any statement — opening plus movements equals closing, and unbroken statement sequence numbers — before accepting the file.
6. You will be able to choose between camt.052, camt.053, MT940 and BAI2 for a controlled reconciliation and defend the choice.
7. You will be able to read the ISO 20022 balance type codes and say whether a given screen is showing CLBD, CLAV or XPCD.
8. You will be able to explain why a payer-supplied reference is evidence of intent rather than a key, and design allocation that does not break when it is truncated or wrong.
9. You will be able to run a three-way reconciliation between internal ledger, processor report and bank statement, and name the characteristic failure of each pair.
10. You will be able to say why twelve failed safeguarding firms averaged a 65 per cent shortfall in client funds, and what the rules coming into force on 7 May 2026 require instead.

Everything in this volume so far has been an argument about what money is. Money is a record. A balance is a row. Every payment is two facts. A bank is a balance sheet whose liabilities you happen to call your savings. Clearing is not settlement. Netting turns twenty-seven million obligations into a few dozen. And an amount of money is an integer in minor units, never a floating-point number.

All of that is true, and none of it is in your hand.

This chapter is. Open your banking app, or take the PDF the bank sent you at the start of the month, and put it in front of you. Every abstraction in the preceding nine chapters is printed on it somewhere, usually in a column you have never looked at. By the end of this chapter you will be able to read the document line by line and say, for each line, which system wrote it, when, on whose authority, and what could still change it.

Then we will do the harder thing, which is to prove that your record of these events and the bank's record of these events are the same record. That exercise is called reconciliation, and it is where young

payments companies actually die — not in a dramatic outage, but quietly, over eighteen months, because nobody could ever say with certainty how much customer money the firm was holding.

The plain version

Here is the first thing to understand, and it is not obvious.

Your bank statement is not your money. It is not even a picture of your money. It is a letter from your bank saying what it believes it owes you, written at a particular moment, by a system that was not finished processing.

Think of a school report. The report says what the teacher thinks of your work. It is written on a Thursday, printed on a Friday, and posted on a Monday. If you handed in a brilliant essay on the Friday afternoon, it is not in the report, and the report is not wrong — it was written before the essay arrived. The report is a true description of a moment that has already passed.

A statement is the same. It is written at a moment. The world kept going.

Three dates, not one

Every line on your statement has at least three dates attached to it, and most statements only show you one or two of them.

Imagine you write an essay on Friday. You hand it in on Monday. The teacher marks it as though it had arrived on Sunday night, because Sunday night was the deadline. Three dates:

The date it happened. Friday, when you wrote it.

The date it was recorded. Monday, when the teacher took it off the pile and wrote your name in the mark book.

The date it counts from. Sunday, because that is the date the rules say applies.

Money works exactly like this. You buy a coffee on Saturday. The shop tells its bank about it on Monday. Your bank writes it into your account on Tuesday. And your bank may decide that, for the purposes of working out interest, the money left you on Saturday.

Three dates for one coffee. Most people have never noticed there is more than one, and then one day they see a purchase they made on the first of the month sitting on a line dated the fifth, and think the bank has made a mistake. It has not. It has shown a different one of the three dates than the one they were thinking of.

Two balances, not one

The second thing that is not obvious is that you have two balances at the same time, and they are usually different.

One is the balance the bank has written down: the sum of everything it has actually recorded. Call it the written-down balance.

The other is the balance you are allowed to spend: the written-down balance, minus anything the bank has promised somebody else on your behalf but has not yet recorded. Call it the spendable balance.

Here is the classic case. You drive onto a forecourt and use the pay-at-pump. The pump has no idea how much fuel you are about to take. It could be five pounds, it could be ninety. So before it releases a drop, it asks your bank a question: if this person tries to spend one hundred pounds, will you cover it?

Your bank says yes, and — this is the important part — it sets one hundred pounds aside. Not spent. Set aside.

You put in forty-seven pounds and thirty pence of fuel and drive away. For the next couple of days, your written-down balance has not moved at all, because nothing has been recorded yet. But your spendable balance is a hundred pounds lower, because a hundred pounds is being held back for a bill that has not arrived. Then the real bill arrives for forty-seven pounds thirty, the bank records it, the written-down balance finally drops, and the hundred-pound reservation is released.

Notice what happened in between. For two or three days, your written-down balance was too high — you had spent forty-seven thirty and it did not show. And your spendable balance was too low — a hundred pounds was reserved when only forty-seven thirty was owed. Both balances were wrong at the same time, in opposite directions, and neither of them was a bug.

A month in one account

Let us make this concrete. Here is a statement. Priya has a current account. This is March.

Date	Description	Out	In	Balance
1 Mar	Opening balance			1,842.60
3 Mar	Direct debit, Thames Water	34.00		1,808.60
5 Mar	Card, Caffè Nero, 1 Mar	3.40		1,805.20
9 Mar	Payment in, J Raman, ref BIRTHDAY		50.00	1,855.20
12 Mar	Card, Sainsbury's, 10 Mar	62.15		1,793.05
18 Mar	Card, Esso, 15 Mar	47.30		1,745.75
20 Mar	Refund, Zara, 14 Feb		28.99	1,774.74
25 Mar	Salary, KedByte Technologies		2,150.00	3,924.74
26 Mar	Standing order, rent	1,100.00		2,824.74
31 Mar	Interest		0.42	2,825.16
31 Mar	Closing balance			2,825.16

Ten lines. Now read them properly.

The coffee on line three was bought on 1 March and written down on 5 March. Four days. Nothing went wrong; that is simply how long it took the coffee shop's bank to send the bill to Priya's bank.

The fifty pounds on line four came from Priya's brother. The word BIRTHDAY is on the statement because her brother typed it into his banking app. No bank wrote it, no bank checked it, and no bank knows what it means. If he had typed nothing, the line would say nothing. If he had typed the wrong thing, the line would say the wrong thing and the money would still have arrived, because the money is addressed by the account number, not by the note.

The Esso line is the forecourt. Purchase 15 March, recorded 18 March. On 16 March, Priya's app would have shown a written-down balance of £1,793.05 and a spendable balance of £1,693.05, because £100 was reserved. Her real position — everything she had actually spent — was £1,745.75. Her written-down balance was £47.30 too high. Her spendable balance was £52.70 too low. On one screen. At the same time.

The refund on line seven is for something bought on 14 February. It is not the February purchase being undone. It is a completely separate payment, going the other way, made by the shop to Priya. This is why refunds are slow when purchases feel instant: nothing is being erased, a second thing is being done.

And the whole statement has one property that matters more than any single line: the opening balance plus everything in, minus everything out, equals the closing balance. £1,842.60 plus £2,229.41 in, minus £1,246.85 out, is £2,825.16. If that sum does not come out, something is missing from the page.

Reconciliation, without the word

Now the second half of the chapter, in plain terms.

Suppose you run a market stall. At the end of Saturday you have a notebook: every sale, what it was, how much. Your card machine's provider also has a record of every sale. And your bank has a record of what landed in your account.

Three records of the same day. Reconciliation is the work of proving all three describe the same Saturday, and explaining — line by line, not in total — every place they differ.

It sounds trivial. It is not. Your notebook says you took £1,240 on Saturday. Your bank account shows £1,213.90 arriving on Tuesday. Those numbers are different and both are correct: the card company kept £26.10 in fees, and it paid you two days later. If your test is "does the bank number equal my number", you fail a test you should pass. And the reverse also happens: your notebook says £1,240, the bank says £1,240, and you are missing a sale, because two errors cancelled out. The total matched and the record is wrong.

That is the whole discipline in one paragraph. Matching totals proves nothing. Reconciliation means matching individual events, one by one, to individual events, and having a defensible explanation for every single one that does not match. Not most of them. All of them.

Do this every day and a mistake is a small puzzle. Do it every month and a mistake is an investigation. Do it never — and this is the thing that kills companies — and one day somebody asks how much of the money in your bank account belongs to your customers, and the honest answer is that nobody knows.

Where the plain version stops being true

The report-card picture is a good one, and the three dates and two balances are real. Four things it hides matter enough to correct.

There is no such thing as "the statement"

The plain version implies one document. In practice, at least three different artefacts are called your statement, they are produced by different systems at different moments, and they routinely disagree.

The screen in the mobile app is usually a merged view: recorded entries from the ledger, plus card authorisations that have not been recorded at all, presented in one list so it looks like a single sequence of events. The PDF or paper statement is normally the ledger alone. The machine-readable file that a business downloads — an ISO 20022 camt.053, a SWIFT MT940, a BAI2 file — is the ledger alone as well, but it is generated at a fixed cut-off and delivered on a schedule, so it can be hours behind the app and days behind reality.

When a customer says "the app shows a payment but the statement does not", nobody has made an error. Two different systems were asked two different questions. This is also why software consuming bank data must be explicit about which artefact it is reading. Building a customer balance from the app's merged view and building it from the camt.053 will give different numbers on most days, and only one of them is a balance in any accounting sense.

"Pending" is not a status; it is the absence of an entry

This is the correction that matters most for anyone building systems.

In the plain version, the pending coffee "becomes" the posted coffee, as though a single record changed state from grey to black. That is not what happens in a card transaction. The pending item is not an entry in the ledger at all. It is a memorandum — an amount held against your available funds, recorded in an authorisation store, with no double entry behind it and no place in the sum that produces your balance. The posted item, days later, is a genuinely new ledger entry created from a completely separate message that arrived through the clearing system.

Three consequences follow, and each of them breaks naive software.

The amounts need not match. The forecourt held £100 and the entry was £47.30. A restaurant may authorise the bill and clear the bill plus a tip. A hotel may authorise nightly and clear once.

The pending item can vanish with no trace whatsoever. If the merchant never submits the transaction for clearing, the hold simply expires. Nothing appears on the statement. There is no reversal line, because there was never a line.

And there is usually no field on your statement that links the two. The authorisation and the clearing record carry identifiers that can be matched by the issuer's systems, but those identifiers are frequently not surfaced to the customer, and are not present in a camt.053 in any standardised place. Matching a pending item to a posted item from the outside is inference, not lookup.

Value date is not "when I can spend it"

The plain version compressed three ideas into "the date it counts from". They come apart.

The value date is an interest and accounting concept: the date from which the amount is treated as being at the account holder's disposal for the purpose of calculating interest. It is not a promise about availability of funds, and in the United Kingdom the two are regulated separately. The Payment Services Regulations 2017 require, at regulation 89, that "the credit value date for the payee's payment account must be no later than the business day on which the amount of the payment transaction is credited to the account of the payee's payment service provider", and, separately, that the payee's provider "must ensure that the amount of the payment transaction is at the payee's disposal immediately after that amount has been credited to that payment service provider's account". Two obligations, one about interest, one about access.

Value dates can also point backwards. An entry booked today can carry a value date of last Tuesday, which means interest is recalculated as though the money had been there since last Tuesday. Back-valuation is normal, particularly for corrections, and it means that yesterday's interest figure was not final when it was printed.

Finally, the money-can-still-go-back problem. A recorded credit is not necessarily a final credit. Cheques paid in under the United Kingdom's Image Clearing System, live since 2019, are typically available to withdraw by 23:59 on the next weekday, but availability and finality are different questions and a cheque can still be returned. Direct debit collections can be represented or refunded. Payments sent in error can be pursued through indemnity and recall processes. So a balance made entirely of booked entries can still overstate what you will keep.

Reconciliation is not matching, it is explaining

The market stall gave the right instinct and the wrong scale. Reconciliation in a real payments business is not a comparison of two lists; it is a continuous process of accounting for differences, most of which are expected.

Differences fall into recognisable species. Timing differences, where both records are right and one is early. Netting differences, where one side reports gross and the other net of fees. Granularity differences, where one side records a thousand sales and the other records one settlement. Sign and direction differences, especially around refunds and chargebacks. Currency differences, where the amount instructed and the amount settled are not in the same currency and neither party rounded the same way.

None of those is an error. Each still has to be identified, categorised and proved, because the only way to find the one real break — the missing payment, the double credit, the customer whose money went to the wrong internal account — is to have rules that eliminate every harmless difference and leave the harmful one standing alone.

The technical version

What the document actually is

An account is a ledger: an ordered, append-only sequence of postings, each of which is one half of a double entry, as chapters three and four established. A statement is not that ledger. A statement is a **report over an interval of that ledger**, and it has a fixed logical shape regardless of format:

An identification of the account and the account servicing institution. A statement identifier and a sequence number. A period, with a start and an end. One or more balances, each labelled with a type and a date. Zero or more entries. And, implicitly, an arithmetic proof: opening booked balance, plus the signed sum of the entries in the period, equals the closing booked balance.

That proof is the single most useful property of the document and it is the first thing an integration should check. If it does not hold, the statement is either truncated, mis-parsed, or reporting a balance type you have misidentified.

The sequence number is the second most useful property. In ISO 20022 it is the `ElctrncSeqNb` element inside `Stmt`; in SWIFT MT940 it is field `:28C:`, the statement number and sequence number. Consecutive statements must have consecutive numbers. A gap means a statement was never delivered, and a firm reconciling only the statements it received will reconcile cleanly to a false position. BAI2 provides no equivalent statement sequence number, which is one of several reasons it is a weaker basis for controlled reconciliation than camt.053; it relies instead on control totals in its account, group and file trailer records.

The formats, and which one you are actually reading

Format	Standard body	Content	Typical timing
camt.052	ISO 20022	Bank-to-customer account report, intraday	Periodically through the day
camt.053	ISO 20022	Bank-to-customer statement, end of day	Once per business day, after cut-off

Format	Standard body	Content	Typical timing
camt.054	ISO 20022	Bank-to-customer debit/credit notification	Per event or per batch
MT940	SWIFT MT	Customer statement message	End of day
MT942	SWIFT MT	Interim transaction report	Intraday
BAI2	Banking Administration Institute	Cash management balance reporting	Previous-day, and intraday variants
PDF / CSV / OFX	None binding	Human or spreadsheet consumption	Varies

The distinction between camt.052 and camt.053 is the one that causes the most trouble in practice. camt.053 is the official end-of-day statement for the previous business day and its entries are booked. camt.052 is an intraday report and may contain items that are not final. Reconciling a firm's books against camt.052 will produce a position that changes underneath you; reconciling against camt.053 produces a position that does not.

An MT940 message is a tagged text format. The fields that matter are `:20:` transaction reference number, `:25:` account identification, `:28C:` statement number and sequence number, `:60a:` opening balance, `:61:` statement line, `:86:` information to account owner, `:62a:` closing balance, `:64:` closing available balance and `:65:` forward available balance.

Field `:61:` carries the format `6!n[4!n]2a[1!a]15d1!a3!c16x[//16x]` with an optional `[34x]` supplementary details line. Read left to right, that is: a six-digit value date as `YYMMDD`; an optional four-digit entry date as `MMDD`; a debit/credit mark of `C`, `D`, `RC` or `RD`; an optional funds code; the amount; a transaction type identification code of one letter and three characters, where `S` denotes a SWIFT transfer and the three characters are the message type, and `N` denotes a non-SWIFT transfer with a mnemonic such as `TRF` or `CHG`; a reference for the account owner; optionally, after a double slash, the account servicing institution's own reference; and optionally a supplementary details line.

Look at that entry date again. `MMDD`. Four digits. **There is no year.** A parser must infer the year from context, and every year in the last week of December and the first week of January, somewhere in the world, a reconciliation engine books an entry twelve months out because it inferred wrongly. This is not a hypothetical class of bug; it is a scheduled one.

BAI2 is a comma-delimited record format used mainly by North American banks and by global banks for cash management reporting. Its records are `01` file header, `02` group header, `03` account identifier, `16` transaction detail, `88` continuation, `49` account trailer, `98` group trailer and `99` file trailer. Balances appear as type codes inside the `03` record: `010` opening ledger balance, `015` closing ledger balance, `040` opening available balance and `045` closing available balance, with `100` and `400` carrying total credits and total debits.

The dates, exactly

ISO 20022 gives an entry three date elements and they are not interchangeable.

`BookgDt` is the booking date: the date the entry was posted to the account. This is the date that determines which statement the entry appears on and which day's closing balance it affects.

`ValDt` is the value date: the date from which the amount is taken to be at the account holder's disposal for interest and accounting purposes.

`Dt` inside the transaction details, and the various dates inside `RltdDts`, carry the underlying transaction's own dates — for a card entry, the date of purchase; for a direct debit, the requested collection date.

The relationship between them is institution-specific and rail-specific, and the only safe assumption is that they are three separate facts. For a Bacs direct credit, booking date and value date will typically both fall on entry day. For a card purchase, the transaction date precedes the booking date by anything from hours to a week. For a back-valued correction, the value date precedes the booking date deliberately.

Store all three, plus the timezone in which the booking date was determined. A very large proportion of reconciliation breaks that appear to be missing transactions are, on inspection, entries booked into the adjacent day because two systems disagreed about when the day ended. Business date is not a derived quantity. It is a value the institution declares, and it changes at a cut-off, not at midnight.

Pending, posted, and the balance type code set

ISO 20022 gives an entry a status element, `Sts`, from a code set whose members are `BOOK` (booked, posted to the account), `PDNG` (pending, not yet posted), `INFO` (informational, no financial impact) and, in current versions, `FUTR` (future). Market practice for an end-of-cycle camt.053 requires `BOOK`; `PDNG` and `INFO` entries belong in intraday reporting. If you are receiving `PDNG` entries in a document you believe to be an end-of-day statement, you are reading a camt.052 or a bank's proprietary variant.

Balances are where the standard is most useful and most ignored. A `Bal` element carries a type code, and the code set is precise:

Code	Meaning
OPBD	Opening booked balance. Always equals the previous report's closing booked balance
CLBD	Closing booked balance. Opening booked balance plus all entries booked in the period
OPAV	Opening balance of money at the account owner's disposal on the date specified
CLAV	Closing balance of money at the account owner's disposal on the date specified
ITBD	Booked balance calculated during the business day, at the time specified, subject to change
ITAV	Available balance calculated during the business day, at the time specified, subject to change
FWAV	Forward available balance at the disposal of the account owner on the date specified
PRCD	Balance at the previously closed reporting period
XPCD	Expected balance: booked entries plus pending items known at the time of calculation, projecting the end-of-day balance if everything is booked and nothing further is posted
INFO	Balance for information only

Two of these deserve emphasis. `XPCD` is the standard's name for the number a customer thinks of as "my balance" in a mobile app — booked plus pending — and it is explicitly labelled as a projection, not a position. And `FWAV` is the mechanism by which a bank tells you that money is arriving on a future

date under a value-dating arrangement; it is a forward-looking figure and belongs in cash forecasting, not in a ledger.

The available balance itself is not defined by any standard, because it is a commercial construct. In broad terms:

Available balance equals the booked balance, minus authorisation holds not yet cleared, minus credits booked but not yet available under the institution's funds availability policy, plus any agreed overdraft or credit facility, plus any same-day credits the institution has chosen to release early.

Every term after the first varies by institution, by product and sometimes by customer. This is why two accounts holding identical entries can show different available balances at the same bank, and why "available balance" is never a safe input to an accounting process.

Authorisation holds

A card authorisation is a message, not an entry. Its economic effect on the cardholder is a reduction in available funds; its accounting effect on the issuer's ledger is nothing at all until a clearing record arrives.

The scheme rules control how long that gap may be, and the answer depends on how the merchant characterised the authorisation. A **final authorisation** asserts that the amount is exactly the amount that will be captured. An **estimated** or **pre-authorisation** asserts that the amount is a good-faith estimate, which is what a fuel pump, a hotel and a car hire desk necessarily send.

Timeframes published by the acquirer Worldline for the European region, as of writing, illustrate the shape of the rules:

Scheme	Authorisation type	Validity period
Visa	Final authorisation	10 days
Visa	Pre-authorisation, lodging, cruise line, vehicle rental	30 days
Visa	Pre-authorisation, other merchant categories	10 days
Mastercard	Final authorisation	3 days, with reversal or clearing still possible up to 7 calendar days subject to fees
Mastercard	Pre-authorisation	30 days
Diners	Final authorisation / pre-authorisation	7 days / 30 days
Discover	Final authorisation / pre-authorisation	10 days / 30 days

Visa's own acceptance documentation states the merchant-side obligation in the same terms: an authorisation flagged as final "must be submitted for capture within seven calendar days of its request", while a pre-authorisation must be submitted for capture within 30 calendar days, and an authorisation that is not captured must be reversed.

Three practical points follow.

First, the hold on the cardholder's available balance and the scheme's authorisation validity period are related but not identical. An issuer may release a hold on its own schedule, typically shorter than the scheme maximum, precisely because customers complain. Once released, the funds are spendable

again — and the clearing record may still arrive afterwards, producing an entry against a balance that no longer expects it.

Second, an estimated authorisation can be adjusted. Incremental authorisations increase the reserved amount as a hotel stay lengthens; partial reversals reduce it when the final amount is known. Whether an issuer reflects those adjustments promptly in the customer-visible available balance is an implementation matter, not a rule.

Third, the amount that clears may legitimately differ from the amount authorised, within tolerances that vary by merchant category. Software that treats such a mismatch as an anomaly will generate an unmanageable volume of false positives against fuel, hospitality and transport merchants.

Why a balance can be wrong in two directions at once

Return to Priya's account on the afternoon of 16 March.

Measure	Amount	Why
Booked balance (CLBD as at 15 March)	1,793.05	Every posted entry, correctly summed
Authorisation hold	100.00	Estimated authorisation at the pump
Available balance	1,693.05	Booked minus the hold
Economic position	1,745.75	What she has actually committed to spend

The booked balance overstates her position by £47.30, because she has bought fuel that the ledger does not yet know about. The available balance understates it by £52.70, because £100 is reserved against a £47.30 obligation. There is no number on the screen that is correct, and every number on the screen is produced correctly.

Generalise it. A balance overstates in three situations: authorised but uncleared card spending; credits booked that remain revocable, including cheques within the return window, direct debits within the indemnity window and credit transfers subject to recall; and fees and interest accrued but not yet applied. It understates in three others: holds that exceed the eventual clearing amount; holds that persist after the corresponding clearing record has already posted, producing a temporary double count; and incoming credits received by the institution but not yet released under its availability policy.

Both categories are usually live at once on a busy account. The correct engineering response is not to pick one balance and trust it. It is to hold, separately, a booked position derived from entries, a set of outstanding holds with their own lifecycle, and a set of credits with a finality date. Any single number you present to a user is a summary of those three, and you should know which one you are presenting.

The reference field, and who wrote it

The most misunderstood column on any statement is the narrative, and the misunderstanding is this: people believe the bank wrote it. Almost none of it is written by the bank.

Different rails give the payer different amounts of room, and the room is small.

For **Bacs**, using the Standard 18 file layout described in the Bacs *Electronic Funds Transfer, File Structures* document, the payment instruction record carries a `service user's reference` in field 10, and it is **18 characters**. The originating account name is field 9, the destination account name field 11, and the

amount in pence field 8, itself limited to 11 digits. Eighteen characters is the whole channel through which a payroll bureau, a utility or an insurer can tell the receiving party what a payment is for.

For **Faster Payments**, which uses an ISO 8583 message, the Open Banking Read/Write specification's field mapping shows field 62 `END TO END REFERENCE` at 31 characters, field 120 `REFERENCE INFORMATION` at 18 characters and field 121 `REMITTANCE INFORMATION` at 140 characters. Note the asymmetry: 140 characters of remittance information exist in the message, but the field most receiving systems key on is the 18-character one, and if the creditor account carries a secondary identification such as a building society roll number, that secondary identification takes field 120 and the payer's reference does not get it.

For **CHAPS**, carried as an MT103, the beneficiary reference is field 70, four lines of 35 characters, with structured tags such as `/ROC/` for the original customer reference.

In a camt.053, the references that reach the statement live in `NtryDtls/TxDtls/Refs` and comprise `MsgId`, `PmtInfId`, `EndToEndId`, `TxId` and, for direct debits, `MndtId`. Remittance data lives in `RmtInf`, as `Ustrd` free text of up to 140 characters per occurrence or as `Strd` structured remittance. The entry itself carries `AcctSvcrRef`, the account servicing institution's own reference, and `AddtlNtryInf`, free text written by the bank.

So the ownership of each string is as follows:

String	Written by	Reliability
`EndToEndId`	The originating party's system	High for payments you sent; unvalidated for payments you received
`RmtInf/Ustrd`	The originating party, usually a human typing	Free text; no validation of any kind
Bacs field 10 reference	The originating service user's file	18 characters, uppercase, restricted character set
`AcctSvcrRef`	The account servicing bank	The bank's own key; stable within that bank
`AddtlNtryInf`	The account servicing bank	Descriptive; format not guaranteed stable across releases
Card merchant descriptor	The acquirer, from the merchant's registered acceptor name and location	Frequently a trading entity's legal name, not its shop sign

The consequence for anybody building a payment system is a single hard rule. **A payer-supplied reference is evidence of intent, not a key.** It can be truncated, uppercased, stripped of punctuation, rewritten by an intermediary, mistyped by the payer or omitted entirely, and the payment will still arrive, because the payment is addressed by account identifier. Systems that allocate incoming funds by exact-matching a reference will misallocate. Systems that allocate by exact match, then fall back to fuzzy match, then fall back to a human queue with the amount, date, payer name and remitting sort code available for judgement, will misallocate far less.

The corresponding rule for outbound payments is to generate the reference yourself, make it structurally checkable — a prefix plus a check character is enough — and keep it within the tightest field length on any rail you use, which in the United Kingdom means 18 characters, not 35 and not 140.

Classification: bank transaction codes

Each entry may carry a `BkTxCd`, a hierarchical classification with a domain, a family and a sub-family, drawn from the ISO 20022 external code set, plus an optional proprietary code.

A received SEPA credit transfer, for example, is `PMNT` / `RCDT` / `ESCT`: payments domain, received credit transfer family, SEPA credit transfer sub-family. The family codes within the payments domain distinguish received from issued and credit transfers from direct debits — `RCDT`, `ICDT`, `RDDT`, `IDDT` — with `MCOP` and `MDOP` for miscellaneous credit and debit operations such as fees and interest.

Two warnings. Banks vary enormously in how carefully they populate this structure, and a great deal of real-world classification lives in the proprietary code rather than the standard one. And the American equivalent, the BAI2 numeric type code, has a different and larger vocabulary, so a firm consuming both formats needs a mapping layer and should expect that mapping to be incomplete.

Reconciliation, properly

Reconciliation is the process of proving that two or more independently maintained records of the same set of events agree, and identifying, categorising and resolving every difference between them.

In a payments business there are rarely two records. There are usually three, and the discipline is described as three-way reconciliation:

The **internal ledger**: what your own system says each customer is owed and what your own system says you instructed.

The **scheme or processor report**: what the acquirer, processor or scheme says happened — authorisations, captures, refunds, chargebacks, fees, and the settlement amounts derived from them.

The **bank statement**: what money actually arrived in or left your bank accounts.

Each pair must tie, and each fails in a characteristic way.

Internal ledger against processor report fails on events the processor knows about and you do not: chargebacks raised, fees levied, and transactions your API call timed out on but which the processor nonetheless completed. That last case produces duplicate charges, and it is why every outbound instruction must carry an idempotency key that survives a retry.

Processor report against bank statement fails on netting. The processor reports gross transactions; the bank receives a net settlement after fees, refunds, chargebacks and reserve movements. If your matching logic looks for a bank credit equal to the sum of the day's sales, it will never find one. The correct approach is to reconstruct the settlement: sum the gross items in the batch, subtract each reported deduction, and prove the result equals the credit on the statement to the penny. Then, separately, prove every gross item in the batch exists in your ledger.

Internal ledger against bank statement fails on timing and on entries you never created: bank charges, interest, returned direct debits, and credits from parties you were not expecting.

A working process has five properties. It runs daily, not monthly. It matches at item level, never at total level alone. It classifies every break into a named category with an owner. It ages breaks and escalates them, so that a break surviving three days becomes somebody's problem by policy rather than by luck. And it treats the arithmetic proofs described earlier — opening plus movements equals closing, sequence numbers unbroken, BAI2 control totals consistent — as automated preconditions, not as things a human notices.

Why this is where firms die

This is not a metaphor and there are figures.

For investment firms holding client money, the FCA's client assets rules are explicit. CASS 7.15.15R requires an internal client money reconciliation "each business day", performed on the previous business day's closing records. CASS 7.15.22R requires an external reconciliation against third-party records "as regularly as is necessary but without allowing more than one month to pass" between reconciliations. And CASS 7.15.29R requires that where an internal reconciliation reveals a shortfall, the firm determines the reason and pays in the difference "by the close of business on the day that the reconciliation is performed". A firm that cannot reconcile daily cannot fund a shortfall daily, and a firm that cannot fund a shortfall daily is in breach on the day it first fails.

For payments and e-money firms, the rules were weaker for years, and the FCA's own evidence on what that produced is stark. In its consultation paper CP24/20, published in September 2024, the regulator recorded that between the first quarter of 2018 and the second quarter of 2023, twelve payments firms that safeguarded relevant funds became insolvent, and that for those firms there was an average shortfall of 65 per cent in funds owed to clients. Two thirds of customer money, gone, on average, across twelve failures. The same paper identified the causes in terms that are almost a description of this chapter: "a lack of documented processes for consistently identifying which funds must be safeguarded" and "inadequate reconciliation procedures to ensure that the correct sums are protected on an ongoing basis". In 2023 alone the FCA opened supervisory cases relating to approximately 15 per cent of firms that safeguard.

The response is Policy Statement PS25/12, published on 7 August 2025, which introduces the interim Supplementary Regime through new Handbook chapters CASS 15, CASS 10A, SUP 3A and SUP 16.14A, together with amendments to the *Payment Services and Electronic Money — Our Approach* document. The rules come into force on **7 May 2026**. Their substance is this chapter's subject matter turned into obligations: internal and external safeguarding reconciliations at least once each reconciliation day, excluding weekends, public holidays and days when relevant foreign markets are closed; a resolution pack sufficient to return customer funds promptly on insolvency; an annual safeguarding audit by a qualified auditor, with an exemption for firms that have not been required to safeguard more than £100,000 of relevant funds over a period of at least 53 weeks; a monthly safeguarding return to the regulator; and documented due diligence on third parties holding relevant funds.

Read that list next to the failure statistic and the causal story is unambiguous. Firms did not fail because their payments went wrong. Their payments mostly worked. They failed because nobody could prove, on any given day, how much of the money in the bank belonged to customers — and by the time anyone asked the question properly, two thirds of it was not there.

Reconciliation is therefore not back-office hygiene. It is the control that converts a database of intentions into a defensible statement of obligation. It is the difference between a firm that knows what it owes and a firm that believes what it owes.

A short list of things to build

For anyone implementing against bank statement data, the practical conclusions of this chapter compress into seven rules.

Parse and store all three dates on every entry, plus the timezone, and never let a report choose between them implicitly.

Verify the arithmetic proof on every statement before accepting it, and verify statement sequence continuity, alarming on a gap rather than proceeding.

Key on the account servicing institution's own reference for idempotency of ingestion, and on your own generated end-to-end identifier for outbound payments. Never key on payer free text.

Model holds as a separate object with their own lifecycle, linked to entries where possible and explicitly unlinked where not.

Distinguish booked, available and projected balances in your data model with different names, and never store one in a field named after another.

Reconcile daily, at item level, three ways, with categorised and aged breaks.

And treat every credit as revocable until the rail-specific finality point has passed — a subject the remaining volumes take apart rail by rail.

Where Volume I leaves you

Trace the argument backwards from the document in your hand.

The statement is a report over a ledger. The ledger is a sequence of double entries, each recording two facts about one event, because a payment is never one fact. Those entries adjust a balance, and the balance is not a quantity of anything; it is the size of a debt a bank owes you, held as a row in the bank's records. The bank can owe you that debt because a bank is an institution whose liabilities circulate as money, and its own money — the money it uses to settle with other banks — is a different and superior kind, issued by a central bank. When your payment reaches another bank, the obligation between the two of them is established by clearing and discharged by settlement, which are different events at different times, and those obligations are compressed by netting before any central bank money moves. Every amount involved is an integer count of minor units in a currency identified by an ISO 4217 code, because it has to be exact.

That is Volume I. It is a description of what money is, and every part of it happened inside one institution's records, or between an institution and the central bank that stands above it.

The rest of this book is about what happens **between** institutions, and it starts with the most familiar object in the whole system.

There is a piece of plastic in your pocket that weighs about five grams. It carries, in several different physical forms, one piece of information: which account to charge. It talks to a terminal it has never met, through a network owned by neither of them, using a message format designed in the 1980s and cryptography designed in the 1990s, and it does this several hundred billion times a year without anybody thinking about it.

Volume II is about that card, and it begins by taking it apart.

10.98 Common wrong ideas

Wrong: A statement is a picture of your money. **Right:** It is a letter from your bank saying what it believes it owes you, written at a particular moment by a system that was not finished processing.

Wrong: There is one document called "the statement". **Right:** At least three artefacts carry that name — the app's merged view, the PDF, and the machine-readable file at a fixed cut-off — produced by different systems at different moments, and they routinely disagree.

Wrong: A pending item becomes the posted item, changing state from grey to black. **Right:** The pending item is a memorandum in an authorisation store with no double entry behind it; the posted item is a genuinely new ledger entry created from a separate clearing message.

Wrong: A cleared amount that differs from the authorised amount is an anomaly worth alerting on. **Right:** The forecourt held £100 against a £47.30 entry, and treating such differences as anomalies produces an unmanageable volume of false positives against fuel, hospitality and transport merchants.

Wrong: If a hold disappears there will be a reversal line to reconcile it against. **Right:** If the merchant never submits the transaction for clearing the hold simply expires, and nothing appears on the statement, because there was never a line.

Wrong: The value date tells you when you can spend the money. **Right:** It is an interest and accounting concept, regulated separately from availability under regulation 89, and it can point backwards, so yesterday's interest figure was not final when it was printed.

Wrong: A booked credit is money you will keep. **Right:** Cheques can be returned, direct debits represented or refunded and credit transfers recalled, so a balance made entirely of booked entries can still overstate what you will keep.

Wrong: The narrative on a statement was written by the bank. **Right:** Almost none of it was; the end-to-end identifier and the remittance text come from the originating party, and a payment arrives on its account identifier whatever the reference says.

Wrong: Reconciliation means proving the two numbers agree. **Right:** Matching totals proves nothing, because two errors cancel out; the discipline is matching individual events and having a defensible explanation for every single difference.

Wrong: The bank credit should equal the day's sales. **Right:** The processor reports gross and the bank receives a net settlement after fees, refunds, chargebacks and reserve movements, so logic that looks for that credit will never find one.

10.99 Chapter summary in 20 lines

1. A statement is not your money and not a picture of it; it is a report over an interval of a ledger, written at a moment by a system that was still processing.
2. Every entry carries at least three dates: when the transaction happened, when it was booked, and the date it counts from for interest.
3. ISO 20022 names them `BookgDt`, `ValDt` and the transaction's own dates, they are three separate facts, and all three should be stored with the timezone.
4. Business date is not a derived quantity but a value the institution declares, and it changes at a cut-off rather than at midnight.
5. You have two balances at once: the booked balance the bank has recorded, and the available balance you are allowed to spend.

6. On the afternoon of 16 March Priya's booked balance overstated her position by £47.30 and her available balance understated it by £52.70, and every number on the screen was produced correctly.
7. The most useful property of any statement is its arithmetic proof: opening booked balance plus the signed entries in the period equals the closing booked balance.
8. The second most useful is the sequence number, because a gap means a statement was never delivered and a firm will otherwise reconcile cleanly to a false position.
9. There is no single statement: the app shows a merged view, the PDF and the machine-readable file show the ledger, and the file is generated at a cut-off and can be days behind reality.
10. camt.053 is the end-of-day booked statement and camt.052 is intraday and not final, so reconciling against the latter gives a position that changes underneath you.
11. MT940's field `:61:` carries an entry date as `MMDD` with no year, which makes the turn of the calendar a scheduled bug rather than a hypothetical one.
12. A pending card item is not a ledger entry but a hold against available funds, and the amounts need not match, the item can vanish silently, and no standardised field links the two.
13. Authorisation validity runs from three days to thirty depending on scheme and on whether the merchant flagged the authorisation final or estimated, and issuers release holds on their own shorter schedules.
14. The ISO 20022 balance type codes are precise, and `XPCD` — booked plus pending — is the number a mobile app calls "my balance" and the standard explicitly calls a projection.
15. Available balance is defined by no standard because it is a commercial construct, which is why it is never a safe input to an accounting process.
16. A balance can overstate and understate at once, so hold booked entries, outstanding holds and credits with finality dates as three separate things and know which one you are presenting.
17. Almost none of the narrative is written by the bank, and the tightest United Kingdom reference field is eighteen characters, so a payer reference is evidence of intent and never a key.
18. Reconciliation is not matching but explaining: timing, netting, granularity, sign and currency differences are all expected, and rules must eliminate every harmless one so the harmful one stands alone.
19. In a payments business it is three-way — internal ledger, processor report and bank statement — run daily, at item level, with breaks categorised, owned, aged and escalated.
20. Between the first quarter of 2018 and the second quarter of 2023 twelve safeguarding payments firms became insolvent with an average 65 per cent shortfall in client funds, which is why reconciliation is the control that turns a database of intentions into a defensible statement of obligation.

Sources used: The Payment Services Regulations 2017, SI 2017/752, regulation 89 (legislation.gov.uk). FCA Handbook, CASS 7.15.12R, 7.15.13R, 7.15.15R, 7.15.20R, 7.15.22R, 7.15.27R, 7.15.29R and 7.15.31R. Financial Conduct Authority, CP24/20 "Changes to the safeguarding regime for payments and e-money firms" (September 2024) and PS25/12 (7 August 2025), together with the FCA's safeguarding requirements page for payment and e-money institutions. ISO 20022 external code sets: BalanceType (OPBD, CLBD, OPAV, CLAV, ITBD, ITAV, FWAV, PRCD, XPCD, INFO) and EntryStatus (BOOK, PDNG, INFO, FUTR); camt.052, camt.053 and camt.054 message definitions, including the SWIFT camt.053.001.02 message definition report. SWIFT MT940 and MT942 specifications, field :61: format 6!n[4!n]2a[1!a]15d1!a3!c16x[//16x][34x]. Banking Administration Institute BAI2 cash management balance reporting format, record codes 01, 02, 03, 16, 88, 49, 98 and 99 and account type codes 010, 015, 040 and 045. Open Banking Limited, Read/Write API Standard v3.1.3, "Domestic Payment Message Formats", covering the Bacs Standard 18 mapping derived from Bacs "Electronic Funds Transfer, File Structures" (PN5011), the Faster Payments ISO 8583 field mapping and the CHAPS MT103 mapping. Visa Acceptance Solutions developer documentation on the final authorisation indicator. Worldline Acquiring documentation, "Authorization validity", scheme validity periods for the European

region. Pay.UK, Image Clearing System.

VOLUME II

The Card

The plastic, the chip that is a computer, the cryptogram it signs, the terminal that reads it, and the token that stands in for it.

CHAPTER

11

What Is Physically on a Card

11.0 What this chapter gives you

1. You will be able to name the six physical forms in which one card carries one piece of information, say which three are obsolete, and explain why all six are still there.
2. You will be able to read a track 2 record character by character, from the start sentinel to the check character, and say what each field is for.
3. You will be able to show that the 79-character and 40-character track limits are not round numbers but what the bit densities allow across 2.80 inches of usable track.
4. You will be able to explain why chip data written onto a blank magnetic stripe produces a card that is declined, and why that is a scheme rule rather than an implementation accident.
5. You will be able to read a three-digit service code and say whether the card is international, whether it carries a chip, and what the terminal should do about a PIN.
6. You will be able to say which of the eight ISO/IEC 7816-2 contacts do anything under EMV, and why C6 is a fossil.
7. You will be able to explain why a terminal accepting six brands runs six transaction state machines behind one tap.
8. You will be able to say why tag `57` is a track 2 record in shape only, and why tag `5F34` matters to anyone reconciling on the account number alone.
9. You will be able to justify the magnetic stripe's survival to 2033 in terms of declined transactions rather than technology, and read the economics off the published timetable.
10. You will be able to describe how a card is laminated, punched, milled and personalised, and say where the boundary falls between plastics manufacturing and cryptographic key management.

Take the card out of your wallet and put it on the table in front of you. It weighs about five grams, and it carries, in six different physical forms, one piece of information that could be written on the back of a receipt: which account to charge.

Six. Not one. The number is stamped into the plastic so it stands proud of the surface. It is printed in ink. It is written magnetically into a strip of iron oxide. It is stored electrically inside a silicon chip. It can be recited over a radio link at 13.56 megahertz. And on many cards it is printed a second time, in small type, on the back.

Three of those six are obsolete. All six are still there.

This chapter is about the object itself: what each layer physically is, which standard governs it, what it holds, and why an industry that is otherwise ruthless about cost has not thrown any of it away.

The plain version

Imagine you run a shop, and you have to leave a note for whoever opens up tomorrow. You do not know who that will be. It might be your grandmother, who only reads handwriting. It might be your colleague, who only checks the noticeboard. It might be the new assistant, who only reads text messages. You cannot teach any of them a new habit before morning, so you write the same note three times: once in pen, once pinned to the board, once as a text. The reader decides which copy they see.

A payment card is that note. It has to be readable by a card machine installed last month in a London coffee shop, by a petrol pump in rural Nebraska that has not been touched since 2009, and by a hotel in a country where the phone line to the bank is unreliable. Those machines cannot be upgraded on your behalf before you next buy a sandwich. So the card carries the same message in several formats at once.

What each copy of the note actually is

The raised numbers on the front. These exist because of a machine most people under forty have never seen. Before card machines had telephone lines, a shop had a metal device with a roller. The shopkeeper laid your card in it, put a paper form on top with carbon paper underneath, and rolled the handle across. The raised digits pressed through the carbon and printed your account number on the form, which went in the post to the bank. That is why the numbers stick out rather than being printed flat: the card was a printing plate.

The brown or black stripe on the back. This is magnetic tape, exactly the same idea as a cassette. It has three parallel lanes running along its length, and a machine reads it by dragging a tiny magnetic sensor along one lane while you swipe. The whole stripe holds less than a short text message. The first lane holds seventy-nine characters, enough for your account number, your name, the expiry date and a few extra digits. The second holds forty and no name. The third holds a hundred and seven and is, in practice, empty.

The gold square. This is not a memory chip. It is a computer, with a processor, memory and a program, powered by the card machine through those gold contacts, which is why nothing happens until you push the card in. Unlike the stripe, the chip does not simply hand over what it knows. The terminal asks it questions and it answers with the results of calculations. That difference is the entire reason chips replaced stripes.

The invisible loop of wire. Somewhere inside the plastic, running around the edge of the card, is a coil of very fine wire you cannot see or feel. When you hold the card near a contactless reader, the reader's own magnetic field induces a current in that loop, and the current powers the same chip. The card has no battery. It borrows power from the reader for the fraction of a second it needs.

The shiny hologram. A shopkeeper in 1990 could not phone the bank for every purchase, and needed a way to tell by eye that the card was real. A hologram was chosen because in 1985 you could not photocopy one, and printing one required equipment a forger with a garage did not have.

The white strip you sign. Once, the shopkeeper compared your signature on the receipt with the one on this strip. That comparison stopped being required years ago, and most people's cards are now unsigned.

A worked example

Suppose your card number is 4111 1111 1111 1111, it expires in December 2026, and your name is on it. Here is what is actually written into the second lane of the magnetic stripe, character for character:

```
;4111111111111111=26122011234567890 12?
```

Read it left to right. The semicolon says "the data starts here". Then sixteen digits of account number. The equals sign says "account number finished". Then `2612`, which is December 2026 written year-first. Then `201`, a three-digit code telling the terminal what the card is allowed to do. Then twelve digits the bank uses for its own purposes. Then a question mark, which says "data ends here", followed by one more character the reader uses to check it did not misread anything.

Thirty-eight characters. That is the whole of what a swipe transmits, and if you can read it you can make a working copy of the card. That single sentence is the reason the chip exists.

The first lane holds a richer version of the same thing, with your name in it:

```
%B4111111111111111^SINGH/SHIKHAR^2612201123456789?
```

The chip, by contrast, holds tens of thousands of characters, most of which it will never show anybody, because most of it is keys.

Where the plain version stops being true

The note-in-three-formats picture is the right shape. Four things about it are wrong, and each of the four matters.

First, the copies are not identical, and the differences are the security. The plain version says the chip and the stripe carry the same message. They carry deliberately different messages. The chip does hold something that looks like the second lane of the stripe, and it even uses the same layout, but one field inside it is set to a different value on purpose. If a criminal reads the chip's copy and writes it onto a blank magnetic stripe, the resulting card will be declined, because the value the issuer expects from a swipe is not the value the chip carries. This is not an accident of implementation. It is a scheme rule, and it is why chip data cannot be trivially downgraded into a stripe clone. Chapter 13 is about the four different check values that make this work.

Second, the chip does not "hold" your account number in the sense the analogy implies. It holds it, yes, and will read it out. But the thing the terminal actually needs from the chip is not stored on it at all: it is computed, freshly, for that transaction, using a secret key that never leaves the silicon. The card is not a container. It is a participant. Treating it as a container is the single most common misunderstanding among people building their first card integration, and it leads directly to designs that try to "read the card and send the data", which is exactly what EMV was built to make impossible.

Third, "obsolete" and "unused" are different words. It is tempting to read the embossing, the hologram, the signature panel and the stripe as vestigial decoration. Three of the four genuinely are. The stripe is not: it is still the fallback when a chip fails to read, still the only thing some unattended machines can handle, and still in daily use in parts of the world. An issuer removing the stripe is not simplifying a card; it is accepting a measurable rate of declined transactions in exchange. That is a commercial decision with a number attached, and it is why removing it has needed a decade-long timetable.

Fourth, the plastic itself is engineered, not just cut to size. The plain version treats the card body as an inert rectangle. It is a laminated stack of polymer sheets built to survive bending, torsion, being sat on, being left on a dashboard, and thousands of passes through a reader, all while keeping a chip module and a wire antenna intact inside it. There are international standards for how far it may bend before it has failed, and how large a burr the punching die may leave on its edge. Card manufacturing is a precision plastics business with a cryptographic key-management business bolted to the end of it.

With those four corrections in place, the technical version will land properly.

The technical version

The card body: ISO/IEC 7810 and what ID-1 means

The format is called ID-1, and it is defined in ISO/IEC 7810, *Identification cards — Physical characteristics*. The nominal dimensions are 85.60 mm by 53.98 mm with a nominal thickness of 0.76 mm. ISO/IEC 7810:2019 states these as tolerance bands rather than nominals: for an unused ID-1 card, width 85.47 mm to 85.72 mm, height 53.92 mm to 54.03 mm, and thickness 0.68 mm to 0.84 mm. Corners are rounded to a radius of 2.88 mm to 3.48 mm. Edge burrs left by the punching die must not stand more than 0.08 mm above the card surface.

The same standard defines the other card sizes you will meet in this industry: ID-000 at 25.10 mm by 15.10 mm maximum, which is the full-size SIM outline; ID-2 at approximately 105 mm by 74 mm; and ID-3 at approximately 125 mm by 88 mm, which is the passport format. All four share the same thickness band.

ID-1 cards are also specified for mechanical behaviour. Under the standard's bending test, an ID-1 card's deformation must fall between 13 mm and 35 mm: too rigid and it cracks around the chip cavity, too limp and it will not feed through a reader. The test procedures themselves live in a separate series, ISO/IEC 10373-1, *Cards and security devices for personal identification — Test methods — Part 1: General characteristics*, and card service life evaluation is covered by ISO/IEC 24789-1:2024 and ISO/IEC 24789-2:2024.

A conventional payment card is not a solid slab. It is a laminated sandwich: overlay, printed front core, an inner inlay carrying the antenna, printed back core, rear overlay. The magnetic stripe sits on or within the rear overlay. The overlays take the abrasion so the print does not.

The magnetic stripe

Two standards families govern the stripe. ISO/IEC 7811 covers the recording technique: what the stripe is, where it is, and how bits are written to it. ISO/IEC 7813 covers what a financial card must put on tracks 1 and 2. Track 3 content is specified separately in ISO/IEC 4909.

ISO/IEC 7811 is published in parts, and the numbering has a history that catches people out:

Part	Title	Status
7811-1	Embossing	Current
7811-2	Magnetic stripe — Low coercivity	Current
7811-3	Location of embossed characters on ID-1 cards	Withdrawn, folded into Part 1
7811-4	Location of read-only magnetic tracks — Tracks 1 and 2	Withdrawn, folded into Part 2
7811-5	Location of read-write magnetic track — Track 3	Withdrawn, folded into Part 2
7811-6	Magnetic stripe — High coercivity	Current
7811-7	Magnetic stripe — High coercivity, high density	Current
7811-8	Magnetic stripe — Coercivity of 51,7 kA/m (650 Oe)	Current

Part	Title	Status
7811-9	Tactile identifier mark	Current

If a supplier quotes you ISO/IEC 7811-4 for track geometry, they are quoting a withdrawn part. The geometry is still correct; the reference is stale.

Where the stripe is

ISO/IEC 7811-2 places the stripe on the back of the card, no closer than 2.92 mm (0.115 in) to the reference edge, with a minimum length of 82.55 mm (3.250 in) and a minimum width of either 6.35 mm (0.25 in) or 10.28 mm (0.405 in) depending on how many tracks are encoded. A stripe 82.55 mm long on a card 85.60 mm wide is effectively edge to edge.

The three tracks are parallel bands measured from the top reference edge. The figures as originally published in ISO/IEC 7811-4 and 7811-5 are:

Track	Encoded data lies between
Track 1	5.66 mm (0.223 in) and 8.46 mm (0.333 in)
Track 2	8.97 mm (0.353 in) and 11.76 mm (0.463 in)
Track 3	12.52 mm (0.493 in) and 15.32 mm (0.603 in)

Each band is therefore about 2.80 mm wide, with a guard gap of half a millimetre or so between them. Longitudinally, the centreline of the first data bit — the start sentinel — sits 7.44 mm $\pm$ 0.50 mm (0.293 in $\pm$ 0.020 in) from one end of the card, and the last data bit must not come closer than 6.93 mm (0.273 in) to the other. That leaves roughly 71 mm, or 2.80 inches, of usable track. Remember that figure; it is what makes the character limits come out where they do.

How bits are written

The recording method is two-frequency coherent-phase encoding, universally abbreviated F2F and also known as Aiken biphase. There is a flux transition at every bit-cell boundary; a cell containing a `1` has an additional transition in the middle of the cell and a cell containing a `0` does not. ISO/IEC 7811-2 states it plainly: a flux transition occurring between clock transitions signifies a one, and its absence signifies a zero.

That is the property which made magnetic stripes work at all in the 1960s. Because every bit cell begins with a transition, the reader recovers the clock from the data itself, so it does not care how fast you swipe, only that you swipe at a roughly constant speed. A self-clocking code was the price of building a reader with no motor and no precision transport.

Average bit densities are:

Track	Density	Bits per character	Character set	Maximum characters
Track 1	8.27 bits/mm (210 bpi)	7	64 alphanumeric	79
Track 2	2.95 bits/mm (75 bpi)	5	16 numeric	40

Track	Density	Bits per character	Character set	Maximum characters
Track 3	8.27 bits/mm (210 bpi)	5	16 numeric	107

Now the arithmetic. Track 1 at 210 bits per inch across 2.80 inches of usable length yields about 588 bits, which at 7 bits per character is 84 characters — comfortably above the 79-character maximum. Track 2 at 75 bits per inch across the same length yields about 210 bits, which at 5 bits per character is 42 characters, against a 40-character maximum. The limits in the standard are not arbitrary round numbers; they are what the physics allows with margin.

Track 1's 64-character set is the printable ASCII range from `0x20` to `0x5F` with 32 subtracted from each value, giving six data bits per character. Track 2 and track 3 use a 16-character set: the digits `0` to `9` plus the six symbols `: ; < = > ?`, which is ASCII `0x30` to `0x3F` with 48 subtracted, giving four data bits per character. In both cases the data bits are followed by one odd parity bit: ISO/IEC 7811-2 requires that the total number of one-bits recorded for each character, including the parity bit, be odd. Bits within a character are recorded least significant first.

After the end sentinel comes a longitudinal redundancy check character. The rule is column parity: taken across every character on the track including the LRC itself, the count of one-bits in each corresponding bit position must be even. The LRC's own parity bit is still odd parity, computed over the LRC character. Between per-character odd parity and per-column even parity, a single misread bit is always detected and its position can usually be located.

Coercivity

Coercivity is the field strength needed to flip the magnetisation of the stripe's oxide particles, and it distinguishes a stripe that survives a handbag magnet from one that does not. The ISO standards are explicit about *not* specifying it: ISO/IEC 7811-6 states that coercivity influences many of the quantities specified in the document but is not itself specified. What the standards specify instead is the resulting signal amplitude, which is what a reader actually cares about.

The commercial vocabulary is nonetheless universal. Low coercivity ("LoCo") stripes are typically brown; high coercivity ("HiCo") stripes are typically black. The figures usually quoted are around 300 Oe for LoCo and 2,750 Oe or 4,000 Oe for HiCo, and card manufacturers and reader vendors quote them with striking consistency, but no public specification states them; they are trade convention rather than published requirement, which is exactly what one would expect of a family of standards that declines to specify coercivity at all. What is verifiable is that a specific intermediate grade has a standard of its own: ISO/IEC 7811-8 is titled *Magnetic stripe — Coercivity of 51,7 kA/m (650 Oe)*. Payment cards are HiCo; hotel keys and transit tickets are usually LoCo, being disposable.

What is actually written: ISO/IEC 7813

ISO/IEC 7813 defines the data content of tracks 1 and 2 on a financial transaction card. Track 1 uses a format code; format code B is the one the financial industry uses, and format code A is reserved for the card issuer's proprietary use.

Track 1, format B, in order:

Field	Content	Length
Start sentinel	%	1
Format code	B	1
Primary account number	Digits	Up to 19
Field separator	^	1
Cardholder name	Surname, /, given name	2 to 26
Field separator	^	1
Expiration date	YYMM	4, or ^
Service code	Three digits	3, or ^
Discretionary data	Issuer-defined	Balance
End sentinel	?	1
LRC	Check character	1

Total: 79 characters maximum.

Track 2, in order:

Field	Content	Length
Start sentinel	;	1
Primary account number	Digits	Up to 19
Field separator	=	1
Expiration date	YYMM	4, or =
Service code	Three digits	3, or =
Discretionary data	Issuer-defined	Balance
End sentinel	?	1
LRC	Check digit	1

Total: 40 characters maximum.

Two details are routinely got wrong. First, the expiry date and service code are not unconditionally four and three digits: ISO/IEC 7813 permits a placeholder character in place of either, ^ on track 1 and = on track 2, for cards where the field does not apply. A parser assuming fixed offsets after the PAN will break on such a card. Second, the PAN is variable length up to 19 digits; its structure, and the issuer identification number inside it, are governed by ISO/IEC 7812 and are the subject of the next chapter.

The discretionary data field is where the interesting things hide. On most cards it carries a PIN verification field and a card verification value. The value encoded on the stripe is CVV1 or CVC1, and it is not the number printed on the back. Chapter 13 explains why there are four of these values and what attack each one defeats.

The service code

The service code is three digits, sits immediately after the expiry date on both tracks, and tells the terminal what the issuer permits. It is worth learning because it is also one of the inputs to the card verification value algorithm, which means an issuer cannot change it without invalidating every CVV already in the field.

Value	Position 1: interchange and technology	Position 2: authorisation processing	Position 3: allowed services and PIN
0	—	Normal	No restrictions, PIN required
1	International	—	No restrictions
2	International, integrated circuit card	By issuer, online	Goods and services only
3	—	—	ATM only, PIN required
4	—	By issuer, online, unless explicit bilateral agreement applies	Cash only
5	National	—	Goods and services only, PIN required
6	National, integrated circuit card	—	No restrictions, prompt for PIN if PIN pad present
7	Private	—	Goods and services only, prompt for PIN if PIN pad present
9	Test	—	—

So `201` is an international chip card, normal authorisation processing, no terminal restrictions. `221` is the same card with the issuer requiring online authorisation. `121` is a stripe-only international card. `702` is a private card, goods and services only, prompt for PIN where a pad exists.

A first digit of `2` or `6` is the stripe telling a swipe terminal *there is a chip on this card, use it*. That is the mechanism by which a chip-capable terminal refuses a swipe on a chip card and asks the customer to insert instead. It is also the field a fraudster with a stripe writer changes first, which is precisely why the service code is fed into the CVV computation.

Track 3

Track 3 was designed to be read *and written* by the terminal, so an offline ATM could decrement a balance held on the card itself. Its content is specified in ISO/IEC 4909. It is, in payments, dead: almost no issuer encodes it and almost no terminal reads it. It survives in the standards, and in the physical geometry of the stripe, as a reserved band of plastic that nothing uses.

The chip

The contact chip is governed by ISO/IEC 7816. Part 1 covers physical characteristics, part 2 the dimensions and location of the contacts, part 3 the electrical interface and transmission protocols, and part 4 the organisation, security and commands for interchange. EMV Book 1, *Application Independent ICC to Terminal Interface Requirements*, states directly that the EMV specification is based on the ISO/IEC 7816 series and then narrows it: EMV is a profile of 7816, not a superset.

The contacts

ISO/IEC 7816-2 defines eight contacts, C1 to C8, in two rows of four on the front face of the card. Each must present a conductive rectangle of at least 2.0 mm by 1.7 mm. Horizontally, measured from the left edge of the card, the four columns occupy bands at 19.23 mm to 20.93 mm, 21.77 mm to 23.47 mm,

24.31 mm to 26.01 mm, and 26.85 mm to 28.55 mm. Vertically, the upper row's required conductive area extends to 10.25 mm from the upper edge and the lower row's begins at 12.25 mm.

The assignments, as EMV Book 1 sets them out:

Contact	Signal	EMV use
C1	VCC	Supply voltage
C2	RST	Reset
C3	CLK	Clock
C4	Auxiliary	Not used; need not be physically present
C5	GND	Ground
C6	VPP	Not used; EMV cards shall not require VPP
C7	I/O	Serial input/output, half duplex
C8	Auxiliary	Not used; need not be physically present

Note C6. On the earliest smart cards, VPP supplied the elevated programming voltage needed to write EEPROM. Modern chips generate that voltage internally with a charge pump, so EMV forbids relying on it. If you have wondered why the gold plate has eight pads when only five do anything, that is the answer: two are reserved and one is a fossil. There is a second fossil: before 1990 an alternative contact position was in use, and ISO/IEC 7816-2 still discusses electrical isolation of the zones so a terminal can accept cards built to either convention.

Electrical behaviour

EMV Book 1 specifies that the card shall operate at 5 V $\pm$ 0.5 V DC with a maximum current requirement of 50 mA, and that the terminal shall supply 5 V $\pm$ 0.4 V DC with a steady-state output current in the range 0 to 55 mA. The clock frequency must be in the range 1 MHz to 5 MHz, and once the card has answered to reset the terminal must hold the clock within $\pm$1% of the frequency used during that answer to reset, for the remainder of the card session.

Two transmission protocols are defined: T=0, a character-oriented asynchronous half-duplex protocol, and T=1, a block-oriented asynchronous half-duplex protocol. EMV requires that a card support one of them and that a terminal support both. This asymmetry is deliberate: it lets issuers choose a chip platform without fragmenting the terminal estate.

What the chip holds

Data on an EMV chip is organised as BER-TLV data objects, each identified by a tag. The identity-carrying ones are:

Tag	Name
`4F`	Application Identifier (AID) — card
`50`	Application Label
`57`	Track 2 Equivalent Data
`5A`	Application Primary Account Number (PAN)
`5F20`	Cardholder Name
`5F24`	Application Expiration Date
`5F25`	Application Effective Date
`5F28`	Issuer Country Code

Tag	Name
5F30	Service Code
5F34	Application PAN Sequence Number
84	Dedicated File (DF) Name
87	Application Priority Indicator
8C	Card Risk Management Data Object List 1 (CDOL1)
8D	Card Risk Management Data Object List 2 (CDOL2)
9F1F	Track 1 Discretionary Data
9F20	Track 2 Discretionary Data

Tag 57, Track 2 Equivalent Data, is the one that causes confusion. It has the same field order as the magnetic stripe's track 2 — PAN, separator, expiry date, service code, discretionary data — but it is packed as compressed numeric, two digits per byte, so the separator is the hexadecimal nibble D rather than the character =, and the final nibble is padded with F if the digit count is odd. It is a track 2 record in shape only. Its discretionary data carries the chip's own card verification value, conventionally set to a different value from the stripe's, so that lifting tag 57 off a chip and encoding it onto a blank stripe produces a card that fails verification.

Note also 5F34, the PAN sequence number. When a bank reissues a card after loss without changing the PAN, this is the field that distinguishes the new plastic from the old. Reconciliation systems keying on PAN alone will merge two physically distinct cards, a bug that only surfaces during a dispute.

Everything above is data the chip will hand over. The keys it uses to generate cryptograms are not in that list, because they are not readable by any command. That is the point of the chip, and the subject of chapter 15.

The contactless antenna

The radio layer is ISO/IEC 14443, *Cards and security devices for personal identification — Contactless proximity objects*, in four parts: part 1 physical characteristics, part 2 radio frequency power and signal interface, part 3 initialisation and anticollision, part 4 transmission protocol. In 14443's vocabulary the card is a PICC (proximity integrated circuit card) and the reader is a PCD (proximity coupling device).

The physical parameters from ISO/IEC 14443-2 are:

Parameter	Value
Carrier frequency	13.56 MHz ± 7 kHz
Minimum operating field, Hmin	1.5 A/m rms
Maximum operating field, Hmax	7.5 A/m rms
Initial bit rate, both directions	fc/128, approximately 106 kbit/s
Subcarrier	fc/16, approximately 847 kHz

Two signalling types are defined and both are in use on payment cards. Type A uses 100% amplitude-shift keying with modified Miller coding from reader to card, and on-off keying of an 847 kHz subcarrier with Manchester coding from card to reader. Type B uses roughly 10% ASK from reader to card, with the modulation index required to lie between 8% and 14%, encoded NRZ-L, and binary phase-shift keying of the subcarrier from card to reader. A terminal must support both; a card implements one.

The 100% modulation of Type A is worth pausing on. Switching the field fully off is a blunt instrument, and during those gaps the card runs on whatever charge it has stored in its supply capacitor. Type B's shallower modulation never removes power. Both survive because both were deployed before either won.

The card side of this is a coil: several turns of fine enamelled copper wire ultrasonically embedded into an inlay layer of the card body, or an etched or printed conductive track on a substrate. ISO/IEC 14443-1 defines antenna size classes; a full-size payment card carries a Class 1 antenna, the largest, running close to the perimeter. The chip module connects to it either galvanically, by welding or soldering the wire ends to terminals on the underside of the module, or inductively, by putting a small coil on the module which couples to a matching loop in the card antenna. The inductive approach — Infineon's Coil on Module, and equivalents from other vendors — removes two solder joints from the most mechanically stressed part of the card.

On top of 14443 sits the EMV contactless specification set. It is published as *EMV Contactless Specifications for Payment Systems* in several books:

Book	Title
A	Architecture and General Requirements
B	Entry Point Specification
C-1 to C-8	Kernel Specifications
D	Contactless Communication Protocol Specification

Book D is the EMV profile of ISO/IEC 14443. Book B, Entry Point, is the logic that decides which kernel to invoke based on the application identifier the card presents. The kernels map to schemes as follows: Kernel 2 (Book C-2) for Mastercard AIDs, Kernel 3 for Visa, Kernel 4 for American Express, Kernel 5 for JCB, Kernel 6 for Discover, and Kernel 7 for UnionPay. Kernel 1 was removed from Book A as of version 2.7. Kernel 8, Book C-8, was published by EMVCo on 5 October 2022 as a scheme-neutral kernel intended to coexist with the legacy kernels and, over time, reduce the number terminals must carry.

That list is the honest answer to "why is contactless acceptance so complicated". A terminal accepting six brands runs six transaction state machines behind one tap.

Embossing

ISO/IEC 7811-1 governs embossing. It specifies three typefaces: ISO 1073-1 OCR-A in sizes I and IV, ISO 1073-2 OCR-B in sizes I and IV, and the type font Farrington 7B — the slightly squarish face you will recognise from every embossed card you have ever held.

Machine-readable embossed characters are 4.32 mm (0.170 in) high with a centre-to-centre spacing of 3.63 mm $\pm$ 0.15 mm (0.143 in $\pm$ 0.006 in). Visually readable characters, used for name and address lines, are set at 2.54 mm $\pm$ 0.15 mm (0.100 in $\pm$ 0.006 in) pitch.

Relief height — how far the character stands proud of the surface — is specified twice, once for unused cards and once for cards returned from the field, because embossing flattens with use:

Character type	Unused card	Returned card
Machine readable	0.40 mm to 0.48 mm	0.30 mm to 0.48 mm
Visually readable	0.36 mm to 0.46 mm	0.26 mm to 0.46 mm

The standard defines two embossing areas. Area 1 is the identification number line and accommodates a maximum of 19 character positions — the same 19 as the maximum PAN length, which is not a coincidence. Area 2 is the name and address area and accommodates four lines of 27 characters each.

Embossing exists because of the manual imprinter, and its machine-readability requirement exists because for a period the industry expected imprinted vouchers to be read back optically. Neither use case survives. Most cards issued today are flat: the PAN is applied by laser engraving, thermal transfer or indent printing, and a growing number carry no PAN on the front at all. Embossing on a modern card is a design choice about how the card feels in the hand, and usually a premium one.

The hologram

The optical variable device was introduced in the 1980s for one reason: a shopkeeper needed a counterfeit test that required no equipment, no phone call, and no training beyond "tilt it". It worked because holographic origination equipment was, at the time, capital-intensive and rare.

The schemes have been steadily reducing it. Visa's guidance is that hologram options were consolidated to a single design: vendors ceased production of the standard and mini dove designs on 31 December 2020, and from 1 January 2021 the updated Visa Product Brand Standards removed those options, leaving the silhouetted dove as the sole hologram. Cards carrying the Premium Visa Brand Mark do not require a dove hologram at all, because the brand mark itself carries the anti-counterfeiting security features.

The direction of travel is the same everywhere, and the reason is straightforward. A security feature that only works when a human inspects it loses its value in proportion to how many transactions involve a human inspecting anything. In a market where the overwhelming majority of face-to-face transactions are a tap and the rest are card-not-present, the hologram is defending a shrinking perimeter.

The signature panel

The panel is a printed receptive coating — a surface that ink will bind to, which the card's glossy overlay will not — usually laid over a tint or repeating pattern chosen so that an attempt to erase and re-sign leaves a visible mark.

Mastercard announced on 17 October 2017 that signatures would no longer be required at the point of sale in the United States and Canada from April 2018, and subsequently made the change global. From April 2019, Mastercard issuers globally were no longer required to include a signature panel on the back of Mastercard products. The other major schemes moved on comparable timelines.

The panel persists on many cards for two reasons that have nothing to do with payments: some jurisdictions and some non-payment uses of the card still expect a signature, and card designs are expensive to change so issuers batch such changes. Where the panel remains, it typically carries the printed card verification value — CVV2, CVC2 or CID depending on scheme — which chapter 13 distinguishes from the three other things called "the security code".

Tactile identifiers

ISO/IEC 7811-9 specifies a tactile identifier mark: a raised feature that lets a cardholder who cannot see the card determine what it is by touch.

The best-known implementation is Mastercard's Touch Card, announced on 25 October 2021. It places a notch on the short side of the card, with the shape encoding the product type: a squarish notch for credit, a rounded notch for debit, and a triangular notch for prepaid. The notch also tells the cardholder

which way round to insert the card, which matters when the chip contacts are on one face and one end only.

This is the one feature on the card that is getting more common rather than less, and the only one whose reader is a person rather than a machine.

Why all of this coexists

The honest answer is that a payment card is a network device with a five-year life, and the network it talks to has a thirty-year life. Every acceptance device in the world constrains what a card may stop carrying, and no issuer controls that estate.

The magnetic stripe is the clearest case, because for once there is a published timetable. Mastercard announced on 23 August 2021 that the stripe would no longer be required on new cards in selected regions including Europe from 2024; that the United States would follow from 2027; that from 2029 no new Mastercard cards would be issued with a magnetic stripe; and that by 2033 no Mastercard cards with stripes would remain in circulation. That is a twelve-year plan to remove a feature already technically obsolete in Europe.

Read that timetable and you can reconstruct the economics. Between 2021 and 2033 the cost of carrying the stripe — a few pence of tape and a persistent cloning risk — was judged lower than the cost of the transactions that would fail without it. The stripe is not there because anybody wants it. It is there because removing it costs more than keeping it, and the crossover point arrives at different dates in different regions. As of 2026 the early milestones are in force in Europe; the later ones remain a stated intention, and the industry has revised such timetables before.

The same logic, with smaller numbers, explains the embossing, the hologram and the signature panel. Each survives in the space between "no longer required" and "actively removed", and that space is measured in years.

Manufacturing and personalisation in outline

Card production splits cleanly into two phases with different security regimes. Card manufacturing produces cards that are identical to one another. Personalisation makes one card belong to one cardholder. Everything expensive about card security governance lives at the boundary between them.

Manufacturing

Artwork is separated and printed onto polymer sheets, many cards to a sheet — typical layouts are 21-up, 24-up or 48-up depending on the plant. Offset lithography remains dominant in secure card printing, with screen printing for metallic and opaque effects and digital printing growing for short runs.

The printed sheets are collated into a stack: overlay, front core, inlay, back core, overlay, with the antenna in the inlay. The stack is tacked together, usually by ultrasonic or thermal welding at the edges, so it cannot shift. Lamination follows: the stack goes between polished steel plates in a press, under heat and pressure, then cools under pressure. The polished plates give the card its finish; the cooling stage under load is what stops it curling.

The laminated sheet is then die-punched into individual ID-1 cards, which is where the 0.08 mm burr limit and the 2.88 mm to 3.48 mm corner radius are either achieved or missed. The magnetic stripe is applied either as part of the overlay film or hot-stamped as a tape after lamination. The hologram is hot-stamped.

For a chip card, a stepped cavity is then milled into the card face: a wide, shallow step to seat the contact plate flush with the surface, and a narrower, deeper pocket beneath it for the die and its bond wires. The module is placed over a hot-melt adhesive film and pressed by a heated die. On a dual-interface card, the module's antenna terminals are joined to the embedded card antenna at this stage, or coupled inductively. Every card is then tested electrically, contact and contactless, and visually.

Personalisation

Personalisation has a logical half and a physical half.

The logical half begins with data preparation inside a hardware security module. The issuer holds an Issuer Master Key; for each card the HSM derives card-specific keys by diversifying that master key over the PAN and the PAN sequence number, so compromising one card yields no information about any other. The output is a personalisation data set: application data, derived keys, the offline PIN, and the private key material used for offline data authentication.

Writing that data into the chip happens over a secure channel. The commands and structures are defined by the EMV Card Personalisation Specification, the channel by GlobalPlatform's secure channel protocols — SCP02, triple-DES-based, deprecated but still widely deployed, and SCP03, AES-based. Where the personalisation bureau is not the key owner, secrets travel under separate transport keys: a data transport key, a PIN transport key and a key transport key. The bureau's HSM decrypts under the transport keys and re-encrypts under the card's session keys, so plaintext PINs and card keys never exist outside an HSM.

The physical half is embossing or laser engraving of the PAN, name and expiry; thermal transfer or indent printing of the CVV2; encoding of the magnetic stripe; and attachment of the card to its carrier for mailing. The PIN mailer, where one is produced, goes separately and is generated inside an HSM without any human seeing the value.

Who polices this

The applicable standard is the PCI Card Production and Provisioning Security Requirements, published by the PCI Security Standards Council in two parts, Physical and Logical. Version 3.0 was announced on 13 January 2022 and covers card manufacturing, magnetic-stripe encoding and embossing, personalisation, chip initialising and embedding, storage, shipping and mailing, and the over-the-air provisioning of card credentials to devices.

One point matters commercially and is frequently misunderstood: while the requirements are maintained by PCI SSC, compliance is managed directly by the payment brands. A vendor works to the PCI document but is assessed and approved through each scheme's own programme.

One card, summarised

Feature	Governing standard	What it holds	Status
Card body	ISO/IEC 7810 (ID-1)	Nothing; it is the substrate	Current
Embossing	ISO/IEC 7811-1	PAN, name, expiry, in relief	Obsolete, common
Magnetic stripe	ISO/IEC 7811-2, -6, -7, -8	Tracks 1 and 2 per ISO/IEC 7813	Obsolete, in use, scheduled for removal
Track 3	ISO/IEC 4909	Nothing, in practice	Dead

Feature	Governing standard	What it holds	Status
Contact chip	ISO/IEC 7816, EMV Books 1 to 4	Application data and keys	Current, primary
Contactless antenna	ISO/IEC 14443, EMV Books A to D	Same chip, radio interface	Current, primary
Hologram	Scheme brand standards	Visual authenticity	Being reduced
Signature panel	Scheme brand standards	Signature, printed CVV2	Optional since 2019
Tactile mark	ISO/IEC 7811-9	Product type by touch	Growing

Six ways of saying one thing, one of which does the work.

The next chapter takes the thing itself — the number, all nineteen possible digits of it — and shows what each part of it means, who assigned it, and how a terminal knows in a millisecond that you have mistyped it.

11.98 Common wrong ideas

Wrong: The chip and the stripe carry the same message in two formats. **Right:** They carry deliberately different values in one field, so chip data copied onto a blank stripe is declined — a scheme rule, not an accident of implementation.

Wrong: The chip is a container you read the account number out of. **Right:** It is a participant; what the terminal needs is computed freshly for that transaction using a key that never leaves the silicon, which is precisely what EMV was built to require.

Wrong: The stripe, the embossing, the hologram and the signature panel are equally vestigial. **Right:** Three of the four are, but the stripe is still the fallback when a chip fails to read and still the only thing some unattended machines handle, so removing it buys a measurable rate of declines.

Wrong: The card body is an inert rectangle cut to size. **Right:** It is a laminated polymer stack with tolerance bands, a specified bending deformation between 13 mm and 35 mm, and a limit of 0.08 mm on the burr the punching die may leave.

Wrong: ISO/IEC 7811-4 is the reference to quote for track geometry. **Right:** Parts 3, 4 and 5 were withdrawn and folded into Parts 1 and 2; the geometry is still correct but the citation is stale.

Wrong: LoCo at 300 Oe and HiCo at 2,750 or 4,000 Oe are figures from the standards. **Right:** ISO/IEC 7811-6 explicitly declines to specify coercivity and specifies the resulting signal amplitude instead; those numbers are trade convention.

Wrong: Expiry and service code are always four and three digits, so track 2 can be parsed at fixed offsets after the account number. **Right:** ISO/IEC 7813 permits a placeholder character in place of either, and the account number itself is variable up to nineteen digits.

Wrong: Tag `57` is the stripe's track 2, so the two can be compared as strings. **Right:** It is compressed numeric at two digits per byte, with the hexadecimal nibble `D` as separator and `F` padding, and its discretionary data carries a different verification value on purpose.

Wrong: One account number identifies one piece of plastic. **Right:** A reissue after loss can carry the same number, distinguished only by tag `5F34`, so reconciliation keyed on the number alone merges two physically distinct cards and the bug surfaces during a dispute.

Wrong: The gold plate has eight contacts because eight are needed. **Right:** Only five do anything under EMV; C4 and C8 are auxiliary and need not be physically present, and C6's programming voltage is a fossil that modern chips generate internally.

11.99 Chapter summary in 20 lines

1. A payment card carries one piece of information — which account to charge — in six different physical forms, three of which are obsolete and all six of which are still present.
2. It works like a note left in several formats at once, because the acceptance estate cannot be upgraded on your behalf before you next buy a sandwich.
3. The card body is ID-1 under ISO/IEC 7810, nominally 85.60 mm by 53.98 mm and 0.76 mm thick, specified as tolerance bands with corner radii and a burr limit.
4. It is not a slab but a laminated sandwich of overlay, front core, antenna inlay, back core and rear overlay, with a bending range specified so it neither cracks round the chip nor jams in a reader.
5. The magnetic stripe carries three parallel tracks recorded in self-clocking F2F encoding, so the reader recovers its clock from the data and does not care how fast you swipe.
6. Track 1 holds 79 alphanumeric characters, track 2 holds 40 numeric characters, and track 3 holds 107 and is, in payments, dead.
7. Those limits are what the physics allows with margin: 210 bits per inch across 2.80 usable inches at seven bits per character gives 84, against a 79-character maximum.
8. Per-character odd parity combined with per-column even parity in the check character means a single misread bit is always detected and can usually be located.
9. The standards specify signal amplitude rather than coercivity, so the familiar LoCo and HiCo figures are trade convention rather than published requirement.
10. ISO/IEC 7813 fixes what tracks 1 and 2 must carry, and the discretionary data field is where the PIN verification field and the stripe's own card verification value hide.
11. The three-digit service code tells the terminal what the issuer permits, and a leading 2 or 6 is the stripe telling a swipe terminal that there is a chip and it should be used.
12. Because the service code feeds the card verification value computation, a fraudster who changes it invalidates the value, and an issuer cannot change it without invalidating every value already in the field.
13. The contact chip is governed by ISO/IEC 7816, of which EMV is a profile rather than a superset, with eight contacts of which five do anything.
14. Chip data is a set of BER-TLV objects, and tag 57 reproduces track 2 in compressed numeric with a deliberately different verification value, so a chip-to-stripe copy fails verification.
15. The keys that generate cryptograms appear in no tag list, because no command will read them out, and that is the point of the chip.
16. The contactless layer is ISO/IEC 14443 at 13.56 MHz, with two signalling types that every terminal must support and each card implements one of.
17. Above it sit the EMV contactless books with a kernel per scheme, which is the honest answer to why contactless acceptance is complicated: six brands mean six state machines behind one tap.
18. Embossing, the hologram and the signature panel each survive in the space between "no longer required" and "actively removed", and that space is measured in years.
19. The tactile identifier mark is the one feature becoming more common rather than less, and the only one whose reader is a person rather than a machine.
20. The stripe is scheduled out over a twelve-year timetable to 2033 because the cost of carrying it was judged lower than the cost of the transactions that would fail without it, which is the economics of every layer on the card.

Chapter sources: ISO/IEC 7810:2019; ISO/IEC 7811 parts 1, 2, 4, 5, 6, 8, 9; ISO/IEC 7813:2006 and earlier editions; ISO/IEC 4909; ISO/IEC 7816-2 (1999 and 2007 editions); ISO/IEC 14443-2; ISO/IEC 10373-1; ISO/IEC 24789-1:2024; EMV Book 1, ICC to Terminal Interface Requirements; EMV Contactless Specifications for Payment Systems, Books A and D; EMVCo tag references and the October 2022 Kernel 8 announcement; PCI SSC Card Production and Provisioning Security Requirements v3.0; Mastercard and Visa published brand and product announcements.

12

CHAPTER

The Number

12.0 What this chapter gives you

1. You will be able to verify a card number by hand, on paper, in about ninety seconds, and to build the check digit rather than merely test it.
2. You will be able to say exactly what a commercial BIN lookup can and cannot honestly tell you.
3. You will be able to explain why hard-coding a six-digit issuer identifier is now a defect, and why six-digit and eight-digit identifiers coexist rather than one having replaced the other.
4. You will be able to say why the schemes route on a nine-digit or eleven-digit account range rather than on the issuer identifier, and what a six-digit classification gets structurally wrong.
5. You will be able to demonstrate the check digit's two known blind spots by hand, and derive why 0 and 9 is the only adjacent transposition it misses.
6. You will be able to explain why anchoring the doubling at the rightmost digit is the only safe implementation, and why anchoring from the left makes every American Express card appear invalid.
7. You will be able to account for the four representations of one expiry date across the printed card, the stripe, the chip and the authorisation message, and say which one carries a day.
8. You will be able to say why an expired card does not fail at the chip, and what the terminal verification results actually record.
9. You will be able to give the five separate forces that make card number length vary, without conflating them.
10. You will be able to store, mask and truncate an account number correctly, and explain why masking and truncation are different controls with different limits.

Take a card out of your wallet and look at the long number on the front of it. Sixteen digits, usually, in four groups of four. Most people assume it is a serial number — that somewhere there is a very large list, and your card is entry number four trillion and something on it.

It is not that. Almost none of those digits are about you. The number is an address. It is structured, it is hierarchical, and it is read from the left in exactly the way a postal address is read from the bottom up: which country, which town, which street, which house. The first digits say which institution in the world is answerable for this card. The middle digits say which account at that institution. And the final digit is not information at all — it is a checksum, a digit whose only job is to catch the mistakes made by human beings and card readers.

This matters more than it sounds. When a terminal in a petrol station in Carlisle reads a card issued by a bank in Manila, nothing in that terminal knows anything about Philippine banking. What it knows is how to look at the leading digits, find a matching row in a table it was given, and work out where to send the message. The number is the routing key for the entire global card system, and everything in

the next several chapters — the cryptogram, the authorisation message, the interchange fee — depends on the leading digits of the number being correct and being interpreted correctly.

This chapter takes the number apart. By the end of it you will be able to verify a card number by hand, on paper, in about ninety seconds, and you will know exactly what a commercial "BIN lookup" can and cannot honestly tell you.

The plain version

Think of a telephone number.

If someone reads you a number starting `+44 20`, you already know two things before you have heard the rest. `44` means the United Kingdom. `20` means London. You have not been told whose phone it is, or whether they are in, or whether they will answer. But you know where to send the call, and that is enough to get started.

A card number works the same way, with one extra trick at the end.

The first eight digits say **which bank or card issuer** this card belongs to. Not which country, exactly, and not which branch — which *institution*, out of the tens of thousands in the world that issue cards. Those eight digits are handed out by a registry, the way telephone dialling codes are handed out, so that no two issuers can ever claim the same ones.

The digits after that say **which account** at that institution. That part is the issuer's own business. It might be sequential. It might be random. Nobody outside the bank needs to know which, and in fact banks deliberately make it hard to guess.

And the very last digit is a **check digit**. It carries no information about you, your bank or your account. It exists solely so that if somebody reads the number out wrong, or types it wrong, or a card reader misreads a smudged digit, the mistake is caught immediately instead of sending a payment to a stranger.

Why a check digit is worth a whole digit

Imagine reading a sixteen-digit number aloud over a bad phone line. Two things go wrong constantly. Either one digit gets heard wrong — a five becomes a nine — or two neighbouring digits get swapped, because human beings swap things. `...4536...` becomes `...4356...` and nobody notices.

You could catch this by reading the number twice. That is slow and people cheat. So instead the last digit is made to *depend* on all the others, by a rule that both ends of the line agree on. If the sixteenth digit does not match what the first fifteen say it should be, something has gone wrong, and you know before you send any money anywhere.

Giving up one digit out of sixteen costs you a tenth of your available numbers. Catching essentially every single-digit typo and nearly every swap is worth vastly more than that.

Working it out by hand

The rule is called the Luhn formula, and it is genuinely simple enough to do at a kitchen table. Here it is on a real card number — a test number published by payment gateways for exactly this purpose, so it belongs to nobody:

4012 8888 8888 1881

Write it out with the positions numbered from the left.

Step one: **double every second digit, starting from the first one on the left.** (That works because this number has sixteen digits, an even count. Come back to that in a moment.)

Step two: **if a doubled digit comes out bigger than nine, subtract nine.** So an 8 doubles to 16, and 16 minus 9 is 7. A 6 doubles to 12, and 12 minus 9 is 3.

Step three: **add up everything.**

Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Digit	4	0	1	2	8	8	8	8	8	8	8	8	1	8	8	1
Doubled?	yes	no	yes	no	yes	no	yes	no	yes	no	yes	no	yes	no	yes	no
Becomes	8	0	2	2	7	8	7	8	7	8	7	8	2	8	7	1

Add the bottom row: 8 + 0 + 2 + 2 + 7 + 8 + 7 + 8 + 7 + 8 + 7 + 8 + 2 + 8 + 7 + 1.

Take it in stages so you can check yourself: the first four come to 12. The next four (7, 8, 7, 8) come to 30, running total 42. The next four (7, 8, 7, 8) come to 30 again, running total 72. The last four (2, 8, 7, 1) come to 18. **Total: 90.**

Ninety ends in a zero. That is the whole test. **If the total ends in zero, the number is well formed. If it does not, somebody made a mistake.**

Watch it catch a mistake

Now break the number deliberately. Swap the thirteenth and fourteenth digits, so `...1881` becomes `...8118`. That is exactly the sort of slip a tired person makes.

Position 13 was a 1, doubled to 2. It is now an 8, which doubles to 16, minus 9 is 7. That is five more than before. Position 14 was an 8, not doubled, contributing 8. It is now a 1, contributing 1. That is seven less than before. Net change: minus two. The new total is 88.

Eighty-eight does not end in zero. **Rejected**, in one pass, before any message goes anywhere.

Try a single-digit error instead. Change the fifth digit from 8 to 3. It is a doubling position, so where it used to contribute 7 (from 16 minus 9) it now contributes 6. Total drops to 89. **Rejected** again.

Building the check digit rather than testing it

The bank does the same arithmetic backwards. It decides the first fifteen digits — `401288888888188` — and then works out what the sixteenth must be.

Run the sum on those fifteen digits with a zero standing in for the check digit. Everything above stays the same except position 16, which is now 0 instead of 1, so the total is 89 instead of 90.

Whatever the check digit is, it sits in a non-doubling position, so it just gets added. You need a total ending in zero. 89 needs 1 more. **The check digit is 1** — which is precisely what is printed on the card.

The general rule: take the total ending in whatever it ends in, and the check digit is the number that brings it up to the next multiple of ten. If your running total ended in 4, the check digit is 6. If it ended in 0 already, the check digit is 0.

Two warnings you can act on tonight

First, you can verify your own card this way in about a minute and a half, on paper. Do it on paper. Do not type your card number into a website that offers to "check if your card number is valid", because such a site has just been given your card number and you have learned nothing you could not have worked out with a pencil.

Second — and this is the part people get wrong — **passing this test does not mean the card exists.** It means the number is spelled correctly. A number can pass the Luhn test perfectly and correspond to no account anywhere on earth. It is a spell-checker, not a lock.

The date

The other number on the front is the expiry date, printed as month and year: `09/28` means September 2028.

Cards expire for boring reasons rather than sinister ones. The plastic wears, the chip contacts corrode, the magnetic stripe demagnetises in a pocket next to a phone, and the bank wants a periodic excuse to check that you still live where it thinks you live. A typical card runs three to five years.

The single fact people get wrong: a card marked `09/28` is good **through the whole of September 2028**, right up to the last second of the 30th. It stops working on 1 October. "Valid thru" means exactly that.

Why some cards are shorter

If you have an American Express card in the drawer, it has fifteen digits, not sixteen, grouped 4-6-5 rather than 4-4-4-4. Diners Club cards have historically had fourteen. Some debit cards abroad have nineteen.

The reason is history rather than logic. The card schemes invented their own numbering before there was an international standard, the standard was written afterwards to accommodate what already existed, and it set a ceiling — no more than nineteen digits — rather than a fixed size. Sixteen won because it was what the two biggest schemes chose, and because acceptance equipment built in the 1980s got used to it.

The arithmetic still works on all of them. But note the trap: because the doubling is anchored at the check digit on the *right*, a fifteen-digit number doubles a different set of positions counted from the left than a sixteen-digit number does. On a fifteen-digit Amex number you double the second, fourth, sixth digit and so on from the left. Get that backwards and every American Express card in the world will appear invalid to your software — which is, reliably, the first bug in every hand-written card validator ever shipped.

Where the plain version stops being true

The telephone-dialling-code picture is the right shape and it is wrong in four specific places, each of which costs money when a working system relies on it.

The issuer prefix is not six digits, and hard-coding six is now a defect

For roughly forty years the leading **six** digits were the issuer's identifier, and an entire generation of payment software, database schemas, fraud rules and merchant reports was written on that assumption. The relevant international standard changed in January 2017. The issuer identifier is now **eight** digits, and both Visa and Mastercard made support for eight-digit issuer identifiers mandatory for their participants from their April 2022 releases.

The plain version above already used the correct figure, which is why it is worth stating the correction loudly: any system built before about 2019 almost certainly assumes six, and a great deal of tooling still in production does. Six-digit and eight-digit identifiers now coexist — the old ones were not withdrawn, they were subdivided — so code that assumes either number in isolation is wrong.

Even eight digits is not how the network actually routes

Here is the correction that surprises practitioners. The issuer identifier tells you the institution. It does not tell you the **product**, and the product is what determines the interchange fee, the debit-or-credit distinction, whether the card is commercial, and which processing rules apply.

For that, the schemes route on something finer than the issuer identifier: an **account range**. Visa's is nine digits; Mastercard's is eleven. A single issuer identifier is deliberately carved into multiple account ranges so that one bank can put its consumer debit, premium credit and corporate cards on the same identifier with different behaviour.

Visa has told its acquirers explicitly, in its own operations bulletins, to process on the complete nine-digit account range record rather than the first six digits. A merchant system that classifies a card by its leading six digits is not merely out of date; it can and does classify cards into the wrong fee category.

Luhn is arithmetic, not security, and it never was

The check digit's rule was patented by an IBM engineer in the 1950s, has been published for over sixty years, and is in the public domain. It is in every standard, every textbook and roughly ten thousand GitHub repositories. It is designed to catch accidents, and it does that superbly. It is not designed to resist an adversary, and against one it is worthless — generating millions of Luhn-valid card numbers is a first-week programming exercise, and "card testing" attacks do exactly that, then hunt for the live ones by firing small authorisations at merchants.

There is a second, subtler failure that follows from the arithmetic itself. Luhn catches every single-digit error and every adjacent transposition **except one pair**, and it misses certain twin-digit substitutions. You can prove both by hand.

Take the four-digit number `1099`. Doubling the first and third: 1 becomes 2, 0 stays 0, 9 becomes 18 which reduces to 9, 9 stays 9. Total $2 + 0 + 9 + 9 = 20$. Valid. Now swap the middle two digits to get `1909`. Doubling the first and third: 1 becomes 2, 9 stays 9, 0 becomes 0, 9 stays 9. Total $2 + 9 + 0 + 9 = 20$. **Also valid.** Swapping an adjacent `0` and `9` is invisible to Luhn, always, everywhere, on every card number in the world.

The same trick works on twins. `1222` gives $2 + 2 + 4 + 2 = 10$, valid. Replace the middle `22` with `55` to get `1552`, and you get $2 + 5 + 1 + 2 = 10$, also valid. The pairs that collide this way are 22 against 55, 33 against 66, and 44 against 77.

None of this makes Luhn a bad check. It makes it a *known* check with *known* blind spots, which is a different thing from a guarantee.

The number does not identify a card, and may not identify a bank account

Three separate collapses of the plain version here.

A single card number can belong to several pieces of plastic. Replacement cards, and additional cardholders on the same account, may carry the same number distinguished only by a separate sequence number that is not printed on the card at all and lives only in the chip.

Worse for anyone building a lookup: the number in front of you may be a **network token** rather than a card number. When a card is added to a phone wallet or stored by a large merchant, the scheme can

issue a substitute number, of the same length, that passes the Luhn test and looks in every respect like a card number, but resolves to the real one only inside the scheme's own vault. This is the subject of a later chapter. Its consequence here is immediate: some of the numbers flowing through your systems are not the numbers printed on any card, and a lookup on their leading digits does not describe the issuer's card product.

And finally, the number does not identify a *bank account*. A card is an access mechanism attached to an account. The mapping is the issuer's private business, it is not one-to-one in either direction, and nothing in the number exposes it.

The technical version

ISO/IEC 7812-1: what the standard actually specifies

The governing document is **ISO/IEC 7812-1, "Identification cards — Identification of issuers — Part 1: Numbering system"**. The current text is the fifth edition, published January 2017, confirmed on systematic review in 2022 and still current as of writing in 2026. Its companion, **ISO/IEC 7812-2**, covers application and registration procedures rather than format.

The standard defines the **Primary Account Number** (**PAN**) as three concatenated components:

Component	Length	Assigned by
Issuer Identification Number (IIN)	8 digits	Registration Authority
Individual account identifier	1 to 10 digits	The issuer
Check digit	1 digit	Computed

The maximum total PAN length is **19 digits**, unchanged by the 2017 revision. The fifth edition's substantive change was to fix the IIN at eight digits rather than six. The mechanics of that change matter: existing six-digit IINs were not invalidated, they were **converted into a block of one hundred eight-digit IINs**. The six-digit IIN `412345` becomes the eight-digit block `41234500` through `41234599`, all still belonging to the same institution. This is why six-digit and eight-digit identifiers coexist rather than one having replaced the other.

Two further consequences are stated in the standard and are frequently missed. Issuers holding eight-digit IINs are required to issue a **minimum PAN length of ten digits** — you cannot have an eight-digit issuer identifier, a check digit and no account identifier at all. And with an eight-digit IIN and a nineteen-digit ceiling, the individual account identifier can be at most ten digits.

The Registration Authority for ISO/IEC 7812 is the **American Bankers Association**; administration passed to CUSIP Global Services acting on the ABA's behalf, reported as of February 2024.

The Major Industry Identifier

The first digit of the IIN is the **Major Industry Identifier** (**MII**), a classification inherited from the earliest days of the standard.

MII	Category
0	ISO/TC 68 and other industry assignments
1	Airlines
2	Airlines and other industry assignments

MII	Category
3	Travel and entertainment
4	Banking and finance
5	Banking and finance
6	Merchandising and banking/finance
7	Petroleum
8	Healthcare, telecommunications and other industry assignments
9	National assignment

Treat this table as archaeology, not as operational data. The MII is a historical classification with almost no present-day predictive value, and the clearest proof is Mastercard's own expansion: when Mastercard ran short of numbers under its traditional 51–55 prefixes it was allocated the range **222100–272099**, announced in November 2014 with participant systems required to be ready by October 2016. Every one of those cards is a mainstream consumer payment card whose MII digit is 2, a value the table above labels "airlines". Any code branching on the first digit to guess what kind of card it is holding is guessing.

The one MII with a live structural rule is 9, reserved for assignment by national standards bodies, where the convention is that the three digits following the 9 are the ISO 3166 numeric country code and the remainder of the number is defined nationally. In practice the national card schemes largely did not take up this space, choosing ranges allocated to them under the banking MIIs instead.

BIN, IIN and account range

The industry says **BIN**, for Bank Identification Number. The standard says **IIN**. For payment cards they denote the same leading digits; BIN is the older, commercial term and it has stuck so firmly that the schemes themselves use it in their bulletins. Where the two words genuinely diverge is scope: IIN is a general card-numbering concept covering non-payment cards too, whereas BIN in scheme usage carries specific commercial and licensing meaning.

The migration, scheme by scheme, as documented in the schemes' own material:

Visa. Effective with the **April 2022 VisaNet Business Enhancements release**, Visa began assigning eight-digit issuer BINs and required all clients to process using the eight-digit BIN structure. Visa will not assign new six-digit issuing BINs after that release, but both six-digit and eight-digit BINs exist in the estate. Notably, **acquirer numerics remain six digits** and were renamed "Acquirer Identifiers" — the expansion applied to the issuing side only. Visa also confirmed that PANs and tokens **remain sixteen digits**, with V PAY in Europe called out as an exception using nineteen-digit PANs in some implementations.

Mastercard. Adoption of the eight-digit standard was announced in 2017, with **April 2022** as the mandatory date for acquirers and service providers to support eight-digit BINs and **eleven-digit account ranges**.

American Express. Fifteen digits, prefixes 34 and 37. Amex is a three-party scheme, and its numbering, allocation and product identification are internal to Amex in a way that four-party scheme numbering is not.

The account-range arithmetic is worth doing explicitly, because it explains the schemes' issuance policies. Take a sixteen-digit PAN with an eight-digit BIN. Subtract the BIN and the check digit and you have **seven digits of individual account identifier: ten million accounts per BIN.** Visa's account range

is nine digits — the BIN plus one more — so a single eight-digit BIN divides into at most **ten** account ranges, which is exactly what Visa's own documentation states. Mastercard's eleven-digit range is the BIN plus three, giving up to a thousand ranges of ten thousand accounts each.

This is also why Visa's BIN utilisation policy, effective from its October 2017 release, instructs issuers not to randomise card issuance across an entire BIN but to **randomise starting at the tenth digit** — the tenth digit is the first digit that is not fixed by the nine-digit account range. The same bulletin instructs acquirers and processors to process transactions "based on complete nine-digit account range records in ARDEF tables rather than relying solely on the first six digits". ARDEF is Visa's **Account Range Definition** file, and it is a restricted product distributed to Visa clients through their account managers, not a public dataset.

Scheme prefixes and lengths

The following table is the conventional industry summary. Treat it as indicative. **The authoritative source for any given range is the scheme's own account range file**, not a published table, because ranges are added, moved and retired continuously.

Scheme	Leading digits	PAN length	Luhn
Visa	4	13, 16, 19	Yes
Mastercard	51–55	16	Yes
Mastercard 2-series	222100–272099	16	Yes
American Express	34, 37	15	Yes
Discover	6011, 644–649, 65	16–19	Yes
Diners Club International	30, 36, 38, 39	14–19	Yes
JCB	3528–3589	16–19	Yes
UnionPay	62	16–19	Yes
Maestro	5018, 5020, 5038, 5893, 6304, 6759, 6761–6763	12–19	Yes
Mir	2200–2204	16–19	Yes
RuPay	60, 65, 81, 82, 508	16	Yes

Two dated observations. Maestro is the widest-ranging length in the table and also the one being withdrawn: Mastercard began phasing Maestro out across Europe from **1 July 2023**, with issuers required to replace expired or lost Maestro cards with a different product. And note the overlap: 65 appears against Discover, RuPay and other schemes, which is a reminder that a two-digit prefix identifies nothing reliably.

The check digit, formally

The check digit is computed by the **Luhn formula**, described in US patent **2,950,048, "Computer for verifying numbers"**, filed by Hans Peter Luhn on 6 January 1954, granted 23 August 1960 and assigned to IBM. The patent has long since expired and the algorithm is specified within ISO/IEC 7812-1 for PAN check-digit computation.

The exact procedure, stated so that an implementation can be written from it without ambiguity:

1. Starting from the **rightmost** digit of the PAN, which is the check digit, and moving left, double the value of every second digit — that is, digits in positions 2, 4, 6 and so on counted from the right.
2. If a doubled value exceeds 9, subtract 9 from it. Equivalently, sum its two decimal digits; the results are identical for all inputs 0 to 9.

3. Sum all resulting values together with the undoubled digits.
4. The PAN is well formed if and only if the sum is congruent to 0 modulo 10.

To **generate** a check digit for a payload of n-1 digits, append a placeholder `0`, run steps 1 to 3, and set the check digit to `(10 - (sum mod 10)) mod 10`. The outer `mod 10` is not decorative: when the running sum already ends in zero the check digit is `0`, and an implementation that computes `10 - 0 = 10` and stores the low digit by accident will pass its own tests and fail in production on roughly one PAN in ten.

Anchoring at the right-hand end is the specification's wording and it is the only safe way to implement it, because it makes the algorithm indifferent to PAN length. Implementations that anchor from the left must branch on odd versus even length, and this is the single most common defect in hand-rolled validators. Worked on a fifteen-digit American Express test number, **3782 8224 6310 005**, the doubling positions counted from the left are 2, 4, 6, 8, 10, 12 and 14:

Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Digit	3	7	8	2	8	2	2	4	6	3	1	0	0	0	5
Becomes	3	5	8	4	8	4	2	8	6	6	1	0	0	0	5

The sum is 60, congruent to 0 modulo 10, and the number is well formed.

The blind spots demonstrated in the previous section are derivable rather than anecdotal. Define $f(d)$ as the doubled-and-reduced value of a digit minus the digit itself. Then $f(0) = 0$, $f(1) = 1$, $f(2) = 2$, $f(3) = 3$, $f(4) = 4$, $f(5) = -4$, $f(6) = -3$, $f(7) = -2$, $f(8) = -1$, $f(9) = 0$. An adjacent transposition changes the checksum by $f(a) - f(b)$, which is zero only when $f(a) = f(b)$, and the only distinct pair for which that holds is **0 and 9**. Every other adjacent transposition is detected.

For twin substitutions, define $g(d)$ as the doubled-and-reduced value plus the digit. Modulo 10 the values are 0, 3, 6, 9, 2, 6, 9, 2, 5, 8 for digits 0 to 9. The collisions are $g(2) = g(5)$, $g(3) = g(6)$ and $g(4) = g(7)$, giving the undetected substitutions **22<->55, 33<->66 and 44<->77**.

Finally, a scoping note. Within ISO/IEC 7812 the check digit is mandatory and a PAN that fails Luhn is malformed. Numbers that merely look like PANs but are issued outside ISO/IEC 7812 — some private-label and closed-loop store cards, for instance — are under no obligation to use it, and refusing them on Luhn grounds is a category error.

Expiry: four representations of one date

The expiry date exists in at least four forms in a live transaction, and they do not all have the same precision.

Printed on the card. Month and year, `MM/YY`, meaning valid to and including the final day of that month.

Magnetic stripe, ISO/IEC 7813:2006. On **track 2**, the fields in order are: start sentinel, PAN of up to 19 digits, field separator, expiration date as **four digits `YYMM`** (or the separator again if absent), a three-digit service code, discretionary data, end sentinel and a longitudinal redundancy check character, with a maximum record length of **40 numeric characters**. On **track 1**, Structure B, the record is up to **79 alphanumeric characters** with the expiry likewise four digits.

Note the capacity arithmetic that follows, because it is the constraint people reach for when explaining PAN lengths. Forty characters on track 2 less the start sentinel, end sentinel and LRC leaves 37 char-

acters of data — which is exactly why ISO 8583 defines its track 2 field as a maximum of 37. Subtract the field separator, four digits of expiry and three digits of service code and you have **29 characters to share between the PAN and the discretionary data**. A nineteen-digit PAN therefore leaves ten digits of discretionary data, which is tight but workable given that the discretionary data conventionally carries a PIN verification value and a card verification value; a sixteen-digit PAN leaves thirteen.

There is a live inconsistency here worth naming rather than papering over. **ISO/IEC 7813:2006 still describes the PAN as comprising a six-digit IIN and a variable individual account number of up to twelve digits**, because it predates the 2017 revision of ISO/IEC 7812-1. Two current ISO standards describe the same field differently. Implementers should follow ISO/IEC 7812-1:2017 for the IIN length and treat the 7813 text as historic in that one respect.

In the chip, EMV. The relevant data elements from the EMV Integrated Circuit Card Specifications, Book 3, Annex A:

Tag	Name	Source	Format	Template	Length
5A	Application Primary Account Number (PAN)	ICC	cn	70 or 77	var. up to 10 bytes
5F34	Application PAN Sequence Number	ICC	n 2	70 or 77	1 byte
5F24	Application Expiration Date	ICC	n 6 YYMMDD	70 or 77	3 bytes
5F25	Application Effective Date	ICC	n 6 YYMMDD	70 or 77	3 bytes
57	Track 2 Equivalent Data	ICC	b	70 or 77	var. up to 19 bytes

Three things to read off that table. First, tag `5A` is compressed numeric: two digits packed per byte, so ten bytes carry up to twenty digit positions of which at most nineteen are PAN, padded on the right with hexadecimal `'F'`. Second, tag `5F24` carries **six digits, `YYMMDD`** — the chip holds a day, where the printed card holds only a month. What the issuer puts in the day field is a personalisation decision; a terminal performs a date comparison and must not assume a particular value. Third, tag `57`, Track 2 Equivalent Data, reproduces the track 2 structure in binary with hexadecimal `'D'` as the field separator in place of the = character and `'F'` padding — which is why the chip and the stripe can produce byte-different but semantically identical data, and why comparing them naively as strings fails.

Tag `5F34`, the PAN Sequence Number, is the field that resolves the "one number, several cards" problem raised earlier: it identifies and differentiates cards carrying the same PAN.

In the authorisation message, ISO 8583. The relevant data elements:

DE	Name	Format
2	Primary account number	n ..19
14	Date, expiration	n 4, YYMM
35	Track 2 data	z ..37
45	Track 1 data	an ..76

DE2's `n ..19` is a variable-length numeric field of up to nineteen digits, carried with a two-digit length prefix. DE14 is fixed at four digits, `YYMM` — the message format drops the day that the chip carried.

What the terminal does with the date

Expiry checking happens in EMV **processing restrictions**, where the terminal compares the Application Expiration Date, tag `5F24`, against the Transaction Date, tag `9A`. The outcome is recorded in the **Terminal Verification Results**, tag `95`, a five-byte field. In byte 2:

Bit	Meaning
b8	Card and terminal have different application versions
b7	**Expired application**
b6	**Application not yet effective**
b5	Requested service not allowed for card product
b4	New card
b3–b1	Reserved for future use

Setting a TVR bit is not the same as declining. The TVR records observations; the decision of whether an observation causes a decline, an online authorisation request or nothing at all is made later, by comparing the TVR against the terminal action codes and issuer action codes. That comparison is the subject of a later chapter. The point to carry forward is that an expired card does not fail at the chip — it fails, if it fails, because somebody's action codes said it should.

Why PAN length varies, definitively

Five separate forces, and it is worth separating them because they are usually conflated.

The standard sets a ceiling and, since 2017, a floor. Nineteen digits maximum; ten digits minimum for issuers holding eight-digit IINs. Everything between is permitted.

History precedes the standard. American Express and Diners Club were issuing numbered cards before ISO/IEC 7812 consolidated the practice. The standard was written to accommodate the installed base, not to replace it, which is why fifteen and fourteen digit numbers remain conformant.

The magnetic stripe imposed a practical squeeze but never forced sixteen. As shown above, track 2 leaves 29 characters for PAN and discretionary data combined. Long PANs are possible and merely leave less room for the values that the stripe also had to carry.

Acceptance infrastructure calcified around sixteen. Merchant databases, printed forms, screen layouts and card-embossing conventions all standardised on four groups of four. This inertia is real enough that Visa, announcing the eight-digit BIN expansion, went out of its way to reassure the market that PANs and tokens **remain sixteen digits** and that card embossing alignment was unaffected. The nineteen-digit capability exists; it is used sparingly, and V PAY in Europe is the exception Visa itself names.

BIN capacity is the modern pressure. The whole reason for the 2017 expansion is that six-digit IINs ran out. Moving to eight digits multiplied the available issuer identifiers by a hundred, at the cost of two digits of account space per PAN — from nine digits of account identifier down to seven on a sixteen-digit PAN. That is the trade the industry made, and it is why the schemes now police BIN utilisation and instruct issuers to fill existing ranges before requesting new ones.

What a BIN lookup can and cannot tell you

A BIN lookup is a table join. Its answer is only ever as good as the table, and the tables differ enormously in provenance.

The authoritative sources are the schemes' own range files, distributed to their licensed participants: Visa's Account Range Definition (ARDEF) file, restricted to Visa clients, and Mastercard's equivalent account range data. Separately, in EMV 3-D Secure, the **Directory Server** supplies card range data to a 3DS Server in the **PRes** message, the response to a PReq, containing `startRange`, `endRange`, `acsStartProtocolVersion`, `acsEndProtocolVersion`, an optional `threeDSMethodURL` and an `actionInd` for each range. That is an authoritative, scheme-supplied range list, and it exists for one narrow purpose: to let the 3DS Server work out which protocol versions the issuer's Access Control Server supports before it attempts an authentication.

Everything else on the market — the commercial BIN databases sold by the gigabyte — is derived, aggregated and of variable freshness.

What a good lookup can tell you, subject to the table being current: the scheme, the issuing institution, the issuer's country, the product identifier and hence the broad product type (credit, debit, prepaid), and whether the card is a consumer or commercial product. Those attributes are genuinely useful, and they are used for real decisions: routing, fee categorisation, eligibility for dynamic currency conversion, fraud rules keyed on issuer country, and 3DS protocol version discovery.

What it cannot tell you, however good the table:

Whether the card exists. A range covers ten million potential accounts and a lookup does not know which are issued.

Whether the card is active, blocked, expired, or has any funds behind it. Those questions have exactly one answer, and it comes from an authorisation.

Anything about the cardholder. Not name, not address, not age, not creditworthiness.

Anything reliable at a granularity coarser than the account range. This is the correction that costs money: a six-digit lookup against a scheme that routes on nine or eleven digits can straddle several products with different funding sources and different interchange. If your table is keyed on six digits and the scheme's is keyed on nine, your answers are not merely stale, they are structurally incapable of being right.

Whether the number is a card number at all. EMVCo's Payment Tokenisation Specification – Technical Framework, which EMVCo publishes for public access, defines a payment token as a variable-length, ISO/IEC 7812-compliant numeric issued from a designated token BIN or token BIN range and flagged as such in the appropriate BIN tables; it passes the same basic validation rules as an account number, including the Luhn check digit, and it never takes the same value as a real PAN. A network token therefore has the length and shape of a PAN and satisfies Luhn, but its leading digits belong to a range set aside for tokens rather than to the issuer's card product, so a conventional lookup describes the token range and not the card.

Which scheme a transaction will actually route on. Co-badged cards, particularly domestic debit schemes in continental Europe, carry more than one acceptance brand on a single PAN, and the routing choice is made by rules outside the number.

Handling the number in a system you are responsible for

Four rules, each with a citable basis.

Store it as a string, never as an integer. IINs under MII `0` may begin with a zero, and integer storage silently destroys leading zeros. Separately, the largest signed 64-bit integer is 9,223,372,036,854,775,807 — itself nineteen digits — so the upper part of the nineteen-digit PAN space does not fit. And no arithmetic operation on a PAN is meaningful.

Accept the full permitted range of lengths. Twelve to nineteen digits will cover the schemes in circulation; ISO/IEC 7812-1 permits up to nineteen. Do not validate against sixteen.

Mask on display, truncate in storage, and know that these are different controls. PCI DSS separates them: **Requirement 3.4.1** governs masking when the PAN is displayed or printed, and **Requirement 3.5.1** governs rendering the PAN unreadable wherever it is stored. The PCI SSC's own guidance puts the distinction plainly: masked data can be unmasked, but there is no un-truncation without recovering the PAN from another source.

Know the truncation limits. PCI SSC FAQ 1091, as updated in June 2022, gives the maximum digits that may be retained:

PAN length	Brands	Minimum removed	Maximum retained
16 digits (6- or 8-digit BIN)	Discover, JCB, Mastercard, UnionPay, Visa	At least 4 digits	First 8, any other 4
15 digits	American Express	At least 5 digits	First 6, last 4
Fewer than 15 digits	Discover	—	First 6, any other 4

Three cautions on that table. It states **maxima permitted where there is a documented business need**, not defaults; first six and last four remains the format accepted by every brand and remains the sensible starting point. Mastercard's own position paper states that its allowable truncation format is not mandatory and that entities are not required to change their existing truncation as a result of the eight-digit BIN migration, and recommends retaining the fewest digits possible. And the PCI SSC warns specifically that holding the **same** PAN truncated **different ways** in different data stores can expose more digits in aggregate than either format permits alone — a failure mode that only appears when two systems are correlated, which is to say during a breach.

The number in one paragraph

A PAN is an eight-digit issuer identification number allocated by a registry, followed by up to ten digits of issuer-assigned account identifier, followed by one Luhn check digit, to a maximum of nineteen digits in total. The schemes route on finer account ranges than the issuer identifier — nine digits at Visa, eleven at Mastercard — because that is where product, funding source and fee category are determined. The check digit catches accidents and nothing else. The expiry date exists as four digits on the stripe and in ISO 8583, six on the chip, and means valid through the final day of the stated month. And a lookup on the leading digits tells you about a *range*, never about a card, and never about the person holding it.

12.98 Common wrong ideas

Wrong: A card number is a serial number, and somewhere there is a very large list. **Right:** It is a structured hierarchical address: which institution is answerable, which account at that institution, and one final digit that is not information at all.

Wrong: The issuer identifier is six digits. **Right:** It has been eight since January 2017, Visa and Mastercard made support mandatory from their April 2022 releases, and the old six-digit identifiers were subdivided into blocks of a hundred rather than withdrawn.

Wrong: The issuer identifier is what the network routes and prices on. **Right:** The schemes route on a finer account range — nine digits at Visa, eleven at Mastercard — because that is where product, funding source and fee category are determined.

Wrong: Passing the check-digit test means the card exists. **Right:** It means the number is spelled correctly; it is a spell-checker rather than a lock, and a well-formed number may correspond to no account anywhere on earth.

Wrong: The check digit gives some protection against an attacker. **Right:** The rule has been public for over sixty years, generating millions of valid numbers is a first-week programming exercise, and card-testing attacks do exactly that before hunting for the live ones.

Wrong: The check digit catches every adjacent transposition. **Right:** Swapping an adjacent 0 and 9 is invisible to it always and everywhere, and the twin substitutions 22 against 55, 33 against 66 and 44 against 77 also pass.

Wrong: The doubling starts at the first digit on the left. **Right:** The specification anchors at the rightmost digit, which makes it indifferent to length; anchoring from the left requires branching on odd against even and is the single most common defect in hand-rolled validators.

Wrong: The first digit tells you what kind of card you are holding. **Right:** The Major Industry Identifier is archaeology, and Mastercard's 222100–272099 range puts mainstream consumer cards under a digit the table labels "airlines".

Wrong: Every number flowing through your systems is a card number. **Right:** Some are network tokens issued from token ranges, of the same length and satisfying the same check, and a lookup on their leading digits describes the token range rather than any issuer's card product.

Wrong: A card number identifies one card and one bank account. **Right:** Replacements and additional cardholders can share a number distinguished only by the chip's sequence number, and the mapping to accounts is the issuer's private business and is not one-to-one in either direction.

12.99 Chapter summary in 20 lines

1. A card number is not a serial number but an address, read from the left in the way a postal address narrows from country to house.
2. ISO/IEC 7812-1 defines the primary account number as an issuer identification number, an individual account identifier and one check digit, to a maximum of nineteen digits.
3. Since the fifth edition of January 2017 that issuer identifier is eight digits rather than six, and each old six-digit identifier became a block of one hundred eight-digit ones.
4. That is why six-digit and eight-digit identifiers coexist, and why any system built before about 2019 almost certainly assumes the wrong figure.

5. Even eight digits is not how the network routes: Visa uses a nine-digit account range and Mastercard an eleven-digit one, because that is where product and fee category live.
6. Visa has told its acquirers explicitly to process on the complete nine-digit account range record rather than on the first six digits.
7. The first digit is the Major Industry Identifier, and it is archaeology, as Mastercard's 2-series consumer cards under an "airlines" digit demonstrate.
8. The last digit is a Luhn check digit: double every second digit from the right, subtract nine where a doubled value exceeds nine, and require the total to be congruent to zero modulo ten.
9. Worked on 4012 8888 8888 1881 the total is 90, and both a transposition of the last two digits and a single-digit change push it off a multiple of ten.
10. To generate rather than test, append a placeholder zero and set the check digit to ten minus the sum modulo ten, all taken modulo ten again, because the outer step is not decorative.
11. Anchoring at the rightmost digit makes the algorithm indifferent to length; anchoring from the left forces a branch on parity and makes every fifteen-digit American Express number appear invalid.
12. The check digit catches accidents superbly and adversaries not at all, and it has known blind spots rather than being a guarantee.
13. Those blind spots are derivable rather than anecdotal, from the doubled-and-reduced value of each digit taken against the digit itself.
14. The expiry date exists in four forms: `MM/YY` printed, `YYMM` on the stripe and in the authorisation message, and `YYMMDD` in the chip, which carries a day the other three drop.
15. A card marked 09/28 is good through the last second of 30 September 2028 and stops working on 1 October.
16. An expired card does not fail at the chip: the terminal records the observation in the terminal verification results, and whether that becomes a decline is decided later by the action codes.
17. Card number length varies for five separate reasons: the standard's ceiling and floor, history that predates the standard, the stripe's capacity squeeze, acceptance infrastructure that calcified around sixteen, and issuer identifier capacity pressure.
18. Track 2's forty characters leave twenty-nine to share between the number and the discretionary data, which is a squeeze rather than a rule forcing sixteen digits.
19. A BIN lookup is a table join whose authoritative sources are the schemes' own range files, and it can give scheme, issuer, country and product type but never whether a card exists, whether it has funds, or anything about the person holding it.
20. Store the number as a string, accept the full permitted range of lengths, mask on display and truncate in storage knowing these are different controls, and remember that the same number truncated two ways in two stores can expose more digits in aggregate than either format permits alone.

Chapter sources: ISO/IEC 7812-1:2017 (5th edition, confirmed 2022) and ISO/IEC 7812-2:2017, with the ISO announcement of the eight-digit IIN change and the ISO/IEC 7812 Registration Authority information sheet published via ANSI; ISO/IEC 7813:2006 (track 1 and track 2 structure); EMV Integrated Circuit Card Specifications for Payment Systems, Book 3, Annex A (tags 5A, 57, 5F24, 5F25, 5F34, 95, 9A); EMV 3-D Secure Protocol and Core Functions Specification (PReq/PRes card range data); EMV Payment Tokenisation Specification – Technical Framework (definitions of Payment Token, Token BIN and Token BIN Range); Visa bulletin on the implementation of eight-digit BINs, Visa article AI09153 on BIN management policies, Visa's eight-digit BIN numerics FAQ and Visa's eight-digit BIN PCI position paper; Mastercard's 2-Series BIN issuer impact checklist and its "8-Digit BIN Expansion and PCI Standards" paper; PCI DSS v4.0.1 requirements 3.4.1 and 3.5.1, PCI SSC FAQ 1091 (updated June 2022) and the PCI SSC masking-versus-truncation FAQ; US patent 2,950,048, Hans P. Luhn, IBM, filed 6 January 1954, granted 23 August 1960. Check-digit arithmetic, track capacity arithmetic and account-range capacity arithmetic in this chapter were computed and verified by

hand rather than quoted.

13

CHAPTER

The Four Things Called the Security Code

13.0 What this chapter gives you

1. You will be able to name the four values that a card carries under the heading "security code" — CVV1, CVV2, iCVV and dCVV — and say which of the four reading interfaces each one belongs to.
2. You will be able to explain why a criminal who copies every byte out of a chip onto a blank magnetic stripe gets a card that declines, and name the single issuer check that makes it decline.
3. You will be able to say why that same attack nevertheless succeeded against a supermarket group and a Georgia liquor retailer in 2020, and what the issuers concerned had failed to do.
4. You will be able to derive all three static values from the same algorithm, and explain why feeding in a service code of `000` rather than `999` changes every digit of the answer with no pattern connecting them.
5. You will be able to tell a caller that no bank anywhere can look up the printed code and read it back, because nothing is stored and every value is recomputed inside a hardware security module at authorisation time.
6. You will be able to correct the folklore that iCVV is what makes a chip transaction impossible to counterfeit, and name the Application Cryptogram as the actual defence.
7. You will be able to read a POS entry mode field and predict which value the issuer will compute, and so diagnose the decline caused by a terminal that reads a chip and reports a swipe.
8. You will be able to trace the printed code through the authorisation message on both Visa and Mastercard, and interpret the result codes `M`, `N`, `P`, `S` and `U` without treating them as the authorisation decision.
9. You will be able to list the places where PCI DSS requirement 3.3.1.2 is most often broken — debug logs, crash dumps, access logs, queues, call recordings, paper and backups — and explain why encrypting a retained code is no defence at all.
10. You will be able to explain why a monthly subscription can never re-present the printed code, and why the merchant must retain the `M` result instead of the value.

Ask a room of payments people what "the CVV" is and you will get one answer: the three digits on the back of the card. Ask the same room what the issuer's authorisation host actually checks, and the answers begin to diverge, because there is not one such value on a modern card. There are at least four. They are computed by the same family of algorithm, by the same machine, from mostly the same inputs, and they come out looking identical — three unremarkable decimal digits. They are not interchangeable. Presenting the wrong one at the wrong interface is precisely the event that a large part of card security is designed to detect.

The four are conventionally named CVV1, CVV2, iCVV and dCVV. Three of them are burned into the card once, at personalisation, and never change. One of them is manufactured fresh for every tap. One of them is printed in ink where a human can read it, and is the only one of the four that a merchant is forbidden to keep. The industry uses different names for all of them depending on which scheme's rulebook is open on the desk, which is why the same value can appear in three different documents as CVV, CVC and CID without anybody being wrong.

This chapter separates them. It explains what each one is a defence against, why removing any one of them would open a specific and historically exploited hole, and — the question that this chapter exists to answer properly — why copying the data off a chip and writing it onto a blank magnetic stripe does not give you a working card, except when an issuer has failed to do the one check that makes it not work.

The plain version

Imagine a building with four different doors: a front door, a side door, a delivery entrance and a gate at the back that opens onto the car park. All four doors belong to the same building. The same security office issues the passes for all four. But each door has its own pass, and the passes are deliberately different from one another, so that a pass for the delivery entrance is refused at the front door even though it was issued by the same office to the same person on the same morning.

Your bank card is that building. The four doors are the four ways a shop can read it.

The **first door is the brown stripe** on the back. A machine drags the card through a slot, reads the magnetism, and gets a line of digits.

The **second door is the printed code**, the three digits on the signature panel. Nobody's machine reads that. A human reads it, off the card in their hand, and types it into a website or reads it down a telephone.

The **third door is the chip**, the little gold rectangle. A terminal pushes the card into a slot, the chip wakes up, and the terminal asks it questions.

The **fourth door is the tap**. The card has an aerial hidden inside the plastic, and holding it near a reader lets the two of them talk by radio for about a fifth of a second.

Four doors, four passes. That is all "the four things called the security code" means.

The pass machine

The bank has a machine — really a locked box in a data centre, but think of it as a machine — that makes these passes. You feed it three things: your card number, your expiry date, and a three-digit **door number**. It thinks for a moment and prints out three digits.

The clever part is that the door number goes *into* the calculation. Change the door number, and every digit of the answer changes completely. There is no pattern connecting them. Knowing the answer for one door tells you nothing at all about the answer for another.

So the bank runs the machine three times for each card it makes. The numbers below are invented for the example — only the bank's machine knows the real ones — but the shape of what happens is exactly right.

What the bank feeds in	Door number	Pass that comes out	Where it is put
Card number and expiry	101 (the stripe's door)	561	Written into the brown stripe
Card number and expiry	000 (the printed door)	348	Printed in ink on the back
Card number and expiry	999 (the chip's door)	907	Stored inside the chip

Three passes. One card. Nothing about 561 helps you guess 348 or 907.

Why this stops a thief

Here is the attack the third row exists to defeat, and it is worth walking through slowly because it is the point of the whole chapter.

A criminal gets hold of a shop's card terminal, or infects the shop's till computer with software that watches everything the terminal sends. A customer inserts a chip card. The criminal's software captures everything the chip said: the card number, the expiry date, and the chip's pass — **907**.

Now the criminal buys a blank plastic card and a stripe-writing machine, both of which are cheap and legal, and writes the stolen data onto the stripe. The card number is right. The expiry is right. The pass written into the stripe is 907, because 907 is what was stolen.

The criminal walks into a shop and swipes it.

The shop's till sends the card number, the expiry date and the pass it just read — 907 — to the bank. The bank looks at the message and sees that this came in through the stripe door. So it runs its machine with the stripe's door number, 101, and gets **561**.

907 is not 561. **Declined.**

The criminal has a perfect copy of the card number and it is worthless at the stripe door, because the pass he stole was cut for a different door. That is the entire idea. It is called iCVV — the "i" is for the chip inside — and it is the reason the industry did not simply copy the stripe's pass into the chip when chips were introduced. Had it done so, every chip terminal in the world would have been a machine for mass-producing stripe clones.

The fourth pass is different

The tap door does not use a stored pass at all. It makes one up each time.

Inside the chip there is a counter. It starts at 1 when the card is made and goes up by one every single time the card is used. Nobody can wind it back. When you tap, the chip takes the current counter value, mixes it with a random number the reader has just invented, does its sum, and produces a fresh three-digit code that has never existed before and will never be valid again.

Somebody actually recorded this happening. A researcher held the same contactless card against a reader four times in a row and wrote down what came out each time. The card number was identical every time. The expiry date was identical every time. The last stretch of digits was different every time:

```
...0000072029191
...0000018129281
...0000054629311
...0000099829771
```

Same card, four seconds apart, four different endings. Copy one of those and try to use it again and the bank sees a counter value it has already seen, and refuses. This is called **dCVV**, for dynamic — a pass that is only good for one entry.

So why four?

Because each door is attacked differently.

The stripe is trivially copyable, so its pass exists only to prove that the copier saw a real stripe rather than merely learning a card number from a receipt.

The printed code is not in the card's data at all, anywhere, in any machine-readable form. That is its whole purpose. If a thief steals a database of card numbers from a shop's computer, or reads them off a stripe with a skimmer glued inside a cash machine, he does not get the printed code, because the printed code was never in the data. He would have had to physically hold the card and turn it over.

The chip's pass exists to keep the chip from becoming a stripe-cloning service, as above.

And the tap's pass exists because radio is broadcast: anyone standing nearby with the right aerial might overhear, so nothing overheard may be worth replaying.

One thing, many names

Visa calls it CVV, for Card Verification Value. Mastercard calls it CVC, for Card Validation Code. American Express calls it CID, and prints four digits on the front of the card rather than three on the back. Discover also says CID. JCB says CAV. China UnionPay says CVN. They are the same idea. When a specification says "CVC 2" and another says "CVV2", they mean the printed code, and a system that handles both must simply know that it is being told the same thing twice in two dialects.

The rule to remember

Of the four, exactly one must never be written down by anybody except the bank: the printed one. A shop is allowed to keep your card number for refunds. A shop is never allowed to keep the printed code — not in a database, not in a spreadsheet, not in a note on an order, not scribbled on a pad by the till. The moment a shop's payment has gone through, that code must be gone. Its only value is that it is scarce, and a code that thousands of shops are storing is not scarce.

Where the plain version stops being true

The building-with-four-doors picture gets you to the right conclusion, and it is wrong in four ways that matter to anyone implementing this.

There is no stored pass anywhere, and no lookup

The plain version implies the bank keeps a list: card 4123..., stripe pass 561, printed pass 348. It does not. Nothing is stored. Each value is *recomputed from scratch* at authorisation time, inside a hardware security module, from the card number, the expiry date, the service code and a pair of secret keys, and the freshly computed answer is compared with whatever arrived in the message. The keys are the only thing kept.

This has a consequence that surprises people, including bank staff: **an issuer cannot look up your CVV2 and tell you what it is.** It can only regenerate it. Some issuers' card-management systems expose that as a function, most deliberately do not, and none of them are reading it out of a column in a table. Similarly, when a call centre says "we can see the last four digits", they mean the card number; there is no screen anywhere with your printed code on it.

Four is a simplification in one direction and an undercount in the other

"Four" is a teaching number. In practice the same personalisation run may produce a CVV1 for the stripe, a distinct value for track 1 and track 2 discretionary data if the issuer chooses to differ them, an iCVV for the contact chip, a separate contactless value, a CVV2 for print, and — for issuers who offer it — a rotating code shown in the banking app that changes every few minutes. Card-not-present verification values used inside 3-D Secure are a separate family again, and are not card verification values at all.

Pulling the other way, the four are not equals. CVV1, CVV2 and iCVV are static three-digit numbers. dCVV and its Mastercard equivalent are dynamic. And none of them is what actually secures a chip transaction.

The security code is not why chip transactions cannot be cloned

This is the correction most worth internalising, because the popular explanation is backwards.

A chip transaction is not made hard to counterfeit by iCVV. It is made hard to counterfeit by the **application cryptogram** — a message authentication code the chip computes over the transaction amount, currency, date, terminal country, a terminal-supplied random number and its own transaction counter, using a key derived from a master key that the chip will not disclose and cannot be read out of it. Chapter 5 of this volume is entirely about that cryptogram. A criminal holding every byte of static data from a chip still cannot produce a valid cryptogram for tomorrow's £47.20 purchase in Leeds, because he does not have the key.

iCVV is not the lock on the chip door. iCVV is the thing that stops data taken from the chip door being reused at the stripe door. It is a *cross-channel* defence, not a chip defence, and it is doing a much narrower job than the folklore suggests.

The codes are only worth anything if somebody checks them, and sometimes nobody does

All of this assumes the issuer validates. In 2020 researchers at Cyber R&D Labs tested chip card implementations from ten banks across Europe and the United States, harvested data from four of them and produced cloned magnetic stripe cards that were used to place real transactions. In the same period, Gemini Advisory documented what it called EMV-Bypass Cloning in two breaches — a supermarket group where more than 720,000 cards were compromised from 16 January 2020, and a Georgia liquor retailer on 29 June 2020 — in which iCVV harvested from chip transactions was written onto counterfeit stripes and worked, because the issuers concerned were not checking that the value arriving on a stripe transaction was the stripe's value.

Visa's own commentary on those events makes the conditionality explicit: the data available from a chip transaction is the account number, iCVV and expiry date, and "provided iCVV is validated properly, the risk of counterfeit fraud was minimal". Provided. The mechanism is sound. Its deployment is uneven, and it is smaller issuers and processors, not the large ones, where the check has been found missing.

There is a fifth, subtler failure in the same family. The service code is an input to the calculation, but it is also a field that travels in the message. Visa has warned acquirers and issuers that service codes of 000 or 999 should never be encoded on a magnetic stripe, and that it "is aware of scenarios in which either 000 or 999 has been encoded on the magnetic stripe of counterfeit cards, resulting in issuer fraud losses". The attack is exactly what it sounds like: if an issuer's validation takes the service code from the incoming track data rather than from its own record of the card, a fraudster can encode 999 on the

stripe alongside a stolen iCVV, and the issuer will helpfully compute the chip's value and find that it matches.

The technical version

Where each value physically lives

The magnetic stripe layout is defined by ISO/IEC 7813, which specifies the data structure and content of tracks 1 and 2. Track 1, Structure B, carries up to 79 alphanumeric characters; track 2 carries up to 40 numeric characters at a lower recording density but a more compact character encoding. Track 3, defined separately in ISO/IEC 4909, holds 107 numeric characters and is unused by the card schemes in practice.

The field order on track 2 is: start sentinel `;`, primary account number of up to 19 digits, field separator `=`, expiry date as YYMM, three-digit service code, discretionary data, end sentinel `?`, and a longitudinal redundancy check character. Track 1 is the same information plus a format code `B`, the cardholder name between `^` separators, and its own discretionary data field.

Discretionary data is where CVV1 lives. ISO/IEC 7813 does not say what goes in it; it is left to the issuer. What the schemes put there is the three-digit card verification value, usually followed by a PIN verification key indicator and PIN verification value, and padding.

For the chip, EMV Book 3 defines tag `57`, Track 2 Equivalent Data, described as containing "the data elements of track 2 according to ISO/IEC 7813, excluding start sentinel, end sentinel, and Longitudinal Redundancy Check", packed two digits to a byte with hexadecimal `D` standing in for the `=` separator and a trailing `F` nibble as padding where the length is odd. Tag `9F1F` is Track 1 Discretionary Data. Tag `5F30` is the service code, formatted `n 3` in two bytes.

The composite picture:

Value	Lives in	Read by	Static or dynamic
CVV1 / CVC1	Discretionary data of the encoded magnetic stripe tracks	A stripe reader	Static
CVV2 / CVC 2 / CID / CAV2 / CVN2	Ink on the card body only	A human being	Static
iCVV / Chip CVC	Discretionary data inside EMV tag `57` and, where present, `9F1F`	A chip terminal, contact or contactless in EMV mode	Static
dCVV / CVC3	Substituted into discretionary data at read time	A contactless reader in magnetic-stripe mode	Dynamic, per tap

The generation algorithm

The card verification value algorithm originates with Visa and is implemented identically in every payment HSM. It is not published by the schemes, but it is documented in the command sets of the HSM vendors and in IBM's cryptographic coprocessor documentation, which describes the Visa CVV as calculated by "the DES-encryption of the PAN, the card expiration date, and the service code using two data-encrypting keys or two MAC keys". Those two keys are the **card verification key pair**, conventionally CVK A and CVK B.

The steps are:

1. Concatenate the primary account number, the expiry date and the three-digit service code as decimal digits, left-justify the result in a 128-bit field and pad to the right with binary zeroes.
2. Split the 128-bit field into two 64-bit blocks, Block 1 and Block 2.
3. DES-encrypt Block 1 under CVK A.
4. Exclusive-OR the result with Block 2.
5. Apply the EDE sequence to that: encrypt under CVK A, decrypt under CVK B, encrypt under CVK A. This is Triple DES with a two-key bundle.
6. Decimalise the 16 hexadecimal digits of the result in two passes. The first pass walks left to right and extracts every digit in the range 0 to 9. The second pass walks left to right again and extracts every digit in the range A to F, subtracting 10 from each so that A becomes 0 and F becomes 5. Concatenate the two extracted strings.
7. Take the leftmost three digits. For American Express, which prints four, take four.

Everything about which value you get is decided by step 1. The keys are the same, the arithmetic is the same, the only thing that varies is what you feed in:

Value	PAN	Expiry	Service code input
CVV1	Yes	Yes	The card's actual service code, for example `101` or `201`
CVV2	Yes	Yes	`000`
iCVV	Yes	Yes	`999`

Note carefully what this does and does not say. The `999` used to compute iCVV is a value *fed into the calculation*. It is not a claim about what is encoded in the service code field of tag `57`. Whether an issuer also encodes a distinguishing service code in the chip's Track 2 Equivalent Data is an issuer decision and implementations differ; Visa's warning about `000` and `999` on counterfeit stripes only makes sense in a world where the encoded field and the calculation input are separable things. Nothing in the algorithm requires the same key pair to be used for all three values either, and an issuer is free to provision a separate card verification key for iCVV; Amazon's payment cryptography documentation, for one, creates the iCVV key as a card verification key in its own right rather than reusing an existing one.

The output space is three decimal digits, so a thousand possibilities. That is deliberately weak against brute force and deliberately strong against the thing it is for. Guessing a CVV2 for a known card number takes on average 500 attempts, which is why the defence against CVV2 guessing is not the code's length but the issuer's velocity controls and the scheme's rules on repeated declines.

CVV1 and the magnetic stripe path

When a card is swiped, the terminal sends what it read. In ISO 8583 that is data element 35 for track 2 and data element 45 for track 1, and the schemes overwhelmingly prefer track 2: Visa's guidance to acquirers is blunt that "Track 1 should not be supported in the authorization message."

The issuer's decision about which value to compute is driven by data element 22, POS entry mode. The values that matter here, in Visa's numbering, are `01` for manual key entry, `02` or `90` for a magnetic stripe read, `05` or `95` for a contact chip read, `07` for contactless using chip data rules and `91` for contactless using magnetic-stripe data rules. A fallback swipe — a chip card at a chip-capable terminal where the chip could not be read — carries its own distinct entry mode so that the issuer can see it happened, which is the basis of every fallback-monitoring rule the schemes operate.

Entry mode is therefore load-bearing in a way that is easy to underestimate. Visa's own instruction to devices is that "if a transaction is processed as magnetic stripe, the track data used in the transaction is read from the magnetic stripe and correspondingly, if a transaction is processed as chip, the track data used should be read from the chip", and that the entry mode field "must be accurate to avoid unnecessary declines". A terminal that reads the chip and reports entry mode 90 will hand the issuer chip-derived track data labelled as a swipe, the issuer will compute CVV1, the iCVV will not match, and the transaction will decline for reasons that will take the merchant's acquirer several days to work out.

The corresponding rule for chip data is that "the terminal must always transmit the full, unmodified contents of the Track 2 Equivalent Data in the chip to the acquirer including the Issuer Discretionary data". Terminals that trim, re-pad or normalise the discretionary data destroy the iCVV in the process. This is a real and recurring integration defect, particularly in software stacks that parse tag 57 into fields and then reassemble it.

CVV2 and the card-not-present path

CVV2 is printed, never encoded. On Visa, Mastercard and Discover it is three digits in or beside the signature panel on the reverse. On American Express it is four digits printed on the front, above and to the right of the embossed account number, and Amex calls it the Card Identification Number.

Its carriage in the authorisation message is scheme-specific and this is one of the places where a generic integration goes wrong. In VisaNet the value travels in Field 126, usage 10, CVV2 Authorization Request Data, and the result comes back in Field 44, usage 10. In Mastercard's Authorization Request/0100 the value goes in data element 48, Additional Data — Private Use, subelement 92, and the result returns in DE 48 subelement 87, Card Validation Code Result.

The result codes are common across both:

Code	Meaning
`M`	CVV2/CVC 2 matched
`N`	Did not match
`P`	Not processed
`S`	The merchant indicated the code was not on the card, but the issuer says it should be
`U`	Issuer is not certified, or has not supplied keys to the scheme
blank	Unknown or not applicable

Two operational points follow. First, the CVV2 result is *not* the authorisation decision. An issuer may return an approval with a result of `N`. Whether the merchant proceeds is the merchant's risk decision, and a great many merchants approve transactions their own issuer has told them did not match. Second, the result is durable evidence. Mastercard's chargeback rules for recurring transactions accept, as part of the merchant's defence, that the CVC 2 was provided in the initial Authorization Request/0100 and that DE 48 subelement 87 came back with a value of `M`. The result code may be retained. The code itself may not.

iCVV and the chip

At personalisation the issuer computes the value with service code input `999` and writes it into the discretionary data portion of tag `57`, and into `9F1F` where track 1 equivalent data is personalised. From that moment the chip's static track data and the stripe's track data carry the same PAN, the same expiry and different verification values.

At authorisation the issuer sees entry mode `05`, `95` or `07`, takes the discretionary data out of the chip track data in DE 35 or from the ICC data in DE 55, recomputes with `999`, and compares.

What this defeats, precisely: the conversion of chip-sourced static data into a working magnetic stripe. Gemini Advisory's name for the attack — EMV-Bypass Cloning — captures it exactly, since it "leverages information from one technology (EMV chips) and converts it into another less-secure technology (magstripe)". The countermeasure is a single comparison, and its failure mode is silent.

Why cloning a stripe does not give you a working chip transaction

The reverse direction is where the asymmetry is total, and it is worth being exact about why, because it is the reason the entire EMV migration was worth its cost.

A chip transaction requires the card to produce an **Application Cryptogram**, EMV tag `9F26`, in response to a GENERATE AC command. The cryptogram is a MAC computed over data the terminal supplies — amount, other amount, terminal country code, terminal verification results, transaction currency, transaction date, transaction type and the Unpredictable Number in tag `9F37` — together with data the card supplies, principally the Application Interchange Profile and the Application Transaction Counter in tag `9F36`. The key is an application cryptogram session key derived from a card-unique key, which is itself derived from an issuer master key using the PAN and the PAN Sequence Number in tag `5F34`. Tag `9F10`, Issuer Application Data, carries the issuer's own proprietary block, including in most schemes the counters and derivation information the issuer needs to reproduce the calculation.

None of that is present in stripe data. The stripe carries a PAN, an expiry, a service code and three static digits. It carries no key, no counter and no ability to compute anything. A terminal presented with a card that has no chip either declines, or performs a fallback swipe that the issuer can see and refuse.

And a criminal in possession of every static byte from a chip is no better off, because he does not have the card's key. He can replay a captured cryptogram, and the Application Transaction Counter will be one the issuer has already seen for that card. He can attempt to build a card that generates cryptograms, and he cannot, because the derivation key never leaves the issuer's HSM or the chip. For cards supporting dynamic data authentication the position is stronger still: the chip holds an RSA private key it will sign challenges with and will not disclose, so even an offline terminal with no network connection can distinguish a genuine card from a copy.

The security codes are three digits of static data. The cryptogram is the actual defence. iCVV exists to make sure that the three digits of static data cannot be smuggled sideways into a channel that has no cryptogram to check.

dCVV, CVC3 and contactless magnetic-stripe mode

For a period, contactless cards in some markets were deployed in a mode designed to reuse existing magnetic stripe message formats end to end, so that acquirers and issuers needed no message changes at all. This is MSD, magnetic-stripe data mode, and it is where the dynamic values live.

The trick is that the card hands the reader a track image with holes in it, plus instructions for filling the holes. In EMVCo's Kernel 2 specification — the Mastercard kernel, Book C-2 of the EMV Contactless Specifications for Payment Systems — the relevant data objects are:

Tag	Name	Source	Length
56	Track 1 Data	Card	variable
9F6B	Track 2 Data	Card	variable
9F60	CVC3 (Track1)	Card	2 bytes
9F61	CVC3 (Track2)	Card	2 bytes
9F62	PCVC3 (Track1)	Card	6 bytes
9F63	PUNATC (Track1)	Card	6 bytes
9F64	NATC (Track1)	Card	1 byte
9F65	PCVC3 (Track2)	Card	2 bytes
9F66	PUNATC (Track2)	Card	2 bytes
9F67	NATC (Track2)	Card	1 byte
9F6A	Unpredictable Number (Numeric)	Terminal	variable
9F6C	Mag-stripe Application Version Number (Card)	Card	2 bytes

The names decode as follows. **PCVC3** is a bit map that "indicates to the Kernel the positions in the discretionary data field of the Track 2 Data where the CVC3 (Track2) digits must be copied". **PUNATC** is a bit map that "indicates to the Kernel the positions in the discretionary data field of Track 2 Data where the Unpredictable Number (Numeric) digits and Application Transaction Counter digits have to be copied". **NATC** "represents the number of digits of the Application Transaction Counter to be included in the discretionary data field of Track 2 Data".

The sequence at the reader is: select the application, read the static Track 1 and Track 2 Data and the position bit maps, generate a numeric unpredictable number, issue a COMPUTE CRYPTOGRAPHIC CHECKSUM command carrying it, and receive back a two-byte CVC3 for each track. The reader then converts the CVC3 to decimal, overwrites the positions the bit maps identified with the CVC3 digits, the unpredictable number digits and the low-order ATC digits, and sends the resulting track image onwards in an ordinary magnetic-stripe-format authorisation. To every system between the terminal and the issuer, it looks like a swipe. Only the issuer, which knows the card's CVC3 key and the expected ATC, can tell that it was not.

A published capture of a Mastercard PayPass card makes this concrete. The card returned application identifier `A0 00 00 00 04 10 10` and, in its magnetic-stripe record, these values:

```
9F6C  02  00 01
9F62  06  00 00 00 00 01 C0
9F63  06  00 00 00 00 00 3C
9F64  01  03
9F65  02  00 E0
9F66  02  00 1E
9F67  01  03
```

`9F64` and `9F67` are both `03`: three ATC digits go into each track. `9F65` is `00E0` and `9F66` is `001E`, two bit maps marking disjoint runs of digit positions in the track 2 discretionary data — three positions for the CVC3, four for the unpredictable number and counter. The static Track 2 Data in `9F6B` began `...D 1103 502 0000000...`, showing the hexadecimal `D` separator, an expiry of March 2011, a service code of `502`, and a discretionary data field of zeroes waiting to be overwritten.

Read four times in quick succession, the same card produced:

```
;5XXXXXXXXXXXXXX2=11035020000072029191?
;5XXXXXXXXXXXXXX2=11035020000018129281?
;5XXXXXXXXXXXXXX2=11035020000054629311?
;5XXXXXXXXXXXXXX2=11035020000099829771?
```

Identical PAN, identical expiry, identical service code, four different discretionary data fields — the CVC3 digits and the counter digits, changing on every read exactly as the bit maps promise. The same capture also shows the contactless interface reporting service code `502` where the physical stripe on the same card carried `101`, and substituting `SUPPLIED/NOT` for the cardholder name, both of which are ordinary personalisation practice and neither of which is a security control.

Visa's equivalent for the MSD path is dCVV, a three-digit dynamic value derived using the ATC and carried in the track 2 discretionary data. Issuer guidance circulated in the United States is explicit about the obligation: "always validate for Dynamic Card Verification Value (dCVV) on POS entry mode 91 transactions and decline if dCVV fails", and where the issuer relies on a scheme service to do the validation, "check the authentication result field and decline if authentication fails."

MSD is a legacy path and is being removed. In the United States and Canada, new point-of-sale terminals have been required since 18 October 2019 to support only EMV mode for contactless, excluding magnetic-stripe mode; Visa set an April 2019 deadline for terminals to retire MSD in favour of qVSDC. The stripe itself is following: Mastercard has said that newly issued credit and debit cards will not be required to carry a magnetic stripe from 2024 in most markets and from 2027 in the United States, that no new Mastercard credit or debit cards will be issued with a stripe from 2029, and that no such cards will have stripes at all from 2033. All of these dates are as of writing and are the kind of commitment that moves.

The naming table

Scheme	Stripe value	Printed value	Chip value	Contactless dynamic value
Visa	CVV	CVV2 (3 digits, reverse)	iCVV	dCVV
Mastercard	CVC 1	CVC 2 (3 digits, reverse)	Chip CVC	CVC3
American Express	CSC	CID (4 digits, front)	—	—
Discover	—	CID (3 digits, reverse)	—	—
JCB	CAV	CAV2 (3 digits, reverse)	—	—
China UnionPay	CVN	CVN2 (3 digits, reverse)	—	—

The dashes are not assertions that the scheme lacks the mechanism; they mark places where the scheme does not publish a distinct public name for it. Generic industry usage covers the printed value with **CSC**, card security code, and PCI documentation uses "card verification code or value" precisely to avoid choosing a dialect.

Why CVV2 must never be stored

PCI DSS divides account data into cardholder data and **sensitive authentication data**. The PCI Security Standards Council defines SAD as "security-related information used to authenticate cardholders and/or authorize payment card transactions", and states that this "includes, but is not limited to, card verification codes, full track data (from magnetic stripe or equivalent on a chip), PINs, and PIN blocks."

The operative requirement in PCI DSS v4.0.1 is 3.3.1: "SAD is not retained after authorization, even if encrypted. All SAD is rendered unrecoverable upon completion of the authorization process." It is broken out into three sub-requirements covering the full contents of any track (3.3.1.1), the card verification code (3.3.1.2 — "The card verification code is not stored upon completion of the authorization process"), and the PIN and PIN block (3.3.1.3). Requirement 3.3.2 permits SAD to exist before authorisation completes only if "stored electronically prior to completion of authorization is encrypted using strong cryptography", and 3.3.3 carves out the only exception, for issuers and issuer processors, limiting any stored SAD "to that which is needed for a legitimate issuing business need" and requiring it to be encrypted with strong cryptography.

Read that carefully. Encryption is not a defence for retained CVV2. Tokenisation of the PAN is not a defence. There is no retention period, no "we keep it for seven days for chargebacks", no exception for a merchant with a good reason. There is no good reason, because the value's only function is to be presented once at the moment of authorisation and its only property of interest is that it is not held anywhere in bulk.

The requirement bites in places that surprise engineers who have only ever thought about the primary database:

- **Application logs.** A payment request logged at debug level with its full body is a PCI DSS 3.3.1.2 violation, and it is by an enormous margin the most common one.
- **Crash dumps and exception reports.** A framework that serialises the request object into an error tracker has just posted the code to a third party.
- **HTTP access logs and proxies.** A card form submitted by GET, or a badly built API that accepts the code as a query parameter, writes it into every log on the path.
- **Message queues, replay buffers and event streams.** Anything with a retention setting is storage.
- **Call recordings.** Telephone orders where the cardholder reads the code aloud require pause-and-resume recording or DTMF suppression; the recording is storage.
- **Paper.** PCI's own guidance addresses this directly, noting that where "card verification codes are stored on paper media prior to completion of authorization, a method of erasing or covering the codes should prevent them from being read after authorization is complete" — literally, cutting the code out with scissors or blacking it out with an opaque permanent marker.
- **Backups and analytics warehouses.** A nightly extract that includes the order payload propagates the violation into systems nobody has scoped.

There is one more consequence that catches product teams. Because the code cannot be retained, **recurring and merchant-initiated payments cannot resupply it.** A subscription billed monthly presents the code once, on the first transaction, and never again. This is why Mastercard's chargeback rules for recurring transactions let the merchant point back at the original authorisation and the `M` result stored against it, and why any product design that assumes "we will just re-verify the CVV each month" is not implementable. Store the result. Never the value.

Failure modes, and what each one is telling you

When these values are wrong in production, the symptom is usually a decline with an unhelpful response code, and the diagnosis lies in matching the interface to the value. This table is the one worth keeping.

Symptom	Likely cause
Chip transactions decline; swipes of the same card work	Terminal or gateway is reporting a chip read with a magnetic-stripe entry mode, so the issuer computes CVV1 against an iCVV
Contactless declines while contact chip works on the same card	Reader is in the wrong operating mode, or the track image was reassembled after the CVC3 digits were substituted
All chip transactions from one terminal model fail at one issuer	The terminal is normalising or re-padding tag 57, destroying the discretionary data that carries the iCVV
CVV2 result `U` on every transaction to a particular issuer	That issuer has not supplied verification keys to the scheme; the check is not being performed at all, by anyone
CVV2 result `S`	The merchant's message said the code was absent or illegible; the issuer's record says the card has one
A cloned stripe works when it should not	The issuer is not validating that a stripe-entry transaction carries the stripe's value, or is taking the service code from the incoming track data rather than its own record

Every one of those is a mismatch between an interface and a value that belongs to a different interface. That is the whole subject. Four values, four doors, and a system whose security depends on nobody confusing them — including, and especially, the people building it.

13.98 Common wrong ideas

Wrong: There is one security code on a card, the three digits on the back. **Right:** There are at least four values from the same family of algorithm — CVV1 in the stripe, CVV2 in ink, iCVV in the chip, dCVV or CVC3 manufactured per tap — and presenting one at another's interface is precisely the event the system is built to detect.

Wrong: The bank keeps your CVV2 in a column somewhere and can look it up. **Right:** Nothing is stored but the keys; each value is recomputed from scratch inside a hardware security module at authorisation time, which is why no issuer can read the printed code back to you and no screen anywhere displays it.

Wrong: iCVV is what stops a chip card being cloned. **Right:** iCVV stops chip-sourced static data being reused at the stripe door and nothing more; what makes a chip transaction impossible to counterfeit is the Application Cryptogram, computed with a key the chip will not disclose.

Wrong: CVV, CVC and CID are different mechanisms. **Right:** They are the same idea in different schemes' dialects — Visa says CVV, Mastercard says CVC, American Express and Discover say CID, JCB says CAV, China UnionPay says CVN — and a system handling several is being told the same thing twice in two languages.

Wrong: A CVV2 result of `N` means the transaction was declined. **Right:** The result is not the authorisation decision; an issuer may return an approval with a result of `N`, and whether to proceed is the merchant's own risk decision.

Wrong: The printed code may be kept encrypted for a few days to handle chargebacks. **Right:** PCI DSS 3.3.1 forbids retention after authorisation even if encrypted, with no retention period and no merchant exception; the defence in a recurring-transaction chargeback is the stored `M` result, never the value.

Wrong: Three digits is far too short to be worth anything. **Right:** The shortness is deliberate — guessing a code for a known card number takes about five hundred attempts on average — and the defence is the issuer's velocity controls and the scheme's rules on repeated declines, not the length.

Wrong: Because iCVV is computed with a service code input of `999`, the chip's track data must carry `999` in its service code field. **Right:** The `999` is fed into the calculation and is not a claim about what is encoded; Visa's warning about `000` and `999` appearing on counterfeit stripes only makes sense because the encoded field and the calculation input are separable things.

Wrong: A terminal may safely normalise or re-pad tag `57` before forwarding it. **Right:** The full unmodified Track 2 Equivalent Data including the issuer discretionary data must be transmitted, and stacks that parse tag `57` into fields and reassemble it destroy the iCVV, which is why every chip transaction from one terminal model can fail at one issuer.

Wrong: A stolen database of card numbers gives a thief everything needed for a card-not-present purchase. **Right:** The printed code is not in the card's machine-readable data anywhere, so it is not in the database and not on a skimmed stripe; the thief would have had to hold the card and turn it over.

13.99 Chapter summary in 20 lines

1. A modern payment card carries at least four values that all look like three unremarkable decimal digits, and they are not interchangeable.
2. They are conventionally CVV1 encoded in the magnetic stripe, CVV2 printed in ink, iCVV stored inside the chip, and dCVV or CVC3 manufactured fresh for every contactless tap.
3. All of them come out of the same algorithm in the same hardware security module, from the primary account number, the expiry date and a three-digit service code.
4. The service code is the only input that varies — the card's own code for CVV1, `000` for CVV2, `999` for iCVV — and changing it changes every digit of the answer with no pattern connecting the results.
5. Nothing is stored except the key pair, so each value is recomputed and compared at authorisation time, which is why an issuer can regenerate your printed code but cannot look it up.
6. CVV1 lives in the discretionary data of the encoded tracks, and exists to prove that whoever presented the card had seen a real stripe rather than merely learned a number from a receipt.
7. CVV2 is printed and never encoded anywhere in machine-readable form, which is its entire purpose: a stolen database or a skimmed stripe does not contain it.
8. iCVV lives in the discretionary data inside the chip's Track 2 Equivalent Data, tag `57`, and differs from the stripe's value so that data lifted from a chip fails when written onto a blank stripe.
9. That makes iCVV a cross-channel defence rather than a chip defence, and the popular explanation that it is what secures chip transactions is backwards.
10. What actually secures a chip transaction is the Application Cryptogram, a message authentication code over the amount, currency, date, terminal country, an unpredictable number and the card's own transaction counter, computed with a key that never leaves the silicon.

11. dCVV and Mastercard's CVC3 are dynamic: the chip mixes its ever-increasing counter with a number the reader has just invented and produces digits that have never existed before and will never be valid again.
12. In magnetic-stripe data mode the card hands the reader a track image with holes in it plus bit maps saying where to write the fresh digits, so everything downstream of the terminal sees an ordinary swipe.
13. A published capture of one contactless card read four times in a row shows an identical account number, expiry and service code with four different discretionary data fields, which is that mechanism made visible.
14. POS entry mode is load-bearing, because it tells the issuer which value to compute, and a terminal that reads the chip but reports a swipe will have chip data checked against the stripe's value and decline.
15. Terminals must transmit tag `57` unmodified, and software that trims, re-pads or normalises the discretionary data destroys the iCVV in the process — a real and recurring integration defect.
16. The mechanism is sound but its deployment is uneven, and in 2020 both Cyber R&D Labs and Gemini Advisory documented issuers who were not validating the stripe's value on stripe transactions, so chip-harvested iCVV worked on counterfeit stripes.
17. A subtler failure in the same family is taking the service code from the incoming track data rather than the issuer's own record, which lets a fraudster encode `999` on a counterfeit stripe and have the issuer helpfully compute the chip's value.
18. The printed code travels in scheme-specific fields — Visa Field 126 usage 10, Mastercard DE 48 subelement 92 — and returns a result of `M`, `N`, `P`, `S` or `U`, which is durable evidence and not an authorisation decision.
19. PCI DSS classes the printed code as sensitive authentication data and forbids retaining it after authorisation even encrypted, which reaches debug logs, crash dumps, access logs, queues, call recordings, paper and backups, and means a recurring payment can never resupply it.
20. Every production failure in this area has the same shape — an interface presented with a value belonging to a different interface — which is the whole subject of the chapter.

Sources: ISO/IEC 7813:2006 track structures; EMV 4.x Book 3 tag definitions (`57`, `5F30`, `5F34`, `9F1F`, `9F10`, `9F26`, `9F36`, `9F37`); EMVCo EMV Contactless Specifications for Payment Systems, Book C-2 (Kernel 2) data objects `56`, `9F60`–`9F67`, `9F6A`–`9F6C`; PCI SSC glossary and PCI DSS v4.0.1 requirements 3.3.1–3.3.3; Mastercard Switch Rules and Security Rules and Procedures (DE 48 subelements 87 and 92); Visa VisaNet authorisation technical specifications (Fields 44.10 and 126.10) and Visa EMV Chip News, October 2019; IBM cryptographic services documentation on the Visa CVV algorithm; Visa and ICBA issuer guidance on POS entry modes and dCVV validation; Gemini Advisory / Recorded Future, "Cybercriminals Deploy EMV-Bypass Cloning" (2020); Cyber R&D Labs card cloning research as reported by Krebs on Security, July 2020; Mastercard magnetic stripe retirement announcement, 2021; Secure Technology Alliance, "Card Payments Roadmap in the United States"; published contactless card capture data (Ambimat Electronics service code analysis).

14

CHAPTER

The Chip Is a Computer

14.0 What this chapter gives you

1. You will be able to state precisely what separates a chip from a magnetic stripe: a stripe is a recording and a chip is a correspondent, so a stripe reader takes and a chip reader asks.
2. You will be able to read a terminal log and name each exchange in it, from application selection through GET PROCESSING OPTIONS and READ RECORD to GENERATE APPLICATION CRYPTOGRAM.
3. You will be able to decode a command by hand from its class, instruction and parameter bytes, and explain why the trailing `00` on almost every EMV command means "send me everything you have" rather than padding or a null argument.
4. You will be able to decode an Application File Locator four bytes at a time, recover the short file identifier from the top five bits, and build the matching READ RECORD parameter byte from it.
5. You will be able to explain why records with short file identifiers 1 to 10 have their template tag and length stripped before they feed the authentication hash while records from 11 to 30 go in whole, and why that defect reaches production so often.
6. You will be able to say what static, dynamic and combined data authentication each prove and each fail to prove — that the data was signed, that the card is present, and that the card, the decision and the transaction are one indivisible statement.
7. You will be able to fill in a Processing Options Data Object List correctly, applying padding and truncation rules that run one way for numeric formats and the other way for everything else.
8. You will be able to look at a GENERATE AC exchange in which the terminal asked for an offline approval and the card returned an online request, and explain why the card may always answer weaker than it was asked but never stronger.
9. You will be able to explain why reading a chip is not an attack, and why any reader obtaining the account number, expiry, name and usage rules from any card is working exactly as designed.
10. You will be able to say why "the chip killed cloning" is true of one attack and one only, and where the fraud went instead.

Hold a card with the contact plate facing you and you are looking at eight gold pads, five of which do anything. Behind them, embedded in a cavity milled into the plastic, is a die perhaps a couple of millimetres square containing a processor, some read-only memory holding a program, some non-volatile memory holding data and keys, and — on any card issued this century — hardware dedicated to arithmetic on numbers hundreds of digits long.

This is the fact most explanations of card payments skip. The magnetic stripe is a *recording*. The chip is a *correspondent*. A stripe reader takes; a chip reader asks. Everything that follows in this volume — the cryptogram, the decision to go online, the cardholder verification list, the contactless tap — follows

from that one structural difference. It is also why counterfeit card fraud collapsed, and not because the chip is hard to read: reading a chip is trivial, and any developer with a fifteen-pound reader can pull a card number and expiry date off one in under a second. The chip works because there is a category of question it answers differently every time, and a category of secret it will never disclose however the question is phrased.

This chapter takes that conversation apart, command by command. By the end you should be able to read a terminal log, name each exchange, decode an Application File Locator by hand, and say precisely what static, dynamic and combined data authentication each prove and each fail to prove.

The plain version

Imagine two ways of proving who you are at a door.

The first way: you hand over a photocopy of your passport. The doorman reads it, sees your name and your photograph, and lets you in. It works. It is also completely useless the moment somebody else gets hold of a photocopier, because a copy of a copy is indistinguishable from the original. That is the magnetic stripe. Everything on it is written down. Anyone who can read it can write it out again, and the new one works exactly as well as the old one.

The second way: there is no photocopy. There is a small clerk in a locked room behind a letterbox. You cannot see him and you cannot go in. You slide questions through on slips of paper and he slides answers back. He is helpful — he will tell you the account number, the expiry date, the name — but he will not, under any circumstances, tell you the secret he uses to sign his answers. And crucially, some of the questions you may ask are ones where the *answer depends on the question*. If you ask "what is 8,391 plus your secret?" and he replies with a number, that number is worthless to anybody who asks him something different tomorrow.

That is the chip.

The conversation, one line at a time

Here is a real coffee. Four pounds twenty, a Tuesday morning, a card inserted into a countertop terminal. What follows is not a metaphor; it is the actual sequence, with the technical names given in brackets so that you can find them again in the second half of the chapter.

"What can you do?" The terminal has no idea what is in front of it. It might be a Visa debit card, a Mastercard credit card, a German girocard, or a transport card that is not a payment card at all. So it asks the chip for its directory of programs. The chip replies with a short list — say two entries, "Visa Credit" and "Visa Debit", each with a priority number — and the terminal picks the highest-priority one it also supports. (*Application selection, using the payment system directory.*)

"Right. I want to talk to Visa Debit." The terminal opens that program. The chip replies with a greeting that includes something surprising: a shopping list. It says, in effect, "before I will start a transaction, tell me nine things, in this exact order and at these exact sizes." (*SELECT of the application; the card returns a Processing Options Data Object List.*)

"Here is your list." The terminal fills it in: the amount is 000000000420, the country is 0826, the currency is 0826, today's date, the type of transaction, a five-byte block of its own findings so far, and — this one matters — a fresh four-byte random number it has just generated and has never used before. (*GET PROCESSING OPTIONS.*)

The chip now knows a transaction has begun. It replies with two things: a two-byte summary of what it is capable of, and a *map* listing which of its internal files the terminal should read, which records in

each, and how many of those records form the material to be authenticated. (*Application Interchange Profile and Application File Locator.*)

"Show me those files." Out come the card number, the expiry date, the cardholder name, the country of issue, rules about what the card may be used for, and a bundle of certificates. (*READ RECORD.*)

"Prove you are not a photocopy." The terminal has a card scheme's public key burned into it. The certificates chain back to that key: the scheme signed a certificate for the issuing bank, and the bank signed one for this chip. The terminal walks the chain, then throws the chip a challenge — a number it has just invented — and asks it to sign. Only this chip can produce the right signature, because only this chip holds the matching private key, and that key has never left the silicon.

"Here is the transaction. What is your verdict?" The terminal hands over the amount, the currency, the date, its own risk findings and another unpredictable number. The chip runs its own checks — how many transactions it has approved offline since it last spoke to its bank, whether the amount exceeds its limit — and produces a cryptogram computed from the transaction with a secret key it shares with the issuing bank. It also says what that cryptogram means: approved offline, declined offline, or *ask the bank*. (*GENERATE APPLICATION CRYPTOGRAM.*)

For a four-pound-twenty coffee most cards say "ask the bank", and the cryptogram goes off into the network. That journey is Volume III; what the cryptogram *is* and what key computes it is chapter 15.

The one thing the clerk will never do

Every command listed above is a question. There is no command in the whole specification that says "give me your private key", and none that says "give me the secret you share with your bank". The chip will sign things with those keys all day long. It will not disclose them.

That is not a policy, and not a setting an issuer can get wrong. No instruction in the instruction set would cause the chip to output them, and the program sits in read-only memory laid down at manufacture. To get the keys you would have to attack the silicon: decapsulate the die and either probe it or infer the key from variations in the power it draws while computing. Those attacks exist, and they are why chips are certified before a scheme will let them be issued. But they cost money, take equipment and destroy the card. Photocopiers do not.

Why the counter matters

One more detail worth carrying to dinner. The chip holds a counter that goes up by one every time a transaction starts and never goes down. It is two bytes wide, so it counts to 65,535, and every cryptogram includes the current value.

This is the quiet killer of replay. Suppose somebody records a complete, valid conversation — every question, every answer, cryptogram included — and plays it back tomorrow. The recorded cryptogram carries counter value 137, but the real card has been used eleven more times since, so the bank has seen 148. A counter that repeats or goes backwards is a card being replayed, and the issuer can see it in the message. The bank does not have to guess whether the transaction is genuine. It can check.

Where the plain version stops being true

The chip is not a computer in the sense you are picturing

It has no clock, no battery and no power of its own. It is parasitic on the terminal, which supplies voltage on one contact and a clock on another; the chip computes only while both are present. Pull the card out mid-transaction and it does not "notice" — it simply stops existing until somebody powers it

again. It has no idea whether the last transaction was four seconds ago or four years ago, it must ask the terminal for the date, and it cannot tell whether the terminal is lying.

Nor can the chip start anything. It has no way to say "excuse me" through the letterbox. It receives a command, computes, returns a response, and waits. This is why "the card decides whether to approve the transaction" is a useful but misleading simplification: the terminal decides what question to ask, and the card can only choose among the answers to *that* question — and, as we will see, it may return an answer weaker than the one requested but never a stronger one.

Offline authentication proves the card is genuine, not that the terminal is

The certificate chain runs one way. The chip proves itself to the terminal; nothing in the ordinary transaction flow requires the terminal to prove anything to the chip, and there is no practical way it could, because a terminal is a commodity device in a shop and the chip cannot check a revocation list or even know today's date.

The consequence surprises people: reading a chip is not an attack. Any reader may power any card, run selection, GET PROCESSING OPTIONS and READ RECORD, and obtain the primary account number, the expiry date, the cardholder name and the card's usage rules. Contactless cards will do this over the air. That is by design — those are the fields a merchant needs. The security does not come from hiding them. It comes from the fact that having them does not let you produce a cryptogram.

"The chip killed cloning" is true of one attack and one only

It killed the counterfeiting of magnetic stripes for use at chip-reading terminals. It did nothing about card-not-present fraud, and the fraud did not stop; it moved. UK Finance's twenty-year retrospective, published on 14 February 2026, records a 95 per cent reduction in losses to counterfeit card fraud over the two decades since Chip and PIN replaced signature — and in the same breath records £215 million lost to remote purchase fraud in the first half of 2025 alone.

There is a second qualification. For years a great many cards described as "chip cards" performed only *static* data authentication, which is a photocopy again — a photocopy with a genuine signature on it. The signature verifies; it is simply not fresh, because the signed material is identical on every transaction, and a device that has recorded it once can present it again. Whether a given card does this is an issuer decision, and the later sections explain exactly what the difference costs.

There is not one conversation; there are several specifications

The dialogue above is the contact interface, governed by the four EMV books. Contactless runs a related but distinct process, with its own entry-point specification and a family of kernel specifications, one per scheme, each with its own command sequence, data elements and decision logic. A Visa tap and a Mastercard tap do not execute the same steps. Treating "EMV" as a single protocol is the commonest error in engineering documentation about cards.

The technical version

What EMV actually is, and what it sits on

EMV is not a standard in its own right. It is a profile of the ISO/IEC 7816 series for integrated circuit cards. The current release, version 4.4, dated October 2022, comprises four books:

Book	Title
1	Application Independent ICC to Terminal Interface Requirements

Book	Title
2	Security and Key Management
3	Application Specification
4	Cardholder, Attendant, and Acquirer Interface Requirements

Book 3 section 1.3 states the relationship precisely: "This specification is based on the ISO/IEC 7816 series of standards and should be read in conjunction with those standards. However, if any of the provisions or definitions in this specification differ from those standards, the provisions herein shall take precedence." Version 4.4 also incorporates two consequential Specification Bulletins: no. 175, on Application Selection Registered Proprietary Data, and no. 243, which introduced Extended Data Authentication and Offline Data Encipherment. The Level 1 electrical and transport requirements now live in a separate document, the EMV Contact Interface Specification, cited by Book 3 as a normative reference.

Chapter 11 covered the physical layer: the ISO/IEC 7816-2 contact assignment, the 5 V supply, the 1 to 5 MHz clock, the answer to reset, and the protocols T=0 and T=1. This chapter starts one layer up.

The APDU: the shape of every question and every answer

Book 3 Figure 1 gives the command APDU as a mandatory four-byte header followed by a conditional body:

```
CLA  INS  P1  P2  [Lc]  [Data]  [Le]
```

and Figure 2 gives the response as a conditional body followed by a mandatory two-byte trailer:

```
[Data]  SW1  SW2
```

`Lc` is the number of bytes in the command data field; `Le` is the maximum expected back. Book 3 is unambiguous about one convention that trips up newcomers: "When Le is present and contains the value zero, the maximum number of available data bytes (up to 256) is expected. When required in a command message, Le shall always be set to '00'." The trailing `00` on almost every EMV command is therefore neither padding nor a null argument. It means "send me everything you have".

ISO/IEC 7816-4 defines four cases by whether data flows in each direction — case 1 neither, case 2 response only, case 3 command only, case 4 both — and EMV inherits them. Under T=0 they do not map cleanly onto single exchanges, which is why logs contain apparent duplicates: a card may answer `61 xx`, meaning "executed, xx bytes waiting, fetch them with GET RESPONSE", or `6C xx`, meaning "your Le was wrong, the correct value is xx". Book 1 Annex A works through all four cases and both procedure bytes. A library that ignores `6C xx` passes every test in the lab and fails intermittently in the field.

Book 3 Table 2 assigns the most significant nibble of the class byte: `'0'` is an inter-industry command defined by ISO/IEC 7816, `'8'` is a command proprietary to EMV, and any other value is out of scope. The least significant nibble carries secure messaging and logical channel indications, again per 7816-4. So `00 A4` is ISO's SELECT used as EMV requires; `80 A8` is an EMV-only command no general-purpose smart card would recognise.

Book 3 Table 3 gives the complete instruction set for financial transaction:

CLA	INS	Command
8x	1E	APPLICATION BLOCK

CLA	INS	Command
8x	18	APPLICATION UNBLOCK
8x	16	CARD BLOCK
0x	82	EXTERNAL AUTHENTICATE
8x	AE	GENERATE APPLICATION CRYPTOGRAM
0x	84	GET CHALLENGE
8x	CA	GET DATA
8x	A8	GET PROCESSING OPTIONS
0x	88	INTERNAL AUTHENTICATE
8x	24	PIN CHANGE/UNBLOCK
0x	B2	READ RECORD
0x	A4	SELECT
0x	20	VERIFY

That is the entire vocabulary: thirteen commands, of which a normal purchase uses five or six. The status bytes are the other half of the protocol. Those worth memorising:

SW1 SW2	Meaning
9000	Command successfully executed
61 xx	Executed; xx bytes available via GET RESPONSE
6C xx	Wrong Le; xx is the correct value
63 Cx	Verification failed; x tries remaining
6985	Conditions of use not satisfied
6A81	Function not supported
6A82	File not found
6A83	Record not found
6A88	Referenced data (data objects) not found

6A82 is not an error in the ordinary sense: it is how a card says "I do not have that application", and it is the expected response during selection.

The Application Identifier

An application is addressed by name, and the name is an Application Identifier. Its structure comes from ISO/IEC 7816-5: a Registered Application Provider Identifier of exactly five bytes, allocated by a registration authority, optionally followed by a Proprietary Application Identifier Extension of up to eleven bytes chosen freely by the provider. The AID is therefore 5 to 16 bytes long. On the card it is tag 4F, or the Dedicated File name, tag 84; in the terminal's configuration it is tag 9F06.

The RID identifies the scheme; everything after it is the scheme's own business — product, region, common debit variant. A working set:

AID	Application
A0000000031010	Visa Debit/Credit (Classic)
A0000000032020	V PAY
A0000000041010	Mastercard Credit/Debit (Global)
A0000000043060	Maestro
A0000000250000	American Express

AID	Application
A00000006510	JCB
A0000001523010	Discover / PULSE D-PAS
A0000002771010	Interac
A000000333010101	UnionPay Debit
D27600002545500100	ZKA girocard

Note the leading conventions — A0 for internationally registered providers, D2 for nationally registered ones — and note that the lengths differ. A terminal that assumes seven bytes fails on UnionPay and girocard, and that is a real and recurring defect.

Application selection

Selection begins with a directory. On contact the entry point is the Payment System Environment, a Directory Definition File named with the fourteen ASCII characters 1PAY.SYS.DDF01 — on the wire:

```
31 50 41 59 2E 53 59 53 2E 44 44 46 30 31
```

On contactless it is the Proximity Payment System Environment, 2PAY.SYS.DDF01, differing by one byte:

```
32 50 41 59 2E 53 59 53 2E 44 44 46 30 31
```

The terminal selects it by name per Book 1 Table 40: P1, the reference control parameter (Table 41), is '04' for selection by DF name, and P2, the selection options parameter (Table 42), is '00' for the first or only occurrence — '02' requests the next occurrence, which is how a terminal enumerates multiple applications sharing a partial AID.

A real trace, from a published analysis of a Visa contactless transaction:

```
C: 00 A4 04 00 0E 325041592E5359532E4444463031 00
R: 6F 37
     84 0E 325041592E5359532E4444463031
     A5 25
       BF0C 22
         61 20
           4F 07 A0000000031010
           50 0B 566973612043726564697 4
           87 01 01
           BF63 04 DF20 01 80
   9000
```

Read that structure carefully; it recurs everywhere. 6F is the File Control Information template; 84 the Dedicated File name, echoing what was selected; A5 the FCI Proprietary Template; BF0C the FCI Issuer Discretionary Data, where Book 3 section 5.1 requires any additional data element in a SELECT response to sit. 61 is a Directory Entry, holding 4F the ADF name, 50 the Application Label (566973612043726564697 4 is ASCII "Visa Credit") and 87 the Application Priority Indicator.

On the contact interface the PSE works slightly differently: the FCI Proprietary Template contains tag 88, the short file identifier of the directory elementary file, and the terminal issues READ RECORD against that SFI to walk the directory. Book 1 Table 46 gives the Payment System Directory Record Format and Table 47 the ADF Directory Entry Format.

If the PSE is absent the card returns 6A82 and the terminal falls back to trying each AID in its own list, one SELECT at a time, per Book 1 section 12.3.3. Against each it holds an Application Selection Indicator saying whether the card's AID must match exactly, including length, or only up to the length of the

terminal's — which is how a terminal holding the seven-byte Visa AID matches a card personalised with a longer, product-specific one. Book 1 section 12.4 governs final selection from the candidate list using the Application Priority Indicator, with cardholder confirmation where required.

Selecting the application itself returns a richer FCI:

```
C: 00 A4 04 00 07 A0000000031010 00
R: 6F 46
     84 07 A0000000031010
     A5 3B
       50 0B 5669736120437265646974
       87 01 01
       9F38 18 9F6604 9F0206 9F0306 9F1A02 9505
               5F2A02 9A03 9C01 9F3704
       5F2D 02 656E
       BF0C 08 9F5A 05 ...
   9000
```

`5F2D` is the Language Preference, `656E` being ASCII "en", `9F5A` the Application Program Identifier, and `9F38` the Processing Options Data Object List — the shopping list from the plain version, made concrete.

Data Object Lists: the card programs the terminal

A DOL is a concatenation of two-part entries, each a one- or two-byte tag followed by a one-byte length. It is *not* TLV data; it is a template telling the terminal what to concatenate, in what order, at what size. Book 3 section 5.4 explains why: to minimise processing in the card the constructed field carries no tags, so the card must know the layout in advance.

The PDOL above decodes to:

Tag	Length	Data element
`9F66`	4	Terminal Transaction Qualifiers
`9F02`	6	Amount, Authorised (Numeric)
`9F03`	6	Amount, Other (Numeric)
`9F1A`	2	Terminal Country Code
`95`	5	Terminal Verification Results
`5F2A`	2	Transaction Currency Code
`9A`	3	Transaction Date
`9C`	1	Transaction Type
`9F37`	4	Unpredictable Number

Thirty-three bytes. Book 3 section 5.4 also gives the rules every implementation gets wrong at least once:

If a tag in the DOL is unknown or names a constructed object, the terminal supplies that many hexadecimal zero bytes; likewise if the object is known but absent or inapplicable. If the DOL length is *shorter* than the actual data, the terminal truncates the leftmost bytes for numeric (`n`) format and the rightmost for everything else. If *longer*, it pads with leading zeroes for numeric, trailing `'FF's` for compressed numeric, trailing zeroes otherwise. Numeric goes one way and everything else the other; reversing that produces a cryptogram that fails at the issuer with no useful diagnostic.

The same mechanism appears three more times: CDOL1 (tag `8C`) and CDOL2 (tag `8D`) for GENERATE AC, the Transaction Certificate Data Object List for TC hashing, and the Dynamic Data Authentication Data Object List (tag `9F49`) for INTERNAL AUTHENTICATE.

GET PROCESSING OPTIONS

Book 3 section 6.5.8, Table 18:

Code	Value
CLA	'80'
INS	'A8'
P1	'00'; all other values are RFU
P2	'00'; all other values are RFU
Lc	var.
Data	PDOL related data
Le	'00'

The PDOL data is wrapped in a command template with tag '83'; if the card published no PDOL, the terminal sends the empty template 83 00. For a £4.20 purchase in the United Kingdom on 17 August 2026, filling in the PDOL above:

```
80 A8 00 00 23
   83 21
      36C04000        Terminal Transaction Qualifiers
      000000000420    Amount, Authorised          (£4.20)
      000000000000    Amount, Other
      0826            Terminal Country Code       (GB)
      0000000000      Terminal Verification Results
      0826            Transaction Currency Code   (GBP)
      260817          Transaction Date            (YYMMDD)
      00              Transaction Type            (purchase)
      7A19C4B2        Unpredictable Number
   00
```

Lc is '23', thirty-five bytes: two for the tag and length of the '83' template plus thirty-three of data. The amount is numeric format, two digits to the byte, right-justified with leading zeroes — 420 minor units, not 4.20. Chapter 9 explained why no floating point number ever touches this field.

The response comes back in one of two formats. Format 1 is a primitive object with tag '80' whose value is the Application Interchange Profile followed immediately by the Application File Locator, with no internal tags — you must know the first two bytes are the AIP and the rest the AFL. Format 2 is a constructed object with tag '77' containing ordinary TLV, AIP as '82' and AFL as '94'. Both are legal and both are deployed. From the same real trace:

```
R: 77 52
     82 02 0000                  Application Interchange Profile
     94 04 10040400              Application File Locator
     57 13 ...                   Track 2 Equivalent Data (PAN masked in source)
     5F20 02 202F                Cardholder Name
     5F34 01 00                  PAN Sequence Number
     9F10 07 06010A03A00000      Issuer Application Data
     9F26 08 1EB2BF209002EFD3    Application Cryptogram
     9F27 01 80                  Cryptogram Information Data (ARQC)
     9F36 02 0089                Application Transaction Counter (137)
     9F6C 02 0280                Card Transaction Qualifiers
     9F6E 04 20700000            Form Factor Indicator
   9000
```

That is a Visa contactless card, and it makes the earlier point concrete: this is not the contact flow. The card returned the cryptogram inside the GET PROCESSING OPTIONS response, before any GEN-

ERATE AC exists, because Visa's contactless kernel compresses the exchange to keep the tap under a few hundred milliseconds. Note too that the AIP is `'0000'`; the cardholder-verification signalling here rides in the Card Transaction Qualifiers, tag `9F6C` — precisely the field the published PIN-bypass attack against this kernel manipulates.

The Application Interchange Profile

Tag `82`, two bytes. Byte 1:

Bit	Meaning
b8	RFU
b7	SDA supported
b6	DDA supported
b5	Cardholder verification is supported
b4	Terminal risk management is to be performed
b3	Issuer authentication is supported
b2	On-device cardholder verification supported
b1	CDA supported

Byte 2 carries, among reserved bits, b8 for EMV mode and b1 for Relay Resistance Protocol support. Book 3 Annex C1, Table 41, is normative.

The Application File Locator

Tag `94`, always a multiple of four bytes, read four at a time:

Byte	Meaning
1	Short File Identifier in the five most significant bits; three least significant bits zero
2	First record to read
3	Last record to read
4	Number of records, starting at the first, included in offline data authentication

The AFL in the trace is `10 04 04 00`: one group. Byte 1 is `0x10`, `00010000` in binary; shift right three and you get `00010`, SFI 2. Records 4 to 4, so a single record. Zero records feed offline data authentication.

READ RECORD

Book 1 section 11.2 and Book 3 section 6.5.11. The command is `00 B2`, with P1 the record number and P2 the reference control parameter, in which the five most significant bits are the SFI and the three least significant are `100` — that is, `P2 = (SFI << 3) | 4`. For SFI 2, record 4:

```
C: 00 B2 04 14 00
R: 70 1A
     5A 08 ...               PAN (masked in source)
     5F28 02 ...             Issuer Country Code
     5F24 03 220731          Application Expiration Date
     9F07 02 C280            Application Usage Control
   9000
```

`0x14` is `00010100`: SFI 2, control bits `100`. The response is always wrapped in the READ RECORD Response Message Template, tag `70`. Short file identifiers are allocated by Book 3 Table 1:

SFI range	Meaning
1 to 10	Governed by the EMV specification
11 to 20	Payment system-specific
21 to 30	Issuer-specific

That boundary is not cosmetic; it is the most commonly mis-implemented rule in offline data authentication. Files with SFI 1 to 10 hold only data the card does not itself interpret, and when their records feed the authentication hash, *the tag `70` and its length byte are stripped* and only the value is used. For SFI 11 to 30 the *entire* record goes in, tag and length included. Get that wrong and every card with issuer-specific files fails offline authentication while every card inside the standard range passes — a defect that reaches production regularly, because test cards tend to use low SFIs.

Note also that a card may answer READ RECORD with `6C xx` if the terminal's Le was wrong, and the terminal must reissue with `Le = xx`.

The rest of the vocabulary

GET DATA (`80 CA`, P1 P2 carrying the tag of the wanted object, Book 3 section 6.5.7) retrieves single primitive objects living outside the file structure. Book 3 Table 32 lists them: the Application Transaction Counter (`9F36`), Last Online ATC Register (`9F13`), PIN Try Counter (`9F17`) and Log Format (`9F4F`).

GET CHALLENGE (`00 84 00 00 00`) asks the card for an eight-byte unpredictable number of its own, used when the card's CDOL demands an ICC Dynamic Number.

VERIFY (`00 20 00` P2) submits an offline PIN. Book 3 Table 24 codes P2 as the qualifier of reference data: `'80'` plaintext, `'88'` enciphered, with Table 25 giving the plaintext block format. Failure returns `63 Cx`, `x` being the tries left; at zero the application blocks. Chapter 17 covers cardholder verification in full.

INTERNAL AUTHENTICATE (`00 88 00 00` Lc Data `00`) submits authentication-related data — in practice the terminal's unpredictable number, laid out according to the DDOL — and returns the card's signature over it.

GENERATE APPLICATION CRYPTOGRAM (`80 AE` P1 `00` Lc Data `00`) is the decision command. P1, the reference control parameter, is coded by Book 3 Table 12:

b8 b7	Cryptogram requested
`0 0`	AAC — Application Authentication Cryptogram (decline)
`0 1`	TC — Transaction Certificate (offline approval)
`1 0`	ARQC — Authorisation Request Cryptogram (go online)
`1 1`	RFU

with b5 set to request a CDA signature and the rest RFU. So `P1 = '00'` asks the card to decline, `'40'` to approve offline, `'80'` to go online, and `'90'` to go online with a combined signature.

The response comes in the same two flavours as GET PROCESSING OPTIONS. Format 1, tag `'80'`, is Cryptogram Information Data, Application Transaction Counter, Application Cryptogram and op-

tionally Issuer Application Data, concatenated without tags. Format 2, tag `'77'`, is TLV, with `9F27`, `9F36` and `9F26` mandatory and `9F10` optional. The Cryptogram Information Data is coded by Book 3 Table 15:

b8 b7	Cryptogram returned
0 0	AAC
0 1	TC
1 0	ARQC
1 1	AAR — Application Authorisation Referral

with b6 and b5 for payment system-specific cryptograms, b4 indicating whether an advice message is required, and b3 to b1 a reason code: `000` no information given, `001` service not allowed, `010` PIN Try Limit exceeded, `011` issuer authentication failed.

A complete GENERATE AC exchange from a published contact-interface trace, the card's CDOL1 being `9F0206 9F0306 9505 5F2A02 9A03 9C01 9F3704 9F4C08 9F4502`:

```
C: 00 84 00 00 00
R: 8D51F46C9F405F71 9000            GET CHALLENGE returns ICC Dynamic Number

C: 80 AE 40 00 25
     000000000001          Amount, Authorised
     000000000000          Amount, Other
     0000000000            Terminal Verification Results
     0978                  Transaction Currency Code (EUR)
     090730                Transaction Date
     21                    Transaction Type
     2C764F65              Unpredictable Number
     8D51F46C9F405F71      ICC Dynamic Number
     D179                  Data Authentication Code
   00
R: 77 1E
     9F27 01 80                    ARQC
     9F36 02 0213                  ATC = 531
     9F26 08 2DF3833C61855BEA      Application Cryptogram
     9F10 07 06842300310208        Issuer Application Data
   9000
```

Read the two decisive values together. The terminal set `P1 = '40'`, asking for a Transaction Certificate — offline approval. The card returned Cryptogram Information Data `'80'`, an ARQC. The card overrode the terminal and forced the transaction online. That is card action analysis in one exchange: the terminal proposes, the card disposes, and the card may always return something weaker than requested but never something stronger.

Finally, `APPLICATION BLOCK`, `APPLICATION UNBLOCK`, `CARD BLOCK` and `PIN CHANGE/UNBLOCK` are the issuer script commands. The terminal never issues them on its own initiative; they arrive from the issuer inside the authorisation response and the terminal relays them. This is how a bank resets a PIN try counter on a card sitting in a wallet three thousand miles away.

Offline data authentication

Everything above establishes what the card says. Offline data authentication establishes whether to believe it, without going online.

The certificate chain

The terminal is loaded with the scheme's Certification Authority public keys, one or more per RID, each identified by a Certification Authority Public Key Index which the card supplies in tag `8F`. The terminal takes the first five bytes of the AID as the RID and, with the RID and the index, looks up the right CA public key in its own table.

The card carries an Issuer Public Key Certificate (tag `90`), an Issuer Public Key Remainder (tag `92`) where the key is too long to fit inside it, and an Issuer Public Key Exponent (tag `9F32`). The certificate is recovered with the CA public key using the message-recovery signature scheme of ISO/IEC 9796-2, and out comes a structure beginning `'6A'` and ending `'BC'`:

Field	Length (bytes)	Value
Recovered Data Header	1	`'6A'`
Certificate Format	1	`'02'`
Issuer Identifier	4	Leftmost 3–8 PAN digits, right-padded with `'F'`
Certificate Expiration Date	2	MMYY
Certificate Serial Number	3	Assigned by the CA
Hash Algorithm Indicator	1	
Issuer Public Key Algorithm Indicator	1	
Issuer Public Key Length	1	
Issuer Public Key Exponent Length	1	
Issuer Public Key or leftmost digits	NCA − 36	Right-padded with `'BB'` if short
Hash Result	20	
Recovered Data Trailer	1	`'BC'`

The terminal checks the header, format byte and trailer, recomputes and compares the hash over the recovered fields concatenated with the remainder and the exponent, checks the issuer identifier against the leading PAN digits, and checks the expiry. Only then does it hold a trusted issuer public key.

The key sizes are bounded. EMV Book 2 states that terminals "shall support keys up to 1984 bits (248 bytes) in length", and its Annex D1 sets the hierarchy: the Issuer Public Key length may be equal to or less than the CA Public Key length up to that 248-byte maximum, and the ICC Public Key and ICC PIN Encipherment Public Key lengths may be equal to or less than the Issuer Public Key length. Keys shrink as you go down the chain; they never grow.

Static Data Authentication

SDA uses the issuer key only. The card carries Signed Static Application Data in tag `93`, produced at personalisation by signing a hash of the card's static data with the issuer's private key. Recovery with the issuer public key yields:

Field	Length (bytes)	Value
Recovered Data Header	1	`'6A'`
Signed Data Format	1	`'03'`
Hash Algorithm Indicator	1	

Field	Length (bytes)	Value
Data Authentication Code	2	Issuer-assigned
Pad Pattern	NI - 26	`'BB'` repeated
Hash Result	20	
Recovered Data Trailer	1	`'BC'`

The authentication input is the concatenation of the records the AFL marked for authentication, assembled with the SFI rule described earlier, followed — where a Static Data Authentication Tag List (tag `9F4A`) is present — by the values of the objects it names. On success the terminal extracts the two-byte Data Authentication Code and stores it as tag `9F45`, which is why `9F45` appears in CDOL1: the card wants proof that the terminal really performed SDA.

What SDA proves: this card's static data was signed by an issuer whose certificate chains to the scheme. What it does not prove: that the card in the slot is the card the data came from. Every byte involved is fixed at personalisation; record it once and you can present it again on any device that speaks the protocol. SDA defeats a forger who alters the data. It does not defeat one who copies it wholesale.

Dynamic Data Authentication

DDA adds a key pair belonging to the individual chip. The card carries an ICC Public Key Certificate (tag `9F46`) signed by the issuer, plus the ICC Public Key Exponent (`9F47`) and Remainder (`9F48`). The recovered certificate has Certificate Format `'04'` and carries the full Application PAN in ten bytes, right-padded with `'F'`, rather than the issuer identifier of the `'02'` format.

Having verified the ICC public key, the terminal issues INTERNAL AUTHENTICATE carrying an unpredictable number of its own choosing. The card signs with its private key and returns Signed Dynamic Application Data, tag `9F4B`, which recovers to:

Field	Length (bytes)	Value
Recovered Data Header	1	`'6A'`
Signed Data Format	1	`'05'`
Hash Algorithm Indicator	1	
ICC Dynamic Data Length	1	LDD
ICC Dynamic Data	LDD	Generated by and/or stored in the ICC
Pad Pattern	NIC - LDD - 25	`'BB'` repeated
Hash Result	20	
Recovered Data Trailer	1	`'BC'`

The hash covers, among other things, the terminal's unpredictable number. That is the whole improvement: the signature is fresh, so a recording is useless, and producing it requires a private key that has never been transmitted. This is the point at which cloning stops being an economic proposition.

What standalone DDA does not do is bind the signature to the transaction. The card proves it is genuine; separately, it produces a cryptogram over the amount. A device sitting between card and terminal can let the authentication succeed and still interfere with the decision, because the two are not cryptographically joined.

Combined DDA/Application Cryptogram Generation

CDA joins them. Instead of a separate INTERNAL AUTHENTICATE, the terminal sets bit b5 of the GENERATE AC reference control parameter and the card returns the Signed Dynamic Application Data (`9F4B`) *in the GENERATE AC response*, alongside `9F27`, `9F36` and `9F10`. The ICC Dynamic Data inside that signature includes the Application Cryptogram itself and a hash over the transaction data, so one signature covers the card's identity, its decision and the transaction it applies to. Substituting any of the three invalidates all of them.

The terminal records the outcome in the Terminal Verification Results, tag `95`, five bytes, whose first byte is:

Bit	Meaning
b8	Offline data authentication was not performed
b7	SDA failed
b6	ICC data missing
b5	Card appears on terminal exception file
b4	DDA failed
b3	CDA failed
b2, b1	RFU

Those bits are then matched against the Issuer Action Codes and Terminal Action Codes to decide whether to approve, decline or go online. That is chapter 16.

Extended Data Authentication

Version 4.4 adds a fourth option. Specification Bulletin no. 243 introduced Extended Data Authentication and Offline Data Encipherment, built on elliptic curve cryptography rather than RSA: Book 3's abbreviation list now carries `XDA`, `ECC` and `EC-SDSA` for the Elliptic Curve Schnorr Digital Signature Algorithm, and Table 31 gives the data required for XDA alongside Table 29 for SDA and Table 30 for DDA and CDA.

The motive is arithmetic, not fashion. EMVCo's explanation is that in 2010 NIST announced it would not support EMV's longest RSA key length, 1984 bits, after 2030; longer keys mean longer computation, and a tap has a hard time budget. ECC reaches comparable strength at a fraction of the size. RSA key lengths remain under annual review.

Why the chip killed stripe cloning

Put the two technologies side by side and the asymmetry is total.

A magnetic stripe is a fixed pattern of magnetised particles encoding a fixed string of characters. The information required to make a working duplicate is exactly the information required to use the original, which is a way of saying the stripe has no secret at all. A skimmer read the tracks and a fifty-pound encoder wrote them to a blank. The counterfeit was not a good imitation of the card; it *was* the card, in every sense the acquiring system could test.

A chip has a secret and never says it. No command in Book 3 Table 3 outputs a private key or a shared symmetric key. The most complete possible read — every SELECT, every READ RECORD against every record named in every AFL, every GET DATA — yields the PAN, the expiry, the name, the certificates and the counters. It yields a card that can answer questions about itself. It does not yield

one that can sign a fresh unpredictable number under DDA, nor one that can compute an Application Cryptogram over a new amount, because both need key material no command will disclose.

Three further locks sit in the same door. The Application Transaction Counter makes replay visible to the issuer, because values that repeat or go backwards are detectable in the authorisation message. The unpredictable number makes replay useless to the terminal, because the challenge changes every time. And, as chapter 13 set out, the chip's Track 2 Equivalent Data (tag 57) is conventionally personalised with a different card verification value from the physical stripe's, so lifting that record onto a blank produces a stripe that fails verification at the issuer.

The result is in the fraud statistics. UK counterfeit card fraud losses were £129.7 million in 2004, up 17 per cent on £110.6 million in 2003, just as migration began: the Northampton trial ran in 2003, national rollout followed, and on 14 February 2006 signature ceased to be accepted for in-store chip card purchases. UK Finance's twentieth-anniversary assessment, published on that date in 2026, records a 95 per cent reduction in counterfeit card fraud losses over the intervening two decades. Chip and PIN now accounts for 30 per cent of UK card payments by value, averaging £93 a transaction against £17 for a contactless tap.

Three qualifications belong with that number. SDA-only cards were never protected in the way the summary implies, and a large installed base of them persisted for years. Magnetic stripe fallback — a terminal accepting the stripe when the chip fails to read — reopened the door wherever permitted, which is why schemes have progressively closed it. And the fraud did not vanish; it moved to the channel where no chip is present. That is Volume V.

The chip in one paragraph

An EMV chip is an ISO/IEC 7816 smart card running a payment application addressed by a 5-to-16-byte Application Identifier, speaking APDUs of the form CLA-INS-P1-P2-Lc-Data-Le and answering with data plus a two-byte status word. The terminal selects a directory (`1PAY.SYS.DDF01` on contact, `2PAY.SYS.DDF01` on contactless), builds a candidate list, selects an application, learns from the PDOL what transaction data the card requires, sends it in GET PROCESSING OPTIONS, gets back a capability profile and a file map, reads the named records, verifies the certificate chain up to a scheme public key it already holds, challenges the card to sign something fresh, and asks for a verdict with GENERATE AC. Static data authentication proves the data was signed; dynamic proves the card is present; combined proves the card, the decision and the transaction are one indivisible statement. Through all of it no instruction exists that will make the chip disclose its keys — which is why a card that answers questions cannot be photocopied, and a card that merely presents data always could.

14.98 Common wrong ideas

Wrong: The chip is a small computer with a clock and power of its own. **Right:** It is parasitic on the terminal for both voltage and clock and computes only while both are present, it cannot start anything, and it must ask the terminal for the date without any way of telling whether the terminal is lying.

Wrong: The card decides whether to approve the transaction. **Right:** The terminal decides which question to ask and the card may only choose among the answers to that question, and it may return an answer weaker than the one requested but never a stronger one.

Wrong: Reading data off a chip is an attack, so a contactless card that gives up an account number over the air is broken. **Right:** Any reader may power any card and obtain the account number, expiry, cardholder name and usage rules; the security does not come from hiding those fields but from the fact that holding them does not let you produce a cryptogram.

Wrong: Offline data authentication proves the card and the terminal to each other. **Right:** The certificate chain runs one way only, and nothing in the ordinary flow requires the terminal to prove anything, because a terminal is a commodity device and the chip cannot check a revocation list or even know today's date.

Wrong: Any card with a chip cannot be cloned. **Right:** A card performing only static data authentication carries a genuine signature over material identical on every transaction, so a device that recorded it once can present it again; freshness arrives only with dynamic authentication.

Wrong: "EMV" is a single protocol, so a Visa tap and a Mastercard tap run the same steps. **Right:** Contact runs the four EMV books while contactless runs an entry-point specification and a family of per-scheme kernels with their own commands, data elements and decision logic, and treating EMV as one protocol is the commonest error in engineering documentation about cards.

Wrong: A `6A82` in a trace is an error. **Right:** It is how a card says "I do not have that application", it is the expected response during selection, and it is what sends the terminal into trying each Application Identifier from its own list one at a time.

Wrong: Application Identifiers are seven bytes long. **Right:** An identifier is a five-byte registered provider identifier plus up to eleven bytes of proprietary extension, so lengths run from 5 to 16, and a terminal that assumes seven fails on UnionPay and girocard.

Wrong: A library that handles `9000` and `61 xx` has covered the protocol. **Right:** A card may also answer `6C xx`, meaning the expected-length byte was wrong and `xx` is the correct value, and an implementation that ignores it passes every test in the lab and fails intermittently in the field.

Wrong: Dynamic data authentication protects the transaction. **Right:** Standalone dynamic authentication proves the card is genuine without binding the signature to the amount, so a device between card and terminal can let it succeed and still interfere with the decision; only combined generation joins identity, decision and transaction into one signature.

14.99 Chapter summary in 20 lines

1. Behind the eight gold pads on a card, five of which do anything, sits a die a couple of millimetres square holding a processor, a program in read-only memory, non-volatile memory for data and keys, and hardware for arithmetic on very long numbers.
2. The structural difference from a magnetic stripe is that a stripe is a recording while a chip is a correspondent, so everything else in card security follows from the fact that a chip answers rather than presents.
3. Reading a chip is trivial and deliberate, because the account number, expiry, cardholder name and usage rules are exactly the fields a merchant needs.
4. What the chip will never do is disclose its private key or the symmetric key it shares with its issuer, because no instruction in the set would output them and the program sits in read-only memory laid down at manufacture.
5. EMV is not a standard in its own right but a profile of the ISO/IEC 7816 series, currently version 4.4 of October 2022 in four books, whose provisions take precedence wherever they differ from those standards.
6. Every exchange is a command of class, instruction and two parameter bytes with an optional body, answered with optional data and a mandatory two-byte status word.
7. The whole vocabulary is thirteen commands, of which an ordinary purchase uses five or six, and the status words carry as much of the protocol as the commands do.

8. An application is addressed by an Application Identifier of 5 to 16 bytes, and terminals that assume a fixed length break on real cards from real schemes.
9. Selection begins with a directory, `1PAY.SYS.DDF01` on contact and `2PAY.SYS.DDF01` on contactless, differing by one byte, and falls back to trying each identifier in turn when the directory is absent.
10. Selecting the application returns a Processing Options Data Object List, the card telling the terminal exactly what to concatenate, in what order and at what size, with no tags, so that the card is spared the work of parsing.
11. The Data Object List padding and truncation rules are asymmetric between numeric and non-numeric formats, and reversing them yields a well-formed command whose cryptogram fails at the issuer with no useful diagnostic.
12. GET PROCESSING OPTIONS returns an Application Interchange Profile saying what the card can do and an Application File Locator saying which records to read and how many of them feed offline data authentication.
13. READ RECORD retrieves those records, and the rule that low short file identifiers have their template tag and length stripped from the authentication input while high ones do not is the most commonly mis-implemented rule in the area.
14. Offline data authentication starts from a scheme public key already held in the terminal, walks a certificate chain through the issuer, and only then trusts anything the card says about itself.
15. Static data authentication proves that fixed data was signed by an issuer whose certificate chains to the scheme, which defeats a forger who alters the data and not one who copies it wholesale.
16. Dynamic data authentication adds a key pair belonging to the individual chip, which signs a number the terminal has just invented, so the signature is fresh and a recording is worthless.
17. Combined generation puts the cryptogram and a hash of the transaction inside that signature, so substituting any one of identity, decision or transaction invalidates all three.
18. Version 4.4 adds Extended Data Authentication on elliptic curves, because NIST will not support EMV's 1984-bit RSA after 2030 and a tap has a hard time budget.
19. GENERATE APPLICATION CRYPTOGRAM is the decision command, and a published trace in which the terminal asked for an offline approval and the card returned an online request is card action analysis in a single exchange.
20. The chip destroyed counterfeit stripe fraud — a 95 per cent reduction in the United Kingdom across two decades — because no command discloses a key, the transaction counter makes replay visible to the issuer and the unpredictable number makes it useless to the terminal, but the fraud did not vanish so much as move to the channel where no chip is present.

Chapter sources: EMV Integrated Circuit Card Specifications for Payment Systems v4.4 (October 2022), Book 3 — sections 1.3, 5.1, 5.3, 5.4, 6.1–6.3, 6.5, 9, 10.3, Tables 1–4, 10–18, 21–25, 29–32, 41, 46 and Figures 1, 2, 4, 5; EMV Book 1 v4.3 (November 2011), sections 10–12 and Tables 38–48; EMV Book 2, sections 5 and 6, 10.2.5.2 and Annex D1, for the certificate and signature formats and the 1984-bit bound; EMVCo Specification Bulletin no. 243 and EMVCo's FAQ "EMV Chip Specifications To Support Elliptic Curve Cryptography" (28 October 2021); EMV Contactless Specifications, Entry Point, section 3.3.1, for PPSE selection; ISO/IEC 7816-4 sections 5.3–5.4, ISO/IEC 7816-5 for RID and PIX, ETSI TS 101 220 for the 5-byte RID plus 11-byte PIX, ISO/IEC 9796-2 for the signature scheme. Traces from Basin, Sasse and Toro-Pozo, "The EMV Standard: Break, Fix, Verify", IEEE Symposium on Security and Privacy 2021, Appendix B, and the CardContact OpenSCDP EMV tutorial. Status words and AIDs cross-checked against EFTlab's references. Fraud figures from UK Finance, "20 years of change in payments — from Chip and PIN to mobiles" (14 February 2026), and APACS 2004 figures as reported by Pinsent Masons. Every hex trace printed here was re-parsed by hand against its stated lengths.

15

CHAPTER

The Cryptogram

15.0 What this chapter gives you

1. You will be able to explain why a criminal holding a perfect byte-for-byte record of everything a card said last Tuesday cannot use that record to buy anything on Wednesday.
2. You will be able to say why the Application Cryptogram is a message authentication code and not a signature, and why the terminal that carries it cannot check a single bit of it.
3. You will be able to name the three cryptogram types and explain why a refusal is sealed as carefully as an approval.
4. You will be able to walk the three-level key hierarchy from an issuer master key, through a card-unique key derived with the account number and sequence number, to a session key derived with the transaction counter.
5. You will be able to assemble the byte string a cryptogram is computed over, pad it correctly, and explain why the mandatory `'80'` byte is appended even when the message is already a multiple of eight.
6. You will be able to read an Issuer Application Data field, pull out the Derivation Key Index, the Cryptogram Version Number and the Card Verification Results, and say why that version byte is the most load-bearing byte in the whole message.
7. You will be able to distinguish the two Authorisation Response Cryptogram methods and explain how the issuer's approval travels inside a MAC so that nobody in the chain can flip it.
8. You will be able to state precisely what a valid cryptogram proves and what it does not — not that a cardholder was present, not that a PIN was entered, not that the amount displayed matched the amount authenticated, and not which merchant was involved.
9. You will be able to give an honest account of the transaction counter: it makes replay detectable rather than impossible, and real issuer hosts apply a window rather than a hard ceiling for defensible reasons.
10. You will be able to diagnose a failing cryptogram from the pattern of which cards fail, knowing that the mathematics has not failed in the field and the byte strings fed to it fail constantly.

Every chapter of this volume so far has described something the card *has*. A number embossed on the front. A stripe with three tracks. A printed code on the signature panel. A chip that runs a program and answers questions. All of it is data, and all data can be copied.

This chapter is about the one thing on the card that cannot be copied, because it does not exist until the moment of the transaction and it is never the same twice. It is eight bytes long. It is called the Application Cryptogram, and it is the reason that a criminal holding a perfect byte-for-byte record of everything your card said last Tuesday cannot use that record to buy anything on Wednesday.

It is also the single most misunderstood object in payments. People who work with cards every day will tell you the chip "signs" the transaction, which is not quite what happens; that the cryptogram proves the cardholder was present, which it does not; or that the terminal checks it, which it emphatically cannot. Getting this right matters more than getting any other detail in this volume right, because when a cryptogram fails there is no diagnostic. The issuer's host sees a mismatch between sixteen hexadecimal characters it computed and sixteen it received, and it has no way of telling you which of the eleven inputs was wrong. Engineers lose weeks to this. The purpose of this chapter is that you should not.

The plain version

Imagine you and your bank have agreed on a secret recipe.

The recipe is not a password and it is not a code word. It is a *method*: a way of taking any piece of writing, stirring it through a fixed sequence of steps, and producing sixteen characters of gibberish at the end. The stirring is designed so that changing anything at all in the writing — one digit, one letter, a single space — changes every one of the sixteen output characters completely and unpredictably. There is no way to work backwards from the gibberish to the writing, and no way to produce the right gibberish for a given piece of writing unless you know the recipe.

You keep the recipe. The bank keeps the recipe. Nobody else has it.

Now you go to buy a coffee.

What gets written down

The till writes out a short statement of exactly what is about to happen. Not a description in words — a fixed list of facts, always the same facts in always the same order, because both you and the bank need to write out the identical statement or the gibberish will not match.

The statement for a coffee in Leeds on 17 August 2026 says:

- The amount is four pounds twenty.
- There is no cashback.
- The till is in the United Kingdom.
- The till's own safety checks found one thing worth mentioning: this purchase is above the amount the shop is allowed to approve on its own.
- The money is pounds sterling.
- The date is the seventeenth of August.
- This is a purchase, not a refund and not a cash withdrawal.
- Here is a number the till just made up on the spot: 9A1BC3D4. It has never used it before and will never use it again.

Then the card adds two facts of its own:

- Here is what I am capable of doing.
- **This is the sixty-first time I have ever been used.**

The card stirs that whole statement through the recipe and produces sixteen characters: `733084432BA24DAE`.

Why those last two facts are the clever part

The till's made-up number is there to defeat the eavesdropper. Suppose somebody has been recording everything your card says. They have last Tuesday's sixteen characters. To reuse them they would need today's statement to be identical to Tuesday's — but today's till invented a different number, so it is not identical, so Tuesday's gibberish is wrong. It is refused.

The card's own count is there to defeat something subtler: the shop itself. A dishonest shop could take a genuine statement and a genuine set of sixteen characters and send the pair to the bank twice, hoping to be paid twice. But the card counts every single use, and the count is inside the statement. The bank sees that use number sixty-one has already been paid for. The second copy is refused.

The count also does something else, which is quieter and more important. **The recipe changes every time the count changes.** It is not literally one fixed recipe; it is a family of recipes, and the count picks which member of the family is used today. So even if somebody somehow worked out the exact recipe used for purchase sixty-one, it would tell them nothing at all about purchase sixty-two.

Three possible answers

When the till hands the statement to the card, it is not just asking for a signature. It is asking a question, and the card gets to answer it. There are three answers, and each comes with its own set of sixteen characters.

The card can say **yes, on my own authority** — I know this cardholder, the amount is small, I have not been used much lately, go ahead without asking the bank. This answer is called a Transaction Certificate.

The card can say **ask the bank** — this needs a decision I am not entitled to make. This answer is called an Authorisation Request Cryptogram. The sixteen characters travel across the world to the bank, which recomputes them and compares.

The card can say **no**. This answer is called an Application Authentication Cryptogram, and — this is the part people find odd — it still comes with sixteen characters. The refusal is sealed too, so that nobody downstream can quietly turn a refusal into an approval.

And here is a detail worth carrying to dinner: the till *asks* for one of these three, and the card is allowed to give a worse answer than the one requested. A till can ask for "yes, on your own authority" and be told "ask the bank". It can ask for "ask the bank" and be told "no". It can never be told something better than it asked for. The terminal proposes; the card disposes.

The bank answers with a seal of its own

The bank recomputes the sixteen characters from its own copy of the recipe. They match. The bank checks the balance and decides to approve.

Now the bank has a problem. If it just sends back the word "approved", anybody in the chain — the shop's till, the shop's software, the phone line — could invent that word. So the bank does the same trick in reverse. It takes the card's sixteen characters, mixes in its answer, stirs the result through the shared recipe, and sends back its own short scramble.

The card receives it, does the same calculation, and if the two agree the card knows something quite specific: *the bank itself, and nobody in between, said yes to this exact transaction.* Only then will it act on instructions from the bank — reset a PIN counter, clear its offline spending allowance, or in the extreme case block itself.

That is the whole mechanism. Two seals, one from the card and one from the bank, over a statement that both of them wrote out identically, using a recipe that changes on every use.

Where the plain version stops being true

The recipe picture gets you to the right conclusion, and it hides four things that matter enormously in practice.

It is not a signature, and the terminal cannot check it

The word "sign" implies that anyone can verify and only one party can produce. That is asymmetric cryptography, and the Application Cryptogram is not that. It is a **Message Authentication Code** computed with a *symmetric* key. Card and issuer hold the same secret. Anybody who can verify a cryptogram can also forge one.

This has an immediate practical consequence that surprises people: **the terminal cannot verify the cryptogram at all.** It has no key and never will. When a terminal receives eight bytes back from a GENERATE AC command it has no idea whether they are correct, random, or fabricated. It copies them into the authorisation message and forwards them. Everything the terminal does to satisfy itself that the card is genuine — static, dynamic and combined data authentication — is a *separate* mechanism using public keys, which was Chapter 4's subject. The cryptogram is not part of it.

Because the key is symmetric, the security of the whole scheme rests on the key never leaving tamper-resistant hardware at either end: a chip on one side, a hardware security module on the other. A single issuer master key extracted from a poorly managed host is not one compromised card. It is every card in that key's range, forever.

There is no such thing as "the data the card signs"

The plain version implies a fixed list. There is not one. EMV Book 2 section 8.1.1 gives a table of ten elements and calls it the "recommended minimum set". It is a recommendation. What is actually included, and in what order, is fixed by the application's **Cryptogram Version Number**, which is a scheme-proprietary value published in Visa's, Mastercard's, Amex's, JCB's and UnionPay's own specifications and not in EMV's.

Two cards from the same issuer with different Cryptogram Version Numbers compute cryptograms over different byte strings. An issuer host that assumes one layout for all its cards will silently fail on the others. This is, by a wide margin, the most common cause of "the cryptogram does not verify and we cannot work out why".

It says nothing about the cardholder, the merchant, or what you agreed to

A valid ARQC proves that a chip holding the correct key participated in a transaction with those exact parameters. It proves nothing else.

It does not prove the cardholder was present. It does not prove a PIN was entered — cardholder verification is reported separately, in the Cardholder Verification Method Results and in the card's own Card Verification Results, and an attacker who can suppress the PIN check can still obtain a perfectly valid cryptogram. It does not prove the amount displayed to the customer matches the amount in the command; it authenticates the amount the *terminal* supplied. And it does not identify the merchant: neither the merchant identifier nor the terminal identifier is in the recommended minimum set.

The counter only stops replay if somebody checks it

The Application Transaction Counter makes replay *detectable*. It does not make it impossible. Detection requires the issuer to remember the highest counter value it has seen for that card and refuse anything at or below it.

Real issuer hosts are much less strict than that, and for defensible reasons: authorisations arrive out of order, offline transactions consume counter values that never reach the host at all, reversals and pre-authorisations complicate the sequence, and a card used at an unattended terminal may burn counter values with no corresponding message. Most hosts therefore apply a window rather than a hard ceiling, and some check nothing at all. A gap in the counter is evidence of offline activity, not of fraud; a repeat is evidence of something wrong, but not necessarily of an attack.

The technical version

The three cryptogram types

The terminal requests a cryptogram with the GENERATE APPLICATION CRYPTOGRAM command — class `'80'`, instruction `'AE'`, P2 `'00'` — and states which type it wants in P1, the reference control parameter. The top two bits carry the request; bit 5 additionally asks the card to return a combined dynamic signature.

P1 bits b8 b7	Requested
`0 0`	AAC — Application Authentication Cryptogram (decline)
`0 1`	TC — Transaction Certificate (approve offline)
`1 0`	ARQC — Authorisation Request Cryptogram (go online)
`1 1`	RFU

The card replies with the cryptogram and with a single byte, the **Cryptogram Information Data**, tag `9F27`, stating which type it actually generated. The top two bits use the same encoding as P1. Bits b6 and b5 flag a payment-system-specific cryptogram, bit b4 indicates that an advice message is required, and bits b3 to b1 carry a reason or advice code: `000` no information given, `001` service not allowed, `010` PIN Try Limit exceeded, `011` issuer authentication failed.

The industry is not consistent about `1 1` in the Cryptogram Information Data. Older editions of Book 3, and a good deal of kernel documentation still in circulation, label it AAR — Application Authorisation Referral, the card asking for a voice referral. Referral processing does not appear in EMV 4.3 Book 2 at all, current terminal specifications do not implement it, and current tag references treat the value as reserved. Treat any card that returns it as broken rather than as requesting a referral.

The response arrives in one of two shapes. Format 1, template tag `'80'`, is a primitive object: Cryptogram Information Data, then Application Transaction Counter, then Application Cryptogram, then optionally Issuer Application Data, concatenated with no tags and no lengths. Format 2, template tag `'77'`, is TLV-encoded, with `9F27`, `9F36` and `9F26` mandatory and `9F10` optional. A Common Core Definitions application must use format 2 and must include a 32-byte `9F10`.

The key hierarchy

Three levels, and everything else in this chapter depends on keeping them straight.

At the top sits the **Issuer Master Key for Application Cryptograms**, IMK-AC. One per issuer per key range, held in the issuer's hardware security module, never in a card. Under two-key Triple DES it is 128 bits; the AES option supports 128, 192 and 256.

From it, at personalisation, the issuer derives one **ICC Application Cryptogram Master Key**, MK-AC, per card. EMV Book 2 Annex A1.4 gives three optional methods, all taking the Application PAN and the PAN Sequence Number as input:

Method	Cipher	Rule
Option A (A1.4.1)	Triple DES only	Concatenate PAN with PAN Sequence Number; take the rightmost 16 digits as an 8-byte number `Y`, left-padded with hexadecimal zeroes if shorter. `ZL := DES3(IMK)[Y]`, `ZR := DES3(IMK)[Y XOR 'FFFFFFFFFFFFFFFF']`, and `MK` is `ZL` followed by `ZR`, with odd parity then forced on every byte.
Option B (A1.4.2)	Triple DES only	If the PAN is 16 digits or fewer, use Option A. Otherwise hash PAN concatenated with PAN Sequence Number under SHA-1, take the first 16 decimal digits of the 20-byte result as `Y`, and if there are fewer than 16 decimal nibbles, decimalise the remaining ones using the table A->0, B->1, C->2, D->3, E->4, F->5, appending from the left. Then continue from Option A step 2.
Option C (A1.4.3)	AES only	Concatenate PAN with PAN Sequence Number, left-pad with hexadecimal zeroes to a 16-byte number `Y`, and encipher under AES.

None of the three is mandatory; Book 2 says so explicitly. Issuers may adopt another method, and some do.

From MK-AC, at transaction time, both card and issuer derive a **session key**, SK-AC. Book 2 Annex A1.3.1 specifies the Common Session Key option. The diversification value `R` for Application Cryptogram and ARPC generation is the two-byte Application Transaction Counter followed by n-2 bytes of `'00'`. Where the key is longer than the block — Triple DES with a 128-bit key, or AES with 192 or 256 bits — two blocks are built from `R`:

```
F1 = R with byte 3 replaced by 'F0'
F2 = R with byte 3 replaced by '0F'
SK = leftmost k bits of { ALG(MK)[F1] || ALG(MK)[F2] }
```

Where key and block are the same size — AES-128 — the derivation collapses to `SK := AES(MK)[R]`. The same session key is used for every command in a single transaction, which is why the second GENERATE AC does not need a fresh derivation.

Not every application uses this. Visa Cryptogram Version 10 performs no session key derivation at all: the card master key *is* the cryptogram key. Visa CVN 18 and CVN 22, and Mastercard CVN 14 and CVN 15, use the Common Session Key method above. Mastercard CVN 12 and CVN 13 use an older Mastercard derivation that folds the terminal's Unpredictable Number into the session key as well as the counter.

A worked derivation

Take an issuer master key of `0123456789ABCDEFFEDCBA9876543210` — a test key, not a real one — a PAN of `4111111111111111` and a PAN Sequence Number of `01`.

Option A concatenates them to `41111111111111101`, eighteen digits, and takes the rightmost sixteen:

```
Y      = 1111111111111101
MK_AC  = A768CE9E1551FDE3A45B4C7C3DE9DC52
```

The card is on its sixty-first use, so the Application Transaction Counter is `003D`. The diversification value and the two derivation blocks are:

```
R      = 003D000000000000
F1     = 003DF00000000000
F2     = 003D0F0000000000
SK_AC  = C471A83E37AEF3FE5D4CEF548575DD9C
```

Everything from here uses `SK_AC` and nothing else. `MK_AC` is not touched again until the next transaction.

What goes into the MAC

Book 2 section 8.1.1 gives the recommended minimum set. Eight elements come from the terminal, two from the card:

Source	Element
Terminal	Amount, Authorised (Numeric)
Terminal	Amount, Other (Numeric)
Terminal	Terminal Country Code
Terminal	Terminal Verification Results
Terminal	Transaction Currency Code
Terminal	Transaction Date
Terminal	Transaction Type
Terminal	Unpredictable Number
Card	Application Interchange Profile
Card	Application Transaction Counter

For a Common Core Definitions application with Cryptogram Version `'5'` or `'6'`, Book 2 replaces that table with a mandatory eleven-element list in a fixed order — the ten above, in exactly that order, followed by the Issuer Application Data. Every real scheme implementation does something equivalent: it appends the card's own **Card Verification Results**, which live inside the Issuer Application Data, to the end of the terminal's eight elements and the card's two.

The terminal does not choose which elements to send. The card publishes **CDOL1**, tag `8C`, a Card Risk Management Data Object List naming the tags and lengths the terminal must concatenate into the first GENERATE AC command, and **CDOL2**, tag `8D`, for the second. The terminal fills the template in the order given. The Data Object List padding and truncation rules of Book 3 section 5.4 apply in full, and they are asymmetric: numeric-format elements are truncated from the left and padded with leading zeroes, everything else is truncated from the right and padded with trailing zeroes, and compressed numeric is padded with trailing `'FF'`s. Getting that backwards produces a well-formed command that yields a cryptogram nobody can verify.

Here is the actual byte string for our £4.20 coffee, purchased in the United Kingdom on 17 August 2026 at a terminal whose floor limit the amount exceeds:

Tag	Element	Bytes	Value
`9F02`	Amount, Authorised (Numeric)	6	`000000000420`
`9F03`	Amount, Other (Numeric)	6	`000000000000`
`9F1A`	Terminal Country Code	2	`0826`
`95`	Terminal Verification Results	5	`0000008000`
`5F2A`	Transaction Currency Code	2	`0826`
`9A`	Transaction Date	3	`260817`
`9C`	Transaction Type	1	`00`
`9F37`	Unpredictable Number	4	`9A1BC3D4`
`82`	Application Interchange Profile	2	`1D00`
`9F36`	Application Transaction Counter	2	`003D`
—	Card Verification Results	4	`03A00000`

Three of those values deserve a sentence each. The amounts are numeric format, twelve digits packed as six bytes, so `£4.20` is `000000000420` and not four hundred and twenty of anything — the currency's minor-unit exponent is not in the string, which is why the Transaction Currency Code has to be. The Terminal Verification Results byte 4 has bit 8 set, which is "transaction exceeds floor limit": that single bit is the reason the terminal asked for an ARQC rather than a TC. And the Application Interchange Profile byte 1 bit 3 is set, meaning "issuer authentication is supported" — the card will accept an EXTERNAL AUTHENTICATE command later.

Concatenated, that is thirty-seven bytes:

```
000000000420 000000000000 0826 0000008000 0826 260817 00
9A1BC3D4 1D00 003D 03A00000
```

The MAC algorithm

Book 2 Annex A1.2.1 specifies the computation for an 8-byte block cipher. It is ISO/IEC 9797-1 with the message padded according to ISO/IEC 7816-4, which the specification notes is equivalent to padding method 2 of ISO/IEC 9797-1: append a mandatory `'80'` byte, then the smallest number of `'00'` bytes that brings the length to a multiple of eight. The `'80'` is mandatory even when the message is already a multiple of eight. Our thirty-seven bytes become forty:

```
...9F36 003D  03A00000  80 0000
```

Split into five 8-byte blocks `X1..X5`. The session key is treated as a left half `KSL` and a right half `KSR`. Chain the blocks in CBC mode using **single DES with `KSL` only**, starting from an initial value of eight zero bytes:

```
Hi := ALG(KSL)[ Xi XOR Hi-1 ],  for i = 1..B,  H0 = '0000000000000000'
```

Then apply the output transformation. Book 2 permits either ISO/IEC 9797-1 Algorithm 1, which is simply `HB+1 := HB`, or Algorithm 3:

```
HB+1 := ALG(KSL)[ ALG-1(KSR)[ HB ] ]
```

Algorithm 3 — encrypt with the left half, decrypt with the right, encrypt with the left again, applied only to the final chaining value — is what every deployed scheme uses, and it is what Book 2 mandates for the Common Core Definitions Cryptogram Version `'5'`. It is the construction the banking world calls the Retail MAC or the ANSI X9.19 MAC. The Application Cryptogram is the s most significant bytes of `HB+1` with $s = 8$, so the whole of it.

For our example:

```
ARQC = 733084432BA24DAE
```

Eight bytes. Sixty-four bits. An attacker with no key guesses correctly about once in eighteen quintillion attempts, and the issuer will have declined and blocked the card long before that.

The AES option, Annex A1.2.2, is a different construction: ISO/IEC 9797-1:2011 Algorithm 5, which is CMAC, over 16-byte blocks, with the two CMAC subkeys `K1` and `K2` derived from the session key and the constant `'10000111'`. Padding is added only when the message is *not* already a multiple of sixteen, and the final block is masked with `K1` when no padding was added and `K2` when it was. Cryptogram Version `'6'` of the Common Core Definitions mandates AES CMAC with $s = 8$, and requires Option C master key derivation to go with it. Cryptogram Version `'5'` mandates Triple DES, Algorithm 3, and Option B.

The Application Transaction Counter

Tag `9F36`, two bytes, binary, maintained entirely by the card. The terminal cannot write it and the issuer cannot reset it. It increments once per transaction — in practice on the first GENERATE AC, or on GET PROCESSING OPTIONS in some contactless kernels — and it does not decrement.

Two bytes means the counter tops out at `FFFF`, 65,535. What a card does on reaching that ceiling is application-defined rather than specified by EMV; the counter is not permitted to be reused, so an application that reaches it has reached the end of its life. At a hundred transactions a month a card would need fifty-four years to get there, so it is not a practical constraint for a payment card, though it is a real one for some transit and stored-value applications.

The counter does three jobs at once, and they are worth separating.

It **diversifies the session key**. This is its cryptographic job and it is the reason it appears in `R`. Every transaction uses a different key, so a compromise of one transaction's key compromises exactly one transaction.

It **is inside the signed data**, as one of the ten recommended minimum elements. So it cannot be tampered with in flight, and its value binds the cryptogram to a specific position in the card's lifetime.

It **is visible to terminal risk management**. The terminal reads it directly with GET DATA — `80 CA 9F 36 00` — along with the **Last Online Application Transaction Counter Register**, tag `9F13`, which records the counter value at the last transaction that actually went online. The difference between the two is the number of transactions the card has completed offline since it last spoke to its issuer. The terminal compares that difference against the card's **Lower Consecutive Offline Limit**, tag `9F14`, and **Upper Consecutive Offline Limit**, tag `9F23`, and sets the corresponding Terminal Verification Results bits, which are what force the transaction online. Velocity checking is exactly this subtraction and nothing more.

At the issuer, the counter is the replay control. The honest statement of what it gives you is: an ARQC arriving with a counter value already seen for that card is either a duplicate or a forgery, and an ARQC arriving with a counter far above the last one seen indicates offline activity the issuer has not been told about. Both are signals. Neither is proof. The gap between "counter arrived out of sequence" and "fraud" is filled by the issuer's own risk rules, not by cryptography.

Issuer Application Data

Tag `9F10`, up to 32 bytes, described by EMV only as "proprietary application data for transmission to the issuer in an online transaction". EMV does not define its contents. The schemes do, and this is where an issuer host learns *how* to verify the cryptogram it has just been handed.

The Visa VIS layout, for Cryptogram Version Numbers 10 and 18, is:

Bytes	Field
1	Length Indicator
2	Derivation Key Index (DKI)
3	Cryptogram Version Number
4–7	Card Verification Results (4 bytes)
8+	Issuer Discretionary Data (optional)

A real one, taken from a Visa test transaction, is `9F10 07 06 00 0C 03A01000`: length indicator `'06'`, derivation key index `'00'`, Cryptogram Version Number `'0C'` — decimal 12, one of Visa's issuer-proprietary versions — and a four-byte Card Verification Results of `03A01000`, whose leading `'03'` is itself a length byte.

The Mastercard M/Chip 4 layout is different in both order and size:

Bytes	Field
1	Key Derivation Index
2	Cryptogram Version Number
3–8	Card Verification Results (6 bytes)
9–10	DAC / ICC Dynamic Number
11–18	Plaintext / Encrypted Counters (optional)

A real one: `9F10 12 02 12 A0000F240000 0000 00000000000000FF`.

A Common Core Definitions application always returns exactly 32 bytes, beginning with a length indicator and a **Common Core Identifier** whose high nibble is the IAD Format Code and whose low nibble is the Cryptogram Version. The remaining layout is given in Book 3's CCD annex.

Three things in that structure carry the whole weight of interoperability.

The **Derivation Key Index** tells the issuer host which of its master keys to load. Issuers rotate master keys across BIN ranges and over time; without the index the host would have to try them all.

The **Cryptogram Version Number** tells the host which algorithm to run — which master key derivation option, whether to derive a session key at all, which elements to concatenate and in what order, and which ARPC method the card expects. It is the single most load-bearing byte in the message.

The **Card Verification Results** are the card's own account of what happened. They report whether offline data authentication was performed and how it ended, whether offline PIN verification succeeded, whether the PIN Try Limit was exceeded, whether the offline counters were exceeded, how the *previous* transaction ended, and whether issuer authentication succeeded last time. The bit layouts are scheme-proprietary and are not published in EMV. Together with the Terminal Verification Results, the Card Verification Results are the two halves of the story: the terminal's version and the card's version of the same transaction, arriving in the same message, and sometimes disagreeing in ways that are extremely informative.

Verification at the issuer

The cryptogram and its companions travel to the issuer inside the ISO 8583 authorisation request, in data element 55, as a TLV list — `9F26` the cryptogram, `9F27` the Cryptogram Information Data, `9F36` the counter, `9F10` the Issuer Application Data, `9F37` the Unpredictable Number, `95` the Terminal Verification Results, `82` the Application Interchange Profile, and the rest. Volume III takes that message apart field by field.

The issuer host does six things, and all the cryptographic ones happen inside a hardware security module:

1. Parse `9F10`. Read the Derivation Key Index and the Cryptogram Version Number.
2. Select the issuer master key identified by the index, as a key token the host itself cannot read.
3. Derive MK-AC from the PAN and PAN Sequence Number using the option the Cryptogram Version requires.
4. Derive SK-AC from the Application Transaction Counter, unless the version says not to.
5. Reassemble the message: the same elements, in the same order, with the same lengths, that the card used. The host is reconstructing the terminal's work from the fields in the authorisation message, and any element it takes from the wrong place — the clearing amount rather than the authorised amount, its own date rather than the terminal's — produces a mismatch.
6. Compute the MAC and compare with the received `9F26`.

If they match, the card is genuine and the transaction data is intact. If they do not, the host has learned only that. Book 2 attaches a recommendation at this point that is easy to skip and important: an ARPC should only be returned if ARQC verification succeeded, and if an issuer insists on returning one anyway, it must not compute it from the ARQC it received from the network.

The ARPC and issuer authentication

The issuer's reply carries an **Authorisation Response Cryptogram**, and Book 2 section 8.2 defines two methods. Both use the same session key SK-AC that verified the ARQC.

Method 1 produces an 8-byte ARPC from the ARQC and the 2-byte Authorisation Response Code, tag `8A`, which is the ISO 8583 response code carried as two alphanumeric characters:

```
X    := ARC || '00' '00' '00' '00' '00' '00'
Y    := ARQC XOR X
ARPC := DES3(SK_AC)[Y]
```

For our transaction, with an ARC of `3030` — the two ASCII characters `0` and `0`, meaning approved:

```
ARQC = 733084432BA24DAE
X    = 3030000000000000
ARPC = 735C24FD87F1BF0E
```

The card verifies by deciphering the ARPC under SK-AC and XORing the result with its own ARQC; if what falls out is the ARC followed by six zero bytes, the issuer is authenticated.

Method 2 produces a 4-byte ARPC and carries far more information back to the card. Instead of a response code it uses a 4-byte binary **Card Status Update** and up to 8 bytes of Proprietary Authentication Data:

```
Y    := ARQC || CSU || Proprietary Authentication Data
ARPC := MAC(SK_AC)[Y], ISO/IEC 9797-1 Algorithm 3, s = 4
```

with the Issuer Authentication Data then formed as `ARPC || CSU || Proprietary Authentication Data`. Whether the proprietary data is included at all is governed by the "Proprietary Authentication Data Included" bit in the Card Status Update itself; if that bit is zero, the field is zero bytes long and, as Book 2 warns explicitly, anything sent in it is not protected by issuer authentication and the card should ignore it.

For our transaction, with a Card Status Update of `00800000`:

```
ARPC = 8F793CCC
Issuer Authentication Data = 8F793CCC 00800000
```

The Card Status Update is where the issuer reaches into the card, and its coding is published rather than proprietary: section C8 of Book 3's Common Core Definitions annex sets it out in Table CCD 11, for cryptogram versions `'5'` and `'6'`. The second byte carries the instructions: bit 8 "Issuer Approves Online Transaction", bit 7 "Card Block", bit 6 "Application Block", bit 5 "Update PIN Try Counter", bit 4 "Set Go Online on Next Transaction", bit 3 "CSU Created by Proxy for the Issuer", and bits 2 and 1 together the "Update Counters" instruction, which tells the card to leave its offline counters alone, set them to the upper offline limits, reset them to zero, or add this transaction to them. When bit 5 is set, the new PIN try count is taken from bits 4 to 1 of the first byte, whose top bit is the Proprietary Authentication Data Included flag; the third byte is reserved and the fourth is issuer-discretionary, defaulting to zero.

Note what bit 8 of byte 2 means. Under Method 2 the issuer's approval is *inside* the MAC. A card that verifies the ARPC and finds that bit clear will return an AAC and decline, no matter what the terminal was told. Under Method 1 the equivalent information is the ARC, also inside the calculation. Either way, the approval decision cannot be flipped in transit — which is precisely the class of attack that unauthenticated response codes leave open.

The Issuer Authentication Data reaches the card as tag `91`, 8 to 16 bytes, and there are two routes for it. If the Application Interchange Profile has byte 1 bit 3 set — "issuer authentication is supported" — the terminal sends an **EXTERNAL AUTHENTICATE** command:

Code	Value
CLA	`'00'`
INS	`'82'`
P1 P2	`'0000'`
Lc	8–16
Data	Issuer Authentication Data
Le	absent

It may be sent only once per transaction; a second attempt returns `6985`. Whatever the outcome, the terminal sets byte 1 bit 5 of the Transaction Status Information, "issuer authentication was performed". If issuer authentication fails, byte 5 bit 7 of the Terminal Verification Results is set, and the card's

CDOL2 will pull that byte into the second GENERATE AC, so the failure is itself covered by the second cryptogram.

If that Application Interchange Profile bit is *not* set, there is no EXTERNAL AUTHENTICATE. The issuer's data must instead be named in CDOL2 as tags `91` and `8A`, and reaches the card as part of the second GENERATE AC command. Mastercard M/Chip 4 takes the EXTERNAL AUTHENTICATE route; Visa VIS 1.4.x takes the second-GENERATE-AC route. An issuer host that assumes one of these for a portfolio containing both will find that half its script processing silently never executes.

The second GENERATE AC

Whatever happened online, the transaction is not finished until the terminal asks the card once more, using CDOL2, for a completion cryptogram. The terminal requests a TC if the issuer approved and it is willing to complete, or an AAC if not. The card performs its own second round of risk management, and again may only respond with something equal to or weaker than what was asked. The commonest surprise in this step is a card that returns an AAC after an approved authorisation: almost always the card failed to verify the ARPC, or verified it and found the approval bit clear.

Because the session key is derived from the Application Transaction Counter, and the counter does not increment between the two commands, both cryptograms in a transaction use the same SK-AC. They differ because the data differs — CDOL2 typically adds the Authorisation Response Code and the updated Card Verification Results.

A TC is not a courtesy. It is the evidence the acquirer presents in clearing that the transaction completed as authorised, and it is what an issuer asks for when a disputed transaction is being reconstructed. Schemes have been steadily tightening the requirement that the cryptogram appear in the clearing message and not only in the authorisation.

Combined data authentication

There is one arrangement in which the cryptogram and the public-key world do meet. Under **Combined DDA/Application Cryptogram Generation** — CDA — the card does not return the cryptogram in clear. It returns tag `9F4B`, Signed Dynamic Application Data, an RSA signature computed over data that includes the cryptogram itself.

For a Common Core Definitions application supporting CDA, the response to GENERATE AC for a TC or an ARQC contains exactly four objects: `9F27` the Cryptogram Information Data, `9F36` the counter, `9F4B` the signature, and `9F10` the 32-byte Issuer Application Data. Tag `9F26` is absent — the cryptogram is inside the signature. The leftmost bytes of the signed ICC Dynamic Data are the ICC Dynamic Number and its length, then the Cryptogram Information Data, then the 8-byte TC or ARQC, then a 20-byte Transaction Data Hash Code covering everything exchanged in the transaction. The signed data format byte is `'05'`, the recovered header is `'6A'`, the trailer is `'BC'`, and the padding pattern is `'BB'`. A card that responds with an AAC does not sign; it returns `9F26` in the clear, and the terminal declines.

CDA closes a specific hole: without it, an attacker sitting between card and terminal can let a genuine card generate a genuine ARQC and then alter the plaintext response, or substitute the card's answer to the second GENERATE AC. With it, altering anything invalidates a signature the terminal *can* check, because the terminal holds the certificate chain. This is the one place where the terminal's opinion of the cryptogram counts for something.

Failure modes

When a cryptogram does not verify, the issuer host learns nothing except that it did not verify. In practice the cause is almost always one of a short list, and it is worth having the list to hand.

Symptom	Usual cause
Every card in a BIN range fails	Wrong issuer master key, or wrong Derivation Key Index mapping
One card type fails, others succeed	Cryptogram Version Number not handled; wrong element list or wrong session key rule
Fails only for long PANs	Option A used where Option B is required, or vice versa
Fails only for non-zero PAN Sequence Numbers	Sequence number omitted from, or wrongly padded in, master key derivation
Fails only with cashback	Amount, Other omitted, or taken from the wrong field
Fails only in some currencies	Amount converted to major units, or the currency code taken from the account rather than the transaction
Fails intermittently at one estate	Unpredictable Number not reproduced byte-for-byte, or Data Object List padding applied the wrong way round
Verifies, but the card declines afterwards	ARPC method mismatch, or the approval bit in the Card Status Update not set
Verifies, but the counter is rejected	Offline transactions the host never saw, or an over-strict replay window

Every one of those is a data-assembly error, not a cryptographic one. The mathematics in this chapter has not failed in the field. The byte strings fed to it fail constantly.

The cryptogram in one paragraph

At personalisation, the issuer derives a card-unique key from its master key and the card's PAN. At transaction time, the card derives a use-unique session key from that card key and its own transaction counter, and computes an eight-byte Retail MAC over a fixed list of terminal-supplied and card-supplied facts — amount, currency, date, country, the terminal's random number, the card's counter and its own results. The type of cryptogram it returns is its verdict: TC to approve, ARQC to ask, AAC to refuse. The terminal cannot check any of it and merely forwards it. The issuer repeats the whole derivation from the PAN, the counter and the Cryptogram Version Number in the Issuer Application Data, and compares. If it approves, it computes a second cryptogram over its own answer so the card can be certain the answer is genuine, and only then will the card act on what the issuer says. Nothing in that chain is a secret the card gives away, which is why recording it is worthless, and why the eight bytes at the centre of it are the strongest single thing on the card.

15.98 Common wrong ideas

Wrong: The chip signs the transaction. **Right:** It computes a message authentication code with a symmetric key the issuer also holds, so anybody who can verify a cryptogram can also forge one, which is why that key must never leave tamper-resistant hardware at either end.

Wrong: The terminal checks the cryptogram before forwarding it. **Right:** The terminal has no key and never will, so when eight bytes come back from GENERATE AC it has no idea whether they are correct, random or fabricated, and it merely copies them into the authorisation message.

Wrong: A valid cryptogram proves the cardholder was present and entered a PIN. **Right:** It proves that a chip holding the correct key participated in a transaction with those exact parameters and nothing else; cardholder verification is reported separately, and an attacker who suppresses the PIN check can still obtain a perfectly valid cryptogram.

Wrong: There is a fixed list of data the card computes the cryptogram over. **Right:** Book 2 offers a recommended minimum set of ten elements, and what is actually included and in what order is fixed by the scheme-proprietary Cryptogram Version Number, which is by a wide margin the commonest cause of unexplained verification failures.

Wrong: The transaction counter makes replay impossible. **Right:** It makes replay detectable, and only if the issuer checks; hosts apply windows rather than hard ceilings because authorisations arrive out of order and offline transactions consume counter values that never reach the host at all.

Wrong: A cryptogram that fails to verify indicates a cryptographic problem. **Right:** In practice every failure is a data-assembly error — the wrong master key, an unhandled cryptogram version, an omitted cashback amount, a currency converted to major units, or Data Object List padding applied the wrong way round.

Wrong: A decline is just a decline, so it needs no cryptogram. **Right:** The refusal is sealed too, precisely so that nobody downstream can quietly turn a refusal into an approval.

Wrong: A terminal that requests an offline approval and receives one has been obeyed. **Right:** The terminal proposes and the card disposes, returning the requested type or a more restrictive one and never a less restrictive one, so a request to approve offline may come back as an online request or an outright refusal.

Wrong: One session key derivation rule covers an issuer's whole portfolio. **Right:** Visa Cryptogram Version 10 derives no session key at all and uses the card master key directly, Visa 18 and 22 and Mastercard 14 and 15 use the common session key method, and Mastercard 12 and 13 fold the terminal's unpredictable number into the derivation as well.

Wrong: Once the issuer approves, the transaction is finished. **Right:** The terminal must ask the card once more using the second data object list, and a card that failed to verify the response cryptogram, or verified it and found the approval bit clear in the Card Status Update, will refuse after an approved authorisation.

15.99 Chapter summary in 20 lines

1. Everything else on a card is data and all data can be copied, but the Application Cryptogram cannot, because it does not exist until the moment of the transaction and is never the same twice.
2. It is eight bytes long and it is a message authentication code computed with a key the card and the issuer share, not a signature anybody can verify.
3. Because the key is symmetric, the terminal cannot check the cryptogram at all and simply forwards eight bytes whose correctness it has no way of assessing.
4. The security of the whole scheme therefore rests on that key never leaving tamper-resistant hardware, since a single master key extracted from a badly managed host compromises every card in its range forever.
5. Three keys stand in a hierarchy: an issuer master key, a card-unique key derived from it with the account number and sequence number, and a session key derived from that with the card's own transaction counter.

6. Book 2 gives three master key derivation options and mandates none of them, and which one a card uses is announced by the scheme's Cryptogram Version Number rather than by EMV.
7. The session key changes with every transaction because the counter does, so compromising one transaction's key compromises exactly one transaction.
8. The data covered by the MAC is a fixed concatenation of eight facts supplied by the terminal, two supplied by the card, and in every real implementation the card's own Card Verification Results as well.
9. The terminal does not choose those elements: the card publishes two Card Risk Management Data Object Lists and the terminal fills them in the order given, under padding rules that run one way for numeric formats and the other for everything else.
10. The computation is CBC chaining under the left half of the session key followed by an encrypt-decrypt-encrypt on the final block, the construction the banking world calls the Retail MAC, with an AES option for newer cryptogram versions.
11. The type of cryptogram returned is the card's verdict: a Transaction Certificate approves offline, an Authorisation Request Cryptogram asks the issuer, and an Application Authentication Cryptogram refuses.
12. The terminal requests a type and the card may return that type or a more restrictive one, never a less restrictive one.
13. The transaction counter does three jobs at once: it diversifies the session key, it sits inside the signed data, and it is visible to terminal risk management as the gap from the Last Online ATC Register.
14. What the counter honestly gives an issuer is a signal and not a proof, since a repeated value means duplicate or forgery and a large jump means unreported offline activity, and the distance from signal to fraud is covered by risk rules.
15. The Issuer Application Data is where the host learns how to verify what it has been handed, carrying the Derivation Key Index, the Cryptogram Version Number and the Card Verification Results in scheme-specific layouts.
16. The Cryptogram Version Number is the single most load-bearing byte in the message, because it decides the derivation option, whether a session key exists, which elements are concatenated and which response method the card expects.
17. The issuer parses that field, loads the master key by index, derives the card key and session key, reassembles the terminal's byte string from the authorisation message and compares — and a mismatch tells it nothing except that there was a mismatch.
18. The issuer replies with its own cryptogram under one of two methods, so that its approval sits inside a MAC and cannot be flipped in transit, which is exactly the attack that unauthenticated response codes leave open.
19. Under combined data authentication the cryptogram is not returned in the clear at all but inside an RSA signature the terminal can check, the one place where the terminal's opinion of the cryptogram counts for anything.
20. When a cryptogram fails, the cause is almost always a data-assembly error rather than a cryptographic one, because the mathematics in this chapter has not failed in the field and the byte strings fed to it fail constantly.

Sources: EMV Integrated Circuit Card Specifications for Payment Systems, Book 2 (Security and Key Management) v4.3, sections 8.1–8.3 and Annexes A1.2, A1.3, A1.4 and the Common Core Definitions part; EMV Book 3 (Application Specification) v4.3, sections 5.4, 6.5.5 and 6.5.7 and the Data Elements Dictionary; ISO/IEC 9797-1 and ISO/IEC 7816-4; EMVCo tag references at emvlab.org; AWS Payment Cryptography and IBM z/OS ICSF documentation for scheme cryptogram version behaviour; CardContact OpenSCDP EMV traces. Worked figures were computed from the Book 2 algorithms using the stated test key and are reproducible.

16

CHAPTER

Online and Offline

16.0 What this chapter gives you

1. You will be able to answer the question the previous chapter skipped — whether anybody actually phones the bank — and say what happens on the occasions when nobody can.
2. You will be able to explain why small purchases keep working during an outage while large ones stop, and why the cut-off is a number sitting in one particular till rather than a national figure.
3. You will be able to perform the three comparisons of Terminal Action Analysis by hand, in order, as a bitwise OR of two policies followed by a bitwise AND with the terminal's own observations.
4. You will be able to decode a five-byte Terminal Verification Results bit by bit, and say why its fifth byte cannot be decoded without knowing which kernel produced it.
5. You will be able to state the asymmetric defaults for absent action codes — all ones for the issuer's online and default lists, all zeros for every terminal action code — and explain why they point in opposite directions.
6. You will be able to explain why setting a terminal floor limit to zero does not straightforwardly mean "always go online".
7. You will be able to distinguish an offline approval from a store-and-forward deferral, and say why calling both "offline" is how merchants end up carrying risk they did not know they had.
8. You will be able to sort a network failure into its three shapes — the request never sent, sent with no answer, and an answer arriving too late — and say which of them makes a reversal mandatory whatever the local decision was.
9. You will be able to read a Terminal Type byte and explain why a supermarket till, a fuel pump, an aircraft seatback and a transit gate behave differently, including why the pump must commit before it knows the amount.
10. You will be able to run the four diagnostics that recover most of this ground when an estate is sending everything online, approving too much offline, failing cryptograms during outages, or producing duplicate debits afterwards.

Chapter 15 ended with the card producing eight bytes and the issuer producing eight bytes back. That description skipped a question, and the question is the one that decides whether your card works in a tunnel, on an aeroplane, at three in the morning when your bank's authorisation host is being patched, and at a petrol pump that has no idea how much fuel you are about to take.

The question is: does anybody phone the bank?

Most people assume the answer is always yes. It is not. A meaningful fraction of card transactions in the world are approved without any message ever reaching the issuer, and much of the *decision* is made locally even when a message does travel. The chip in your card is a small computer running its

own risk policy, and the terminal is running a different risk policy, written by a different organisation, with different money at stake. Both are expressed in the same format: a five-byte bitmap. Both are consulted on every transaction. Neither can see the other's reasoning.

This chapter is about those two policies, the bitmap they share, and what happens when the line to the bank is not there.

The plain version

Imagine a new employee on a supermarket till in Leeds. On her first day she is handed a laminated card from head office saying, in effect: *here are the situations in which you must ring us before taking the money, here are the situations in which you must refuse outright, and here is what to do if the phone is dead.*

That is the shop's policy. Now imagine that every customer also hands over a second laminated card, printed by their own bank, saying the same three things in the same three categories but with the bank's own opinions on them. The employee now has two policies to satisfy at once, and she does not get to choose between them.

She also has a tick sheet of forty boxes, each a specific observation: *the card would not prove it was genuine. The card has expired. The customer could not enter a PIN because the keypad is broken. The amount is above the limit the shop may approve on its own.* She fills it in as she goes, and most boxes stay empty on most transactions.

At the end she lays the tick sheet next to the two policy cards and asks three questions in order.

First: *do any of the ticked boxes appear on either "refuse outright" list?* If yes, she refuses, and no phone call happens.

Second, if not: *do any of the ticked boxes appear on either "ring us" list?* If yes, she rings.

Third, if she cannot ring — the line is dead, the shop has no line, the call timed out: *do any of the ticked boxes appear on either "if the phone is dead" list?* If yes, she refuses. If no, she approves it herself.

That is the whole mechanism. Two policies, one tick sheet, three questions.

A real transaction, with real numbers

A customer buys £42.50 of groceries. The till's limit for approving things on its own — head office calls it the floor limit — is £30.00.

The employee starts ticking. The amount is over £30, so the box marked *transaction exceeds floor limit* gets a tick. The card also tells her it has been used sixty-one times in its life and last spoke to its own bank on use number fifty-five. That is six uses ago, and the card's instructions say it prefers to check in after five, so a second box is ticked: *the card has gone too long without ringing home.* Everything else is fine, and thirty-eight of the forty boxes stay empty.

Question one: does either "refuse outright" list mention either of those? Neither does. She does not refuse.

Question two: does either "ring us" list mention them? The bank's card says yes to the floor limit. It happens not to mention the six-uses box at all — that bank does not consider it urgent — but one match is enough. She rings, the bank approves, and the transaction goes through in about half a second.

The same transaction with the phone line cut

Now run it again with the shop's broadband down. Questions one and two go the same way: no outright refusal, and the answer to "should I ring?" is still yes. But she cannot ring, so she falls through to question three and reads the third list.

The bank's third list *does* mention the floor limit, so the transaction is declined. Not because the customer has no money, not because the card is stolen, but because that customer's bank decided in advance that a £42.50 offline purchase it cannot see is a risk it does not want to take.

Had the basket been £8.40, no box would have been ticked, no phone call would have been needed, and the sale would have gone through with the line still dead. That is the most useful thing to understand about card payments during an outage: small purchases keep working and large ones stop, and the cut-off is not a national figure but a number sitting in that specific till for that specific scheme.

Two risk managers, and a spot check

One twist surprises people, and since it was the subject of the previous chapter it needs only a sentence here. The employee does not actually approve anything. She *proposes*. The card gets the final word, and it may be more cautious than she was but never less: she can say "I'd approve this offline" and the chip can answer "no, ring them", but she can never say "ring them" and have the chip answer "don't bother". There are two risk managers in the conversation, not one.

One last wrinkle explains something you may have noticed: occasionally a £2 coffee takes longer and clearly goes online for no visible reason. That is deliberate. A terminal may pick transactions at random and send them online even when nothing is wrong, because if small transactions *never* went online a thief would learn to keep every purchase under the floor limit and would never once be checked. The chance is weighted, so a purchase near the floor limit is far more likely to be selected than a tiny one.

If you take one thing to dinner, take this. Your card and the shop's till each carry a written policy about when the bank must be consulted, both policies are consulted every time, the strictest wins, and there is a third policy that only comes out when the line is down.

Where the plain version stops being true

The tick-sheet story gets you to the right answer for a supermarket till on an ordinary day. It hides four things, and each of them is where real implementations go wrong.

The two policies are merged, not consulted separately, and the merge only ever adds

The employee reads two lists and matches against either. That is nearly right, but the actual operation is a bitwise OR of the two policies followed by a bitwise AND with the tick sheet. Because the two are ORed before the comparison, *neither party can ever remove a condition the other has set*: an issuer cannot tell a terminal to stop going online for something the acquirer insists on, and an acquirer cannot relax a condition the issuer has demanded. Every configuration change either adds strictness or does nothing. There is no negotiation, no priority, no override.

That catches out people diagnosing live estates. When a terminal goes online it cannot tell you *why*: it has a five-byte mask, a five-byte tick sheet and a non-zero AND result, and it records neither which bit caused it nor which policy contributed that bit. To find out, you fetch the sheet and both policies out of the transaction record and do the arithmetic yourself.

An absent policy is also not a neutral policy. It is a defined value, and the value differs depending on which list is missing. An issuer that ships a card with no "ring us" list has not made it permissive; it

has made it maximally strict, because the fallback for the missing online list is all ones. An acquirer that leaves the terminal's lists unconfigured has done the opposite, because the fallback for a missing terminal action code is all zeros. The two defaults point in opposite directions, and that asymmetry causes a great many surprises in the field.

The terminal never approves, and the tick sheet is not a report

Calling the tick sheet a record of what went wrong is misleading twice over. First, not every ticked box is a problem. "Online PIN entered" is one of the forty boxes and is set on perfectly successful transactions; so is "transaction selected randomly for online processing". The sheet records *observations*, some neutral and some adverse, and it is the two policies — not the sheet — that decide which matter.

Second, the tick sheet is not a report at all. It is an *input*: its five bytes are handed to the chip inside the command that asks for a cryptogram, and they form part of the data the cryptogram is computed over. Two things follow. A terminal that sets a bit after sending the sheet has produced a transaction whose cryptogram covers a different sheet from the one in the authorisation message, and the issuer's verification will fail with no diagnostic. And the issuer is verifying the terminal's *claims* about its own behaviour: nothing prevents a compromised terminal from ticking boxes untruthfully, and the cryptogram merely proves the card saw the same lies the issuer did.

"Floor limit" is at least four different limits owned by four different parties

The plain version has one number, £30, sitting in the till. Real transactions are governed by several limits at once, and confusing them is the most common error in this whole area.

There is the terminal floor limit, loaded by the acquirer, specific to a particular scheme's application on that terminal, doing exactly what the story says. There are contactless reader limits, separate values with separate names governing whether a tap is allowed at all, whether it must go online, and whether it must carry a cardholder verification. There are the card's own offline limits, set by the issuer, which count *transactions* and in some schemes *cumulative amounts* rather than looking at a single purchase. And there are the scheme and regulatory limits — the figure a cardholder thinks of as "the contactless limit" — enforced somewhere else entirely.

The card's limits are the ones that usually bite, and they are worded more precisely than most people realise. The chip carries two consecutive-offline limits, and they are not "soft" and "hard" versions of the same thing. One states the issuer's preference for a terminal that *has* online capability; the other for a terminal that has *not*. Reading them as a graduated pair will make your velocity behaviour wrong at exactly the unattended terminals where it matters.

Setting the terminal floor limit to zero, incidentally, does not straightforwardly mean "always go online". It means the floor-limit observation is recorded every time; whether that produces an online request still depends on whether either policy lists it under "ring us". With a hand-configured pair that both leave the bit clear it does not, and you have a terminal with a zero floor limit approving everything offline.

Offline approval and store-and-forward are different things, and "the network went down" is at least three things

The plain version treats "cannot ring" as a single event with a single response. It is not.

Offline approval is a *decision*: nobody is going to ask the issuer, the card has sealed an approval in a cryptogram, and the money will be claimed later through clearing. Store-and-forward is a deferral: the terminal holds a cryptogram that says "please ask the bank", cannot ask now, and intends to ask later. The customer has already left with the goods, and if the answer comes back "no" three hours

later there is no customer to hand it to. The two put the loss in different places under different rules, and calling both "offline" is how merchants end up carrying risk they did not know they had.

The failure itself also has at least three shapes with three different correct responses. The request may never have left the terminal, in which case nothing has happened anywhere and the terminal falls through to its third policy. It may have left with no answer coming back, in which case the issuer may well have approved and debited a shadow balance, and the terminal is obliged to send a reversal *regardless of what it decides locally*, precisely because it does not know. Or an answer may arrive after the terminal has given up, leaving an approval in the issuer's records for a transaction the terminal has declined and the card has refused to certify. Treating those three as one case is the classic way to produce duplicate debits during an outage, and it is a failure of the acquirer's software, not of EMV.

The technical version

Two decisions, in this order

EMV places the online/offline determination in two processes with two owners, both running on every contact transaction that reaches the cryptogram stage. **Terminal Action Analysis** is EMV Integrated Circuit Card Specifications for Payment Systems, Book 3 (Application Specification), section 10.7: the terminal's decision, producing a *request* for one of three cryptogram types via the reference control parameter of the GENERATE APPLICATION CRYPTOGRAM command. **Card Action Analysis** is Book 3 section 10.8, the card's decision; the chip may return the requested type or a more restrictive one, never a less restrictive one, and reports what it did in the Cryptogram Information Data, tag 9F27. Book 4, Table 3, sets out what the terminal must do with each answer. Everything else in this chapter feeds those two decisions.

Terminal Verification Results

The tick sheet is the Terminal Verification Results, tag 95, five bytes, binary, generated by the terminal and defined in Book 3, Annex C5. Bits run b8 (most significant) to b1 within each byte; reserved-for-future-use bits are coded as zero and neither party checks them.

Byte	Bits, b8 downwards
1	b8 offline data authentication was not performed; b7 SDA failed; b6 ICC data missing; b5 card appears on terminal exception file; b4 DDA failed; b3 CDA failed; b2 SDA selected
2	b8 ICC and terminal have different application versions; b7 expired application; b6 application not yet effective; b5 requested service not allowed for card product; b4 new card
3	b8 cardholder verification was not successful; b7 unrecognised CVM; b6 PIN Try Limit exceeded; b5 PIN entry required and PIN pad not present or not working; b4 PIN entry required, PIN pad present, but PIN was not entered; b3 online PIN entered
4	b8 transaction exceeds floor limit; b7 lower consecutive offline limit exceeded; b6 upper consecutive offline limit exceeded; b5 transaction selected randomly for online processing; b4 merchant forced transaction online

Byte	Bits, b8 downwards
5	b8 default TDOL used; b7 issuer authentication failed; b6 script processing failed before final GENERATE AC; b5 script processing failed after final GENERATE AC

Byte 5 is where contact and contactless diverge. In EMV Contactless Specifications for Payment Systems, Book C-2 (Kernel 2), byte 5 additionally carries the relay resistance outcomes: bits for "relay resistance threshold exceeded" and "relay resistance time limits exceeded", and a two-bit field in b2 and b1 encoding whether the relay resistance protocol was not supported, not performed or performed. A decoder handling both interfaces cannot decode byte 5 without knowing which kernel produced it. Note too that byte 1 b2, "SDA selected", is an observation rather than a failure: it exists so an issuer can express a policy about static data authentication being used where dynamic authentication was available.

Action codes: three lists, two owners, five bytes each

Every one of the six action codes has exactly the same five-byte layout as the Terminal Verification Results, bit for bit.

The **Issuer Action Codes** live on the card and are read during the READ RECORD stage: `9F0D` for Default, `9F0E` for Denial, `9F0F` for Online. Book 3 is precise about scope. Denial specifies the issuer's conditions that cause denial *without attempt to go online*; Online, the conditions that cause a transaction to be transmitted online; Default, the conditions that cause rejection *if it might have been approved online but the terminal is unable to process the transaction online*.

The **Terminal Action Codes** are acquirer-configured and terminal-resident. On the contact interface they have no EMV tag because they never cross the card interface; Book 4, Table 7 lists all three as application-dependent terminal data elements, required if non-zero values are used. Contactless kernels carry them as configuration objects — Kernel 2 holds Terminal Action Code – Denial in tag `DF8121`.

The defaults when a code is absent must be memorised, because they are asymmetric.

Action code	Value used if absent
Issuer Action Code – Denial	`00 00 00 00 00`
Issuer Action Code – Online	`FF FF FF FF FF`
Issuer Action Code – Default	`FF FF FF FF FF`
Any Terminal Action Code	`00 00 00 00 00`

Book C-5 (Kernel 5) states the three issuer defaults explicitly for cards that do not carry the codes, matching Book 3, and Book 4 confirms that in the absence of a Terminal Action Code the default is all bits set to 0. Book 4 attaches a strong recommendation to that default: the bits for "offline data authentication was not performed", "SDA failed", "DDA failed" and "CDA failed" should be set to 1 in both Terminal Action Code – Default and Terminal Action Code – Online. Those are byte 1 b8, b7, b4 and b3, so the minimum implied is `CC 00 00 00 00`. An acquirer shipping all-zero terminal action codes has followed the letter of the specification and ignored its advice.

The three comparisons

Book 3 section 10.7 defines the decision as three bitwise tests in a fixed order. In each, the two five-byte action codes are ORed and the result ANDed with the Terminal Verification Results; a non-zero result means the condition applies.

Denial first: if any bit is set in `TVR AND (TAC-Denial OR IAC-Denial)`, the terminal declines offline and requests an AAC, and no online message is attempted. Online second, and only if the terminal has online capability: if any bit is set in `TVR AND (TAC-Online OR IAC-Online)` the terminal requests an ARQC, otherwise a TC. Default third, used when the terminal is offline-only so the online test never ran, or is online-capable but unable to go online: if any bit is set in `TVR AND (TAC-Default OR IAC-Default)` the terminal requests an AAC, otherwise a TC.

Book 4 section 6.3.6 confirms the bookkeeping. On an offline approval the terminal sets the Authorisation Response Code to "Offline approved"; on an offline decline, to "Offline declined"; and when it transmits online it sets no value at all, leaving the field for the response message.

A complete worked example

Take a supermarket till in Leeds on 17 August 2026. Terminal Type `9F35` is `22`, attended, merchant-controlled, offline with online capability. The floor limit is £30.00, so `9F1B` is `00 00 0B B8`; the basket is £42.50, so Amount, Authorised, `9F02`, is `00 00 00 00 42 50`; Transaction Currency Code `5F2A` and Terminal Country Code `9F1A` are both `0826`; Transaction Date `9A` is `26 08 17` and Transaction Type `9C` is `00`. The card reports an Application Transaction Counter, `9F36`, of `00 3D` — sixty-one — a Last Online ATC Register, `9F13`, of `00 37`, and a Lower Consecutive Offline Limit, `9F14`, of `05`.

Terminal risk management runs. £42.50 exceeds £30.00, so TVR byte 4 b8 is set. The counter is six ahead of the last online register, which exceeds five, so byte 4 b7 is set. Data authentication succeeded, the application is in date, the PIN verified offline. The resulting TVR is:

```
95 05 00 00 00 C0 00
```

The card presents these issuer action codes:

```
9F0E 05 00 10 18 00 00      IAC - Denial
9F0F 05 FC 68 BC 98 00      IAC - Online
9F0D 05 FC 40 AC 80 00      IAC - Default
```

The acquirer has loaded `CC 00 00 00 00` as both Terminal Action Code – Online and Terminal Action Code – Default, and all zeros as Terminal Action Code – Denial.

Denial test: `(00 00 00 00 00) OR (00 10 18 00 00)` is `00 10 18 00 00`, ANDed with the TVR gives all zeros. No denial.

Online test: `(CC 00 00 00 00) OR (FC 68 BC 98 00)` is `FC 68 BC 98 00`, ANDed with `00 00 00 C0 00` gives `00 00 00 80 00` — non-zero, from the floor-limit bit alone. Note what did *not* contribute: byte 4 of this issuer's online code is `98`, that is b8, b5 and b4, and does not include b7. This issuer does not regard the lower consecutive offline limit as a reason to go online. The terminal requests an ARQC.

Now suppose the line is dead, and the terminal falls through to the default test: `(CC 00 00 00 00) OR (FC 40 AC 80 00)` is `FC 40 AC 80 00`, ANDed with `00 00 00 C0 00` gives `00 00 00 80 00`. The terminal requests an AAC and sets the Authorisation Response Code to `Z3`. Same basket, same card, same till, different answer, and the only thing that changed is which of the three lists was consulted.

Terminal risk management: what sets those bits

None of this runs unless the card asks for it. Application Interchange Profile, tag `82`, byte 1 b4 is "terminal risk management is to be performed". A terminal must also perform it whenever its own floor limit checking demands, but the AIP bit is the card's request.

Floor limit checking. Terminal Floor Limit, tag `9F1B`, is four bytes binary, terminal-sourced, held per application identifier — one physical terminal can carry different floor limits for different schemes. Book 4, Table 7 makes it required on an offline terminal or an offline terminal with online capability. The terminal may keep a log of recent transactions keyed on Application PAN and Application PAN Sequence Number so a customer cannot defeat the limit by splitting a purchase; where a log is kept, the current amount is summed with the logged amounts before comparison.

Random transaction selection. Required on offline terminals with online capability, using three acquirer-configured values: Target Percentage to be Used for Random Selection, Threshold Value for Biased Random Selection, and Maximum Target Percentage to be Used for Biased Random Selection. Below the threshold the selection probability is the target percentage; between the threshold and the floor limit it is interpolated linearly:

```
interpolation      = (amount - threshold) / (floor limit - threshold)
transaction target = ((max target - target) x interpolation) + target
```

The terminal generates a random number in the range 1 to 99 and selects the transaction for online processing if that number is less than or equal to the transaction target percent, setting TVR byte 4 b5. With a threshold of £10.00, a target of 10 per cent, a maximum target of 40 per cent and a floor limit of £30.00, a £22.50 purchase interpolates at 0.625 and yields a target of 28 per cent. That is why a purchase at £29.99 is very nearly always selected.

Velocity checking. The card supplies Lower Consecutive Offline Limit, tag `9F14`, and Upper Consecutive Offline Limit, tag `9F23`, one byte each. These are not soft and hard versions of one control. Book 3 is explicit: the lower limit is the issuer's preference for the maximum number of consecutive offline transactions *in a terminal with online capability*, the upper limit the same preference *in a terminal without online capability*. If either is absent from the card, velocity checking is skipped entirely. The terminal retrieves Application Transaction Counter `9F36` and Last Online ATC Register `9F13` with GET DATA — class `80`, instruction `CA`, tag in P1-P2 — and the difference between them is the number of transactions since the card last went online. Exceeding the lower limit sets TVR byte 4 b7; exceeding the upper limit sets byte 4 b6 as well.

Exception file. If the Application PAN matches an entry in the terminal's exception file, TVR byte 1 b5 is set. Exception files are largely historical in online-capable estates and largely useless in offline-only ones, because they cannot be updated.

The terminal also maintains Transaction Status Information, tag `9B`, two bytes, recording which *processes ran* rather than what they found: byte 1 b8 offline data authentication performed, b7 cardholder verification performed, b6 card risk management performed, b5 issuer authentication performed, b4 terminal risk management performed, b3 script processing performed, byte 2 reserved. The TSI says a step happened; the TVR says what it observed. A terminal reporting "issuer authentication failed" in the TVR while leaving "issuer authentication was performed" clear in the TSI has a bug.

Card Verification Results

The card runs its own risk management and keeps its own tick sheet: the Card Verification Results. It is not a standalone tag. It lives inside Issuer Application Data, tag `9F10`, binary, zero to thirty-two bytes, returned by the card in the GENERATE AC response.

There is no single layout. The Issuer Application Data is proprietary to the payment system and its internal structure is selected by the Cryptogram Version Number carried inside it. EMV defines a Card Verification Results only for Common Core Definitions applications, in Book 3, Annex C; everything else is published by the schemes.

Two layouts cover most of the estate. For Visa VSDC with Cryptogram Version Numbers 10 and 18, the Issuer Application Data is a length indicator, a one-byte Derivation Key Index, a one-byte Cryptogram Version Number, a four-byte Card Verification Results and optional Issuer Discretionary Data, so `06 00 0C 03 A0 10 00` carries a Card Verification Results of `03 A0 10 00`. For Mastercard M/Chip, an eighteen-byte Issuer Application Data is a one-byte Key Derivation Index, a one-byte Cryptogram Version Number, a *six*-byte Card Verification Results, a two-byte DAC or ICC Dynamic Number and eight bytes of counters.

What it reports, across all the schemes, is the card's own account of the transaction and its recent history: the cryptogram type returned at each GENERATE AC; whether offline PIN verification was performed and succeeded; whether the PIN try limit is exceeded; whether the previous transaction went online and completed; whether issuer authentication was attempted and failed; whether the card's velocity counters are exceeded; whether an issuer script was received and failed; and counts of consecutive offline transactions and, in some schemes, cumulative offline amounts. Bit positions differ between schemes and between cryptogram versions within a scheme, so there is no generic Card Verification Results decoder; any tool claiming to be one is guessing at the version. And for most cryptogram versions the Issuer Application Data is included in the data over which the Application Cryptogram is computed, so the Card Verification Results is sealed. That asymmetry matters: the terminal's tick sheet is a claim the terminal makes about itself, whereas the card's is signed.

The GENERATE AC exchange

The first GENERATE AC is formatted according to Card Risk Management Data Object List 1, tag `8C`, supplied by the card, which for a typical application includes `9F02` Amount Authorised, `95` Terminal Verification Results, `5F2A` Transaction Currency Code, `9A` Transaction Date, `9C` Transaction Type and `9F37` Unpredictable Number. The second is formatted from Card Risk Management Data Object List 2, tag `8D`, whose critical addition is Authorisation Response Code, tag `8A`, two bytes. The terminal must never modify an Authorisation Response Code obtained from a response message; Book 4, Annex A6, is explicit. But when the transaction is *not* authorised online the terminal generates the code itself, and Book 4, Table 35, defines exactly four values:

Authorisation Response Code	Value
Offline approved	`Y1`
Offline declined	`Z1`
Unable to go online, offline approved	`Y3`
Unable to go online, offline declined	`Z3`

The distinction between `Y1` and `Y3`, and between `Z1` and `Z3`, is precisely the distinction between the second comparison and the third: `Y1` and `Z1` mean the terminal decided offline because its policies told it to, `Y3` and `Z3` mean it decided offline because it had no choice. Issuers price offline risk on that difference, and a terminal reporting `Z1` when it meant `Z3` is misrepresenting an outage as a policy decline.

Where an online response arrives containing Issuer Authentication Data, tag `91`, the terminal delivers it to the card — through EXTERNAL AUTHENTICATE or inside the second GENERATE AC, depend-

ing on the application — and sets TSI byte 1 b5; if the card rejects it, TVR byte 5 b7 is set. Book 4 section 12.2.2 covers the case that trips people up: when the response contains no Issuer Authentication Data at all, the terminal must *not* execute EXTERNAL AUTHENTICATE and must set the TSI bit to 0, continuing on the Authorisation Response Code alone. That is a downgraded authorisation, not a failure, and the TVR bit must stay clear.

Why a fuel pump, an aircraft seatback and a supermarket till behave differently

EMV encodes the answer in one byte. Terminal Type, tag `9F35`, is two digits: operational control and environment. Book 4, Annex A1, gives the full table.

	Financial institution	Merchant	Cardholder
Attended, online only	11	21	—
Attended, offline with online capability	12	22	—
Attended, offline only	13	23	—
Unattended, online only	14	24	34
Unattended, offline with online capability	15	25	35
Unattended, offline only	16	26	36

There are no attended cardholder-controlled types. Terminal types `14`, `15` and `16` with cash disbursement capability — Additional Terminal Capabilities, tag `9F40`, byte 1, cash bit set to 1 — are ATMs; nothing else is, and any logic that decides "is this an ATM" from the merchant category code rather than from those two data elements is doing it wrong.

The supermarket till is `22`: attended, merchant-controlled, offline with online capability, with a modest floor limit, a working keypad, a person behind it and a network it expects to be up. All three comparisons are available to it, and on an ordinary day it goes online above the floor limit and approves the rest itself.

The fuel pump is `25` — unattended, merchant-controlled, offline with online capability — with merchant category code `5542`, automated fuel dispensers, as distinct from `5541`, service stations. Its problem is not connectivity. It is that *the amount is not known when the decision has to be made*: the pump must be authorised to dispense before anyone knows how much fuel will come out.

That breaks the arithmetic. The Amount, Authorised, tag `9F02`, that goes into the cryptogram is an *estimate*, and the cryptogram covers the estimate and only the estimate. The final amount, which is what actually clears, was never seen by the chip. Every scheme therefore wraps automated fuel dispensers in its own rules rather than leaving it to EMV: the Visa Core Rules and Visa Product and Service Rules carry a section on Automated Fuel Dispenser Transactions with tables specifying status check authorisations, maximum amount initial authorisations and maximum allowed amounts, while Mastercard classifies the same devices through the cardholder-activated terminal levels in DE 61, subfield 10 of the authorisation message.

Everything else follows from that estimate: a hold for an amount unrelated to the fuel taken, a reconciliation between hold and clearing, and a figure on the cardholder's statement that never appeared on the pump. It is not a bug. It is the only way to run a device that must commit before it knows.

The aircraft seatback is `26` or `36` — unattended, offline only. "Offline only" is a specification term with a precise meaning, given in Book 4, Table 1: the transaction can only be completed offline by the terminal. On such a terminal the second comparison never happens; Terminal Action Code – Online and Issuer Action Code – Online are read and never used. The decision reduces to the denial test and

then the default test, which is exactly why Issuer Action Code – Default falls back to all ones when absent, and why a card carrying no Issuer Action Code – Default is declined by an aircraft for the smallest observation.

Mastercard's terminology for this environment is CAT level 4, in-flight commerce, alongside level 1 for automated dispensing machines, level 2 for self-service terminals, level 3 for limited amount terminals and level 9 for mobile point-of-sale acceptance devices; the Mastercard Switch Rules manual sets out the requirements for each. Two further differences bite on an aircraft: the exception file can be no fresher than the last time someone updated it on the ground, and there is no attendant to compare a signature, so cardholder verification collapses to whatever the card's CVM List permits an unattended device to do — in practice No CVM Required or offline PIN.

Transit gates are a fourth case, frequently confused with the third. A gate is unattended and often offline at the moment of the tap, but it is usually not an offline-only terminal: it is an online-capable terminal deliberately deferring the online request under a scheme-sanctioned transit model, which is not an EMV offline approval. The regulatory position differs too. Article 12 of Commission Delegated Regulation (EU) 2018/389, the regulatory technical standards on strong customer authentication, exempts unattended terminals *for the purpose of paying a transport fare or a parking fee* with no monetary ceiling, whereas an ordinary contactless purchase falls under Article 11 with its ceiling of EUR 50 for a single transaction, EUR 150 cumulative since the last strong authentication, and no more than five consecutive transactions. A seatback screen selling a gin and tonic is not paying a transport fare, and does not get Article 12. The United Kingdom's own figures were £100 single and £300 cumulative until the Financial Conduct Authority removed the mandatory caps on 19 March 2026, leaving issuers with adequate fraud controls to set their own.

Offline approval and its risk

The risk in offline approval is not that the cryptogram is weak; it is exactly as strong as an online one. The risk is that *nobody with a balance in front of them has looked at it*. Four things are unknown at that moment: whether the account has funds; whether the card has been reported lost or stolen since the exception file was last updated; whether the cryptogram is even valid, because as Chapter 15 established the terminal has no key and cannot check; and whether this is the fifteenth offline transaction on that card in an hour at fifteen terminals, none of which can see the others.

The controls against all four are the card's own. Offline data authentication establishes that the chip is genuine before any offline approval is contemplated, which is why Book 4 recommends so firmly that the authentication-failure bits be set in Terminal Action Code – Default and – Online, and why offline approval following a failed authentication is the single most dangerous configuration in EMV. Beyond that, the consecutive offline limits force a card back online after a defined number of unseen transactions, and scheme-proprietary cumulative amount limits do the same on value. The gap between the Application Transaction Counter and the Last Online ATC Register is literally a measure of the issuer's current exposure on that card.

Liability follows configuration, and configuration is auditable after the fact: the Terminal Verification Results, Terminal Type, Terminal Capabilities and cryptogram all travel in the clearing record. A terminal that approved offline above its scheme-permitted floor limit, or after data authentication failed, has departed from the rules visibly, and scheme chargeback rules attach consequences to exactly that.

Store-and-forward

Store-and-forward is what an online-capable terminal does when it cannot reach the acquirer and its policies still permit completion: it stores the full transaction, cryptogram included, and transmits it when connectivity returns. Some acquirers call it deferred authorisation. It is not offline approval, because the answer is still outstanding.

The cryptogram in a stored transaction is an ARQC nobody has verified, or a TC generated against an Authorisation Response Code of `Y3`, and either way the issuer receives it late. Hosts enforcing a strict Application Transaction Counter ceiling will reject it, because by then the card has almost certainly been used elsewhere and the host has seen higher counter values. This is one of the concrete reasons issuer hosts apply a counter *window* rather than a hard ceiling.

The obligation to send a reversal is not discretionary. Book 4 section 12.2.3 covers the response that is not received, is received too late, or is malformed: after any permitted repeats the terminal processes the transaction as unable to go online, runs the default comparison, and issues the second GENERATE AC. Then, where online data capture is performed by the acquirer, the terminal *shall send a reversal message regardless of the final decision on the transaction*, so that if the host did receive the request and send a response, that transaction is cancelled. If the card's final answer was a TC, the terminal must also create a financial record for the acquirer.

Third, storing a transaction before authorisation means holding cardholder data before authorisation, the one window in which PCI DSS permits sensitive authentication data to exist at all, and only under specific protection; such an implementation must be designed against the standard's pre-authorisation storage requirements, not retro-fitted to them. Fourth, presentment windows are set by scheme rules and measured in days, and a queue surviving a long outage may hold transactions that are no longer presentable.

The messages are ordinary ISO 8583 — authorisation 0100 and 0110, financial 0200 and 0210, advice 0220, reversal 0400 with 0410 or advice-form 0420 with 0430. Whether a stored transaction goes as a financial request or as an advice depends on the acquirer's specification, and determines whether the issuer is asked a question or told a fact.

When the network dies mid-transaction

Sort the failures by where the transaction was when the line went.

Before the request was sent. Book 4 section 12.2.1: the terminal compares the Terminal Verification Results with Terminal Action Code – Default and Issuer Action Code – Default, decides, and issues the second GENERATE AC with an Authorisation Response Code of `Y3` or `Z3`. Nothing has happened at the issuer, no reversal is required, and the card's answer in the Cryptogram Information Data is final.

After the request was sent, with no response, or with a response that arrives too late. Section 12.2.3. The terminal may repeat the request if the acquirer requires it — repeat requirements are explicitly outside the scope of EMV — and then must treat the transaction as unable to go online. Same default comparison, same `Y3` or `Z3`. But the reversal is now mandatory under online data capture, because the issuer may have approved and debited. The late response is the case most often implemented wrongly, because the response *is there* and a naive implementation will process it after the card has already been given a `Z3` and returned an AAC. The transaction is then declined at the card and approved at the issuer, and only the reversal reconciles them.

Response arrives without Issuer Authentication Data. Section 12.2.2, and not a network failure at all. The terminal does not execute EXTERNAL AUTHENTICATE, sets the TSI issuer-authentication bit to

0, leaves the TVR issuer-authentication-failed bit clear, and proceeds on the Authorisation Response Code.

The card leaves mid-transaction. Common on a contactless reader: the cardholder lifts the device between command and response. Kernel 2 long addressed it with a Torn Transaction Log, governed by Max Lifetime of Torn Transaction Log Record, tag `DF811C`, an integer in seconds, and Max Number of Torn Transaction Log Records, tag `DF811D`, together with the RECOVER AC command, which retrieves from the card the last transaction it completed so that a returning card can be matched against a stored torn record and the transaction finished rather than restarted — which would burn a second Application Transaction Counter value. That machinery is now historical. EMVCo's Specification Bulletin number 261 of October 2021, applying to Book C-2 version 2.10 and effective from the first of January 2022, states that torn transaction recovery is no longer supported in Kernel 2, retires the Torn Transaction Log, deletes the RECOVER AC section on the grounds that the command is no longer used, and strikes the torn data objects, tags `DF811C` and `DF811D` among them, from the data dictionary. The revision log of Book C-2 version 2.11, published in June 2023, records that bulletin as incorporated, and the current Kernel C-2 conformance statement offers implementers only mag-stripe mode and data exchange as options. A reader still running an older kernel will meet the mechanism; a newly approved one will not.

The issuer is down but the network is up. Not a terminal problem at all, and the terminal will never know. The scheme answers on the issuer's behalf — Visa calls it Stand-In Processing, Mastercard calls it Stand-In Processing Service — applying issuer-supplied parameters and reconciling afterwards. To the terminal this is an ordinary online authorisation, typically arriving without Issuer Authentication Data, which returns us to the downgraded authorisation case and a TSI with the issuer authentication bit at zero. If a large share of an estate's transactions show issuer authentication not performed, stand-in is the first place to look, not a terminal fault.

What to check when it goes wrong

Every terminal runs the denial test. Only online-capable terminals run the online test; offline-only terminals go straight from denial to default. And every terminal reaches the default test eventually, which is the point most often missed: an online-only terminal is not exempt from it, it simply expects never to get there. When it does, it needs a sensible Terminal Action Code – Default loaded, and online-only estates are precisely the ones that never bothered. With the fallback of all zeros and a card whose Issuer Action Code – Default is present and permissive, such a terminal will approve offline in exactly the circumstances its operator assumed impossible.

Four diagnostics recover most of the ground in this chapter. If an estate is sending everything online, pull the Terminal Verification Results and both online action codes from a sample of transactions and do the AND yourself; almost always the card carries no Issuer Action Code – Online, so the all-ones default applies, or data authentication is silently failing and byte 1 lights up every time. If an estate is approving too much offline, check whether the Terminal Action Codes were ever loaded, because all-zero terminal action codes plus a permissive card will pass every certification test and lose money in production. If cryptograms fail at the issuer only during outages, check whether the terminal is building the GENERATE AC data from a stale copy of the Terminal Verification Results after it has finished setting bits. And if duplicate debits appear after an outage, check the reversal path first: when online data capture is performed and the terminal was unable to obtain a valid response, a reversal is sent regardless of what the card finally decided.

The deeper point behind all four is the one this chapter opened with. There is no single authority in a card transaction: there is a terminal with an acquirer's policy, a card with an issuer's policy, a

merged bitmap neither party can see in full, and a network that is sometimes there. Everything that looks arbitrary about card acceptance — why the pump held £100, why the aeroplane refused a perfectly good card, why the corner shop kept working when the supermarket stopped — falls out of that arrangement, one five-byte comparison at a time.

16.98 Common wrong ideas

Wrong: The terminal reads the issuer's list and the acquirer's list and matches against either. **Right:** The two five-byte codes are ORed together and the result ANDed with the Terminal Verification Results, so neither party can ever remove a condition the other has set and every configuration change either adds strictness or does nothing.

Wrong: A card carrying no Issuer Action Code – Online is permissive. **Right:** The fallback for a missing issuer online or default list is all ones, making such a card maximally strict, whereas an unconfigured Terminal Action Code falls back to all zeros — the two defaults point in opposite directions.

Wrong: The Terminal Verification Results is a report of what went wrong. **Right:** It is an input, handed to the chip inside the command that asks for a cryptogram and covered by that cryptogram, and several of its bits — "online PIN entered", "transaction selected randomly for online processing" — are set on perfectly successful transactions.

Wrong: A terminal can tell you why it went online. **Right:** It holds a five-byte mask, a five-byte sheet of observations and a non-zero AND result, and records neither which bit caused it nor which policy contributed that bit, so you must fetch all three from the transaction record and do the arithmetic yourself.

Wrong: The floor limit is one number. **Right:** There is the acquirer's per-application terminal floor limit, a set of contactless reader limits, the card's own offline limits set by the issuer, and the scheme and regulatory limits, and confusing them is the most common error in this whole area.

Wrong: The lower and upper consecutive offline limits are soft and hard versions of the same control. **Right:** The lower is the issuer's preference for a terminal that has online capability and the upper for one that has not, and reading them as a graduated pair makes velocity behaviour wrong at exactly the unattended terminals where it matters.

Wrong: Setting the terminal floor limit to zero means every transaction goes online. **Right:** It means only that the floor-limit observation is recorded every time, and whether that produces an online request still depends on whether either policy lists that bit, so a hand-configured pair that both leave it clear gives a zero floor limit approving everything offline.

Wrong: Store-and-forward is a kind of offline approval. **Right:** Offline approval is a decision nobody will revisit, while store-and-forward holds a cryptogram that still says "please ask the bank" and defers the question, and the two put the loss in different places under different rules.

Wrong: When no response arrives, the terminal simply decides locally and moves on. **Right:** The issuer may well have approved and debited a shadow balance, so under online data capture a reversal must be sent regardless of what the terminal and the card finally decided, and skipping it is the classic way to produce duplicate debits during an outage.

Wrong: An online-only terminal never needs a Terminal Action Code – Default. **Right:** Every terminal reaches the default test eventually, and an online-only estate that never loaded one falls back to all zeros and, against a permissive card, approves offline in exactly the circumstances its operator assumed impossible.

16.99 Chapter summary in 20 lines

1. A meaningful fraction of card transactions in the world are approved without any message ever reaching the issuer, and much of the decision is made locally even when a message does travel.
2. Two organisations write risk policies for the same transaction: the acquirer's, loaded into the terminal, and the issuer's, carried on the card.
3. Each policy is three five-byte lists — denial, online and default — laid out bit for bit like the terminal's own record of what it observed.
4. That record is the Terminal Verification Results, five bytes of specific findings, most of which stay clear on most transactions and some of which are neutral observations rather than faults.
5. Terminal Action Analysis runs three comparisons in a fixed order, each ORing the two matching action codes and ANDing the result with those observations.
6. Denial comes first and produces an offline decline with no online attempt; online comes second, and only where the terminal has online capability; default comes third, for an offline-only terminal or one that cannot reach the network.
7. Because the two policies are ORed before the comparison, neither party can relax what the other has demanded, and there is no negotiation, no priority and no override.
8. The defaults for absent codes are asymmetric — a missing issuer online or default list counts as all ones, a missing terminal action code as all zeros — and Book 4 separately recommends that the four data-authentication failure bits be set in the terminal's online and default codes.
9. The observations are not a report but an input, since their five bytes go into the command that asks for a cryptogram, so a terminal that sets a bit after sending them produces a cryptogram the issuer cannot verify.
10. They are also a claim the terminal makes about its own behaviour, whereas the card's account of itself, the Card Verification Results inside the Issuer Application Data, is sealed by the cryptogram.
11. Whatever the terminal requests, Card Action Analysis lets the chip return the same type or a more restrictive one and never a less restrictive one, so there are two risk managers and not one.
12. Terminal risk management sets the decisive bits: floor limit checking, random selection biased towards amounts near the limit, velocity checking on the gap between the transaction counter and the Last Online ATC Register, and the exception file.
13. In the worked example a £42.50 basket at a till with a £30 floor limit goes online on the floor-limit bit alone, and the identical basket with the line dead is declined by the default comparison — same card, same till, a different list consulted.
14. "Floor limit" is really several limits owned by different parties, and the card's two consecutive-offline limits are preferences for terminals that do and do not have online capability rather than a soft and hard pair.
15. Terminal Type, one byte of operational control and environment, is why a supermarket till, a fuel pump, an aircraft seatback and a transit gate behave differently.
16. A fuel pump's difficulty is not connectivity but that the amount is unknown when the decision must be made, so the cryptogram covers an estimate and every scheme wraps such devices in rules of its own.
17. An offline-only terminal never runs the online comparison at all, which is exactly why the issuer's default list falls back to all ones and why a card carrying none is refused by an aircraft for the smallest observation.
18. Offline approval is cryptographically as strong as an online one and commercially riskier, because nobody with a balance in front of them has looked at it and the only controls are the card's own.
19. Store-and-forward is a deferral rather than a decision, and the three shapes of network failure have three correct responses, one of which makes a reversal mandatory whatever the terminal and card

decided locally.

20. There is no single authority in a card transaction — a terminal with an acquirer's policy, a card with an issuer's policy, a merged bitmap neither party can see in full, and a network that is sometimes there — and everything that looks arbitrary about card acceptance falls out of that arrangement.

Sources: EMV Integrated Circuit Card Specifications for Payment Systems, Book 3 (Application Specification) and Book 4 (Cardholder, Attendant, and Acquirer Interface Requirements), EMVCo; EMV Contactless Specifications for Payment Systems, Book C-2 (Kernel 2) and Book C-5 (Kernel 5), EMVCo; Visa Core Rules and Visa Product and Service Rules; Mastercard Switch Rules and Mastercard Developers authorisation message documentation; Commission Delegated Regulation (EU) 2018/389; UK Finance and Financial Conduct Authority guidance on contactless limits; ISO 8583 and ISO 18245.

17

CHAPTER

Proving It Is You

17.0 What this chapter gives you

1. You will be able to explain why the Application Cryptogram is a proof and the three bytes of tag `9F34` are only a claim, and why that difference is where every serious EMV attack has landed.
2. You will be able to walk a CVM List from the top for a given amount, terminal and currency, and say which rule wins and why nothing below it is ever read.
3. You will be able to decode a real tag `8E` object byte by byte: amounts X and Y, the method codes, the condition codes and the continuation bit in byte 1.
4. You will be able to tell offline PIN from online PIN by their key hierarchy, block format, comparator, counter and liability, instead of calling both of them "PIN".
5. You will be able to build an ISO 9564-1 format 0 PIN block by hand and say exactly which twelve PAN digits go into the PAN block.
6. You will be able to diagnose a `6985` on every VERIFY, a terminal that never prompts for a PIN, and an online PIN rejected across a whole estate.
7. You will be able to explain why three wrong guesses at a supermarket till lock the chip while the same card still works at a cash machine.
8. You will be able to say why the schemes could drop signature in 2018 without losing any evidence at all.
9. You will be able to explain why a phone pays for a £400 television at a reader that refuses a £120 plastic card.
10. You will be able to read a contactless trace kernel-aware, so that tag `9F66` does not decode as confident nonsense.

The previous chapter was about proving the card is real. This one is about proving *you* are.

They are different problems solved by different machinery, which is the first thing to get straight. The Application Cryptogram is eight bytes of mathematics that a chip and an issuer host compute independently and compare: either the bytes match or they do not. Cardholder verification has no such property. It produces three bytes, tag `9F34`, and those three bytes are not a computation. They are a *statement*, written by the terminal, about what the terminal believes happened at the keypad. Nobody in the chain is obliged to check whether the statement is true, and for most of the last thirty years nobody did.

Once you see that cardholder verification is a claim rather than a proof, the rest follows: why the card publishes a wish list rather than a rule, why the terminal gets the casting vote, why signature survived for decades without anyone ever comparing two signatures, why the two great EMV attacks of 2010

and 2020 both struck this exact seam, and why the industry's answer was not to fix the seam but to move verification onto a device the criminal does not have.

The plain version

Imagine a members' club with a door.

The club has a rule about identification, but it is not a single rule. It is a list, carried by every member, in order of preference. The first entry the doorman can actually perform, under the circumstances that apply tonight, is the one that gets used. Nothing further down the list is consulted.

A member's card might read like this:

1. Ask the bank to check my four-digit number. Use this one only at cash machines.
2. Ask my own chip to check my four-digit number, sending it in a sealed envelope. Use this whenever the door can do it.
3. Ask my own chip to check my four-digit number, sent openly. Use this whenever the door can do it.
4. Take my signature. Use this whenever the door can do it.
5. Let me in with nothing. Use this only if it is not a cash transaction.

The doorman has his own list, much shorter: it says what he is equipped to do. Some doors have a keypad and some do not. Some have a sealed-envelope machine and some do not. Some can only take a signature.

The whole of cardholder verification is these two lists being walked down together, top to bottom, until they touch.

A worked example, in pounds

You are buying £47.50 of groceries in a Leeds supermarket. Attended till, keypad, full capability. Your card's list has two amounts written at the top of it, which the conditions can refer to: **X is £30.00** and **Y is £100.00**. Neither matters here, because none of your card's five rules happen to mention them, but they are there.

Walk the list.

Rule 1 says *ask the bank, but only at an unattended cash machine.* This is a supermarket till. The condition is not met, so the rule is not applicable. Move on. Note carefully: this is not a failure. The rule simply does not apply tonight.

Rule 2 says *ask the chip, sealed, whenever the door can do it.* Can this till do it? Yes: it has an encrypting keypad and knows how to seal a number for a chip. Condition met, method supported. **This rule wins.** The till stops walking. It asks you for four digits, seals them, hands them to the chip, and the chip says yes or no.

Rules 3, 4 and 5 are never read. The signature line is never reached. This is why two cards in the same shop, on the same day, for the same amount, can behave completely differently: they carry different lists.

Now change one thing. Suppose the till is old and cannot do the sealed-envelope trick. Rule 2 becomes inapplicable and rule 3 wins: same digits, same chip, but sent in the open, which is fine because the wire between keypad and chip is a few centimetres long and inside a tamper-resistant box.

Change one more thing. Suppose the till has no keypad, only a screen and a printer. Rules 2 and 3 both drop out and rule 4 wins: signature. And if the till cannot even print, rule 5 wins: you are let in with nothing at all, which sounds alarming until you remember it is exactly what happens every time you tap for a coffee.

Four ways to prove it, and a fifth

There are, in the classical scheme, four families.

The chip checks the PIN. Your card stores your PIN. Not your bank: your card. The till sends the digits to the chip over the contacts, the chip compares them to what it holds, and answers. The bank is not involved and does not need to be, which is why this method exists: it works when the shop's line to the bank is down, on an aeroplane, in a car park machine at three in the morning. It comes in two flavours, one where the digits travel in the open between keypad and chip and one where they are sealed with the chip's own public key first.

The bank checks the PIN. The chip is not involved at all. The till locks your digits into a box that only the bank's cryptographic hardware can open and sends it with the authorisation request. This is what happens at every cash machine and at most tills outside Europe.

Signature. The till prints a slip, you sign it, and somebody is theoretically supposed to compare it to the strip on the back of the card.

Nothing. The card and the terminal agree, in writing, that this transaction needs no verification.

And the fifth, which arrived with the phone: **the device checks you.** Your phone or watch takes your face, fingerprint or passcode, satisfies itself, and tells the till that verification has already been done. The till never sees anything. Nor, importantly, does the bank.

Three strikes

The chip keeps a counter, usually starting at three. Every wrong PIN takes one off it; a correct PIN puts it back. At zero the chip refuses to check any more PINs, permanently, until the bank sends a specific instruction to reset it — which it can only do while you are online, which is why "put your card in a cash machine and try again" is genuinely useful advice.

The counter lives on the card, not on the account. Three wrong guesses at a supermarket till will lock the chip, and yet the same card will still work at a cash machine using the bank-checks-the-PIN route. Two counters, two places.

The box the PIN travels in

Whenever a PIN moves anywhere, it moves inside a fixed-size scrambled box: sixteen hexadecimal characters, eight bytes, containing in order a digit saying which recipe was used, a digit saying how many digits the PIN has, the PIN itself, and filler.

For a PIN of 1234 the raw box is `2 4 1234 FFFFFFFFFF`: recipe 2, four digits, one-two-three-four, ten `F`s of padding. That is the box used when the chip does the checking, and it is deliberately dull.

For a PIN going to the bank the box is built differently, and this is the clever bit: the account number is mixed in before it is locked. Two people with the same PIN on two different cards produce completely different boxes, so an attacker who collects a million boxes cannot spot that a hundred thousand of them are `1234`. The box for PIN 1234 on card 4321987654321098 7 is `0412AC89ABCDEF67`, and on any other card it would be something else entirely.

The three-byte receipt

When it is over, the till writes down three bytes: *what method was used*, *under what condition*, and *did it work*. `01 00 02` means "the chip checked a plaintext PIN, the condition was always, and it succeeded". That is the whole record, and the only record.

Here is the sentence to carry to dinner: **nothing in the system forces those three bytes to be true.** The till writes them. The card does not check them. The bank receives them and, historically, believed them.

Where the plain version stops being true

The list-walking picture is correct as far as it goes. Four things it hides matter enormously.

The card proposes; the terminal disposes; and nobody audits

The CVM List is a *preference*, not an instruction. The terminal performs the walk, decides which rule was applicable, performs the method, and authors the result. The card contributes only the list.

Two consequences follow, and both have been exploited. First, if the terminal lies about the outcome, the card has no independent way of knowing. In the 2010 Cambridge attack (Murdoch, Drimer, Anderson and Bond, *Chip and PIN is Broken*, IEEE Symposium on Security and Privacy), a small device between card and terminal swallowed the VERIFY command and answered `9000` — success — on the card's behalf. The terminal recorded a successful PIN. The card, never having been asked, recorded that no PIN had been performed. Both statements went to the issuer in the same message, and issuers were not comparing them.

Second, the result *does* often get bound into the cryptogram — tag `9F34` appears in most CDOL1s — and that binding creates a false sense of safety. The cryptogram authenticates the terminal's claim. It does not make the claim true.

There is also a switch that turns the whole subject off. The Application Interchange Profile, tag `82`, has a bit meaning "cardholder verification is supported". If it is clear, the terminal does not walk the list at all, whatever the list says.

Offline PIN and online PIN are not two flavours of one thing

Calling them both "PIN" hides that they share almost nothing.

Offline PIN travels a few centimetres over the card contacts, is protected either not at all or by the card's own RSA or elliptic-curve public key, is compared by the chip, is counted by the chip, and is never seen by the issuer. Online PIN travels thousands of kilometres, is protected by symmetric keys shared between terminal, acquirer, switch and issuer, is compared inside a hardware security module in a data centre, is counted by the issuer's host, and is never seen by the card.

Different PIN block formats, different key hierarchies, different failure modes, different counters, and in most markets different liability. A terminal estate can support one and not the other; an issuer can permit one and not the other. Treating them as interchangeable is the commonest conceptual error in this area.

The list barely applies to the transaction you actually did today

Amounts X and Y are expressed in the *application* currency, and conditions `'06'` through `'09'` are explicitly conditioned on the transaction being in that currency. Present a sterling-denominated card at a euro-denominated terminal and every amount-based rule silently evaporates. That is not an edge case; it is every foreign holiday.

More fundamentally, most cardholder verification decisions in a modern economy are contactless, and contactless decisions are dominated not by tag `8E` but by limits configured in the reader and by kernel-specific data objects. When your card asks for a PIN after a run of taps, that decision came from counters inside the card application and from reader limits, not from the walk described above.

Signature did not die of insecurity. It died of pointlessness

The protocol's definition of a successful signature CVM is not "the signatures matched", nor even "a signature was obtained". It is, effectively, "the terminal is capable of handling signature". The capability bit is set, the condition is met, the method is deemed performed, and the result byte is set to successful. There is no comparison step anywhere in EMV, because comparison is a human act the specification cannot describe.

Signature was therefore always a null CVM dressed as a real one, and what killed it was the schemes noticing that it created queues, receipts and disputes while providing no evidence. Note the mirror-image error too: "No CVM required" is not the absence of security. It is a deliberate, listed, cryptogram-covered choice by the issuer, appropriate below a threshold the issuer picked.

The technical version

The objects involved

Tag	Name	Length	Source
`82`	Application Interchange Profile	2	ICC
`8E`	Cardholder Verification Method (CVM) List	10 to 252	ICC
`95`	Terminal Verification Results (TVR)	5	Terminal
`99`	Transaction PIN Data	var.	Terminal
`9F17`	PIN Try Counter	1	ICC
`9F2D, 9F2E, 9F2F`	ICC PIN Encipherment Public Key certificate, exponent, remainder	var.	ICC
`9F33`	Terminal Capabilities	3	Terminal
`9F34`	Cardholder Verification Method (CVM) Results	3	Terminal
`9F35`	Terminal Type	1	Terminal
`9F66`	Terminal Transaction Qualifiers (kernels 3, 6, 7)	4	Terminal
`9F6C`	Card Transaction Qualifiers (kernel 3)	2	ICC

The relevant Application Interchange Profile bits, from EMV 4.4 Book 3 Annex C1, are byte 1 bit 5 (`0x10`), "Cardholder verification is supported", and byte 1 bit 2 (`0x02`), "On device cardholder verification is supported". That second bit was reserved for the contactless specifications in EMV 4.3 and only became a named contact-specification bit in 4.4. Byte 2 carries `0x80` "Contactless EMV mode supported", `0x40` "Mobile phone", `0x20` "Contactless transaction" and `0x01` "Relay Resistance Protocol supported".

The CVM List, tag '8E'

The value field is a fixed eight-byte header followed by a variable number of two-byte Cardholder Verification Rules:

Field	Length	Format
Amount X	4	binary
Amount Y	4	binary
CV Rule 1 … CV Rule n	2n	binary

Both amounts are binary integers in the minor units of the application currency, with the decimal point implied by the Application Currency Exponent, tag `9F44`. A list is always at least ten bytes and always an even length; anything else is malformed.

CV Rule byte 1 carries the method in bits 6 to 1, a continuation flag in bit 7, and a reserved bit 8. Bit 7 set means *apply the succeeding CV Rule if this CVM is unsuccessful*; bit 7 clear means *fail cardholder verification if this CVM is unsuccessful*. The method codes, from EMV 4.4 Book 3 Annex C3 Table 43:

Code	Meaning
`'00'`	Fail CVM processing
`'01'`	Plaintext PIN verification performed by ICC
`'02'`	Enciphered PIN verified online
`'03'`	Plaintext PIN verification performed by ICC and signature (paper)
`'04'`	Enciphered PIN verification performed by ICC
`'05'`	Enciphered PIN verification performed by ICC and signature (paper)
`'06'–'0F'`	Biometric CVMs, in pairs: facial `'06'`/`'07'`, finger `'08'`/`'09'`, palm `'0A'`/`'0B'`, iris `'0C'`/`'0D'`, voice `'0E'`/`'0F'`, the even code verified offline by the ICC and the odd code verified online
`'1E'`	Signature (paper)
`'1F'`	No CVM required
`'20'–'2F'`	Reserved for use by the individual payment systems
`'30'–'3E'`	Reserved for use by the issuer
`'3F'`	Not available for use

The biometric codes `'06'` to `'0F'` are an EMV 4.4 addition. In 4.3 that whole range was reserved. If your parser labels `'08'` as RFU, it is a 4.3 parser.

CV Rule byte 2 is the condition:

Value	Condition
`'00'`	Always
`'01'`	If unattended cash
`'02'`	If not unattended cash and not manual cash and not purchase with cashback
`'03'`	If terminal supports the CVM
`'04'`	If manual cash

Value	Condition
'05'	If purchase with cashback
'06'	If transaction is in the application currency and is under X value
'07'	If transaction is in the application currency and is over X value
'08'	If transaction is in the application currency and is under Y value
'09'	If transaction is in the application currency and is over Y value
'0A'–'7F'	RFU
'80'–'FF'	Reserved for use by the individual payment systems

Only condition '03' explicitly tests terminal support, but the terminal must skip any rule whose method it cannot perform regardless, so '03' is the condition you will see on almost every rule that is not amount- or environment-specific.

Decoding a real list

Take this object:

```
8E 12 00 00 0B B8 00 00 27 10 42 01 44 03 41 03 5E 03 1F 02
```

Eighteen bytes of value: eight of header, ten of rules.

Amount X is 00000BB8 = 3000, which with a sterling application currency and exponent 2 is £30.00. Amount Y is 00002710 = 10000, or £100.00.

Rule	Byte 1	Bit 7	Method	Byte 2	Condition
1	42	set	'02' Enciphered PIN verified online	01	If unattended cash
2	44	set	'04' Enciphered PIN verification performed by ICC	03	If terminal supports the CVM
3	41	set	'01' Plaintext PIN verification performed by ICC	03	If terminal supports the CVM
4	5E	set	'1E' Signature (paper)	03	If terminal supports the CVM
5	1F	clear	'1F' No CVM required	02	If not unattended cash, manual cash or cashback

Now give the terminal a Terminal Capabilities of E0 F8 C8. Byte 1 E0 is manual key entry, magnetic stripe and IC with contacts. Byte 3 C8 is SDA, DDA and CDA. Byte 2, the CVM capability field, is F8:

Mask	Bit	Meaning
0x80	b8	Plaintext PIN for ICC verification
0x40	b7	Enciphered PIN for online verification
0x20	b6	Signature (paper)
0x10	b5	Enciphered PIN for offline verification (RSA ODE)
0x08	b4	No CVM required
0x04	b3	Online Biometric
0x02	b2	Offline Biometric
0x01	b1	Enciphered PIN for offline verification (ECC ODE)

F8 sets the top five: all four classical methods plus no-CVM, and no biometrics. Bits b3, b2 and b1 are EMV 4.4 definitions; in 4.3 they were RFU.

For a £47.50 attended purchase in sterling with Terminal Type '22' (operational control: merchant; environment: attended, offline with online capability), the walk goes: rule 1 skipped, condition not met; rule 2 applicable and supported, so **offline enciphered PIN is performed** and the walk stops. If the PIN matches, the terminal writes CVM Results 44 03 02. Note that byte 1 carries the *entire* CV Rule byte 1 that was applied, continuation bit included: 44, not 04.

The VERIFY command and offline plaintext PIN

Offline PIN is carried by the ISO/IEC 7816-4 VERIFY command: class '00', instruction '20', P1 '00'. P2 is the qualifier of the reference data, and EMV assigns two values: '80' for plaintext PIN and '88' for enciphered PIN. A complete plaintext VERIFY for PIN 1234 is:

```
00 20 00 80 08 24 12 34 FF FF FF FF FF
```

Lc is 08 and the data is an ISO 9564-1 format 2 PIN block: control field '2', length nibble '4', the four PIN digits, then 'F' padding to eight bytes. Format 2 exists precisely for this case. ISO describes it as being for local use with offline systems only, and it deliberately contains no account number: the card already knows its own PAN, and there is no benefit in diversifying a block that never leaves the reader.

Status words matter here more than usual:

SW1 SW2	Meaning
9000	PIN correct
63Cx	PIN incorrect; x attempts remain
6983	Authentication method blocked (try counter exhausted)
6984	Referenced data invalidated
6985	Conditions of use not satisfied

6985 is by far the commonest complaint from engineers writing their first EMV client, and it almost always means one of two things: the command sequence is wrong — VERIFY may be issued any time after Read Application Data and before terminal action analysis completes, not before GET PROCESSING OPTIONS — or the card does not support plaintext offline PIN and you should be sending '88'.

Read the PIN Try Counter *before* attempting VERIFY, with GET DATA:

```
80 CA 9F 17 00
```

If it returns zero, EMV requires the terminal not to attempt offline PIN, to set "PIN Try Limit exceeded" in the TVR, and to treat the method as unsuccessful — which, because bit 7 of the rule is usually set, sends the walk to the next rule rather than failing outright. Resetting the counter requires the issuer script command PIN CHANGE/UNBLOCK, instruction `'24'` under the proprietary class, delivered after online authorisation.

Offline enciphered PIN

The RSA form works as follows. The terminal issues GET CHALLENGE, class `'00'`, instruction `'84'`, P1 and P2 both `'00'`, and the card returns an eight-byte unpredictable number. The terminal then assembles a block for encipherment:

Field	Length	Value
Data Header	1	`'7F'`
PIN Block	8	ISO 9564-1 format 2 PIN block
ICC Unpredictable Number	8	The value returned by GET CHALLENGE
Random Pad Pattern	N(IC) - 17	Generated by the terminal

N(IC) is the byte length of the modulus of the key used, so the assembled block is exactly one RSA block. The key is the ICC PIN Encipherment Public Key, recovered from the certificate in tag `9F2D` with the exponent in `9F2E` and, where the modulus does not fit in the certificate, the remainder in `9F2F`. If the card carries no separate encipherment key, the ICC Public Key from `9F46`, `9F47` and `9F48` is used instead. The result goes in the data field of VERIFY with P2 `'88'` and Lc equal to N(IC). The card decrypts, checks that the recovered header is `'7F'`, checks that the recovered unpredictable number equals the one it issued, and only then parses the PIN block.

That unpredictable number is the entire replay defence. Without it, a recording of one enciphered VERIFY would work forever.

EMV 4.4 adds an elliptic-curve form, Offline Data Encipherment, signalled by Terminal Capabilities byte 2 bit 1. It has no random pad, because the ECC construction supplies its own freshness. EMVCo's worked example (document EMV-SWG-NJ24r95, version 1.0a, December 2022) gives the plaintext for PIN 1234 as:

```
Data Header               7F
PIN block (ISO format 2)  24 12 34 FF FF FF FF FF
ICC Unpredictable Number  4B 61 72 6D 65 6C 69 74
```

The terminal generates an ephemeral P-256 key pair, does Diffie-Hellman against the card's public key, derives four AES keys via CMAC and AES, encrypts in counter mode under K1, MACs under K2 with CMAC truncated to eight bytes, and sends the ephemeral public key's x-coordinate concatenated with the ciphertext. The card repeats the derivation with its private key, verifies the MAC, decrypts, checks the trailing unpredictable number and the leading `'7F'`, and — in EMVCo's own words — parses the eight bytes following the header as an ISO-2 PIN block.

Online PIN and PIN block formats

Online PIN never touches the card. It is captured on a PCI PTS POI approved device, formatted into a PIN block, encrypted under a key shared with the acquirer, and carried in the authorisation message. Between capture and the issuer's hardware security module it is translated — decrypted and re-encrypted under the next zone's key — inside HSMs at each hop, and must never exist in the clear outside a secure cryptographic device. That is the substance of the PCI PIN Security Requirements,

currently version 3.1 (March 2021), and of the PCI PTS POI Modular Security Requirements, currently version 7.0.

The formats are defined by ISO 9564-1. The first nibble of the block identifies the format.

Format	Construction	PAN mixed in	Notes
0	`0` | `L` | PIN | `F` padding, XORed with a PAN block	Yes	Equivalent to ANSI X9.8; the workhorse
1	`1` | `L` | PIN | random padding	No	For use where no PAN is available
2	`2` | `L` | PIN | `F` padding	No	Offline use with smart cards only
3	`3` | `L` | PIN | random `A`–`F` fill, XORed with a PAN block	Yes	Format 0 with random rather than fixed fill
4	`4` | `L` | PIN | `A` fill | random, combined with a PAN block under AES	Yes	16 bytes; introduced in ISO 9564-1:2017 for AES

Formats 0 to 3 are eight bytes, matching the 64-bit block size of TDEA. Format 4 is sixteen, matching AES.

Format 0 is worth building by hand once. For PIN 1234 and PAN 43219876543210987:

```
PIN block   0 4 1 2 3 4 F F F F F F F F F F
PAN block   0 0 0 0 9 8 7 6 5 4 3 2 1 0 9 8
XOR         0 4 1 2 A C 8 9 A B C D E F 6 7
```

The PAN block is four zero nibbles followed by the twelve rightmost PAN digits *excluding the check digit*. Getting that exclusion wrong is a classic bug: it produces a block that decrypts to garbage at the issuer and a `55` invalid-PIN response for every cardholder on the estate.

Format 4 is a different animal. The PIN block is `4` | length | PIN | fill digits `A` | sixteen random nibbles, making 32 nibbles. The PAN block is a length indicator (`'0'` to `'7'`, meaning PAN length 12 to 19), the PAN, and zero padding, also 32 nibbles. The PIN block is encrypted with AES, the result XORed with the PAN block, and that encrypted with AES again. For PIN 1234 on PAN 432198765432109870 the clear PIN block begins `44 12 34 AA AA AA AA AA` and the PAN block is `64321987654321098700000000000000`. Migration has been slow enough that PCI SSC published a dedicated information supplement on implementing it and suspended its own compliance dates.

How the issuer actually checks

The issuer's HSM decrypts the PIN block, extracts the PIN, and compares it. It does not store the PIN. There are two dominant methods, both older than EMV by decades.

IBM 3624 with offset. The HSM encrypts validation data — conventionally the PAN, padded to sixteen characters with a chosen pad character — under a PIN verification key, takes the leftmost four hexadecimal digits of the result, and maps them through a decimalisation table to produce the *natural PIN*. Because a natural PIN cannot be chosen by the cardholder, an *offset* is stored: the digit-by-digit modulo-10 difference between the chosen PIN and the natural PIN. At verification the HSM recomputes the natural PIN, applies the offset, and compares. The decimalisation table is a parameter, and a well-known family of attacks in the early 2000s exploited HSM APIs that let an attacker supply a degenerate one.

Visa PVV. A Transformed Security Parameter is built from the eleven rightmost PAN digits excluding the check digit, a single-digit PIN Verification Key Indicator in the range 0 to 6, and the four PIN digits: sixteen digits, eight bytes. That is encrypted under the PIN verification key pair and the result scanned left to right for decimal digits; if four are not found, the scan repeats over the remaining digits with ten subtracted. The four-digit PIN Verification Value is stored on the account.

Neither scheme stores a PIN and neither can recover one. Both store a four-digit residue, which is why an issuer can tell you your PIN is wrong but genuinely cannot tell you what it is.

The CVM Results, tag '9F34'

Three bytes, defined in EMV 4.4 Book 4 Annex A4 Table 33.

Byte	Content
1	CVM Performed: the CV Rule byte 1 that was applied, or `'3F'` meaning "CVM not performed"
2	CVM Condition: the CV Rule byte 2 that was applied
3	CVM Result: `'00'` unknown, `'01'` failed, `'02'` successful

Value `'3F'` in byte 1 is the interesting asymmetry. In a CV Rule it is explicitly "not available for use"; in CVM Results it is the legitimate way of saying nothing was done. A parser sharing one lookup table between rules and results will reject valid CVM Results.

EMVCo's own ECC worked example carries `9F34` as CDOL1 data with the value `01 00 02`: plaintext PIN verified by the ICC, condition Always, successful.

The Terminal Verification Results, byte 3

The whole of TVR byte 3 is cardholder verification. Under EMV 4.4:

Mask	Bit	Meaning
`0x80`	b8	Cardholder verification was not successful
`0x40`	b7	Unrecognised CVM
`0x20`	b6	PIN Try Limit exceeded
`0x10`	b5	PIN entry required and PIN pad not present or not working
`0x08`	b4	PIN entry required, PIN pad present, but PIN was not entered
`0x04`	b3	Online CVM captured
`0x02`	b2	Biometric required but Biometric capture device not working
`0x01`	b1	Biometric required, Biometric capture device present, but Biometric Subtype entry was bypassed

Bit 3 was named "Online PIN entered" in EMV 4.3 and renamed "Online CVM captured" in 4.4 to accommodate online biometrics. The name changed; the bit did not. An unrecognised CVM code

sets bit 7 and the method is treated as unsuccessful, which is how a card personalised with a scheme-proprietary code fares on a terminal that has never heard of it.

These bits are what issuer and terminal action codes actually test. An Issuer Action Code – Denial with byte 3 bit 8 set means "decline anything where cardholder verification failed", and that is where the CVM decision becomes a financial outcome.

Contactless: a different machine wearing the same words

The contactless world is not the contact world with an aerial. EMV Contactless Book B defines an Entry Point that selects a kernel, and Books C-1 through C-7 define kernels 1 to 7, each belonging to a payment system. Kernel 2 is Mastercard's; kernel 3 is Visa's. Book C-8, Kernel 8, was published by EMVCo in October 2022 as a common kernel licensable on the same royalty-free terms as the contact specification, with elliptic-curve card authentication and explicit support for biometric and mobile cardholder verification methods.

The tag collision problem. The most dangerous fact in contactless decoding is that the same tag means different things in different kernels. Tag `9F66` is the Terminal Transaction Qualifiers for Entry Point and for kernels 3, 6 and 7, but kernel 2 defines `9F66` as PUNATC(Track2). Tag `9F6B` is the Card CVM Limit in kernel 3 and Track 2 Data in kernel 2. Tag `9F6E` is the Form Factor Indicator in kernel 3, Third Party Data in kernel 2 and Enhanced Contactless Reader Capabilities in kernel 4. A decoder that does not know which kernel produced the trace will produce confident nonsense.

Kernel 3, the Visa model. The reader supplies Terminal Transaction Qualifiers, tag `9F66`, four bytes, usually through the PDOL:

Byte	Bit	Meaning
1	b8	MSD supported
1	b7	VSDC supported
1	b6	qVSDC supported
1	b5	EMV contact chip supported
1	b4	Offline-only reader
1	b3	Online PIN supported
1	b2	Signature supported
1	b1	Offline Data Authentication for Online Authorisations supported
2	b8	Online cryptogram required
2	b7	CVM required
2	b6	(Contact chip) Offline PIN supported
3	b8	Issuer Update Processing supported
3	b7	Consumer Device CVM supported

The card replies with Card Transaction Qualifiers, tag `9F6C`, two bytes:

Byte	Bit	Meaning
1	b8	Online PIN Required
1	b7	Signature Required

Byte	Bit	Meaning
1	b6	Go Online if Offline Data Authentication Fails and Reader is online capable
1	b5	Switch Interface if Offline Data Authentication fails and Reader supports VIS
1	b4	Go Online if Application Expired
1	b3	Switch Interface for Cash Transactions
1	b2	Switch Interface for Cashback Transactions
2	b8	Consumer Device CVM Performed
2	b7	Card supports Issuer Update Processing at the POS

That is the whole contactless CVM negotiation for Visa: two bits, one saying "ask for a PIN", one saying "the device already did it". The CVM List is not consulted.

Kernel 2, the Mastercard model. Here reader configuration drives the decision. Three tags matter, all six-byte numeric amounts:

Tag	Name
DF8124	Reader Contactless Transaction Limit (No On-device CVM)
DF8125	Reader Contactless Transaction Limit (On-device CVM)
DF8126	Reader CVM Required Limit

The kernel instantiates a single working Reader Contactless Transaction Limit from DF8125 if the card indicates on-device cardholder verification is supported and from DF8124 otherwise — which is exactly why a phone can pay for a £400 television at a reader that will refuse a £120 plastic card. Separately, if the amount reaches DF8126, the kernel instantiates the CVM capability field of Terminal Capabilities with "CVM Required", forcing a method to be selected.

The card's own counters do the rest: consecutive-transaction and cumulative-amount counters live in the card application, not the terminal, and when they trip the card asks for verification. That is why the PIN prompt appears at an apparently random shop.

Consumer Device CVM

CDCVM is what happens when the phone, watch or laptop verifies you and the terminal takes its word. EMVCo defines it as authentication performed on the consumer's own device: passcode, password, pattern, or a biometric such as fingerprint, iris, voice or facial recognition.

Mechanically, in kernel 3 the device sets Card Transaction Qualifiers byte 2 bit 8, "Consumer Device CVM Performed", and the reader stops asking. In kernel 2 the device signals on-device CVM support and the reader promotes itself to the higher limit. In both cases the terminal performs no verification whatsoever.

The security consequence is precisely stated by EMVCo's own CDCVM task force chair: *unlike an online PIN, a CDCVM cannot be seen by the issuer.* The issuer receives an assertion that some unspecified

mechanism, on some unspecified device, from some unspecified manufacturer, succeeded. Hence three pieces of EMVCo scaffolding: the **EMV CDCVM Security Requirements**, a **Security Evaluation Process** for software-based mobile payment, and an **EMV CDCVM Solution ID and Database** assigning each registered solution a short identifier plus metadata — name, provider, type, operating system — so an issuer can at least know *which* CDCVM it is being asked to trust. EMVCo has also worked with GlobalPlatform on the Secure Element Broker Interface and with the FIDO Alliance since 2016 so that FIDO's biometric certification covers EMVCo's performance requirements.

The regulatory position is more generous than the technical one. Under the PSD2 regulatory technical standards, Commission Delegated Regulation (EU) 2018/389, Article 11 permits contactless payments without strong customer authentication where the individual amount does not exceed EUR 50, the cumulative amount since the last strong authentication does not exceed EUR 150, or the number of consecutive contactless transactions since the last strong authentication does not exceed five. CDCVM satisfies strong customer authentication in its own right, which is why digital wallets have never been subject to those caps.

In the United Kingdom the corresponding caps were £100 per transaction and £300 cumulative, in force from 15 October 2021. From **19 March 2026** the Financial Conduct Authority removed both, allowing firms with sufficiently strong fraud controls to set their own limits or none at all; in practice the major banks retained £100 when the rules changed, and raising it also requires changes at acceptance terminals and in the industry rules that govern them. Mobile wallets were never inside that regime.

The death of signature, precisely

Mastercard announced in October 2017 that from April 2018 merchants in the United States and Canada would no longer be required to obtain a cardholder signature for in-store credit and debit transactions, and confirmed in 2018 that from April 2019 issuers globally would no longer be required to include a signature panel on Mastercard products. Visa, American Express and Discover made comparable changes for North America over the same period.

It could be dropped so cleanly because it was never verified in the protocol. CVM code `'1E'` under condition `'03'` asks only whether the terminal supports signature; if the capability bit is set, the method is applicable, the terminal obtains the signature and reports success. There is no field for a comparison, no field for its result, and no obligation on anyone to perform one. Removing signature removed a receipt, a queue and a dispute category. It removed no evidence, because there was none.

What the attacks actually taught

Three pieces of work define the state of understanding.

Chip and PIN is Broken (Murdoch, Drimer, Anderson and Bond, IEEE S&P 2010) inserted a wedge between card and terminal that intercepted VERIFY and returned `9000` without forwarding it. The terminal recorded a successful offline PIN in `9F34`. The card, never asked, recorded in its own Card Verification Results that offline PIN verification had not been performed. Both went to the issuer in the same authorisation message. Issuers comparing the terminal's claim with the card's own account would have caught it; most did not.

The EMV Standard: Break, Fix, Verify (Basin, Sasse and Toro-Pozo, IEEE S&P 2021) did the same to contactless. A man-in-the-middle Android application modified the Card Transaction Qualifiers in flight: clearing byte 1 bit 8, so the terminal no longer believed online PIN was required, and setting byte 2 bit 8, so it believed the Consumer Device CVM had been performed. Demonstrated on real terminals with Visa Credit, Visa Electron and V Pay cards. The CTQ is not integrity-protected, which is the whole of the vulnerability.

Card Brand Mixup (same authors, USENIX Security 2021) made a Mastercard card behave as a Visa transaction so that the Visa PIN bypass applied. It was demonstrated on Mastercard credit, Mastercard debit and Maestro cards, including a transaction above 400 US dollars. Mastercard's response was to check at authorisation that the Application Identifier is consistent with the card brand.

In every case the cryptogram was valid and the card genuine. What failed was that the record of *who was standing there* is authored by one party rather than agreed between two, and is not cryptographically bound to the act it describes.

A short field guide to failures

Symptom	Usual cause
6985 on every VERIFY	Command out of sequence, or card does not support that flavour of PIN
6984 or 6983 immediately	PIN Try Counter already at zero; read 9F17 first
Terminal never prompts for PIN	AIP byte 1 bit 5 clear, or no rule's method is in Terminal Capabilities byte 2
CVM works at home, not abroad	Conditions '06'–'09' inert because transaction currency differs from application currency
Parser rejects a valid 9F34	Byte 1 value '3F' treated as an invalid CV Rule instead of "CVM not performed"
9F66 decodes as nonsense	Kernel 2 trace parsed with a kernel 3 tag table
Phone allowed above the card limit	DF8125 in force rather than DF8124; working as designed
Enciphered offline PIN always fails	ICC Unpredictable Number not echoed byte-for-byte, or wrong public key
Online PIN rejected across an estate	PAN block built from the wrong twelve digits, or check digit not excluded
Biometric CVM codes shown as RFU	EMV 4.3 tables in use; codes '06'–'0F' are 4.4

The chapter in one paragraph

The card carries an ordered wish list, tag 8E, of ways it will accept being satisfied that you are you, with two amounts and a condition attached to each entry. The terminal walks that list from the top, skipping entries whose condition is unmet or whose method it cannot perform, and stops at the first one it can do. If the method is offline PIN, the digits go to the chip in an ISO 9564-1 format 2 block inside a VERIFY command and the chip answers, decrementing its own try counter on failure. If the method is online PIN, the digits never touch the chip; they go into a format 0 or format 4 block, encrypted under keys shared with the acquirer, and are checked in a hardware security module against a stored offset or PIN verification value. If the method is signature, nothing is verified at all. Whatever happens, the terminal writes three bytes — method, condition, outcome — and forwards them. Contactless replaces the walk with reader limits and two bits in a kernel-specific object, and the phone replaces the whole exercise with a face the terminal cannot see and the issuer cannot verify. The eight bytes of the cryptogram are proof. These three bytes are testimony. Every serious attack on EMV in the last two decades has been an attack on the difference.

17.98 Common wrong ideas

Wrong: the card decides how you are verified. **Right:** the card publishes an ordered preference in tag 8E; the terminal performs the walk, decides which rule was applicable, performs the method and authors the result.

Wrong: because tag 9F34 is bound into the cryptogram, its contents must be true. **Right:** the cryptogram authenticates the terminal's claim without making the claim true, which is exactly what the 2010 Cambridge wedge exploited.

Wrong: offline PIN and online PIN are two flavours of one thing. **Right:** they share almost nothing — different block formats, different key hierarchies, different comparators, different counters and in most markets different liability.

Wrong: three wrong PINs lock the account. **Right:** the try counter lives on the card, so the chip refuses further offline PIN checks while the same card still works at a cash machine on the online route.

Wrong: signature was dropped because it was insecure. **Right:** code '1E' under condition '03' asks only whether the terminal supports signature; there is no comparison step anywhere in EMV, so removing it removed a queue and a receipt but no evidence.

Wrong: "No CVM required" is the absence of security. **Right:** it is a deliberate, listed, condition-bound and cryptogram-covered choice by the issuer, appropriate below a threshold the issuer picked.

Wrong: the amounts X and Y govern your spending wherever you are. **Right:** conditions '06' to '09' only apply when the transaction is in the application currency, so every amount-based rule silently evaporates on a foreign holiday.

Wrong: the PIN prompt after a run of taps came from walking the CVM List. **Right:** it came from counters inside the card application and from limits configured in the reader; kernel 3 settles the question with two bits and kernel 2 with DF8124, DF8125 and DF8126.

Wrong: a value of '3F' in tag 9F34 byte 1 is invalid. **Right:** '3F' is "not available for use" in a CV Rule but is the legitimate way of saying "CVM not performed" in CVM Results, and a parser sharing one table between the two will reject valid data.

Wrong: Consumer Device CVM gives the issuer what an online PIN gives it. **Right:** a CDCVM cannot be seen by the issuer, which receives only an assertion — hence EMVCo's security requirements, evaluation process and solution database so an issuer at least knows which mechanism it is trusting.

17.99 Chapter summary in 20 lines

1. Proving the card is genuine and proving the cardholder is entitled are separate problems solved by different machinery.
2. The Application Cryptogram is eight bytes computed independently by chip and issuer and compared; cardholder verification produces three bytes of terminal-authored testimony in tag 9F34.
3. The card carries an ordered wish list, the CVM List at tag 8E, of the ways it will accept being satisfied that you are you.
4. That list opens with two amounts, X and Y, in the minor units of the application currency, followed by two-byte rules pairing a method with a condition.
5. The terminal walks the list from the top, skipping any rule whose condition is unmet or whose method it cannot perform, and stops at the first one it can do.
6. Nothing below the winning rule is ever read, which is why two cards in the same shop for the same amount can behave completely differently.
7. Byte 1 of a rule carries the method and a continuation bit deciding whether failure moves to the next rule or fails cardholder verification outright.
8. There are four classical families — offline PIN checked by the chip, online PIN checked by the issuer, signature, and no CVM — plus the biometric codes added in EMV 4.4 and, in practice, the device.

9. Offline PIN travels a few centimetres in an ISO 9564-1 format 2 block inside a VERIFY command and is compared by the chip against a counter the chip itself keeps.
10. Online PIN never touches the chip: it goes into a format 0 or format 4 block, is translated inside HSMs at every hop, and is checked in the issuer's own hardware security module.
11. Issuers store no PIN at all, only an IBM 3624 offset or a Visa PIN Verification Value, which is why they can tell you the PIN is wrong but genuinely cannot tell you what it is.
12. The PIN Try Counter lives on the card rather than the account, and only an issuer script delivered after online authorisation can reset it.
13. Enciphered offline PIN binds the card's own unpredictable number into the encrypted block, and that number is the entire replay defence.
14. Signature was never verified anywhere in the protocol: success is declared on terminal capability alone, so it was always a null CVM dressed as a real one.
15. That is why it could be dropped from 2018 onwards so cleanly — removing it removed a receipt, a queue and a dispute category, and no evidence.
16. Whatever happens, the terminal writes down the rule applied, the condition applied and the outcome, and nothing in the system forces those three bytes to be true.
17. The 2010 Cambridge wedge answered VERIFY on the card's behalf and the 2021 contactless work rewrote the Card Transaction Qualifiers in flight; in both the cryptogram was valid and the card genuine.
18. Contactless replaces the walk entirely, with reader limits in kernel 2 and two bits of a kernel-specific object in kernel 3.
19. Consumer Device CVM moves verification onto a device the terminal cannot see and the issuer cannot verify, which is why wallets were never inside the contactless caps.
20. Every serious attack on EMV in two decades has been an attack on the gap between eight bytes that are proof and three bytes that are only testimony.

Sources: EMV ICC Specifications for Payment Systems v4.4, Book 3 sections 6.5 and 10.5 and Annexes C1 and C3, Book 4 Annexes A1 and A4, and Book 2 on PIN encipherment; EMV Contactless Books B, C-2, C-3 and C-8; EMVCo document EMV-SWG-NJ24r95 v1.0a, December 2022, for the ECC worked examples; EMVCo on CDCVM, October 2020; ISO 9564-1:2017 and ISO/IEC 7816-4; PCI PIN Security Requirements v3.1 and PCI PTS POI v7.0; Commission Delegated Regulation (EU) 2018/389 Article 11; FCA and UK Finance contactless guidance, March 2026; Mastercard newsroom, 2017 and 2018; Murdoch, Drimer, Anderson and Bond (IEEE S&P 2010) and Basin, Sasse and Toro-Pozo (IEEE S&P 2021, USENIX Security 2021). Bit tables cross-checked against openemv/emv-utils, lumag/emv-tools, CardContact OpenSCDP and emvlab.org.

18

CHAPTER

Contactless

18.0 What this chapter gives you

1. You will be able to explain where a card with no battery gets its power, and why the honest description is mutual inductance in the near field rather than radio reception.
2. You will be able to say why a card works at two centimetres and does nothing at fifteen, and why that short range is a power-budget consequence and not a privacy guarantee.
3. You will be able to describe how a card that cannot transmit still answers, by shorting a load across its own coil and letting the reader read the dips in its own field.
4. You will be able to name every layer a single tap traverses, from ISO/IEC 14443-2 up through T=CL, ISO/IEC 7816-4 APDUs, the EMV entry point and the kernel books.
5. You will be able to explain why the tap limit is a lost-and-stolen control with no relationship to the radio, and name the three independent places that enforce it.
6. You will be able to trace the UK limit from £10 in 2007 to £100 and £300 in 2021 and to the removal of the mandated caps on 19 March 2026, and say what "the limit" now means.
7. You will be able to separate eavesdropping, which reaches metres and yields a transaction bound to one amount and one counter, from relaying, which creates a new transaction.
8. You will be able to explain why a Frame Waiting Time of nearly five seconds leaves the protocol no obstacle whatever to a relay.
9. You will be able to say what the Relay Resistance Protocol measures, why both entropies end up inside the cryptogram, and why it still works less well than the design suggests.
10. You will be able to explain why a ticket gate cannot ask your bank, what it verifies before it opens, and why your statement shows one number, a day late, matching no fare on any poster.

There is a moment, at the till, when nothing appears to happen. You hold a card near a lit square of plastic, the square beeps, and you walk away. No slot, no keypad, no paper. The whole event is shorter than the sentence describing it.

That moment is the most heavily specified two-thirds of a second in retail. It involves an international standard for radio, a second layer for how bytes are framed on that radio, a third for how smart cards talk at all, a fourth from EMVCo describing which of eight payment kernels should wake up, and a fifth of scheme rules and regulatory technical standards deciding whether you should have been asked for a PIN. Almost all of it is invisible, and almost all of the public understanding of it is wrong in the same three ways.

This chapter is about all five layers, in order, from the magnetic field upward.

The plain version

Start with the card. It has no battery. Cut one open and you will find a chip the size of a grain of rice and a loop of very thin wire, or a printed spiral of aluminium, running around the inside edge of the plastic. That is the whole machine. Nowhere for a battery to go.

So where does the power come from? The reader.

Think of a guitar. Pluck a string on one guitar, hold a second guitar close by, and the matching string on the second guitar starts to hum on its own. Nobody touched it. The energy came across the air, and the second string was built to be sympathetic to exactly that note.

The reader in the shop is the plucked string. It hums a note — not a sound, but a magnetic note, oscillating 13,560,000 times every second — constantly, all day, into empty air, whether or not anyone is there. The loop of wire in your card is the sympathetic string. Bring it into the hum and a small current begins to flow around the loop. That current is rectified, smoothed and fed to the chip, and the chip wakes up. It has no idea what time it is or how long it slept. It only knows it now has power.

Now the harder half. Once the card is awake, how does it answer?

You might assume it hums back. It cannot: it has barely enough power to run its own arithmetic, let alone to broadcast. So it does something stranger. It interferes with the reader's own hum.

Imagine standing at a window at night, looking at a lighthouse. You cannot signal back — you have no lamp. But if you could put your hand in front of the lighthouse's beam, and take it away, and put it back, in a rhythm, then someone standing next to the lighthouse would see its own light flicker. You would be talking using their light.

That is what the card does. It repeatedly shorts and unshorts a load across its own coil. Each time it does, it drags a tiny amount of energy out of the reader's field, and the reader — which constantly monitors how much energy its own field is costing it — sees a dip. Dip, no dip, dip, dip, no dip. That is the card talking. It is speaking in the reader's voice.

Why you have to hold it so close

The hum falls away brutally fast. Not gently, the way sound fades as you walk from a speaker, but savagely: double the distance and you have roughly an eighth of the strength. This is why a card works at two centimetres and does nothing at fifteen. Nobody imposed that rule. It is the shape of the physics.

A worked example: £8.40 for coffee

You order a flat white and a pastry. £8.40. The barista taps the total into the till and the till pushes the number to the reader, whose little screen lights up. You present your card. Roughly this happens, in this order.

The card enters the field and powers up. The reader has been calling out, in effect, "anyone there?" — repeatedly, several times a second, all day long. Your card answers with a short serial number so the reader can tell it apart from any other card in range. The reader picks it and says "you, then."

The reader asks for the card's list of payment applications, and picks the highest-priority one it also supports.

The reader hands over the facts of the sale: eight pounds forty, pounds sterling, United Kingdom, today's date, and a random number it has just invented and will never use again.

The card does its sum. Using a secret key that has never left the chip and can never be read out of it, it computes an eight-byte answer depending on the amount, the currency, the date, that random number, and a counter inside the card that goes up by one every time the card is used. It hands the answer back with the counter's current value.

The reader beeps. You leave. The reader sends that answer to your bank, and your bank — which holds the other copy of the secret key — does the identical sum and checks it got the identical answer.

Those eight bytes are the whole security model, and here is the point people miss: they are exactly the eight bytes you would get if you had pushed the card into the slot and typed your PIN. Same chip, same key, same maths. The radio changed nothing about the proof that the card is real.

So why is there a tap limit?

Because the proof that the *card* is real is not the same as the proof that *you* are entitled to use it.

When you type a PIN you are answering a second question. Contactless, below a threshold, skips that question for the sake of speed. So the threshold is not protecting you against anything happening in the air. It protects you against a much older and duller crime: someone taking your card out of your coat pocket and spending it in a newsagent before you notice.

The limit is therefore built as a leaky bucket. Two numbers, not one. In the United Kingdom, from 15 October 2021, the rules said: no single tap above £100, and no more than £300 of taps in total since the last time you actually proved who you were.

Work it through with a thief. They lift your card at 09:00 and buy £95 of gift cards. Running total: £95. Again: £190. Again: £285. On the fourth £95 attempt the running total would reach £380, so the answer comes back: not without a PIN. The thief does not have the PIN. The bucket has emptied, and it will not refill until somebody types four digits or inserts the card into a machine.

Three hundred pounds is the size of the hole the industry decided to leave open. Everything above it meets the same wall as always.

Two thieves and a very long wire

There is one attack the physics genuinely does invite, and it matters because it is the only one where the radio is the point.

Suppose two thieves. One stands near you on a train platform with a device in a bag; the other stands at a till a mile away. The two devices are connected by a phone call. When the shop's reader asks its question, the thief in the shop relays it down the line to the thief on the platform, whose device asks your pocket. Your card, which cannot see, answers honestly. The answer goes back down the phone and into the shop reader.

Nobody forged anything. Nobody broke any cryptography. The thieves simply made a very long extension lead between a real reader and a real card. This is a relay attack, and the defences are not about secrecy. They are about time and about the PIN — which is exactly why the tap limit matters more than people think.

Buses and barriers

Finally, the special case that breaks the whole model: the ticket gate.

A shop's reader can wait a second or two while the message goes to your bank and back. A ticket gate cannot. Behind you there are forty people walking at three miles an hour, and the gate has a few

hundred milliseconds to decide before somebody's nose meets a barrier. There is no time to ask your bank.

So it does not ask. It takes the card's cryptographic answer, checks it against a locally held list of cards known to be bad, opens the gate, and files the answer away. Then it does the same at your exit gate, and again tomorrow. At the end of the day a computer adds up everything you did, works out the cheapest legal price for that set of journeys — applying the daily cap, the weekly cap, the off-peak rates — and sends your bank a single request for one total.

This is why the entry on your statement is one number matching no fare on any poster, and why it arrives a day or two late. The gate was never charging you. It was collecting evidence.

Where the plain version stops being true

The card is not "powered by radio waves", and the distinction has consequences. The guitar analogy is closer than it sounds: this is resonant magnetic coupling in the near field, not radio reception. At 13.56 MHz a full wavelength is about 22 metres and the boundary between near field and far field sits at roughly 3.5 metres. Everything between card and reader happens deep inside that boundary, where energy is exchanged by mutual inductance between two coils rather than radiated away. That is why the working range is centimetres. But it also means the plain version's reassurance is wrong in a specific way: the range at which a card can be *powered and transacted with* is not the range at which it can be *overheard*. Radiated leakage is a separate phenomenon with a much longer reach, and published work has recovered the reader-to-card channel at several metres. Short range is not a privacy guarantee; it is a power-budget consequence.

The £100 was never a technical fact, and as of 2026 it is not a rule either. Nothing in any radio standard, chip specification or EMV kernel knows what a pound is. The limit lives in three places at once — configured amounts inside the terminal, counters inside the card chip, velocity rules inside the issuer's authorisation host — and it exists because a regulator wrote it down. The UK's £100 and £300 came from Article 11 of the onshored Strong Customer Authentication regulatory technical standards. On 19 December 2025 the Financial Conduct Authority announced it was removing those mandated caps, and the change took effect on 19 March 2026, leaving firms free to set their own limits provided they have adequate fraud controls. Most major UK issuers did not immediately move. So the correct statement in 2026 is not "the limit is £100" but "your issuer's limit is probably still £100, and it is now their choice."

"Contactless is less secure" and "contactless is exactly as secure" are both wrong. The cryptogram is identical; the interface changes nothing about proving the card is genuine. But the seam is real and it is not where people look. It is in cardholder verification: the data elements that tell a terminal *whether* verification happened are, in some scheme configurations, not covered by the cryptogram. That is a specific, documented, demonstrated flaw with a paper trail, and it has nothing to do with anyone reading your card through your trousers.

The gate has not taken an IOU. The transit picture implies the barrier trusts you and sorts it out later. It does not trust you. Before that gate opened it verified a digital signature generated by your card's own private key and obtained a full transaction cryptogram over data it supplied. What is deferred is not the proof but the *authorisation*, the question of whether your bank will honour it. Those are different things, they fail differently, and the transit operator carries the gap between them as a real financial risk with a name.

The technical version

The radio: ISO/IEC 14443

The physical and link layers of every contactless payment card in the world are defined by ISO/IEC 14443, a four-part standard originally titled *Identification cards — Contactless integrated circuit cards — Proximity cards* and retitled in later editions to *Cards and security devices for personal identification — Contactless proximity objects*. Part 1 covers physical characteristics; Part 2, radio frequency power and signal interface; Part 3, initialization and anticollision; Part 4, transmission protocol.

The standard's vocabulary matters, because EMV documents use it throughout. The reader is a **PCD**, a Proximity Coupling Device. The card is a **PICC**, a Proximity Integrated Circuit Card. Neither term says anything about payments; the same standard carries transport tickets, passports and door badges.

Part 2 sets the numbers everything else hangs from.

Parameter	Value
Carrier frequency, fc	13,56 MHz ± 7 kHz
Minimum operating field, Hmin (Class 1)	1,5 A/m rms
Maximum operating field, Hmax (Class 1)	7,5 A/m rms
Subcarrier frequency, fs	fc/16, approximately 847 kHz
Initialisation bit rate	fc/128, approximately 106 kbit/s
Higher bit rates	fc/64 approx. 212, fc/32 approx. 424, fc/16 approx. 848 kbit/s
Elementary time unit	1 etu = 128 / (D x fc); 9,44 µs at D = 1

The field-strength window is the most under-appreciated engineering constraint in the system. A card must work at 1,5 A/m and survive 7,5 A/m: five to one in field terms and considerably more in delivered power. The chip therefore carries a shunt regulator whose job at close range is to throw energy away as heat and whose job at long range is to run a public-key operation on almost nothing. Later editions of Part 2 define six PICC classes with different antenna sizes and correspondingly different windows, from Class 1 at 1,5 to 7,5 A/m up to Class 6 at 4,5 to 18 A/m.

The two signalling variants are not compatible and both are mandatory for payment readers.

Type A. Reader to card: amplitude shift keying at 100 % modulation depth — the field is switched fully off in short pauses — with modified Miller coding. Card to reader: on-off keying of the 847 kHz subcarrier, Manchester coded. The 100 % pauses mean the card loses power entirely during each pause and must ride through on a reservoir capacitor.

Type B. Reader to card: amplitude shift keying at roughly 10 %, with the modulation index specified as between 8 % and 14 %, NRZ-L coded. The field never drops out, so the card's supply is steadier. Card to reader: BPSK modulation of the same subcarrier.

In both directions the card's reply is **load modulation**. The card switches an impedance across its coil; the reader observes the reflected change in its own antenna's load, appearing as sidebands at fc ± fs. The card never generates a carrier of its own. This matters enormously to relay attacks and their defeat: the card's timing is not free-running but locked to the reader's carrier, because that carrier is also the card's clock.

Above Part 2 sits Part 3. For Type A, the reader continuously issues REQA (or WUPA for halted cards), a card in the field responds with ATQA, and a bitwise anticollision loop over the UID resolves any

collision, cascading through up to three levels for 4-, 7- or 10-byte UIDs; the reader then issues SELECT and receives SAK. Type B uses REQB/WUPB, ATQB and ATTRIB with a slot-marker scheme instead. Both terminate at the same place: one selected card, everything else told to be quiet. Two cards in a wallet do not "confuse" the reader in the way folklore suggests; they cause a collision the standard resolves deterministically, ending in a clean read or a "present one card only" prompt.

Part 4 defines the half-duplex block transmission protocol, universally called **T=CL**. The reader sends RATS (Request for Answer To Select); the card returns ATS, which declares its capabilities. Two of those set the shape of everything above.

The first is frame size. FSDI and FSCI index a fixed table: 16, 24, 32, 40, 48, 64, 96, 128 and 256 bytes. Anything longer must be chained.

The second is the **Frame Waiting Time**:

FWT = (256 x 16 / fc) x 2^FWI

with FWI in the range 0 to 14 and a maximum FWT of approximately 4 949 ms. If the card needs longer than the FWT it has declared, it sends S(WTX) with a Waiting Time eXtension Multiplier in the range 1 to 59, and the reader grants a temporarily extended FWT, capped at FWTMAX.

That maximum figure — nearly five seconds — should be held in mind. It is the reason the protocol, as designed, offers no obstacle whatever to a relay.

On top of T=CL, the card speaks ISO/IEC 7816-4 command and response APDUs, exactly as it would over the contacts. From the payment application's point of view the radio is a transport. On top of that sits EMV.

EMVCo layers its own requirements over ISO/IEC 14443 in the **EMV Level 1 Specifications for Payment Systems, EMV Contactless Interface Specification**, which reached Version 3.1 in December 2020. Specification Bulletin 245, announced by EMVCo on 2 June 2021, required all compliant contactless terminals to decode IQ modulation — to demodulate the in-phase and quadrature components of the card's load modulation rather than amplitude alone. The stated motivations were reliability, positioning tolerance and transit gate throughput, which tells you where the pressure comes from.

Power harvesting, in detail

The card's antenna is a planar coil, typically a small number of turns of etched or wire-embedded conductor running around the perimeter of an ID-1 card body of 85,60 mm x 53,98 mm. It forms a resonant tank with a capacitance that is partly on-chip and partly distributed, tuned so that the loaded resonance sits at or slightly above 13,56 MHz.

Inside the chip, four things happen in parallel from the moment the field arrives. A rectifier converts the induced AC into DC and charges a reservoir. A shunt regulator clamps the supply and dissipates surplus energy — the closer the card, the more it must waste. A clock extractor derives the system clock by dividing the 13,56 MHz carrier, which is why the card has no independent notion of elapsed time. A power-on-reset circuit holds the logic in reset until the supply is stable.

The consequences shape the payment protocol above. There is no clock across power cycles, so the card cannot enforce "no more than one transaction per minute" by itself. There is no battery-backed counter, so any anti-replay counter must live in non-volatile memory and survive the field being pulled away mid-write, which is why torn-transaction recovery is a named feature of contactless kernels. And there is a hard energy budget for asymmetric cryptography: an RSA or ECC operation at 1,5 A/m

is genuinely difficult, which is why contactless offline data authentication was optimised into a fast variant rather than reusing the contact flow unchanged.

The EMV layer: entry point and kernels

A contactless terminal begins not by selecting an application directly but by selecting the **Proximity Payment System Environment**, whose file name is the ASCII string `2PAY.SYS.DDF01` — the contactless counterpart of `1PAY.SYS.DDF01`. The response is a directory of the card's applications, each with an ADF Name (tag `4F`) and priority.

The terminal matches those against its own configured list, and the match determines which **kernel** processes the transaction. EMVCo publishes the architecture in *EMV Contactless Specifications for Payment Systems*: Book A for architecture and general requirements, Book B for the entry point, Books C-1 through C-8 for the individual kernels, and Book D for the contactless communication protocol.

Book	Kernel	Scheme
C-2	Kernel 2	Mastercard
C-3	Kernel 3	Visa
C-4	Kernel 4	American Express
C-5	Kernel 5	JCB
C-6	Kernel 6	Discover
C-7	Kernel 7	UnionPay
C-8	Kernel 8	EMVCo (scheme-neutral)

Book C-1 defines Kernel 1, built on the EMV common core definitions; the scheme mapping quoted for it in secondary sources is inconsistent and is omitted here rather than guessed. Book C-8, *Kernel 8 Specification*, Version 1.0, was published on 5 October 2022 to reduce the number of kernels the industry maintains; it specifies elliptic curve cryptography, a secure channel, biometric verification and optional on-card data storage.

The practical significance of the split is that "the contactless limit" is not one setting. It is a different named data object in each kernel. In Kernel 2 the terminal-side amounts are carried in proprietary tags in the `DF81xx` range:

Tag	Kernel 2 data object
`DF8123`	Reader Contactless Floor Limit
`DF8124`	Reader Contactless Transaction Limit (No On-device CVM)
`DF8125`	Reader Contactless Transaction Limit (On-device CVM)
`DF8126`	Reader CVM Required Limit
`DF811B`	Kernel Configuration

Their meanings, in the vocabulary the US Payments Forum uses: above the **contactless floor limit** the transaction must be authorised online by the issuer; above the **CVM required limit** cardholder verification must be performed using the full capability of the terminal; above the **contactless transaction limit** the transaction cannot use the contactless interface at all and the customer is directed to insert or swipe. That last limit is split according to whether on-device cardholder verification is available, which is the mechanism by which a phone can pay £400 where a plastic card cannot.

Kernel 3 does the equivalent work with two well-known tags. The **Terminal Transaction Qualifiers**, tag `9F66`, four bytes, is sent to the card in the PDOL and states what the terminal can and will do.

The **Card Transaction Qualifiers**, tag `9F6C`, two bytes, comes back and states what verification the card believes should happen. Tag `9F6E`, four bytes, carries the Form Factor Indicator, which is how an issuer learns whether it was a card, a phone or a watch.

The transaction data itself is the same set of objects used on the contact interface: Application Interchange Profile (`82`), Application File Locator (`94`), Amount Authorised (`9F02`), Transaction Currency Code (`5F2A`), Terminal Country Code (`9F1A`), Unpredictable Number (`9F37`), Application Transaction Counter (`9F36`), Application Cryptogram (`9F26`), Cryptogram Information Data (`9F27`), Issuer Application Data (`9F10`) and CVM Results (`9F34`). Offline data authentication is normally **fDDA** (fast DDA) or **CDA** (Combined Data Authentication), both producing a dynamic signature under the card's own private key rather than the static signature of SDA. That distinction is essential in transit.

In the authorisation message the interface is signalled by POS entry mode: in common scheme usage, `05` for contact chip, `07` for contactless chip, `91` for the legacy contactless magnetic-stripe mode. Issuers risk-score off that field, which is why an issuer can decline contactless while approving contact for the same card.

Why the tap limit exists, and what it actually protects

The limit is a **lost-and-stolen control**. It is not a countermeasure against skimming, eavesdropping or cloning, and it has no relationship to the radio.

Its legal basis in Europe and, by onshoring, in the United Kingdom is Article 11 of Commission Delegated Regulation (EU) 2018/389, the SCA-RTS. The Article permits a payment service provider not to apply strong customer authentication for a contactless transaction at the point of sale provided the individual amount does not exceed EUR 50, and either the cumulative amount since the last application of strong customer authentication does not exceed EUR 150, or the number of consecutive such transactions since then does not exceed five.

The UK version raised the figures without changing the shape:

Date	Single transaction	Cumulative
2007	£10	—
2010	£15	—
2012	£20	—
2015	£30	—
1 April 2020	£45	set under SCA-RTS Article 11
15 October 2021	£100	£300, or five consecutive transactions
19 March 2026	set by the firm	set by the firm

The 2020 increase was accelerated as part of the industry's response to Covid-19; the 2021 figures then held for four and a half years. On 14 March 2025 the FCA published an engagement paper on contactless limits, with responses due by 9 May 2025; on 19 December 2025 it announced that it would remove the prescribed limits and allow firms with strong fraud controls to set their own, encouraging them to let customers set personal limits or switch contactless off entirely; the change took effect on 19 March 2026. As of mid-2026 the major UK issuers had publicly retained £100. Consumer protection was explicitly unchanged: unauthorised transactions on a lost or stolen card must still be reimbursed.

Enforcement happens at three independent points, and a practitioner needs to know which one is refusing. The **terminal** refuses using the kernel data objects above: over the CVM required limit it demands verification, over the transaction limit it refuses the interface entirely, producing "please

insert your card" with no network round trip. The **card** refuses using issuer-personalised counters that force a contact transaction once the number or cumulative amount of contactless transactions passes a threshold; this implements the £300 bucket at chip level, and is why the bucket resets when you insert the card and type a PIN. The **issuer host** refuses using velocity rules against its own record of the card, and is the only one of the three that sees transactions at other merchants in real time.

The fraud numbers put the control in proportion. UK Finance reported contactless fraud losses of £41,5 million in 2023, a 19 % increase on 2022, against total unauthorised fraud of £708,7 million — about 5,9 %. Its Annual Fraud Report of 15 June 2026, covering 2025, reported total fraud losses of £1,28 billion, of which contactless fraud was £46,8 million, up 8 %; lost and stolen card fraud £109,8 million, down 2 %; and remote purchase fraud — card-not-present, which no tap limit touches — £423,5 million, up 3 %. The tap limit governs a category roughly one ninth the size of the one it cannot govern at all.

Relay attacks

The design flaw, if it is one, is implicit in the standard: ISO/IEC 14443 has no concept of distance, only of *field strength sufficient to operate*, which the attacker supplies himself. Nothing in the protocol binds the card's physical location to anything.

A relay therefore requires no cryptanalysis. The attacker builds two devices — a fake card presented to a genuine reader, a fake reader presented to a genuine card — and forwards APDUs between them over any channel with adequate bandwidth. Gerhard Hancke demonstrated a practical relay against ISO 14443 proximity cards in 2005; Saar Drimer and Steven Murdoch published the case for distance bounding against smartcard relays in 2007; relays using unmodified NFC handsets followed. The problem was solved in principle much earlier, by Brands and Chaum's distance-bounding protocol in 1993 and Hancke and Kuhn's practical variant in 2005, both timing single bits at the physical layer and deriving an upper bound on distance from the speed of light.

Two things are frequently conflated. **Eavesdropping** is passive interception of an existing transaction, and it reaches much further than the operating range because it depends on radiated leakage rather than coupling. Hancke's experiments at RFIDsec'08, using a wide-range receiver and a commercial antenna kit, recovered the reader-to-card (forward) channel of ISO 14443 Type A at 5 m in an entrance hall and 4 m in a corridor, and the card-to-reader (backward) channel at 1 m; for Type B, the forward channel at 3 m and the backward channel at up to 4 m. Eavesdropping yields the transaction as it happened, not a reusable credential, because the cryptogram is bound to that amount, that unpredictable number and that ATC. **Relaying** is active and creates a new transaction. It is the dangerous one.

The Relay Resistance Protocol

The industry's answer is a timing bound. Mastercard's Relay Resistance Protocol has been in EMV Book C-2 since 2016, in that book's sections 3.10, 5.3 and 6.6. The same protocol is specified in publicly available form in the European Card Payment Association's CPACE Terminal Kernel Specification v1.0 of 12 July 2018, from which the following field-level detail is taken.

The terminal issues an **EXCHANGE RELAY RESISTANCE DATA** command immediately after application selection:

Field	Value
CLA	`80`
INS	`EA`
P1	`00`

Field	Value
P2	`00`
Lc	`04`
Data	Terminal Relay Resistance Entropy, 4 bytes, binary
Le	`00`

The card returns four data elements: **Device Relay Resistance Entropy** (4 bytes, binary), **Min Time For Processing Relay Resistance APDU** (2 bytes), **Max Time For Processing Relay Resistance APDU** (2 bytes) and **Device Estimated Transmission Time For Relay Resistance R-APDU** (2 bytes). All three timing values are expressed in units of hundreds of microseconds.

The terminal starts a timer when it sends the command, stops it when the response arrives, and subtracts its own known transmission overheads to isolate the card's contribution:

Measured Relay Resistance Time = Timer - Terminal Transmission Time For Relay Resistance Command - Expected Minimum Transmission Time For Relay Resistance Response

floored at zero, where the expected minimum is the lesser of the card's declared estimate and the terminal's own configured response transmission time. The CPACE defaults for the terminal's transmission times are `0012` and `0018` in hundreds of microseconds: 1,8 ms for the command, 2,4 ms for the response.

Three checks then run. If the measured time is below the card's declared minimum by more than the configured tolerance, the transaction fails outright — a card that answers *too fast* is as suspicious as one that answers too slowly, because it suggests a pre-computed answer. If the measured time exceeds the declared maximum plus tolerance, a Terminal Verification Results bit is set and the protocol may be retried, the relay resistance counter permitting a maximum of two attempts. And if the card's estimated transmission time and the terminal's own differ by more than a configured percentage, or the measured time exceeds the declared minimum by more than a configured difference limit, further TVR bits are set for the issuer to act on.

The terminal-side configuration lives in Kernel 2 proprietary tags:

Tag	Kernel 2 data object
`DF8132`	Minimum Relay Resistance Grace Period
`DF8133`	Maximum Relay Resistance Grace Period
`DF8134`	Terminal Expected Transmission Time For Relay Resistance C-APDU
`DF8135`	Terminal Expected Transmission Time For Relay Resistance R-APDU
`DF8136`	Relay Resistance Accuracy Threshold
`DF8137`	Relay Resistance Transmission Time Mismatch Threshold

Crucially, both entropies are subsequently included in the data over which the Application Cryptogram is computed. A relay cannot pre-compute the exchange, and cannot substitute its own timing values, because the values it saw are bound into the cryptogram the issuer will verify.

How well it works

Honestly: less well than the design suggests, because the measurement is taken at the wrong layer.

Researchers at the Universities of Birmingham and Surrey, working under the NCSC-funded TimeTrust project, measured the RRP exchange on a test card and found a mean round-trip of about 53 000 µs on the first exchange with a standard deviation near 13 170 µs, and about 40 100 µs on subsequent exchanges. Against a card-declared maximum of 79,62 ms they relayed successfully in 67,79 to 77,05 ms. The jitter inherent in an application-layer measurement forces the acceptance window so wide that a competent relay fits inside it. Their conclusion was that distance bounding is more reliable at Level 1, at the ISO/IEC 14443-A layer where timing is locked to the carrier, than at Level 3 where it is at the mercy of APDU processing; they proposed a Level 1 relay protection protocol, L1RP, and verified it formally in Tamarin.

The same group showed why relay protection matters beyond plastic: a non-standard sequence of bytes preceding the standard ISO 14443-A wake-up command would convince Apple Pay that it was speaking to a transport reader, unlocking the Express Transit path. Combined with a Visa credential and a relay, they published video of £1 000 taken from a locked iPhone. Mastercard on Apple Pay and Visa on Samsung Pay were not affected.

The defences that remain, in descending order of effectiveness: the CVM required limit, because a relay attacker still cannot produce the PIN; issuer velocity rules, because a relay is a physically slow crime; the card's own offline counters; and RRP, which raises the bar without closing the door.

Why contactless is not less secure than chip and PIN in the way people assume

The popular model is that chip and PIN has two locks and contactless has one. The accurate model is that both have exactly the same lock on the *card*; the difference is entirely in the *cardholder* lock and in offline authorisation policy.

The Application Cryptogram at tag `9F26` is a MAC computed over the transaction data, the AIP and the ATC using a session key derived from the card's issuer master key. It is the same computation on both interfaces, and it is bound to one amount, one currency, one unpredictable number and one value of a counter that never repeats. It cannot be replayed for a different amount, at a different terminal, or twice.

So the "digital pickpocket" scenario — a criminal walking through a crowd with a reader in a bag — fails structurally. What can be read without interaction is limited, and what would be needed to monetise it is absent: no CVV2, no pair usable for card-not-present, no cryptogram usable for anything but the single transaction it was generated for. To take money the criminal must run a transaction against a real acquirer, which means being a merchant, which means being findable.

The real weaknesses, when they have appeared, have been in the layer that decides *whether verification happened*, and they are worth stating precisely because they are the seam.

In 2014, Emms, Arief, Freitas, Hannon and van Moorsel of Newcastle University presented at ACM CCS an attack showing that Visa contactless cards would approve foreign-currency transactions for any amount up to €999 999,99 without the cardholder's PIN. The UK limit then in force was £20. The flaw was not in the radio or the cryptogram: the limit check was performed against the terminal's own currency and no check existed for others.

In 2021, Basin, Sasse and Toro-Pozo of ETH Zurich presented "The EMV Standard: Break, Fix, Verify" at IEEE Security and Privacy, describing a PIN bypass on Visa contactless. The Card Transaction Qualifiers, tag `9F6C`, tells the terminal what verification the card expects, and in the affected configurations

it is neither authenticated nor cryptographically protected against modification. A man-in-the-middle clears bit 8 of the first byte, removing the online-PIN requirement, and sets bit 8 of the second byte, asserting that on-device cardholder verification was performed; in their published log `8200` became `0280`. The terminal then believes the cardholder has already been verified. They demonstrated it with an Android proof of concept in a real shop for approximately $190. Mastercard was not vulnerable to the same manipulation, because there the equivalent capability is signalled in the second bit of the first byte of the Application Interchange Profile, and the AIP *is* covered by the cryptogram.

The lesson generalises. Any data element that steers cardholder verification but is not covered by the Application Cryptogram is a candidate vulnerability, whatever the interface. Contactless made the attack cheaper to mount, but it did not create the seam. The seam was in the data model.

Transit and aggregated fares

Transit breaks the standard model for one reason: latency. The US Payments Forum's technical solution for transit contactless open payments states the requirement as a go/no-go customer entry prompt within a sub-second window, typically no more than **500 milliseconds** from a valid tap. No online authorisation meets that budget reliably, at every gate, at 08:15.

So the gate runs the transaction offline and pushes everything else downstream. The architecture has four named components.

Offline data authentication at the terminal. The gate performs dynamic ODA — fDDA or CDA — using the card's public key certificate chain, verified against the scheme CA public key held in the reader. This defeats counterfeit cards, skimming and man-in-the-middle at the gate itself, with no issuer connection. It is also where the energy budget bites: the card must complete an asymmetric operation inside the tap.

A deny list. Every gate holds a locally cached list of blocked cards and checks the credential against it before opening. The forum's framing is that the ability to hot-list a card quickly, so it is declined at every point of entry, is critical to preventing recurring fraud.

Deferred authorisation. Formally, a request that occurs when a merchant captures transaction information while connectivity is interrupted and holds it until connectivity is restored. Transit uses the mechanism deliberately rather than by accident.

Aggregation. Multiple taps combined into a single authorisation for a single amount. This is what makes fare capping possible: the back office cannot know that today's fifth journey has hit the daily cap until the fifth journey exists, so it cannot price any journey at the moment of travel.

The financial consequence has a name: **first tap risk**, the exposure created when a deferred authorisation is eventually declined for insufficient funds after the passenger has travelled. **Second tap risk** is the follow-on, the same card used again before it reaches the deny list. Neither sits with the issuer or the passenger; both sit with the operator, which is why transit operators are unusually exercised about deny-list latency.

Two data-level points a practitioner must get right. The Amount Authorised in tag `9F02` as sent by the gate will not equal the amount finally cleared, because the cleared amount is the aggregate. And an issuer must validate the cryptogram using only the data carried in the authorisation message's ICC data field — in ISO 8583 terms, Field 55 — and must not cross-check it against other fields whose values legitimately differ under aggregation. Issuers that get this wrong produce mysterious declines that appear only on transit.

Regulation accommodates this explicitly. Article 12 of the SCA-RTS exempts from strong customer authentication any electronic payment transaction initiated by the payer at an unattended payment terminal for the purpose of paying a transport fare or a parking fee. Without it, the cumulative counter in Article 11 would demand a PIN from every commuter twice a week, at a gate with no keypad.

London is the reference implementation. Contactless bank cards were accepted on London buses from 13 December 2012, with 2 586 journeys on the first day, and on the Tube and rail network from 16 September 2014. By December 2022 buses alone carried about 1,7 million contactless journeys a day, with more than 2,5 billion journeys across all modes since launch and credentials from over 180 countries presented at the gates.

For the passenger the visible artefacts follow directly. Charges appear grouped rather than per journey, and a day or more late, because aggregation runs after the travel day closes. The same card or device must be used to touch in and out, because the back office identifies a journey by matching two taps from the same credential — a card and the phone holding that card are, to the fare engine, two travellers. An incomplete journey draws a maximum fare because the back office cannot infer where you went. Caps are applied by the back office, never by the gate. And a device that runs flat mid-journey costs its owner a maximum fare, which is why Express Transit modes with a power reserve exist, and why those modes are where the interesting attacks live.

The sequence, end to end

The layers a single tap traverses, and the specification governing each:

Step	Governed by
Reader field present, card powers up	ISO/IEC 14443-2; EMV Contactless Interface Specification (Level 1)
Polling, REQA/WUPA or REQB/WUPB, anticollision, SELECT	ISO/IEC 14443-3
RATS/ATS, frame size and FWT negotiation, T=CL	ISO/IEC 14443-4
APDU command and response structure	ISO/IEC 7816-4
SELECT `2PAY.SYS.DDF01`, candidate list, kernel activation	EMV Contactless Book B (Entry Point)
EXCHANGE RELAY RESISTANCE DATA, where supported	EMV Contactless Book C-2
GET PROCESSING OPTIONS with PDOL, READ RECORD, fDDA/CDA	EMV Contactless Books C-1 to C-8
GENERATE AC, cryptogram returned	EMV Contactless Books C-1 to C-8
Authorisation request carrying the cryptogram	ISO 8583 or ISO 20022, per scheme

The last four rows are identical to the contact interface. Everything above them is radio. The money never notices the difference.

18.98 Common wrong ideas

Wrong: the card is powered by radio waves it picks out of the air. **Right:** at 13,56 MHz the exchange happens deep inside the near field, where energy passes by mutual inductance between two coils rather than being radiated, which is why the range is centimetres.

Wrong: the short operating range means nobody can listen in. **Right:** the range at which a card can be powered and transacted with is not the range at which it can be overheard, and published work has recovered the reader-to-card channel at several metres.

Wrong: the £100 limit is a technical property of the chip, the radio or the kernel. **Right:** nothing in any radio standard or EMV kernel knows what a pound is; the figure came from Article 11 of the SCA-RTS and since 19 March 2026 it is the firm's own choice.

Wrong: contactless is less secure than chip and PIN because it has one lock instead of two. **Right:** the Application Cryptogram is the identical computation on both interfaces; the whole difference is in cardholder verification and offline authorisation policy.

Wrong: contactless is exactly as secure, so there is no seam. **Right:** the seam is real and documented — data elements that steer whether verification happened, in some configurations not covered by the cryptogram, which is precisely the 2021 Visa PIN bypass.

Wrong: a criminal walking through a crowd with a reader in a bag can harvest usable card details. **Right:** there is no CVV2, nothing usable card-not-present, and a cryptogram bound to one amount, one unpredictable number and one counter value; to monetise anything the criminal must be a real, findable merchant.

Wrong: two cards in a wallet confuse the reader. **Right:** they cause a collision that Part 3's anticollision loop resolves deterministically, ending either in a clean read or a "present one card only" prompt.

Wrong: the ticket gate lets you through on trust and settles up afterwards. **Right:** before it opened it verified a dynamic signature from the card's own private key and obtained a full cryptogram; what is deferred is the authorisation, and the gap is first tap risk carried by the operator.

Wrong: the transit charge on your statement is a fare. **Right:** it is an aggregate computed after the travel day closes, which is why the Amount Authorised sent by the gate never equals the amount finally cleared.

Wrong: the Relay Resistance Protocol closes the relay problem. **Right:** measured at the APDU layer the jitter forces an acceptance window wide enough for a competent relay, which is why the same researchers argued for timing at the ISO/IEC 14443-A layer instead.

18.99 Chapter summary in 20 lines

1. A contactless card has no battery: a chip the size of a grain of rice and a loop of wire around the inside of the plastic is the whole machine.
2. The reader hums a magnetic note at 13,56 MHz all day, and the card's coil draws its power out of that field by mutual inductance.
3. The card cannot transmit, so it answers by repeatedly shorting a load across its own coil and letting the reader see the dips in its own field.
4. Field strength falls away savagely with distance, so the working range is centimetres — a consequence of the power budget rather than a privacy feature.
5. ISO/IEC 14443 defines the whole radio layer: Part 2 the field and modulation, Part 3 anticollision, Part 4 the block protocol universally called T=CL.
6. Above T=CL the card speaks ordinary ISO/IEC 7816-4 APDUs, so from the payment application's point of view the radio is only a transport.
7. EMVCo layers its own Level 1 requirements over that, and above them sit Book B's entry point and the eight contactless kernel books.
8. Which kernel wakes up determines what the limits are called and where they live, which is why "the contactless limit" is never one setting.
9. The Application Cryptogram is the same computation on both interfaces, bound to one amount, one currency, one unpredictable number and one value of a counter that never repeats.

10. So the radio changes nothing about proving the card is genuine, and the tap limit exists to answer an entirely different question.
11. That limit is a lost-and-stolen control built as a leaky bucket: a single-transaction cap plus a cumulative cap since the last strong authentication.
12. It is enforced at three independent points — reader data objects, counters inside the card application, and issuer velocity rules — and knowing which one refused is the practitioner's first question.
13. Its legal basis was Article 11 of the SCA-RTS; the UK figures climbed from £10 in 2007 to £100 and £300 in 2021, and the FCA removed the mandated caps on 19 March 2026.
14. The fraud figures keep it in proportion: contactless losses are a small fraction of remote purchase fraud, which no tap limit touches at all.
15. Eavesdropping is passive, reaches further than the operating range, and yields a transaction that cannot be replayed; relaying is active and creates a new one.
16. Nothing in ISO/IEC 14443 binds a card to a place, and a maximum Frame Waiting Time of nearly five seconds leaves ample room for a very long extension lead.
17. The industry's answer is the Relay Resistance Protocol: a timed exchange whose entropies are afterwards bound into the cryptogram, so the timing cannot be forged.
18. It helps less than it looks, because application-layer jitter widens the acceptance window enough for a competent relay to fit inside it.
19. Transit breaks the model on latency alone: the gate performs offline dynamic data authentication, checks a local deny list, opens, and defers both the authorisation and the price.
20. Aggregation is what makes fare capping possible and what puts first tap risk on the operator, and it is why your statement shows a single number, a day late, matching no fare on any poster.

Sources: ISO/IEC 14443 parts 2, 3 and 4; EMVCo Level 1 EMV Contactless Interface Specification v3.1 and Specification Bulletin 245; EMVCo Book C-8 v1.0; European Card Payment Association CPACE Terminal Kernel Specification v1.0; Commission Delegated Regulation (EU) 2018/389 Articles 11 and 12; FCA contactless limits engagement paper (2025) and press releases (2025); UK Finance contactless limit FAQs and Annual Fraud Report 2026; US Payments Forum papers on contactless limits (2020) and transit contactless open payments (2018); Transport for London press material; and the published research of Hancke (2005, 2008), Emms et al. (CCS 2014), Chothia et al. (FC 2015), Basin, Sasse and Toro-Pozo (IEEE S&P 2021), and Radu, Chothia, Newton, Boureanu and Chen on practical EMV relay protection.

19

CHAPTER

The Token in Your Phone

19.0 What this chapter gives you

1. You will be able to explain why the last four digits printed on a phone receipt differ from the last four on your plastic, and prove it to yourself by buying the same coffee twice.
2. You will be able to state exactly what a Payment Token is — 13 to 19 Luhn-valid digits from a real Token BIN Range, carrying its own expiry — and why it therefore crosses the existing network with nobody changing anything.
3. You will be able to say why a stolen token is close to worthless, and name the three separate walls that make it so.
4. You will be able to correct the claim that your card number never leaves the phone, and state the narrower claim that is actually true.
5. You will be able to say who mints the token and who owns the vault, and why many issuers cannot enumerate their own outstanding tokens without calling a network API.
6. You will be able to explain what the Payment Account Reference is for, why the industry had to invent it, and why it cannot itself initiate a payment.
7. You will be able to tell a secure element wallet from a host card emulation wallet by their availability and failure modes rather than by their cryptography.
8. You will be able to explain the green, yellow and red provisioning paths, and why yellow is where the whole design lives or dies.
9. You will be able to say why a reissued card leaves the token working and a lost phone leaves every standing order untouched.
10. You will be able to read a tokenised contactless trace, including why tag `9F6E` must be parsed kernel-aware or it will produce plausible nonsense.

The previous chapter left a phone held against a reader, an antenna pulling power out of the air, and a two-byte answer travelling back at 106 kilobits per second. It deliberately did not say what number was in that answer.

Here is the thing almost everybody gets wrong, including people who work in payments: when you tap your phone, your card number is not transmitted. Not encrypted, not obscured, not hashed. It is simply not there. The number the shop receives is a different number, sixteen digits long, beginning with the same digit as your card and looking in every respect like a card number, because it is one — it is just not yours. It was minted for that phone, it works only from that phone, and if somebody steals it they will find it worth almost nothing.

That substitution is called payment tokenisation, and it is the single largest change to card security since the chip. It is also the reason your bank can kill your Apple Pay card at four in the morning and

still let you buy petrol with the plastic on the way home. The plastic and the phone stopped being the same credential some time around 2014, and most cardholders have never been told.

This chapter is about that second credential: what it is, who makes it, what stops it being used anywhere else, and why a stolen one is close to worthless.

The plain version

Think about a hotel.

You check in and they give you a key card for room 412. It opens your room, it opens the pool, and it stops working on Friday. It is not the master key. It has its own serial number printed on the back, and if you lose it in a taxi the hotel does two things: it cancels that serial number, and it cuts you a new card. Room 412 does not change. The lock does not change. Nobody has to re-key the building.

Now suppose you want a second key so your brother can come and go. The hotel does not photocopy your card. It cuts a *new* card, with a *different* serial number, which opens room 412 and the pool but not the gym, and only between nine and five. Two cards, two serial numbers, one room, different powers.

Your bank account is room 412. Your plastic card is the first key. The card in your phone is the second key: a different number, cut for a specific purpose, cancellable on its own.

The number is genuinely different

Say your card is **4659 1234 5678 9011**, expiring 11/27. That number is embossed on the plastic, printed on your statement, and known to every shop you have ever typed it into.

When you add that card to your phone, your bank and the card network create a second number for you — say **4658 7712 3456 7892**, expiring 03/29. It is stored on a small, sealed chip inside the phone. It is not printed anywhere. You cannot read it out. The phone shows you the last four digits and nothing more.

Both numbers point at the same account and the same money. But they are not interchangeable, and this is the whole idea.

A worked example: £180 of headphones

You are in a shop in Manchester buying headphones for £180. You hold your phone to the reader and look at it, and your face unlocks the payment.

Three things happen in the two seconds that follow.

First, the phone sends **4658 7712 3456 7892** and the expiry 03/29. It does not send 4659 1234 5678 9011. The shop never learns that number. Neither does the shop's till software, nor its payment provider, nor the company that stores the day's takings.

Second, the phone sends a short one-time code — eight bytes of arithmetic, computed fresh for this tap, using a secret key that never leaves the sealed chip and a counter that ticks up by one every single time. Yesterday's code will not work today. This tap's code will not work at the next shop. It is the digital equivalent of a wax seal that can only be pressed once.

Third, the phone says: *the person holding this device proved who they were, on the device, just now.* That is why £180 goes through without a PIN pad, when a plain contactless card would have stopped dead at £100. The reader does not see your face. It simply receives a statement that a face was seen, from a device the bank has decided to trust.

Your bank receives all three, notices that 4658 7712 3456 7892 stands for account 4659 1234 5678 9011, checks the one-time code against its own copy of the maths, sees that verification was done on the device, and approves.

Now look at your receipt. The last four digits printed on it will be **7892**, not 9011. Most people assume this is a misprint or a strange rounding of their card number. It is neither. It is the token's last four digits, and it is the easiest way to prove to yourself, tonight, that your phone is not sending your card number: pay for the same coffee twice, once with the plastic and once with the phone, and compare the two receipts. They will end in different digits.

What a thief gets

Six months later the shop is hacked and somebody walks off with a year of transaction records, including **4658 7712 3456 7892** and 03/29.

They cannot use it. Three separate walls stand in the way.

The number is registered as working only for contactless taps from that one phone. Type it into a website and it is refused before any money is considered, because the network can see that this number is not permitted to be typed into a website. That restriction is baked in at the moment the number is created.

The number is worthless without the one-time code, and the code cannot be produced without the sealed chip in your phone. The thief has the envelope; the seal is in Manchester in your pocket.

And it does not identify you. It is not on your statement, your bank does not use it on the phone to you, and it cannot be traced back to 4659 1234 5678 9011 by anyone outside the network.

Compare that to what a thief gets from a stolen card number: a working credential, usable in dozens of channels, that stays valid until the bank reissues the plastic and posts it to you.

Losing the phone

You leave the phone on a train. You call the bank.

The bank cancels **4658 7712 3456 7892**. That is one row in one table. It takes seconds, needs no post, and costs nothing.

Your plastic card still works. Your card number has not changed. Every standing order, every subscription, every card-on-file at every shop that has your real number is untouched. The direct debit for your gym does not fail. You buy dinner on the way home with the card in your wallet.

This is the practical payoff, and it is the sentence to carry to dinner: **the number in your phone can be destroyed without destroying your card, because it was never your card.**

Where the plain version stops being true

The hotel key analogy is sound in outline, and wrong in four ways that matter.

The token is not what makes the transaction safe

A payment token is, on its own, just a number. It has no cryptography in it. If you place a token in a channel where nobody demands a cryptogram and nobody checks the restrictions, it spends exactly like a card number — that is the point of the design, since the whole scheme was built to travel over infrastructure that already existed and could not be changed.

What makes the tap safe is the combination: a token *plus* a per-transaction cryptogram *plus* restrictions enforced by whoever is doing the checking. Take away any one of the three and you have a card number with an unusual value. Merchant card-on-file tokens, which are the same EMVCo mechanism, routinely operate with no per-transaction cryptogram at all; they rely on merchant and channel restriction alone.

So the correct statement is not "tokens are safe". It is "tokens shrink the blast radius of a breach, and cryptograms stop a stolen credential being replayed, and the two are separate mechanisms that people habitually merge into one word".

Your card number does leave the phone — once

Apple's and Google's public claims are carefully worded, and the careful wording is doing work.

When you add a card, you type the real number into the phone, or the phone reads it with the camera, or your bank's app pushes it across. That number is then encrypted and sent to the card network and your bank, because they have to know which account you are asking to tokenise. It leaves the device. It travels.

What is true is narrower and still meaningful: the wallet provider cannot decrypt the token it receives back; the token is written into the phone's sealed chip; and the wallet's servers do not store it. Apple states that the Device Account Number is "never stored on Apple Pay servers or backed up to iCloud". That is a real property. It is not the same as "Apple never sees your card details", which is what people repeat.

There is also a residue. The wallet keeps an opaque identifier derived from the funding card so that it can tell two tokens on the same card apart from two tokens on different cards. Apple exposes it to apps as `primaryAccountIdentifier`. It is not the card number, but it is a persistent handle on the card number, and it exists because the ecosystem could not function without one.

The issuer is not really in charge; the network is

The analogy implies the hotel cuts the keys. In practice, for the overwhelming majority of device tokens, the Token Service Provider is Visa or Mastercard, not your bank. They own the vault. They own the mapping table. They decide the token's BIN range, they mint the value, they hold the keys used to validate cryptograms, and they perform the substitution back to your real number before the authorisation reaches your bank's systems.

Your bank's role at provisioning time is to answer one question — approve, decline, or ask for more proof — and its role afterwards is to consume a data feed. Many issuers cannot enumerate their own outstanding tokens without calling a network API. When the network's token service is degraded, tokens stop working even though the bank is perfectly healthy and the plastic is fine. This is a real operational dependency, and it is not visible from the cardholder's side of the counter.

"Device token" describes a shrinking share of tokens

The chapter title says *phone*, and phones are where tokenisation is most visible. But EMV payment tokens are also issued in enormous numbers to merchants for stored cards, to wallets for e-commerce, and to acquirers on behalf of merchants who never asked. Those tokens live in databases, not secure elements. There is no biometric, no secure chip, no per-tap cryptogram of the kind described above.

The consequence shows up in a place nobody anticipated. Once the same person carried five different numbers at five different merchants, the systems that needed to know "these are all the same account" — fraud scoring, loyalty, anti-money-laundering, transit fare capping — broke. The industry's answer

was to invent yet another identifier, the Payment Account Reference, whose entire job is to undo just enough of the separation to make those systems work again. That is a strange thing to have to build, and its existence tells you the analogy was too clean.

The technical version

Vocabulary, exactly

The governing document is the **EMV Payment Tokenisation Specification – Technical Framework**, published by EMVCo, royalty-free and publicly available. Version 2.4 was published on 9 July 2026, superseding v2.3; v1.0 dates from 2014. A companion document, *EMV Payment Tokenisation – A Guide to Use Cases*, carries the worked scenarios.

Note the spelling and the scope. EMVCo's document says *tokenisation*, and it defines a **Payment Token** — not a DPAN. The terms this chapter's title uses are industry vocabulary that grew up alongside the specification:

Term	Expansion in common use	Status
FPAN	Funding PAN	Industry usage; not an EMVCo term
DPAN	Device PAN, or device primary account number, or (in Google's documentation) dynamic primary account number	Industry usage; expansions disagree
Payment Token	—	EMVCo term, covers device *and* non-device tokens

Apple's PassKit framework uses both abbreviations openly: `PKSecureElementPass` exposes `deviceAccountIdentifier` (documented as an opaque identifier of the DPAN), `deviceAccountNumberSuffix` (the last four digits of the DPAN), and `primaryAccountIdentifier` (an opaque identifier of the FPAN). Google's device tokenisation documentation writes "dynamic primary account number (DPAN) or device token". When you see DPAN in a scheme manual, check which expansion that manual means; they are not always the same thing.

The roles are stable across versions:

Role	Definition
Token Service Provider (TSP)	Operates the Token Vault and related processing; maps Payment Tokens and token expiry dates to PANs and PAN expiry dates; issues tokens to Token Requestors; performs de-tokenisation; sets aside licensed ISO BINs as Token BINs
Token Requestor	The entity that submits Token Requests. Apple, Google, Samsung, a merchant, a wallet, an acquirer
Token Vault	The repository holding token-to-PAN mappings and token attributes
BIN Controller	The entity that controls a BIN range, registered with EMVCo for PAR purposes

EMVCo runs two registration programmes rather than a certification programme. The **TSP Registration Programme** assigns a three-digit **TSP Code** so that every TSP issuing payment tokens is globally identifiable; as of writing the fee is USD 11,500 for initial registration and USD 2,850 per calendar year thereafter. The **BIN Controller Registration Programme** assigns a **BIN Controller ID** (**BCID**), which becomes the first four characters of every PAR that controller generates. EMVCo is explicit that it "does not evaluate, approve or otherwise endorse TSPs"; it maintains a list of codes. There is no EMVCo tokenisation test programme, and EMVCo says plainly that one would be impractical.

The token itself

Element	Specification
Payment Token	A surrogate value for a PAN: a 13 to 19 digit numeric value that must pass basic account-number validation rules including the Luhn check digit
Token BIN Range	A unique identifier consisting of the leading 6 to 12 digits of the Token BIN; carried in BIN routing tables
Token Expiry Date	The token's own expiry, mapped by the TSP to the PAN expiry date; need not equal it
Token Requestor ID (TRID)	An 11-digit numeric value identifying each unique combination of token requestor and token domain
Token Cryptogram	"A cryptogram generated using the Payment Token and additional transaction data to create a transaction-unique value." Calculation and format vary by use case
Payment Account Reference (PAR)	A fixed-length 29-character uppercase alphanumeric value: 4 characters of BCID assigned by EMVCo, plus 25 unique characters

Three properties of that last row are load-bearing and frequently misstated. PAR has a one-to-one relationship with the PAN and a one-to-many relationship with the payment tokens issued against it. It cannot be reverse-engineered to obtain either a PAN or a payment token. And it cannot itself initiate a financial transaction. It is an identifier, not a credential.

The Token BIN Range point deserves emphasis for anyone building routing: token BINs are real ISO BINs, carved out of the issuer's or the TSP's allocation and published into routing tables. That is precisely why a token traverses the existing network without any acquirer changing anything. It is also why a badly maintained BIN table will route a token to the wrong place and produce a decline that looks like a network fault.

Where the token travels in the message

In ISO 8583, the payment token occupies the primary account number field, **DE 2**, and the token expiry date occupies **DE 14**. Chip data travels in **DE 55**. Point-of-service entry mode is **DE 22**; Visa's own acquirer implementation guidance specifies DE 22 = `07` for a qVSDC contactless chip transaction and `91` for a contactless magnetic-stripe (MSD) transaction.

Payment Account Reference is carried in EMV tag `9F24` within DE 55 for chip transactions. For non-chip transactions the Secure Technology Alliance's primer places it in data element 56 of the authorisation response. Treat the tag `9F24` path as the reliable one and confirm the DE 56 behaviour against your acquirer's own manual, because it varies.

The token-specific attributes that the issuer receives — token requestor ID, token assurance data, wallet identifiers, the last four digits of the underlying PAN — travel in scheme-private data elements whose numbering differs between networks and which are specified in each scheme's own message manual, not in any EMVCo document. Do not carry a field number learned on Visa across to Mastercard. This is the single most common source of integration errors in token projects.

In the ordinary model, the network de-tokenises before the authorisation reaches the issuer's host: the issuer sees the FPAN in DE 2, plus token metadata alongside. Some issuers host their own TSP and receive the token itself. Establish which model you are in before writing a single line of authorisation logic.

The chip data a phone actually presents

A tokenised contactless tap is an ordinary EMV contactless transaction whose PAN happens to be a token. The tags below are the base EMV set, defined in the EMV ICC Specifications for Payment Systems, Book 3, Annex A.

Tag	Name	What it holds on a phone
5A	Application Primary Account Number (PAN)	The token, not the funding PAN
57	Track 2 Equivalent Data	Token, field separator, token expiry, service code, discretionary data
5F24	Application Expiration Date	The token's expiry
82	Application Interchange Profile	Card capabilities
95	Terminal Verification Results	Terminal's view of what happened
9A	Transaction Date	
9C	Transaction Type	First two digits of the ISO 8583 processing code
9F02	Amount, Authorised (Numeric)	
9F10	Issuer Application Data	Proprietary data for the issuer, including the card verification results
9F1A	Terminal Country Code	ISO 3166
9F26	Application Cryptogram	The eight bytes produced in response to GENERATE AC
9F27	Cryptogram Information Data	Cryptogram type and required terminal actions
9F34	Cardholder Verification Method Results	Three bytes of testimony, as chapter 17 explained
9F36	Application Transaction Counter (ATC)	Increments every transaction
9F37	Unpredictable Number	Terminal-supplied variability
5F2A	Transaction Currency Code	ISO 4217

Two kernel-specific objects appear in Visa qVSDC flows and are not in Book 3 Annex A:

Tag	Name	Length
9F66	Terminal Transaction Qualifiers (TTQ)	4 bytes
9F6C	Card Transaction Qualifiers (CTQ)	2 bytes

And then there is `9F6E`, which is a trap.

Scheme	Meaning of `9F6E`	Length
Visa (qVSDC)	Form Factor Indicator (FFI)	4 bytes, fixed
Mastercard (M/Chip)	Third Party Data	5 to 32 bytes, variable

The same tag number carries two entirely different structures depending on which kernel produced the trace. A Visa FFI of `20700000` decodes as version `001`, form factor "standard card", with signature panel, hologram and CVV2 capability set. Feed those four bytes to a Mastercard Third Party Data parser and you will get plausible nonsense. Feed a seven-byte Mastercard value such as `06430000303000` to a Visa FFI decoder and it will either reject the length or truncate it. If your terminal estate handles both schemes, your TLV parser must be kernel-aware, and any documentation that lists `9F6E` with a single name is incomplete.

The Form Factor Indicator is where "this was a phone, not a card" is stated in the chip data, which is why it matters to issuers doing risk scoring on tokenised taps.

The secure element

Apple's implementation is the best-documented and is worth stating precisely, because it is also the most misdescribed.

The Secure Element is, in Apple's words, "an industry-standard, certified chip running the Java Card platform, which is compliant with financial industry requirements". It is "certified in accordance with the EMVCo Security Evaluation process", for which EMVCo issues unique IC and platform certificates, and it is separately "certified based on the Common Criteria standard". The Device Account Number lives there, in a payment applet, and Apple's servers manage those applets without being able to read the number.

The NFC controller sits between the application processor, the Secure Element and the reader, routing frames. Below it is ordinary contactless plumbing: ISO/IEC 14443 Type A or Type B at the radio layer, ISO-DEP (ISO/IEC 14443-4) for the transport, and application protocol data units per ISO/IEC 7816-4 above that. Applications are addressed by Application Identifier — up to 16 bytes, consisting of a five-byte Registered Application Provider Identifier issued by the ISO/IEC 7816-5 registration authority followed by a proprietary extension. `A000000003` is Visa and `A000000004` is Mastercard, so the credit and debit applications your phone advertises are `A0000000031010` and `A0000000041010` respectively — the same identifiers a plastic card presents. From the reader's point of view, a phone is a card.

Authorisation on Apple devices is a two-chip conversation. The Secure Enclave and the Secure Element communicate over a serial interface using a shared secret established at runtime through Elliptic Curve Diffie-Hellman key agreement. When you authenticate, "the Secure Enclave then sends signed data about the type of authentication and details about the type of transaction (contactless or e-commerce) to the Secure Element, tied to an Authorization Random (AR) value". The AR is generated in the Secure Enclave when the first card is provisioned and persists while Apple Pay is enabled, protected by Secure Enclave encryption and an anti-rollback mechanism. A payment "can be made only after the Secure Element receives authorization from the Secure Enclave".

The cryptogram itself is "computed using a transaction counter and a key", with the counter incremented for each transaction, and it incorporates the terminal's Unpredictable Number for NFC transactions or an Apple Pay server anti-replay value for in-app and web transactions. That last distinction

is the answer to a question practitioners ask constantly: an in-app tokenised payment has no terminal and therefore no terminal-supplied nonce, so the freshness has to come from somewhere else.

Host card emulation

Android took a different route. From **Android 4.4** onwards the platform supports host-based card emulation, in which "the data is routed directly to the host CPU instead of being routed to a secure element". A `HostApduService` receives command APDUs through `processCommandApdu()` and returns response APDUs; Android's implementation supports a single logical channel, so the exchange is strictly single-threaded and half-duplex. AIDs are declared in groups, and Android guarantees that all AIDs in a group route to your service or none do.

The security consequence is direct. Without a secure element, the payment keys are in ordinary application memory on a device that may be rooted. The industry's answer is not to pretend otherwise but to make the keys nearly worthless: **limited-use keys**, provisioned over the network and bounded by thresholds — a time-to-live, a maximum number of transactions, a cumulative transaction amount, or a combination — and replenished from the cloud. A key that expires after a handful of taps or a few days is a poor prize.

Around this sits EMVCo's **EMV Mobile Payment: Software-Based Mobile Payment Security Requirements** and the associated **SBMP Security Evaluation Process**, introduced in 2018. It evaluates software development kits, trusted execution environments, consumer device cardholder verification methods including biometrics and authenticators, attestation mechanisms and software protection tools. EMVCo issued its hundredth SBMP security evaluation certificate on 29 November 2022, with nine accredited laboratories performing the evaluations.

The practitioner-facing difference between the two architectures is not really cryptographic. It is availability:

	Secure element	Host card emulation
Key storage	Certified hardware, Java Card applet	Application memory, protected by software and platform features
Key lifetime	Long-lived, provisioned once	Limited-use, replenished
Behaviour offline	Works indefinitely	Works until keys are exhausted, then fails
Failure mode	Hardware failure	"Unable to pay, please connect to the internet"
Certification	EMVCo Security Evaluation, Common Criteria	EMVCo SBMP Security Evaluation

A user who leaves an HCE wallet in a drawer for a month and then tries to tap in a car park with no signal will be declined, and the decline will be correct.

Provisioning: green, yellow, red

Provisioning is the security-critical moment in the whole design. A criminal who successfully provisions a stolen card into their own phone has manufactured a fresh, fully working credential with strong cryptography protecting *them*.

Apple's flow uses three server-side calls — Required Fields, Check Card, and Link and Provision — with sessions protected by TLS 1.2 or later. The device sends the card details together with information

about the account and the device, and the issuer or its network returns an encrypted Device Account Number along with the key material used to generate the per-transaction codes.

For provisioning initiated inside a bank's own app, Apple's PassKit provides `PKAddPaymentPassViewController`, configured with `PKAddPaymentPassRequestConfiguration`, which returns a `PKAddPaymentPassRequest` carrying three fields:

Field	Contents
`encryptedPassData`	Payload encrypted by the issuer host under a key derived from Apple's public certificates and a generated ephemeral private key
`ephemeralPublicKey`	The public half of a key pair generated for this provisioning attempt only
`activationData`	A cryptographic one-time value generated by the issuer host per the network's specification

`EV_ECC_V2` is the mandatory encryption scheme.

On Android, the equivalent path is Google's TSP integration with push provisioning from the issuer app. Where a challenge is required, app-to-app identification and verification hands control to the issuer's app via an Android Intent, with a base64-encoded JSON payload in `Intent.EXTRA_TEXT`; the issuer app authenticates the cardholder and returns `BANKING_APP_ACTIVATION_RESPONSE` with a value of `approved`, `declined` or `failure`, optionally accompanied by `BANKING_APP_ACTIVATION_CODE`. Google's documentation is explicit that the issuer app must verify the caller using `Activity.getCallingPackage()` and confirm it is Google Wallet, `com.google.android.gms`. Skip that check and you have built an activation oracle for anybody who can install an app.

The issuer's decision is described everywhere in the industry by colour:

Path	Meaning	Typical use
Green	"Token creation has been approved. The token is created in an active state and is ready to be used for payment."	Push provisioning from an authenticated banking session; trusted devices
Yellow	"Cardholder is subject to additional authentication requirements (called stepped-up authentication)."	Most manual card entry
Red	"Token request is declined. No token is created."	Reserved for unacceptable risk; in Google's guidance, "typically a very small fraction of attempts"

Yellow is where the design lives or dies, because a token stuck awaiting activation is a card the customer believes they have added and cannot use. The resolution methods in general use are: a one-time code by SMS; a one-time code by email; a call to the issuer's contact centre; login to the issuer's mobile banking app, including the app-to-app flow above; and, for Visa customers, web-based secure authentication. Google's published guidance is that "well thought out and high performing ID&V processes yield token activation rates of +90% on Google Pay", and that most issuers route manual provisioning attempts to yellow except on a trusted device. Issuers receive account and device risk scores from the TSP and are encouraged to override them when their own data says the user is legitimate.

The specification's own vocabulary for what yellow accomplishes is **identification and verification** (**ID&V**), with five defined cases: No ID&V Performed; Account Verification; Token Service Provider Assurance; Token Service Provider Assurance with Requestor Data; and Card Issuer Verification of the Cardholder.

One historical correction that catches people reading older material. Version 1.0 of the framework defined a two-digit **Token Assurance Level**, where "a value of 00 would indicate no ID&V and 99 would indicate the highest level of assurance". Version 2.0 replaced that concept with **Token Assurance Method**, based on the type and outcome of the ID&V performed, the entity that performed it, the domain in which the token is used, and supporting token assurance data. If a vendor document still talks about assurance *levels* as though they were current, that document has not been revised since 2016 and everything else in it should be treated with suspicion.

Mastercard's Digital Enablement Service exposes the mechanics under its own names: a `tokenUniqueReference` identifying the token for its lifetime, a `panUniqueReference` for the funding account, and operations including `GET /{id}/token/activate`, `GET /{id}/token/resendactivationcode` and `GET /{id}/token/activationmethods`. A token awaiting the yellow path carries a provisioning status of "Awaiting Activation".

Domain restriction, precisely

EMVCo defines **Token Domain Restriction Controls** as "a set of parameters that are applied during Token Processing to constrain a Payment Token to the permitted usage scenarios". Three inputs do the constraining:

Token Requestor ID. Does this token belong to this token requestor? A token minted for Apple Pay presented by a different requestor is refused.

POS Entry Mode. The specification calls for "the use of POS Entry Mode values that are carried in the POS Entry Mode Code field to limit the use of Tokens to only those POS Entry Modes agreed to during Token Requestor Registration". This is what prevents a contactless device token being keyed into a web form: the entry mode does not match the registration.

Merchant information. "Merchant-related data elements, such as the Card Acceptor ID in a combination with Acquirer-identifying data elements SHOULD be used to limit the use of a Payment Token." This is the mechanism behind merchant-scoped card-on-file tokens.

To these the framework adds the requirement for a valid token cryptogram in domains where one is expected. Validation may be performed by the TSP or delegated to the issuer; either way, a cryptogram that fails or a transaction that violates the domain controls is refused before the account is even considered.

The critical property, and the one the plain version hides, is that **domain restriction is enforced by whoever is checking, and it is only as strong as that party's implementation.**

The clearest public demonstration of this came in 2021. Andreea-Ina Radu, Tom Chothia, Christopher J. P. Newton, Ioana Boureanu and Liqun Chen, of the University of Birmingham and the University of Surrey, published *Practical EMV Relay Protection*, showing that an iPhone with a Visa card set up in Express Transit mode could be made to authorise **£1,000** while locked. Transport for London ticket gates broadcast a non-standard byte sequence before the standard ISO/IEC 14443 wake-up command — the researchers called them the "magic bytes" and deliberately did not publish their value. Replaying that sequence convinced the phone it was at a transit gate, where Express Transit deliberately suppresses the biometric prompt. The attack then set bits in the Terminal Transaction Qualifiers sent in the GET

PROCESSING OPTIONS command and set the "consumer device CVM performed" bit in the Card Transaction Qualifiers, so that the shop reader believed a cardholder verification had happened that had not. Relay success improved from roughly 40 per cent to effectively complete once READ RECORD responses were cached. The combination was specific: it did not affect Mastercard on iPhone, or Visa on Samsung Pay. Apple and Visa have since addressed it.

The lesson is not that tokenisation failed. The token worked exactly as designed. What failed was one exemption inside one domain control, on one platform, for one scheme. That is the shape of every real tokenisation failure: not broken cryptography, but a restriction that somebody decided to relax.

Lifecycle: why the token dies and the plastic lives

Tokens carry state independent of the account. Mastercard's states are the ones most commonly encountered:

State	Meaning	Reversible
Active	Provisioned, activated, usable	—
Suspended	Temporarily blocked	Yes, by unsuspend
Deactivated	Permanently ended	No

Four lifecycle events are worth committing to memory because they generate most of the support tickets.

The card is reissued. Your plastic is lost and a new card arrives with a new PAN and expiry. The token in your phone keeps working, because the TSP updates the token-to-PAN mapping in the vault. You do not re-add the card. This surprises people and is the correct behaviour: EMVCo's own material describes tokenisation as reducing "the need for card replacements" and enabling issuers "to control and replace payment tokens for specific merchants, devices or transaction types, often without any interaction from the cardholder". Note the corollary — if the card was reissued *because of* fraud, the automatic mapping update means the token survives too, unless the issuer suppresses it. Suppression is a configuration choice, and getting it wrong in either direction produces either angry customers or continued fraud.

The device is lost. The token is suspended or deactivated. The account is untouched. Apple's implementation lets Lost Mode in Find My suspend the ability to pay, or a remote erase remove it, and Apple notes that the issuer suspends the cards "even if your device is offline and not connected to a cellular or Wi-Fi network" — because suspension happens in the vault, not on the phone.

The token is compromised. A merchant breach exposes tokens. The TSP deactivates the affected tokens and the requestor provisions replacements. No plastic is reissued, no card numbers change, and nothing else the cardholder has set up is disturbed. This is the single largest operational saving in the whole design, and it is the reason issuers accepted tokenisation despite the loss of control described earlier.

The customer deletes the card from the wallet. The token is deactivated and cannot be revived; adding the card again mints a new one, with a new value, a new expiry and a new last four digits.

What it is worth

Some figures, all dated, all from the schemes themselves.

Visa announced its ten billionth token on 4 June 2024, ten years after launch, stating that tokenisation "can reduce the rate of fraud by up to 60%", that it had saved USD 650 million in fraud in the preceding

twelve months, that 29 per cent of all Visa transactions used tokens as of April 2024, and that tokens contributed a six basis-point increase in global payment approval rates.

Mastercard reported in June 2025 that nearly half of its European e-commerce transactions were tokenised, up by more than a third year on year, and has stated a goal of 100 per cent tokenisation of European e-commerce by 2030 alongside phasing out manual card entry.

The PCI Security Standards Council's position is the one to quote at auditors: payment tokens are "created by TSPs that are registered with EMVCo", are "issued to a cardholder in lieu of a PAN", and "during a Payment Token transaction, the merchant and acquirer do not receive or have access to the corresponding PAN". That last clause is the compliance argument in a single sentence, and it is why tokenisation is a scope conversation and not only a fraud conversation.

A short field guide to failures

Symptom	Usual cause
Token declines at exactly one merchant	Domain restriction on Card Acceptor ID; check the merchant data the acquirer is actually sending
Card shows in wallet but every tap declines	Yellow path never completed; token in Awaiting Activation, not Active
Receipt last four does not match the plastic	Correct behaviour; that is the token's suffix, not the PAN's
9F6E decodes as rubbish	Kernel mismatch: Visa Form Factor Indicator parsed as Mastercard Third Party Data or vice versa
PAR absent from the response	BIN controller not registered for a BCID, or PAR not populated on that route
Works contactless, fails in a browser	POS entry mode outside the registered domain; the token was never permitted there
HCE wallet declines after weeks unused	Limited-use keys exhausted or expired; device must reach the network to replenish
Issuer cannot locate a token by PAN	Search by `tokenUniqueReference`; the vault is keyed on the token, not the account
New card arrived, token stopped working	Issuer suppressed the automatic token update on reissue — usually deliberate, occasionally a misconfiguration
Token metadata field missing after a scheme switch	Scheme-private data element numbering differs; you cannot port a field number between networks

The chapter in one paragraph

Payment tokenisation replaces the primary account number with a Payment Token: a 13 to 19 digit Luhn-valid surrogate, drawn from a Token BIN Range of 6 to 12 leading digits, carrying its own expiry, minted by a Token Service Provider registered with EMVCo under a three-digit TSP Code, on behalf of a Token Requestor identified by an 11-digit Token Requestor ID. On a phone the token is written into a secure element — a Java Card platform chip evaluated under the EMVCo Security Evaluation process and Common Criteria — or, where there is no secure element, held in software under limited-use keys bounded by time, count or amount and evaluated under EMVCo's Software-Based Mobile Payment requirements. Provisioning is gated by identification and verification, whose outcome the industry calls green, yellow or red, and whose rigour the framework records as a Token Assurance Method, having abandoned the older two-digit Token Assurance Level at version 2.0. At the till the

token occupies DE 2, its expiry DE 14, and its chip data DE 55, where tag `5A` holds the token rather than the account, tag `9F26` holds a cryptogram computed from a key inside the device and a counter that only ever increases, and tag `9F24` may carry a 29-character Payment Account Reference whose first four characters are an EMVCo-assigned BIN Controller ID and which links every token on the account back together without exposing the account. Token Domain Restriction Controls constrain all of it by token requestor, by POS entry mode and by merchant identity, enforced by whichever party is checking — which is why the only serious public break of the model, the 2021 Express Transit relay, was not a break of the cryptography but the exploitation of a deliberately relaxed control. And because the token is a separate credential with a separate lifecycle, it can be suspended or deactivated in the vault in seconds, leaving the plastic, the account, the standing orders and the card-on-file arrangements entirely untouched. That is the whole point. The phone does not carry your card number. It carries a second one, cut for that phone, that the bank can throw away.

19.98 Common wrong ideas

Wrong: the phone sends your card number in encrypted or obscured form. **Right:** the card number is simply not there; a different sixteen-digit number, minted for that phone, occupies the account number field.

Wrong: tokens are safe. **Right:** a token on its own is just a number with no cryptography in it, and safety comes from the token plus a per-transaction cryptogram plus restrictions enforced by whoever is checking — merchant card-on-file tokens routinely run with no cryptogram at all.

Wrong: your card details never leave the device. **Right:** the real number is encrypted and sent to the network and your bank at provisioning; what is true and narrower is that the wallet cannot decrypt the token it receives back and its servers do not store it.

Wrong: your bank issues and controls the token in your phone. **Right:** for the overwhelming majority of device tokens the Token Service Provider is Visa or Mastercard, which owns the vault and the mapping table, mints the value, holds the validation keys and de-tokenises before the message reaches the issuer.

Wrong: tokenisation is a phone technology. **Right:** tokens are issued in enormous numbers to merchants, wallets and acquirers, living in ordinary databases with no secure element, no biometric and no per-tap cryptogram.

Wrong: the Payment Account Reference is a kind of substitute card number. **Right:** it is a 29-character identifier that cannot be reverse-engineered to a PAN or a token and cannot initiate a financial transaction.

Wrong: a receipt whose last four digits do not match the plastic is a misprint. **Right:** those are the token's last four digits, and they are the easiest available proof that the phone is not sending your card number.

Wrong: a reissued plastic card means re-adding the card to the wallet. **Right:** the TSP updates the token-to-PAN mapping in the vault and the token keeps working, which is also why a fraud reissue needs the issuer to suppress that update deliberately.

Wrong: host card emulation is a software secure element. **Right:** its keys sit in ordinary application memory and are made nearly worthless by limited-use thresholds, so an HCE wallet left unused and offline will eventually decline — correctly.

Wrong: the 2021 Express Transit attack broke tokenisation. **Right:** the token and the cryptography worked exactly as designed; what failed was one deliberately relaxed domain control, on one platform, for one scheme.

19.99 Chapter summary in 20 lines

1. When you tap your phone, your card number is not transmitted at all: a different number is.
2. That substitution is payment tokenisation, specified by EMVCo in a royalty-free, publicly available technical framework.
3. A Payment Token is a 13 to 19 digit Luhn-valid surrogate drawn from a real ISO BIN range, which is exactly why it traverses the existing network unchanged.
4. It carries its own expiry date, mapped by the vault to the account's expiry and not obliged to equal it.
5. It is minted by a Token Service Provider registered with EMVCo under a three-digit code, on behalf of a Token Requestor identified by an eleven-digit ID.
6. The vault holding the token-to-PAN mapping usually belongs to the network rather than to your bank, which is a real operational dependency invisible from the cardholder's side.
7. On a phone the token is written into a secure element, a certified Java Card platform chip, and the key that signs each tap never leaves it.
8. Every tap produces an eight-byte cryptogram from that key, a counter that only increases and the terminal's unpredictable number, so it cannot be replayed anywhere.
9. The device also asserts that the holder was verified on the device, which is why a phone pays £180 where a plain contactless card stops at £100.
10. A stolen token is close to worthless because of three separate walls: domain restriction, the missing cryptogram key, and the absence of any link back to you.
11. Domain restriction is enforced by token requestor ID, by POS entry mode and by merchant identity — and only as strongly as the checking party implements it.
12. Your real card number does leave the phone once, at provisioning, encrypted to the network and the issuer, because they must know which account you are asking to tokenise.
13. Provisioning is therefore the security-critical moment, gated by identification and verification whose outcome the industry calls green, yellow or red.
14. Yellow is where the design lives or dies, because a token stuck awaiting activation is a card the customer believes they have added and cannot use.
15. Where there is no secure element, Android's host card emulation holds keys in application memory and neutralises the risk with limited-use keys bounded by time, count or amount.
16. The practical difference between the two architectures is availability rather than cryptography: an HCE wallet that cannot reach the network eventually stops paying.
17. Because one account now wears many numbers, the industry had to invent the Payment Account Reference to tie them back together without exposing the account.
18. The token has a lifecycle of its own and can be suspended, deactivated or remapped in the vault in seconds, leaving the plastic, the account and every card-on-file arrangement untouched.
19. The only serious public break of the model, the 2021 Express Transit relay, exploited a deliberately relaxed exemption rather than any cryptography.
20. The phone does not carry your card number; it carries a second one, cut for that phone, that the bank can throw away.

Sources: EMV Payment Tokenisation Specification – Technical Framework, EMVCo, v2.4 (9 July 2026) and v2.3 glossary; EMVCo Token Service Provider and BIN Controller ID registration programme pages; EMV ICC

Specifications for Payment Systems Book 3 Annex A for base tag definitions; EMV Mobile Payment: Software-Based Mobile Payment Security Requirements and the SBMP Security Evaluation Process, EMVCo; EMVCo Payment Account Reference: A Primer, Secure Technology Alliance, April 2018; EMV Payment Tokenization Primer and Lessons Learned, U.S. Payments Forum, June 2019; Technologies for Payment Fraud Prevention: EMV, Encryption and Tokenization, Smart Card Alliance, October 2014; Industry Perspectives on the Evolution of EMV Payment Tokenization, Federal Reserve Bank of Boston, 2019; Apple Platform Security — card provisioning, Apple Pay component security and payment authorisation; Apple PassKit in-app provisioning documentation; Google Pay device tokenisation developer documentation, including ID&V best practices and app-to-app ID&V; Mastercard Developers, MDES pre-digitisation and customer service documentation; Visa Smart Debit/Credit and Visa payWave U.S. Acquirer Implementation Guide for DE 22 values; Android host-based card emulation developer guide; ISO/IEC 14443, ISO/IEC 7816-4 and ISO/IEC 7816-5; Radu, Chothia, Newton, Boureanu and Chen, Practical EMV Relay Protection, University of Birmingham and University of Surrey, 2021; Basin, Sasse and Toro-Pozo, The EMV Standard: Break, Fix, Verify, IEEE S&P 2021; PCI Security Standards Council FAQ on acquiring, issuer and payment tokens; Visa newsroom, 4 June 2024; Mastercard newsroom, June 2025. Tag collisions on `9F6E` *cross-checked against EFTlab's EMV and NFC tag list and emvlab.org.*

20

CHAPTER

The Terminal

20.0 What this chapter gives you

1. You will be able to explain why a card machine wipes its own keys the instant its case is opened, and why the result is a device that will never take another payment.
2. You will be able to say what tamper response protects and what it does not, and why overlays, shimmers and prompt manipulation make the enclosure the wrong place to look.
3. You will be able to name the three certification regimes, say who runs each and what each actually tests, and stop hearing "EMV certified" as a security statement.
4. You will be able to read a PCI approval listing and see that it attaches to a hardware revision, a firmware version and an expiry date rather than to a product name.
5. You will be able to explain why deployment takes months, and point at the stage of the process that repeats per scheme and per configuration.
6. You will be able to separate minor terminal settings from major ones, and say which change invalidates the description of an approved configuration.
7. You will be able to pick the right PCI approval class for a form factor, from a countertop terminal to a fuel dispenser to a PIN pad module inside somebody else's kiosk.
8. You will be able to explain why an unattended fuel transaction runs as a pre-authorisation and a completion, and why the enclosure that matters there is the dispenser's door.
9. You will be able to say what changed with softPOS: that PCI lists a whole solution rather than a phone, and that attestation and monitoring replaced the mesh.
10. You will be able to point at the four kernel data objects where the contactless limits actually live, and explain why a regulator can change a rule overnight and an estate cannot.

The previous chapters were about the card. Each of them had a second party standing just off the page, holding out a slot or a radio field and asking the questions. This chapter is about that second party.

It is the least glamorous object in payments and the most heavily regulated. A card is a piece of plastic a bank posts to you. A terminal is certified hardware built under audit, attacked in a laboratory with a scalpel and an oscilloscope, carrying an approval number valid until a stated date, and becoming an inert brick the moment somebody opens the case. Three separate certification regimes sit on top of it, run by three different sets of people, none of whom accepts the others' paperwork. That is why it takes months rather than days to put a new one into a shop, and why the machine in a corner shop in Leeds and the one at a motorway fuel pump in Perth are not the same class of object at all.

The plain version

Think of the card machine as a **locked translation booth**.

Two parties who cannot speak to each other need to do business. One is your card, which speaks a compressed technical language and will only answer certain questions in a certain order. The other is your bank, hundreds of miles away, which speaks a different language entirely. Between them sits a booth with a window on each side and a translator inside, who listens to the card, writes the question down in the bank's language, sends it off, waits, and tells everybody what happened.

Now the important part. The translator also hears your PIN. That single fact changes everything: it is why the booth is locked, why the walls are lined with sensors, and why there are rules about who may build one, who may deliver one, and what must happen if anybody ever gets inside.

The dye pack

Banks used to put dye packs into bags of cash. Force the bag open away from the counter, a small charge goes off, the notes are stained bright red, and the robbery has produced a bag of worthless paper. The money simply stops being money. The card machine works on exactly that principle, and it is the single most useful thing to understand about it.

Inside every one of these machines are secret numbers, called keys. They let the machine scramble your PIN so nobody along the wire can read it, and without them it cannot take a payment at all. The keys are worth stealing; the plastic case is not.

So the case is wired. A fine mesh, thinner than a hair, is printed around the inside carrying a small current. There are little switches under the screws, and sensors that notice if the machine gets very cold, or if somebody slows its clock down, or if light reaches a place inside where light never reaches. All of it runs day and night, even while the terminal sits unplugged in a stockroom.

Cut the mesh, undo the wrong screw, freeze it, drill it: the machine wipes its keys. Not "logs an alert". Wipes them, in the instant, permanently. Switch it back on and the screen says something like TAMPER, and it will never take a payment again. It goes back to the manufacturer to be rebuilt and re-keyed.

The thief has the bag. The money is red.

The family of machines

The booths do not all look the same, because the jobs differ.

There is the **countertop** one by the till, and the **portable** one the waiter carries to your table: the same machine with a battery and a radio, docking into a base near the bar. There is the **little reader that pairs with a phone**, which market traders and plumbers use: the phone does the amount and the internet, the reader does the card and the PIN.

There is the **unattended** one: the fuel pump, the car park machine, the vending machine, the ticket kiosk. Same job, no human anywhere near it, and that absence is the difficulty. In a shop, if somebody kneels at the counter for ten minutes with a screwdriver, the shopkeeper notices. At three in the morning at pump four, nobody notices.

There is the **integrated till**, where the card machine and the shop's computer talk so nobody keys the amount twice. That sounds like a small convenience. It is where most of the interesting failures happen, because now there are two computers and only one of them is certified.

And lately there is the machine that has no machine: **the shop's own phone is the terminal**, and you tap your card on the back of it. That required an entirely new rulebook, because every rule until then assumed a sealed case with a mesh in it.

A worked example: £34.60 in a bakery

Here is a sale in a small bakery with one countertop terminal. The baker keys **£34.60** into it; the screen shows 34.60 and the contactless symbol lights. You tap your card. The radio side wakes, powers your card through the air and asks which payment applications it carries; your card answers with a list, and the terminal picks one.

Now the terminal checks its own rulebook. It has been given four numbers by whoever runs its payments:

The rule	What it means	This bakery's setting
Contactless floor limit	Above this, always ask the bank	£0 — always ask
Contactless transaction limit, no phone check	Refuse a plain tap above this	£100
Contactless transaction limit, phone checked you	Refuse a phone tap above this	£5,000
CVM required limit	Above this, demand a PIN	£100

£34.60 is under £100, so no PIN. The floor limit is zero, so the bank gets asked anyway.

The terminal asks your card to sign the transaction. The card produces an eight-byte code — computed from the amount, the date, the currency, a random number the terminal invented on the spot and a counter inside the card — and hands it over. The terminal packs that code, the card number, the amount and about forty other small facts into a message and sends it down the line. Two seconds later the bank approves, the terminal beeps and prints, and the baker gives you your bread.

Now change one thing. Make it **£134.60**, because you bought a cake. Same tap, same card, but 134.60 is above the CVM required limit of £100, so the terminal asks for a PIN. You put the card in the slot and type four digits. Those digits never leave the sealed part of the machine as digits: they go straight into the secure processor, which scrambles them with one of its keys into a block of gibberish. The gibberish travels; the PIN does not.

If somebody has glued a fake keypad over the real one — an *overlay* — they get your PIN and the machine never knows, which is the first hint that the dye pack is not the whole story.

Why a new machine takes months

The baker is switching payment provider, and assumes a new terminal is like a new kettle: order Tuesday, plug in Thursday. It is not, because three different bodies must each be satisfied by their own tests.

Test one: does the plug fit? Can this reader physically talk to a card — the right voltages in the right order in the slot, a radio field of the right strength so that a card held at a sensible distance actually answers? Electrical and mechanical, done once, by the manufacturer, for the model.

Test two: does the software know the rules? Inside the terminal is a program that knows the whole question-and-answer sequence a chip card expects: what to ask, in what order, what to do with each answer, when to go online, when to decline. It is tested against a very large book of scenarios, again by the manufacturer.

Test three: does *this* setup, talking to *this* bank, work end to end? This is the one that takes months, because it must be redone for the bakery's particular combination of terminal, software version, payment provider and card schemes. Every scheme has its own test list; every test is a scripted transaction with a test card, checked field by field. One wrong value and that section starts again. Then someone must load the secret keys, two people present, in a room built for it, before the machine can take its first real payment.

That is the sentence to carry to dinner: **the card machine is a locked booth that destroys its own secrets if you open it, and it takes months to deploy not because the hardware is hard but because three separate examiners must each pass it, and only the last of them cares about your particular shop.**

Where the plain version stops being true

The booth is a good picture, and it is wrong in four ways that matter to anyone who buys, builds or certifies one.

There is no such thing as "the terminal"

The plain version speaks of one object with one approval. In the paperwork there is no such object. There is a contact interface module, separately approved; a proximity coupling device — the radio front end — separately approved; a contact kernel and up to eight contactless kernels, each separately approved; a PIN entry device with its own security approval, which may be a physically separate module bought from another vendor; a payment application on top of the kernels, which is nobody's type approval and everybody's problem; and the till, certified by no one.

Approvals attach not to a product name but to a product name **plus a hardware revision plus a firmware version plus a checksum**. Change the firmware and you may or may not still be approved, depending on which requirements the change touches; the process for answering that has a name, *delta evaluation*, and a laboratory must run it. Where a device embeds other approved devices, the renewal date of the whole is the earliest among all of them: one expiring component expires the terminal. So "is this terminal certified?" is not a yes or no question. It is a query against several lists, each with its own dates.

Tamper response protects the keys, not the transaction

Tamper response guarantees one thing: an attacker who penetrates the enclosure gets no usable cryptographic material. It guarantees nothing about attacks that never penetrate it, and those are the ones that take money. Overlays: a moulded plastic keypad sitting on top of the real keypad, recording digits, defeated by nothing inside the box. Shimmers: a paper-thin device pushed into the card slot to sit between card and reader. Prompt manipulation: persuading the terminal to display "ENTER PIN" when the software behind it is not doing what the cardholder thinks. And at unattended sites the enclosure that matters is the *dispenser's* door, not the payment module's.

PCI's requirements do address prompt control, and its technical FAQ spends considerable space on touchscreens precisely because a bezel can conceal an overlay. But the honest statement is narrower than the analogy: opening the box is one of the harder ways in, which is why intelligent attackers stopped trying.

The three certifications test three unrelated things

"EMV certified" is used in the trade as though it were a security statement. It is not. EMVCo's functional approvals assess performance and compatibility: Level 1 asks whether the electrical and radio interface conforms, Level 2 whether the kernel implements the specified processing logic. Neither asks whether the device resists attack.

The security standard for the terminal is PCI PTS POI, from a different organisation, tested in different laboratories, against a different document. Level 3 is a third thing again: not run by EMVCo, not a security test, and not a conformance test in the usual sense, but an integration test whose content each payment system writes for its own acquirers. Merchants routinely assume that possession of any one implies the others. Nothing implies anything.

With softPOS, the phone is not the terminal

The last form factor breaks the analogy rather than stretching it. When the shopkeeper's own phone accepts the tap there is no sealed enclosure, no tamper mesh, no key store you control and no PCI PTS approval, because PCI PTS approves hardware and there is no hardware to approve. What replaced it is a different architecture: cryptographic work pushed into whatever secure element the phone already has, plus continuous *attestation* — the phone proving to a back-end service, repeatedly, that it is the device it claims to be and has not been rooted, hooked or emulated — plus *monitoring*, a server-side system that can refuse to let a suspect device transact. The certified thing is the whole solution including its servers, not the handset.

The security boundary therefore moved off the counter and into somebody's data centre, and a softPOS solution can be switched off remotely in a way a countertop terminal never could: a real gain and a real new dependency, and the booth analogy has nothing to say about either.

The technical version

What is physically inside

A conventional attended terminal contains a contact card reader, a contactless front end, a secure processor with a protected key store, a keypad, a display, communications and usually a magnetic stripe head and a printer.

The **contact interface** is governed by ISO/IEC 7816: part 2 defines the eight contacts and their positions, part 3 the electrical characteristics and the transmission protocols, of which EMV uses the character-oriented T=0 and the block-oriented T=1. The reader must handle the class A (5 V), class B (3 V) and class C (1.8 V) supply classes cards may demand, negotiate the answer-to-reset, and cope with cards inserted crookedly or withdrawn mid-transaction. In EMVCo's vocabulary this component is the **Interface Module**, or IFM.

The **contactless front end** is governed by ISO/IEC 14443, at 13.56 MHz with an initial data rate of 106 kbit/s, supporting Type A and Type B modulation and framing, and supplies the card's power through the field. In EMVCo's vocabulary it is the **Proximity Coupling Device**, or PCD, specified in the EMV Contactless Interface Specification with the protocol layer in Book D of the EMV Contactless Specifications for Payment Systems.

The **secure processor** holds the keys, performs PIN block encryption, executes firmware whose authenticity it checks at boot and drives the tamper circuitry. It is permanently powered from a coin cell or supercapacitor, so the tamper sensors run while the terminal sits unplugged.

The catalogue of form factors

Form factor	Typical PCI PTS approval class	Notes
Countertop	PED	Attended, mains, fixed line or Ethernet
Portable, with base	PED	Same silicon, battery, Bluetooth or DECT to a base
mPOS reader plus phone	SCRP, or PED with SRED	Reader takes card and PIN; phone does UI and comms
Unattended kiosk, fuel, parking, vending	UPT, containing an EPP	Cardholder-operated, no attendant
PIN pad for an ATM or kiosk	EPP	A module, not a complete terminal
Card reader with no PIN	SCR, or Non PED	PIN handled elsewhere or not at all
softPOS on a merchant's own phone	none, outside PTS	Covered by PCI MPoC instead

PCI SSC's approval classes, exactly as the approved-devices listing filters them, are PED, EPP, HSM, Multi-tenant HSM, UPT, Non PED, SCR, SCRP, KLD (key loading device) and RAP (remote administration platform). The PED/EPP distinction matters commercially: an EPP is a module supplied to an ATM or kiosk manufacturer, and a UPT built around an approved EPP still requires its own laboratory evaluation, because the final enclosure introduces cardholder interfaces the EPP vendor never saw. The OEM's module cannot itself receive a UPT approval; the final form factor vendor's product does, and the module is listed separately as an approved component.

EMV classifies the environment in a single byte, tag `9F35`, Terminal Type, defined in Annex A of EMV Book 4. The first digit is who controls the terminal: 1 a financial institution, 2 a merchant, 3 the cardholder. The second digit carries environment and capability together, 1 to 3 attended and 4 to 6 unattended, and within each triple the order is online only, offline with online capability, offline only.

Operated by	Attended	Unattended
Financial institution	11 12 13	14 15 16
Merchant	21 22 23	24 25 26
Cardholder	not defined	34 35 36

A supermarket lane is typically 21 or 22, a fuel dispenser 24 or 25, a bank-operated ATM 14. Terminal Type is a *major* setting: change it and the Level 2 approval of the configuration no longer describes the device.

The PIN entry device as a certified object

The security standard is the **PCI PIN Transaction Security Point of Interaction Modular Security Requirements**, published by the PCI Security Standards Council. Version 7.0 was published on 29 May 2025, superseding v6.2, and PCI SSC describes it as containing 59 requirement changes and 23 pieces of additional guidance arising from two requests for comment during 2024. Among them: a new requirement for the physical and logical security of biometric interfaces; a new requirement permitting third-party applications, that is, app stores, on a POI device; an allowance for keys not to be zeroised on tamper where forward secrecy is used and extraction would require destroying the

processing element; and a rule that terminal security keys, such as firmware authentication and tamper or storage keys, must use cryptography with an effective key strength of 128 bits or stronger.

The standard is *modular*, which is the point of its name. The evaluation modules are: POI Device Core Requirements, covering core logical and physical requirements; Device Integration Requirements, which apply whenever previously approved components are combined; Open Protocols, covering the interface to open networks; Secure Reading and Exchange of Data, or SRED, covering encryption of account data inside the POI; and Device Management Security Requirements, covering how the device is produced, controlled, transported, stored and used across its life. Products need not support Open Protocols or SRED, but a product that implements either must be evaluated against that module.

Within those modules the requirements are lettered rather than numbered straight through: the core physical requirements run from A1, the core logical requirements from B1, and the SRED requirements are the K series, which is why practitioners talk about "requirement K20" rather than a section number.

Security is measured, not asserted. The standard uses an attack-costing framework expressed in points, combining the effort to identify an attack with the effort to exploit it. The headline physical requirement, A1, requires that defeating the device's tamper protections to disclose sensitive material demands an attack potential of at least **26 points for identification and initial exploitation, with a minimum of 13 for initial exploitation**. The same figures recur through the physical requirements, including side-channel monitoring during PIN entry.

Some specifics from PCI's technical FAQ that decide whether a device passes:

Open Protocols. New POI evaluations using the Internet Protocol suite must support TLS 1.2 or higher, with cipher suites providing at least 112 bits of security. Suites using triple DES are no longer permitted, on the grounds that a 64-bit block size does not protect large volumes encrypted under one key. Where a device runs Android, the version must be one officially supported with security patches; reports submitted with an unsupported version are rejected, and where Google no longer patches, the vendor must evidence its own monthly patching, validated by the laboratory.

Key loading. The initial terminal master key must be loaded by asymmetric techniques or manual ones such as the keypad, an IC card or a key loading device; plaintext keys and their components are never permitted over a network connection. Key blocks are mandatory: ASC X9 TR-31 for triple DES keys, TR-31 or ISO 20038 for AES, or a demonstrably equivalent scheme that has passed independent expert review. Remote key distribution using asymmetric techniques follows ASC X9 TR-34, in either its two-pass form using nonces or its one-pass form using timestamps, and binding the device to its key distribution host is a prerequisite for both. PIN block formats are constrained too: ISO 9564 format 4, the AES format, is treated explicitly, with its use distinguished under DUKPT, fixed key and master/session key schemes. Where a device is compromised, keys are not reloaded by any methodology; the device is withdrawn.

Approval identity and expiry. A PCI listing identifies a device by model name and number, a hardware number that may contain wildcard positions, a firmware number, an approval number and an expiry date, alongside a vendor-published security policy setting out the roles the device supports and the services each may use. The device must perform only its designed functions; no hidden functionality is permitted. Approvals expire, and the vendor must have the model re-evaluated against the current version before the renewal date. PCI SSC's listing records that version 3 POI approvals expired on 30 April 2021 and version 4 on 30 April 2024. Version 5 approvals were due to expire on 30 April 2026, and a Council bulletin of September 2025 extended them by one year to **30 April 2027**, which is now the earliest expiry date on the approved-devices list. These dates govern new deployments rather

than the instant removal of installed units, but they are why estates are replaced on a cycle unrelated to hardware wearing out, and why the version of an approval matters as much as its existence.

Tamper: what actually happens

PCI's technical FAQ leaves no room for a cheaper design. A card reader or non-PIN device cannot meet the physical requirements through tamper resistance and tamper evidence alone; it must have permanently active tamper detection that monitors for intrusion and responds with the immediate erasure of sensitive information, rendering the device inoperative. More broadly, in PCI's words: "A device cannot meet PTS POI requirements without having an active tamper response mechanism to zeroize secret and private keys during a penetration attack regardless of which modules of the PTS POI standard the device is designed to comply with."

The mechanisms in a real device are layered. A flexible printed circuit or serpentine trace forms a mesh wrapping the secure area, monitored for open circuit, short circuit and resistance change, so that cutting it, bridging it or drilling through it all register. Switches detect case separation, light sensors an opened enclosure, and further sensors out-of-range temperature, supply voltage and clock frequency, defeating the classic laboratory tricks of freezing memory so that it retains its contents or under-clocking a processor to make its behaviour observable. Where the device offers a service hatch, requirement K20 of the SRED module obliges the design either to make sensitive material physically unreachable from that hatch or to erase it when the hatch is opened.

The response is defined as tightly as the detection. On tamper the device must become **immediately inoperable** and must automatically erase secret information such that recovery becomes infeasible. The FAQ closes the obvious loopholes: keys that could be used to download further keys must be zeroised or rendered unusable for that purpose, covering both symmetric key-encrypting keys and the private keys used in asymmetric key loading; and where a device encrypts other keys under an internally generated key held in a register, erasing that one register suffices, provided the encrypted keys are protected under compliant algorithms and key sizes.

A tampered PIN-acceptance device must immediately stop processing all PIN-based transactions. If a reset is implemented at all, only one is permitted unless the device is removed for inspection and repair, and any intervention that re-enables transactions requires on-site presence, dual control, logged user identities with timestamps and actions, and authentication values that cannot be replayed on the same or another device. Decommissioning is a security event too: PCI PIN Security requires that a device removed from service has all keys destroyed that were, or could have been, used for any cryptographic purpose — firmware validation keys, display prompt control keys and keys used during loading included — and if that cannot be assured, the device is physically destroyed.

The practical consequences are unglamorous and expensive. Terminals tamper for reasons unconnected with attack: dropped on a tiled floor, left in a van overnight in February, battery flat for months in a drawer, or a well-meaning engineer removing a case screw. Each produces a device that cannot be repaired in the field, must be returned and must be re-keyed under dual control. It cannot be swapped for a spare unless the spare was keyed in advance, so estates carry keyed spares, and keyed spares are themselves controlled inventory.

EMV certification, level by level

Level 1 tests the electrical, mechanical and radio-frequency interfaces. For contact the object of the test is the IFM; for contactless, the PCD. EMVCo defines the processes and recognises the laboratories but performs no testing itself: laboratories test with qualified tools and submit reports, and EMVCo issues a Letter of Approval when a report demonstrates sufficient conformance. Contactless Level 1 includes

range and positioning tests, measuring how close a card must be held before the exchange succeeds — a functional question with a large effect on how the terminal feels to use.

Because a phone's antenna is not designed to be a payment reader's antenna, EMVCo launched a **Reduced Range PCD Level 1 approval process** on 10 September 2024, defining two reduced-range compliance levels with different read-range and positioning requirements, specifically so TapToMobile devices could be measured at all.

Level 2 tests the kernel. For contact, the governing document is the EMV Integrated Circuit Card Specifications for Payment Systems, version 4.4 of October 2022, in four books: Book 1 on the application-independent ICC-to-terminal interface, Book 2 on security and key management, Book 3 on the application specification, Book 4 on cardholder, attendant and acquirer interface requirements. EMVCo now calls this the Contact Kernel Approval Process; the vendor registers once, declares the product in an Implementation Conformance Statement on a versioned form — 4.4c and 4.4d are the current ones — has a recognised laboratory run the Terminal Level 2 Test Cases, and receives a **four-year** Letter of Approval.

For contactless the architecture is different and routinely misdescribed. The EMV Contactless Specifications for Payment Systems comprise Book A (Architecture and General Requirements), Book B (Entry Point), Books C-1 to C-8 (the kernel specifications) and Book D (the contactless communication protocol). Books A and B and the C-x kernel books stood at v2.12, published 6 July 2026. Books C-1 to C-7 are the individual payment systems' kernels, published by EMVCo on their behalf, with test plans managed by the relevant scheme rather than by EMVCo. Book C-8 is EMVCo's own kernel, separately versioned at v1.2 and licensable on the same terms as the contact specification; it introduces a split architecture allowing processing in different locations including the cloud, Secure Channel, Blinded Diffie-Hellman and elliptic-curve cryptography, and it may be tested either inside a full acceptance device or standalone.

A contactless product approval requires the successful validation of the PCD, of Entry Point (Books A and B) and of at least one kernel, and ends with a **three-year** Letter of Approval — one year shorter than the contact kernel's. Where a manufacturer's architecture complies with EMVCo's Modular Architecture Requirements, an EMVCo-qualified compliance auditor must validate that architecture and report to EMVCo before approval, and separate lighter processes exist for a product change, a derivative product and a renewal.

The granularity of an actual approval surprises people. A contact Level 2 Letter of Approval names a kernel version, tested on a named terminal model, with a named PIN pad running named firmware, on a named operating system. It carries approval numbers of the form `2-05820-1-1S-0426-4.4c`, in which the trailing element is the specification version tested against and the four digits before it the month and year of approval, plus a renewal date four years out, a laboratory report identifier, an eight-hex-digit kernel checksum and a separate configuration checksum. EMVCo states in it that the samples sufficiently conform, and that the letter is valid while the approval number is posted on the EMVCo website — which is to say, EMVCo can end it.

Attached is the Implementation Conformance Statement, and it is the ICS, not the letter, that tells you what the terminal can do. It enumerates, item by item: manual key entry, magnetic stripe and contact IC input; plaintext PIN, enciphered online PIN, paper signature, offline enciphered PIN, no-CVM and biometric CVMs; SDA, DDA, CDA and Extended Data Authentication; whether the terminal performs floor limit checking, random transaction selection and velocity checking; whether it holds a transaction log or exception file; and whether it supports PIN bypass.

That granularity is why re-certification is a live risk rather than a formality. Terminal settings divide into *minor* ones — Terminal Country Code, tag `9F1A`, or Transaction Currency Exponent, tag `5F36` — changeable freely, and *major* ones, such as Terminal Type, tag `9F35`, whose change invalidates the description of the approved configuration.

Level 3 is the integration test, and EMVCo describes its own role narrowly: L3 validates the integration of an approved acceptance device with its acceptance infrastructure, and the test plans are provided not by EMVCo but by the **Participant Systems** — the payment systems and other entities that elect to use the framework. What EMVCo supplies is the **EMV L3 Testing Framework**: standardised test-tool components, a qualification process for L3 test tools, and a Participant System Identifier service that lets domestic schemes place their own requirements inside a qualified tool alongside the international schemes', so the tool selects and runs the right tests for the right system. In practice your acquirer or processor obtains a qualified tool, loads the relevant test plans and runs scripted transactions against your terminal and your host, scheme by scheme and configuration by configuration, each result inspected field by field.

Why deployment takes months

Stage	Who does it	Repeats for
Select hardware; confirm PTS approval class, version and expiry	Merchant or acquirer	Each model
EMV L1 for IFM and PCD	Manufacturer, before you buy	Each hardware revision
EMV L2 for contact and each contactless kernel	Manufacturer, before you buy	Each kernel and configuration
Payment application build and configuration	Terminal estate owner	Each acquirer and each terminal type
EMV L3 integration testing	Acquirer or processor with a qualified tool	Each scheme, each configuration
Key injection under dual control	Key injection facility or acquirer	Each device
PCI DSS or P2PE scoping of the surrounding estate	Merchant	Each deployment model
Estate rollout, terminal management system, remote updates	Estate owner	Continuously

Two features of that table explain the months. First, the repeat column: L3 is per scheme and per configuration, so an estate supporting four schemes across attended, unattended and mobile configurations is not running one certification but a matrix of them. Second, the delta problem: a change to the payment application, the kernel version, the terminal type or the CVM configuration can reopen the L3 matrix and, if it touches security, the PTS evaluation as well. A laboratory test cycle cannot be paused and resumed either: if it ends with discrepancies, it ends.

Unattended terminals

The unattended payment terminal is not a countertop terminal in a metal box. PCI treats the UPT as its own approval class, covering cardholder-operated devices that read, capture and transmit card information in conjunction with an unattended self-service device, with fuel dispensers, ticketing machines and vending machines named explicitly, and notes that UPTs may have a compound architecture combining payment with the delivery of goods or services. That compound architecture is the difficulty: the payment module and the thing being sold share an enclosure, a controller and often a

maintenance regime, and the people who service the dispenser are not the people who understand the payment module.

Three consequences follow. The first is **prompt control**: in an attended environment the cardholder can see who is asking for the PIN, while in an unattended one the display and keypad may be driven by a device controller outside the secure boundary, which is why PCI constrains what may be displayed during PIN entry, and why an EPP with no display of its own must be integrated with a display it can trust.

The second is **inspection**. An overlay on a countertop keypad has a shopkeeper looking at it all day; an overlay inside a fuel dispenser has nobody. Estate owners answer with tamper-evident seals, dispenser door alarms wired into the site controller and scheduled physical inspection — none of it in the terminal standard, all of it in the operational regime around it.

The third is **the transaction shape**. An unattended fuel transaction cannot know its final amount before authorisation, because the fuel has not been dispensed; it runs as a pre-authorisation for an estimated amount followed by a completion for the actual amount, so the terminal must support offline data capture and later completion messaging, and the terminal type byte reflects that offline capability. Vending and parking behave differently again, often as small online transactions with no CVM at all. Fuel also has its own integration standard, maintained by IFSF, between forecourt controller, dispenser and payment application.

softPOS: when the phone is the terminal

The standards arrived in three steps. PCI SSC published **Software-based PIN Entry on COTS** (**SPoC**) in 2018 for PIN entry on an ordinary phone paired with a secure card reader, and **Contactless Payments on COTS** (**CPoC**) in 2019 for contactless acceptance on the phone's own NFC with no PIN. It then combined them into **Mobile Payments on COTS** (**MPoC**), first published in November 2022, with version 1.1 in November 2024, whose principal addition is allowing both PIN entry and contactless card data on the same commercial off-the-shelf device.

"Objective-based", PCI's own description of MPoC, is doing real work. PCI PTS specifies a physical enclosure and tests it with tools; MPoC cannot, because the enclosure is a phone the solution provider did not design and cannot control. It therefore states security objectives and requires evidence that the solution meets them, across the software, the back-end, the attestation mechanism and the monitoring system. Solutions, not devices, are listed.

Apple's implementation is the best-documented public example. In Tap to Pay on iPhone the Secure Element hosts the payment acceptance applets and the kernel configurations, downloaded from Apple's servers and validated for integrity and authenticity before installation. The applet performs the card read and then encrypts and signs the card data, so the card number is, in Apple's words, "secured by the Secure Element and isn't visible to the acceptance device", and is released only to the merchant's payment service provider. The Secure Element also captures the PIN digits and builds the encrypted PIN block; screenshots and screen recording are blocked during PIN entry; and because the phone has been facing the payer, the merchant must re-authenticate with Face ID, Touch ID or a passcode to get control of it back. Apple states that on iOS 18.4 or later Tap to Pay on iPhone is a PCI MPoC validated solution listed by PCI SSC, and that its servers monitor device security compatibly with MPoC. Store-and-forward transactions do not support PIN entry.

On the EMVCo side nothing is waived. A TapToMobile solution still needs Level 1 for its reader, which is why the Reduced Range PCD process exists, Level 2 for its kernels, which is why kernel configurations are provisioned onto the Secure Element at all, and Level 3 with each acquirer. So

softPOS is not "a phone with PCI approval" — PCI PTS does not approve phones — but a listed solution of which the phone is one component, and it does not remove certification effort so much as move it: the hardware burden goes and a continuous attestation, monitoring and lifecycle burden takes its place.

The integrated till, and the interface nobody certifies

An integrated deployment connects the electronic cash register to the payment terminal so the basket total flows across automatically and the result flows back. In a *fully integrated* design the ECR drives the payment device directly and is therefore in scope for card data. In a *semi-integrated* design — now the common architecture — the ECR sends only an amount and receives only a result, card data never traverses it, and the till leaves PCI DSS scope for that data, which is very often the entire commercial reason the architecture exists.

The interface between them is covered by no EMVCo or PCI approval. In practice it is either proprietary to the terminal vendor or acquirer, or an industry protocol: **nexo standards**' Retailer protocol, built on ISO 20022 messages, specifying the interface between payment application and ECR, with the companion nexo FAST specification defining the payment application itself; or IFSF's equivalents in fuel. Choosing an open protocol does not shorten Level 3, which tests the messages that reach the acquirer, not those that cross the counter.

The terminal's own configuration

Everything a terminal decides is driven by data loaded into it, per acquirer and per application identifier. These are the terminal-side data objects a practitioner meets most often, as defined in EMV Book 3 Annex A.

Tag	Name	Length	Note
9F1A	Terminal Country Code	2	ISO 3166-1 numeric
5F2A	Transaction Currency Code	2	ISO 4217 numeric
5F36	Transaction Currency Exponent	1	Minor unit position
9F35	Terminal Type	1	Table above
9F33	Terminal Capabilities	3	Card input, CVM, security capability
9F40	Additional Terminal Capabilities	5	Transaction types, data input and output
9F1B	Terminal Floor Limit	4	Per application
9F1C	Terminal Identification	8	Unique within a merchant
9F1E	Interface Device Serial Number	8	Unique to the physical device
9F15	Merchant Category Code	2	ISO 18245
9F16	Merchant Identifier	15	Assigned by the acquirer
9F4E	Merchant Name and Location	variable	
9F09	Application Version Number (terminal)	2	Per AID
9F06	Application Identifier (terminal)	5 to 16	The AID the terminal supports

Tag	Name	Length	Note
9F1D	Terminal Risk Management Data	1 to 8	Optional; Kernel 2 uses 8
9F02	Amount, Authorised	6	n12, minor units
9F03	Amount, Other	6	Cashback
9F37	Unpredictable Number	4	Generated per transaction
95	Terminal Verification Results	5	Output: what the terminal found
9B	Transaction Status Information	2	Output: what the terminal did
9F34	Cardholder Verification Method Results	3	Output: how, and whether it worked

Alongside these sit the **Terminal Action Codes** — Default, Denial and Online, five bytes each, held per application identifier. They are terminal-resident acquirer policy, bit-masked against the Terminal Verification Results together with the card's Issuer Action Codes to decide whether to decline offline, go online or approve offline. Getting them wrong is the classic cause of a terminal that approves offline what it should have sent online, and it is exactly what Level 3 exists to catch.

The contactless reader carries a second set of limits, held by the kernel rather than the contact application. In the Kernel 2 configuration these are the Reader Contactless Floor Limit at tag DF8123, the Reader Contactless Transaction Limit without on-device CVM at DF8124, the Reader Contactless Transaction Limit with on-device CVM at DF8125, and the Reader CVM Required Limit at DF8126. Those four values, per application and per currency, decide whether a tap goes through, whether it demands a PIN and whether it is refused outright.

Which is where this chapter ends, because those four numbers are also where policy lands. From 19 March 2026, following an FCA announcement in December 2025, United Kingdom firms with strong fraud controls have been permitted to set their own contactless limits in place of the old fixed pair of £100 per transaction and £300 in the background. UK Finance's guidance observes that most banks were likely to keep existing limits at first, and that "separate changes would also need to be made to card acceptance terminals and the industry rules that govern them to process contactless card payments above £100". A regulator can change a rule overnight. Changing DF8124 across a national estate is a deployment programme, and mobile wallets were never inside the cap in the first place, because they are governed by DF8125.

20.98 Common wrong ideas

Wrong: a terminal is one object with one approval. **Right:** it is a contact interface module, a proximity coupling device, a contact kernel and up to eight contactless kernels, a PIN entry device and an uncertified payment application, each with its own paperwork and its own dates.

Wrong: the approval belongs to the product. **Right:** it belongs to a product name plus a hardware revision plus a firmware version plus a checksum, and where approved devices are embedded in others the earliest renewal date among them expires the whole terminal.

Wrong: tamper response protects the transaction. **Right:** it guarantees only that an attacker who penetrates the enclosure gets no usable cryptographic material, and the attacks that take money — overlays, shimmers, prompt manipulation — never go inside the box.

Wrong: "EMV certified" means the device is secure. **Right:** EMVCo Level 1 tests the electrical and radio interface and Level 2 the kernel logic; neither asks whether the device resists attack, and the security standard is PCI PTS POI from a different organisation entirely.

Wrong: Level 3 is another conformance test. **Right:** it is an integration test whose plans each payment system writes for its own acquirers, run in a qualified tool, repeated per scheme and per configuration.

Wrong: a tampered terminal can be reset on site. **Right:** it cannot be repaired in the field; it goes back to be re-keyed under dual control, which is why estates carry pre-keyed spares as controlled inventory.

Wrong: terminals tamper because somebody attacked them. **Right:** most tamper because they were dropped, left in a van overnight in February, allowed to go flat in a drawer, or opened by a well-meaning engineer — and the response is identical.

Wrong: an approved encrypting PIN pad inside a kiosk makes the kiosk approved. **Right:** the final form factor needs its own evaluation, because the finished enclosure introduces cardholder interfaces the module vendor never saw.

Wrong: softPOS is a phone with a PCI approval. **Right:** PCI PTS approves hardware and there is no hardware to approve; what is listed is a whole solution, including its attestation and monitoring servers, of which the phone is one component.

Wrong: the FCA removing the contactless caps raised the limit. **Right:** the limits live in reader data objects across a national estate, so changing them is a deployment programme accompanied by changes to the industry rules — and wallets were never inside the cap anyway.

20.99 Chapter summary in 20 lines

1. The terminal is the second party standing just off the page in every chapter about the card: the thing holding out a slot or a radio field and asking the questions.
2. It is best pictured as a locked translation booth, listening to the card in one language and to the bank in another.
3. What makes it a certified object rather than a peripheral is that the translator also hears your PIN.
4. Inside sits a secure processor holding the keys, checking its own firmware at boot and driving a tamper circuit powered from a coin cell so it runs even while the terminal is unplugged in a stockroom.
5. A fine mesh, switches under the screws and sensors for light, temperature, supply voltage and clock frequency all report to that circuit.
6. On tamper the device zeroises its keys immediately and becomes inoperable, like a dye pack turning a bag of notes into worthless paper.
7. That protects the keys and nothing else: overlays, shimmers and prompt manipulation take money without ever entering the enclosure, which is why intelligent attackers stopped trying to open it.
8. The form factors differ by job — countertop, portable, mPOS reader, unattended kiosk, PIN pad module, softPOS — and each maps to its own PCI approval class.
9. EMV records the environment in a single byte, Terminal Type at tag `9F35`, encoding who operates the device and whether it is attended and online-capable.
10. The security standard is the PCI PTS POI Modular Security Requirements, now version 7.0, modular by design and measuring attack potential in points rather than asserting it.
11. Approvals name a hardware revision, a firmware version and an expiry date, so "is this terminal certified?" is a query against several lists rather than a yes or no.

12. EMVCo's Level 1 and Level 2 approvals assess conformance and processing logic; neither is a security test.
13. Level 3 is not EMVCo's test at all but the Participant Systems' integration test plans, run by an acquirer or processor in a qualified tool.
14. A Level 2 letter names a kernel version on a named terminal with a named PIN pad and firmware, and the attached Implementation Conformance Statement, not the letter, is what tells you what the terminal can do.
15. Settings divide into minor ones that may be changed freely and major ones, such as Terminal Type, whose change means the approval no longer describes the device.
16. Deployment therefore takes months because Level 3 repeats per scheme and per configuration, because a delta can reopen the matrix and the security evaluation with it, and because keys must then be injected under dual control.
17. Unattended terminals are a class of their own: nobody to notice an overlay, a display and keypad possibly driven from outside the secure boundary, and an enclosure shared with the goods being sold.
18. A fuel transaction cannot know its own amount before dispensing, so it runs as a pre-authorisation followed by a completion, and the terminal type byte must reflect that offline capability.
19. softPOS removes the enclosure and replaces it with continuous attestation and server-side monitoring, moving the security boundary off the counter and into a data centre without removing any EMVCo obligation.
20. Everything the terminal decides comes from data loaded into it, and the four contactless reader limits are where policy finally lands — which is why a rule can change overnight and a national estate cannot.

Sources: EMVCo specification, approval-process and knowledge-hub pages, published Letters of Approval, the EMV Contactless Specifications for Payment Systems and the EMV Integrated Circuit Card Specifications v4.4; PCI Security Standards Council publications, including the PTS POI Modular Security Requirements v7.0 announcement, the PTS Device Testing and Approval Program Guide, the PTS POI technical FAQs, the approved PTS devices listing and the MPoC standard pages; Apple's Tap to Pay on iPhone security documentation; UK Finance guidance on contactless card limits; ISO/IEC 7816 and ISO/IEC 14443; nexo standards and IFSF documentation.

21

CHAPTER

Point-to-Point Encryption

21.0 What this chapter gives you

1. You will be able to explain why a shop can take card payments for a decade and never once hold a readable card number, and why that is a design decision rather than an achievement.
2. You will be able to tell a PCI-listed P2PE solution from a non-listed encryption solution, and say which of the two entitles a merchant to the short questionnaire.
3. You will be able to read a key serial number, name its fields in both the TDEA and the AES layouts, and explain why real-world KSNs so often begin `FFFF`.
4. You will be able to work out why a DUKPT device gets 1,048,575 transactions under TDEA and 2,448,023,842 under AES, rather than the powers of two those counters would suggest.
5. You will be able to explain why extracting a terminal's key state exposes every future transaction and no past one, and why the remedy is re-injection rather than investigation.
6. You will be able to say why the card number and the PIN in the same transaction travel under different keys, and point at the derivation data that makes it so.
7. You will be able to explain to a merchant why encryption cannot support a refund six weeks later, and name the mechanism that can.
8. You will be able to list the four conditions that silently destroy a merchant's scope reduction while the shop carries on working perfectly.
9. You will be able to diagnose the standard failures — key or KSN mismatch, counter exhaustion, a format 4 PIN block meeting a format 0 host, key block version mismatch — from their symptoms.
10. You will be able to explain why perfectly functional card readers were replaced in 2024, and what the supplier meant when it said "compliance".

There is a garden centre outside Norwich that takes about eleven hundred card payments a week and has never held a card number. Not in the till. Not in the back-office PC that runs the stock system. Not in the cloud service that sends the loyalty emails. Not in the nightly export that goes to the accountant. Card numbers have passed through that building at a rate of one every four minutes since 2016, and not one of them has been legible to anybody on the premises, including the owner.

This is not because the garden centre is unusually sophisticated. It is because the card reader on the counter arrived from the supplier already loaded with a secret, and turns every card it reads into an unreadable block before the till software is allowed near it. The shop is not trusted with card numbers. It is not asked to be trustworthy. The design assumes that sooner or later somebody will get into the shop's network, and arranges for that to be uninteresting.

The previous chapter was about the terminal as a certified object. This chapter is about what the terminal does with what it reads, and about the single commercial fact that pays for all of it. Encryption

in the terminal is not bought for elegance. It is bought because a merchant who cannot read card numbers has a much smaller compliance problem than a merchant who can, and the difference is measured in weeks of audit and thousands of pounds a year.

The plain version

Imagine you run a shop, and imagine that the most dangerous thing in the shop is not the cash in the till but a notebook by the door in which every customer's card number is written down. You do not want the notebook. You never wanted the notebook. But every card machine you have ever seen had to read the number in order to ask the bank for the money, and if the machine read it, something in your shop knew it.

Now imagine a different machine. This one has a slot for the card and a little sealed box inside it. When a customer inserts a card, the number goes straight into the sealed box and is scrambled there. What comes out of the machine is not a card number. It is a block of nonsense, thirty-two characters long, that looks like somebody sat on a keyboard. Your till receives the nonsense. Your network carries the nonsense. Your broadband router, your stock system, your accountant's export, the hacker who eventually gets into your Wi-Fi: all of them get nonsense.

Somewhere else entirely, in a building you will never visit, there is a matching sealed box. It is the only thing in the world that can turn that nonsense back into a card number, and it does so for a fraction of a second, inside its own metal case, in order to ask the bank for the money. There are exactly two points in this story where a readable card number exists: inside the sealed box in your card reader, and inside the sealed box in that distant building. Between them there is a pipe full of nonsense. That is why it is called point-to-point encryption. The two points are the only places worth attacking, and neither of them is in your shop.

A worked example

A customer buys £46.80 of compost and bark chippings. She taps her card. Inside the reader, the chip hands over a card number, sixteen digits long, and the reader immediately does two things with it.

First, it keeps a small, deliberately useless summary: the first six digits and the last four, so the till can print `---- ---- ---- 0016` on the receipt and so the software knows the card was a Visa debit rather than an American Express. Second, it takes the full number and scrambles it with a key.

Here is the part that is cleverer than people expect. The reader does not have one key. It has a book of keys, and it uses a different page for every single transaction. The compost sale uses page 1. The next customer, buying a rose bush at 10:14, uses page 2. The man who buys two bags of gravel at 10:19 uses page 3. And each page, once used, is torn out and burnt: the reader physically overwrites the key it just used so that it no longer exists inside the machine. The book has room for roughly a million pages in older readers and around two thousand million in newer ones, which at eleven hundred transactions a week is more than the reader will live to spend.

So the block of nonsense that arrives at the till carries one extra thing with it: a page number. Not a key, a page number. Something like "device 90123456, transaction 1". The distant sealed box holds the master recipe from which every page of that particular reader's book can be worked out, so when it is told "device 90123456, page 1", it can reconstruct page 1, use it once to unscramble the compost sale, and throw it away again.

Count what the thief gets if he owns your shop completely. He gets the nonsense blocks. He gets the page numbers, which are not secret and are useless without the master recipe. He gets `---- 0016`

and the fact that a Visa debit card spent £46.80 on compost. He cannot manufacture a card, buy a television, or sell a list of numbers, because the shop never had a list.

And the boring reason anybody bothers

Everyone who takes card payments has to answer to a rulebook called PCI DSS. If your systems handle readable card numbers, the version of that rulebook you have to satisfy is enormous: hundreds of individual requirements about firewalls, patching, anti-virus, logging, password rules, penetration tests and network segmentation, applied to every computer that the card number could conceivably touch, which in a small business means every computer.

If you use one of these encrypting readers, and if the encrypting reader is part of a scheme that has been formally inspected and published on a list, you are allowed to answer a much shorter questionnaire instead. It asks whether your card readers are the ones you think they are, whether anybody has glued a fake one on top, whether you followed the manual the supplier gave you, and whether you have written down who is responsible. That is very nearly the whole of it, because there is no computer in the shop holding card numbers for the long questionnaire to protect.

That reduction is the product. The cryptography is the mechanism.

Where the plain version stops being true

The reader does not encrypt everything, and it does not use one key

The plain version implies that nothing readable leaves the sealed box. In practice the box deliberately emits readable things, because the shop cannot function otherwise. Almost every deployment returns the first six or eight digits and the last four in clear, so that the till can route by card range, apply the right surcharge rules and print a receipt a human can match to a card. Many return the expiry date and the cardholder name in clear too. That is a choice made by the solution designer, and it is a choice with consequences: a BIN plus last four is not a card number, but it is not nothing either.

More importantly, the PIN is not protected by the same key as the card number, and never was. Card data and PIN travel under separate keys derived from the same hierarchy, so that a compromise of the account-data key does not hand over PINs, and so that PIN handling stays inside the older, stricter regime that governs PINs specifically. There is not one encrypted blob in the message. There are two, with different keys, different formats and different rulebooks.

Encrypting is not the same as being out of scope

This is the correction that costs merchants the most money. A vendor who says “our terminals encrypt end to end” has told you something about their engineering and nothing about your paperwork. The short questionnaire is available only where the whole arrangement — device, firmware, key loading, key management, decryption environment, and the manual the merchant is handed — has been assessed against the P2PE Standard by a qualified assessor and published on the Council’s list of validated solutions.

Solutions that encrypt in hardware but have never been through that process are called non-listed encryption solutions, and there is a separate, formal, and distinctly less generous route for them: an assessor documents what the solution actually achieves, and the merchant’s acquirer decides how much scope reduction, if any, to grant. It may be a great deal. It is not automatic, and it is not the merchant’s decision.

One-time keys are not one-time pads, and the protection runs backwards in time only

The plain version's book of pages suggests that each transaction is sealed off from every other. That is half true, and the half that is false matters.

The keys are derived from one another in a chain that cannot be run in reverse. An attacker who extracts the working state from a reader cannot compute the keys it used yesterday, so yesterday's captured traffic stays safe. But the state he extracts is precisely the material from which every future key of that device is derived. From the moment of compromise until the device is taken out of service and re-keyed, he can decrypt everything it sends. The scheme protects the past, not the future.

And behind every device's book of pages sits one master secret, held by the solution provider, from which the initial key of every reader in the family is derived. Lose one reader and you lose one shop's future traffic. Lose that master secret and you lose every reader derived from it, retrospectively and prospectively, at which point the only remedy is to re-inject the entire estate. This is why the master secret never exists in one piece, in one person's hands, outside a tamper-responsive box.

The second point is not your bank, and encryption gives you nothing to hold on to

"Point to point" invites the assumption that the far point is the merchant's bank. Usually it is not. The decrypting box normally belongs to whoever runs the encryption scheme — a gateway, a processor, a terminal estate manager — and the card number is decrypted there before an authorisation message is built and sent onward. The merchant's acquirer may be a further hop away, and may never see the encrypted form at all.

There is also something encryption structurally cannot do. Because the ciphertext changes on every transaction, it is useless as an identifier. It cannot tell you that the woman buying compost today is the same woman who bought a hosepipe in May, cannot be stored for a refund six weeks later, and cannot be used to bill a subscription. For those a merchant needs a stable substitute for the card number, valid for months, safe to store, and that is a different mechanism with a different name and a different scope story. Encryption protects data in flight. Tokenisation replaces it at rest. Most real estates use both, and confusing them is the commonest error in this field.

The technical version

What "P2PE" means in the Council's own words

Point-to-point encryption in the ordinary engineering sense is any arrangement that encrypts account data at the point of capture and decrypts it somewhere trusted. P2PE with a capital P is narrower: it is a PCI Security Standards Council programme, with a standard, an assessor qualification, a reporting template and a public list.

The Council's own summary is worth quoting because the wording is load-bearing. A PCI-listed P2PE solution "cryptographically protects account data from the point where a merchant accepts the payment card to the secure point of decryption", with the effect that account data — "cardholder data and sensitive authentication data" — "is unreadable until it reaches the secure decryption environment". The current standard is the *PCI Point-to-Point Encryption Standard* v3.2, published in June 2025, superseding v3.1 of September 2021; the accompanying Program Guide reached v3.2 in July 2026.

Four defined objects do most of the work in that programme, and practitioners who mix them up write incoherent contracts:

A **P2PE solution** is the complete package — devices, applications, key management, decryption — offered to merchants and listed as a unit. A **P2PE application** is "software with access to clear-text

account data, intended to be loaded onto a PCI-approved point of interaction (POI) device and used as part of a P2PE solution". A **P2PE component** is a service that is assessed and listed on a stand-alone basis and can be incorporated by reference into somebody else's solution. A **P2PE solution provider** is "an entity, usually a third-party such as a processor, acquirer (merchant bank), or payment gateway, that designs, implements, and manages the P2PE solution", and — the sentence that decides liability arguments — "may outsource certain responsibilities, but will always retain overall responsibility for the P2PE solution".

The merchant is not on that list, with one exception: a merchant that builds and runs its own encryption estate is assessed as a merchant-managed solution, against its own reporting template, and cannot then use the short questionnaire as if it were somebody else's customer.

Five domains and the paperwork that maps onto them

The v3 standard organises its requirements into five domains, numbered 1 to 5, and assessments are reported as a set of Reports on Validation, abbreviated P-ROV. The mapping is set out explicitly in the summary table of the Solution P-ROV template:

Assessed function	Domains
P2PE Solution Management	Domain 3
Encryption Management Services (EMS)	Domains 1 and 5
Decryption Management Services (DMS)	Domains 4 and 5
Key Management Services (KMS)	Domain 5, where not already covered by EMS or DMS
P2PE Application	Domain 2

Read that table as an architecture diagram rather than a filing system. Domain 1 is the encryption side: the devices, their deployment, their firmware, their chain of custody. Domain 2 is any software running on the device with sight of clear-text account data. Domain 3 is the governance of the solution as a whole, including the instructions given to merchants. Domain 4 is the decryption environment. Domain 5 is cryptographic key operations, and it is shared, which is why it appears twice: the same key-management requirements bind the party that injects keys into terminals and the party that decrypts with them.

Requirements are identified in a compound style — Domain 1's requirement `1B-1.1.1` and Domain 2's `2A-3.1.2`, for instance, both concern devices and applications that facilitate merchant-managed decryption — and anyone drafting a gap analysis should cite them in that form rather than by paraphrase.

The programme also lists individual components, referred to throughout the templates by abbreviation: EMCP, PDCP, PMCP, KMCP and KLCP, along with key injection facilities (KIF) and certification and registration authorities (CA/RA). A solution provider assembles validated components and inherits their assessed scope. Where remote key distribution is used, its requirements are additional to, not instead of, the rest.

Two more artefacts matter operationally. The **P2PE Instruction Manual** (PIM) is the document the solution provider must produce and the merchant must follow; deviating from it is the most common way a merchant silently loses eligibility for the short questionnaire. The **Attestation of Validation** (AOV) is the signed summary that accompanies the P-ROV and precedes listing.

The device: PTS POI approval and the SRED module

P2PE encryption happens in a PCI PTS approved point-of-interaction device. The relevant standard is the *PCI PIN Transaction Security (PTS) Point of Interaction (POI) Modular Security Requirements*, v7.0 as of May 2025, with v6.2 (January 2023) and v6 (June 2020) still supporting large fielded populations. Its structure is modular, and the module that makes P2PE possible is Evaluation Module 4, *Secure Reading and Exchange of Data* — SRED — whose requirements are lettered K under the heading "Account Data Protection".

SRED is where the sealed box of the plain version is specified. Requirement numbering has been revised between POI versions, so cite the version you are certifying against, but the substance of the K series has been stable since it was introduced:

All account data must be "either encrypted immediately upon entry or entered in clear-text into a secure device and processed within the secure controller of the device", with no method of accessing the clear-text data. Encryption must use "only ANSI X9 or ISO-approved encryption algorithms (e.g., AES, TDES)" and should use approved modes of operation. Secret and private keys held in the device to support account data encryption must be "unique per device" — a single requirement that rules out the entire class of shared-key deployments that used to be normal. Encrypting or decrypting arbitrary data with an account-data key or key-encrypting key is forbidden, which closes the obvious decryption-oracle attack. When operating in encrypting mode there must be "no mechanism in the device that would allow the outputting of clear-text account data", and the secure controller may release clear-text account data only to authenticated applications executing within the device. Where the device generates surrogate PAN values for output, it must not be possible to determine the original PAN from the surrogate, and where a hash is used the input must include a salt of at least 64 bits. Devices must also have characteristics that "prevent or significantly deter" their use for exhaustive PAN determination — that is, for BIN-range brute forcing by a criminal with a stolen terminal.

Determining the account-data encryption keys by penetrating the device or by monitoring its emanations, including power fluctuations, must require an attack potential above a defined threshold: SRED is tested against side channels, not only screwdrivers.

Note what SRED does not cover. PINs are governed by the PIN-related modules of the same document and by the PCI PIN Security Requirements. A device can be PTS approved without SRED, which is why "PCI approved terminal" is not a synonym for "encrypting terminal", and why terminal estates contain both.

What counts as account data

The scope of encryption is account data, not merely the PAN: the primary account number, the full track data or track-equivalent data from chip or stripe, the expiry date, the service code, and the card verification values where present. Sensitive authentication data — full track data, card verification codes, PINs and PIN blocks — must not be retained after authorisation at all, encrypted or otherwise, a prohibition that survives every scope reduction in this chapter and is stated in PCI DSS v4.0.1 at Requirement 3.3.1.

Where a PAN is displayed, the maximum that may be shown is the BIN and the last four digits (Requirement 3.4.1); where it is stored, it must be rendered unreadable (Requirement 3.5.1). Those two rules, not P2PE, are the source of the `---- 0016` convention on receipts.

DUKPT, part one: the TDEA scheme

Derived Unique Key Per Transaction is the key-management scheme that makes per-transaction keys practical without a network round trip. It is specified in ANSI X9.24 — Part 1 for the original triple-DEA scheme, Part 3 for the AES scheme. Every practitioner meets the TDEA version first because it is what most fielded devices still use.

The structure is a three-level hierarchy. At the top is the **base derivation key** (BDK), a double-length TDEA key held only inside the solution provider's HSMs. From the BDK and a device-specific serial value, the provider derives one **initial PIN encryption key** (IPEK) per device, injects it, and does not keep it. The device then never uses the IPEK for a transaction: it uses it to populate a set of twenty-one future-key registers, and from those it derives the key for each transaction in turn, erasing each key after use through a transformation that cannot be inverted. That non-reversibility is the whole security argument: the device's present state does not reveal its past keys.

Every message carries a **key serial number** (KSN) so the receiving HSM knows which key to reconstruct. The TDEA KSN is ten bytes. In the conventional layout it is read as a 24-bit key set identifier, a 19-bit device identifier and a 21-bit transaction counter, left-padded with `F` nibbles to fill the ten bytes — which is why real-world KSNs so often begin `FFFF`. Field engineers also describe the same thing as "6-5-5" after the hexadecimal digit groups; on that reading a scheme permits roughly sixteen million BDKs, five hundred thousand devices per BDK and a million transactions per device.

The million is not `2^21`. The algorithm increments the counter so that no more than ten of its twenty-one bits are ever set at once, because the count of set bits determines how many derivation steps are needed. The usable counter values therefore number 1,048,575, and a device that reaches the end of its counter must be re-injected rather than rolled over.

DUKPT, part two: the AES scheme, with real numbers

ANSI X9.24-3-2017 replaces the TDEA construction with an AES one, and changes the vocabulary: the per-device key is now the **initial key** (IK) rather than the IPEK, since it derives data and MAC keys as well as PIN keys. Key types may be two-key TDEA, three-key TDEA, AES-128, AES-192 or AES-256, and the same tree serves all purposes.

The AES KSN is twelve bytes: a 32-bit BDK identifier, a 32-bit derivation identifier and a 32-bit transaction counter. The first eight of those bytes together are the **initial key ID**, which is what the IK is derived from.

Derivation is a keyed function over a sixteen-byte block of derivation data whose layout is fixed:

Bytes	Field	Example
0	Version	`01`
1	Key block counter	`01`
2–3	Key usage indicator	`1000`
4–5	Algorithm indicator	`0002`
6–7	Key length in bits	`0080`
8–15	Initial key ID, or derivation ID plus transaction counter	`90123456 00000001`

The key block counter exists so that keys longer than one AES block can be produced from more than one derivation block. The algorithm indicator distinguishes key types: `0000` for two-key TDEA and

`0002` for AES-128 in the standard's own test vectors. The key usage indicator is the field that separates purposes, and its values are worth memorising because they appear in HSM command traces:

Usage	Meaning
`0002`	Key encryption key
`1000`	PIN encryption
`2000`	Message authentication, generation
`2001`	Message authentication, verification
`2002`	Message authentication, both ways
`3000`	Data encryption, encrypt
`3001`	Data encryption, decrypt
`3002`	Data encryption, both ways
`8000`	Key derivation
`8001`	Key derivation, initial key

The published test vectors accompanying X9.24-3-2017 make the whole mechanism concrete, and they are the fastest way to prove a new implementation correct. Take the base derivation key `FEDCBA98 76543210 F1F1F1F1 F1F1F1F1` and the initial key ID `12345678 90123456`. The derivation data for the initial key is `01018001 00020080 12345678 90123456` — version 1, block counter 1, usage `8001` for an initial key, AES-128, 128 bits, then the full initial key ID — and it yields the initial key `1273671E A26AC29A FA4D1084 127652A1`.

Now take transaction 1. The device's derivation key for that transaction is `4F21B565 BAD9835E 112B6465 635EAE44`, and from it three working keys fall out, differing only in two bytes of input:

Purpose	Derivation data	Resulting key
PIN encryption	`01011000 00020080 90123456 00000001`	`AF8CB133 A78F8DC2 D1359F18 527593FB`
MAC generation	`01012000 00020080 90123456 00000001`	`A2DC23DE 6FDE0824 A2BC321E 08E4B8B7`
Data encryption	`01013000 00020080 90123456 00000001`	`A35C412E FD41FDB9 8B69797C 02DCD08F`

Note the last eight bytes: not the whole initial key ID but its low four bytes, `90123456`, followed by the transaction counter `00000001`. And note that the PIN key and the data key for the very same transaction are different keys. When this chapter said earlier that the card number and the PIN travel under separate keys, that is the arithmetic that makes it so.

The AES counter is 32 bits and, as in the TDEA scheme, only values with a limited number of set bits are used — up to sixteen one-bits rather than ten. That yields 2,448,023,842 usable transactions per device, which is the practical meaning of "the reader will not live to spend the book".

	TDEA DUKPT	AES DUKPT
Standard	ANSI X9.24-1	ANSI X9.24-3-2017
Per-device key	IPEK	IK (initial key)
KSN length	10 bytes	12 bytes
KSN layout	24-bit key set ID, 19-bit device ID, 21-bit counter	32-bit BDK ID, 32-bit derivation ID, 32-bit counter

	TDEA DUKPT	AES DUKPT
Counter constraint	at most 10 bits set	at most 16 bits set
Transactions per device	1,048,575	2,448,023,842
PIN block	ISO 9564-1 formats 0, 1, 3	ISO 9564-1 format 4

The PIN block row is a migration trap in its own right. Format 4 is the AES format and is required with AES keys; a host expecting format 0 will decrypt an AES-era PIN block into rubbish and reject perfectly good PINs.

Why a different key every time

State the threat model precisely, because "more keys is more secure" is not an argument.

An attacker with a year of a shop's traffic holds a year of ciphertexts, each under a distinct key. No volume of traffic helps him: he cannot accumulate material under one key, and breaking one transaction gains him one transaction. With a single static terminal key — the arrangement DUKPT replaced — the same capture is one key away from a year of card numbers.

An attacker who steals a terminal and successfully extracts its live key state gains every future transaction of that terminal and no past ones, for the reason given earlier: the derivation runs one way. In practice extraction is what SRED's tamper and side-channel requirements are there to prevent, and the commercial consequence of a suspected extraction is a re-injection, not an investigation.

An attacker inside the merchant's network gains ciphertext plus KSNs. The KSN is not secret; it is an index. Publishing it is not a weakness, and in fact its counter is useful defensively: a host can detect duplicate KSNs, replayed transactions and impossible counter jumps, because the counter of a healthy device only ever increases.

The party with real exposure is the solution provider holding the BDK, which can regenerate any key of any device in its family from the KSN alone. Hence the requirements in Domain 5, and hence HSMs.

Key blocks, and why integrations broke in 2023 and 2025

A key in transit needs its attributes bound to it. A bare 16-byte value carries no statement of what it is for, and a host that will use whatever it is sent as whatever it is told can be manipulated into using a PIN key as a data key, or a data key as a key-encrypting key. Key blocks solve this by encrypting the key together with a header that states its usage, algorithm, mode of use and exportability, and authenticating the whole structure.

The format that the payments industry standardised on was ANSI X9 TR-31, since superseded by **ANSI X9.143-2022**; the two are generally interoperable and the names are used interchangeably in vendor documentation. The key that protects a key block is a key block protection key (KBPK), distinct in principle from a plain key-encrypting key because it protects the header as well as the key.

Key blocks stopped being optional on a published timetable. PCI PIN Security Requirement 18-3 states that encrypted symmetric keys must be managed in key blocks, and the Council's revision bulletin of 17 July 2020 fixed the phases as follows:

Phase	Scope	Effective date
1	Internal connections and key storage within service provider environments, including all applications and databases connected to HSMs	1 June 2019
2	External connections to associations and networks	1 January 2023
3	All merchant hosts, point-of-sale devices and ATMs	1 January 2025

Those dates explain a decade of integration pain. Phase 2 broke host-to-network links that had exchanged variant-format keys for twenty years. Phase 3 pushed the same change out to terminal estates and merchant hosts, which is why perfectly functional readers were replaced in 2024 by suppliers who described the reason as "compliance" rather than explaining it.

Underneath sits the algorithm question. PCI's requirements are expressed in terms of effective key strength, and double-length TDEA at 112 bits still satisfies them; NIST, in SP 800-131A Rev. 2, treats three-key TDEA encryption as deprecated through 2023 and disallowed thereafter. The practical result is a long overlap in which AES DUKPT and TDEA DUKPT coexist in the same estate, sometimes in the same shop, and every host must support both.

Key injection

An initial key has to get into a device without ever existing in the clear where a person could read it. There are two accepted routes.

The first is a **key injection facility**: a physically secured room, assessed against the relevant domain requirements, where devices are connected to a key loading device driven by an HSM. Access is under dual control — no single person can perform a key operation — and the BDK components are held under split knowledge, so that no individual has enough material to reconstruct the key. Components are usually delivered as printed component sheets held by named custodians in tamper-evident envelopes, combined only inside the HSM during a witnessed and logged key ceremony.

The second is **remote key distribution**, in which the device authenticates itself cryptographically to a distribution host and receives its initial key over an untrusted network. The mechanism is asymmetric: ANSI X9.24-2 specifies the approach, and TR-34 is the protocol implementation used in practice, with a key distribution host and a key receiving device exchanging signed and enveloped material. The POI requirements insist on mutual authentication between host and device for remote key distribution, and the P2PE templates treat remote key distribution requirements as additional to the baseline. Remote injection is what makes it economically possible to ship a reader to a merchant by courier rather than through a depot.

Chain of custody is assessed as seriously as cryptography. Devices must be shipped with tamper-evident packaging, tracked by serial number, inspected on receipt against a manifest, and stored securely before deployment. A device whose whereabouts cannot be accounted for between the injection facility and the shop counter is, formally, an unknown device.

The decryption environment

At the far point, decryption occurs inside a hardware security module. This is not an ordinary server with a cryptographic library: it is a tamper-responsive device that erases its secrets when opened, holds a local master key that never leaves it, and refuses to output the keys it protects. Payment HSMs are certified against the *PCI PTS Hardware Security Module (HSM) Modular Security Requirements* — v4.0 of December 2021 for most currently fielded units, with v5.0 published in May 2026 — and typically also against FIPS 140-3 (formerly 140-2) at Level 3 under the NIST validation programme. ISO 13491 covers secure cryptographic device requirements generally, and ISO 11568 covers retail key management, both of which the P2PE and PIN standards reference.

The properties that matter architecturally are simple to state and expensive to achieve. Clear-text account data exists only inside the HSM's boundary. Keys are used, not read: the application asks the HSM to decrypt a block under the key derived from a given KSN, and receives plaintext, never the key. Key operations require dual control and are logged. The BDK exists in the clear nowhere.

In P2PE terms this is the Decryption Management Services function, assessed against Domain 4 with the shared key requirements of Domain 5. It is also the point at which the merchant's scope reduction is paid for by somebody else's scope: everything the merchant no longer has to protect, the solution provider protects instead, under audit, and prices accordingly.

Scope reduction, stated honestly

The commercial case is real, and it is narrower than sales decks imply.

A merchant using a PCI-listed P2PE solution, and following the PIM, may validate using SAQ P2PE instead of the long self-assessment questionnaires or a full Report on Compliance. What remains is drawn chiefly from the physical-security and policy requirements: knowing which devices you have and where, checking them for tampering and substitution, controlling physical access to them, handling any paper records that carry card numbers, training staff, maintaining an incident response plan, and managing the third parties involved. What disappears is the mass of network, patching, malware, logging and segmentation requirements — not because they stopped being good ideas, but because there is no in-scope system in the shop for them to apply to.

Four conditions routinely destroy that position, and all four are common:

The solution must be listed, and listed for the specific device types and applications in use. A validated solution running an unlisted application on a listed device is not a validated solution.

The merchant must not have another route by which clear-text account data enters its systems. A shop with a beautiful P2PE estate on the counter and a member of staff typing telephone orders into a browser has a cardholder data environment again, and the browser is in it.

The PIM must be followed. It specifies device inspection frequency, permitted device configurations, what to do about a suspected tamper, and how devices must be shipped and stored. Ignoring it is not a technical failure; it is loss of eligibility.

Encrypted account data that the merchant stores is out of scope only if the merchant genuinely has no means of decrypting it. A merchant who holds ciphertext and also holds, anywhere, the key or the decryption service is back in scope for that data.

For non-listed solutions, the Council's *Assessment Guidance for Non-listed Encryption Solutions* (June 2020) sets out how a QSA documents the solution's effect so that the acquirer can decide what scope

reduction to accept. This route is widely used and perfectly respectable. It is simply not the same product, and it does not entitle a merchant to SAQ P2PE.

Tokenisation at the acquirer versus encryption in the terminal

These are complementary mechanisms that solve different problems, and the confusion between them causes real architectural mistakes.

Encryption in the terminal protects a card number in flight, from capture to decryption, and produces a value that changes on every transaction. Tokenisation replaces a card number with a surrogate that persists, so it can be stored, quoted, refunded against and billed again.

There are three distinct kinds of token in payments, and only one of them belongs in this chapter. **Acquirer or gateway tokens** are issued by the merchant's acquirer or gateway after decryption: the vault stores the PAN, returns a surrogate, and the merchant stores the surrogate. **EMVCo payment tokens** are network tokens, issued by a token service provider, with their own device and domain controls, covered in the chapter on the token in your phone. **Device-generated surrogate PANs** are produced inside the POI itself, under the SRED requirements quoted earlier, and are useful for local matching where the merchant needs no server round trip.

	Terminal encryption (P2PE)	Acquirer tokenisation
What it produces	Ciphertext, different every transaction	A stable surrogate value
Where the secret lives	Solution provider's HSMs (BDK)	Vault operator's database and HSMs
Useful for refunds and recurring billing	No	Yes
Protects data in flight	Yes	No, on its own
Formal scope effect for the merchant	SAQ P2PE, where the solution is listed	Reduces stored-data obligations; no equivalent listed programme
Fails how	Key compromise, device compromise	Vault compromise, or a token usable by anybody who steals it

The right architecture for a merchant that needs both is a listed P2PE solution for capture and an acquirer token returned in the authorisation response for storage. The terminal sends ciphertext plus KSN; the provider decrypts, authorises, and returns a token; the merchant stores the token and never the number. Refunds, subscriptions and analytics all run on the token.

Two cautions. A token a thief can present for payment without further authentication has merely moved the value rather than removed it, which is why tokens are scoped to one merchant and one channel. And format-preserving tokens that pass the Luhn check are convenient for legacy software and treacherous for governance, because nothing in a schema distinguishes them from real card numbers.

What actually goes on the wire

The encrypted blob does not travel on the card networks. It travels from the terminal to the decrypting party over that party's own protocol, and it dies there.

A typical terminal-to-host request carries the ciphertext as a hexadecimal or base64 string, the KSN in the clear, an indication of the encryption scheme and format in use, a masked PAN, the expiry if the

solution exposes it, and the ordinary transaction fields — amount, currency, the EMV data from the chip. The host passes the ciphertext and KSN to its HSM, which derives the transaction key from the BDK and the KSN, decrypts inside its boundary, and returns the account data.

Only then does the message the rest of this book is about get built. The clear PAN goes into the ISO 8583 primary account number field, DE 2; track data, where present, into DE 35 or DE 45; the PIN block, encrypted under a different key and translated by the HSM from the terminal's key to the acquirer's working key, into DE 52, with key management data in DE 53. The scheme never sees the P2PE ciphertext, has no knowledge of the BDK, and cannot tell from the authorisation message whether the merchant's terminal was encrypting or not. That handoff is the subject of the next chapter.

A short field guide to failures

Key or KSN mismatch. Decryption returns plausible-length rubbish. Almost always the host has been configured with the wrong BDK for that key set ID, or the KSN is being parsed with the wrong field boundaries. Check the key check value of the BDK before suspecting anything subtle.

Counter exhaustion. A device that has quietly reached the end of its counter stops encrypting. It is not broken; it needs re-injection. On TDEA devices this is reachable inside a decade at high volume.

Format assumptions. An AES-era device sending an ISO 9564-1 format 4 PIN block to a host expecting format 0 produces PIN verification failures that look like customer error. The same class of fault arises when a host assumes the encrypted field includes the expiry and it does not.

Key block version mismatch. A host that only accepts one key block format will reject an otherwise valid key. This is the ordinary consequence of the 2023 and 2025 phases arriving at different times in different estates.

Tamper trips in transit. Tamper-responsive devices sometimes respond to being dropped, frozen or X-rayed. A device that arrives dead has done its job.

Silent loss of eligibility. The most expensive failure has no error message. Somebody installs an unlisted application, or starts keying card numbers into the back office, or stops doing the device inspections the PIM requires, and the shop stays fully functional while its right to the short questionnaire lapses. This is discovered at the next assessment, or after a breach.

The through-line of this chapter is that none of the cryptography here exists to protect the merchant from the merchant's enemies. It exists to remove the merchant from the problem entirely. A shop that cannot read card numbers cannot leak them, cannot be socially engineered out of them, and cannot be sued for them, and that is a stronger security property than any amount of diligence in a shop that can.

21.98 Common wrong ideas

Wrong: an encrypting reader means nothing readable leaves the sealed box. **Right:** almost every deployment deliberately emits the first six or eight digits and the last four in clear, and many emit the expiry date and cardholder name too, because the till cannot route, surcharge or print a receipt otherwise.

Wrong: "our terminals encrypt end to end" means the merchant qualifies for SAQ P2PE. **Right:** the short questionnaire is available only where the whole arrangement has been assessed against the P2PE Standard and published on the Council's list; anything else is a non-listed encryption solution, and how much scope reduction it earns is the acquirer's decision, not the merchant's.

Wrong: the card number and the PIN are protected by the same key. **Right:** they are two separate encrypted blocks under keys derived for different usages, with different formats and different rulebooks, which is why compromising the account-data key does not hand over PINs.

Wrong: per-transaction keys mean every transaction is sealed off from every other. **Right:** derivation runs one way only, so an attacker who extracts a device's live state cannot compute yesterday's keys but can compute every future one until the device is re-injected.

Wrong: the far point of point-to-point encryption is the merchant's bank. **Right:** the decrypting box normally belongs to whoever runs the scheme — a gateway, a processor, a terminal estate manager — and the acquirer may be a further hop away and never see the encrypted form at all.

Wrong: encryption and tokenisation are two names for the same protection. **Right:** encryption protects data in flight and produces a value that changes on every transaction; tokenisation replaces the number at rest with a stable surrogate that can be stored, refunded against and billed again.

Wrong: a PCI-approved terminal is an encrypting terminal. **Right:** the encryption requirements live in Evaluation Module 4, SRED, which is optional; a device can be PTS approved without it, which is why terminal estates contain both kinds.

Wrong: the key serial number has to be kept secret. **Right:** it is an index, not a key, and its counter is defensively useful, because a healthy device's counter only ever increases and duplicates or impossible jumps are detectable.

Wrong: storing only encrypted account data always keeps a merchant out of scope. **Right:** it does so only if the merchant genuinely has no means of decrypting it; a merchant holding ciphertext and also holding, anywhere, the key or the decryption service is back in scope for that data.

Wrong: a card reader that arrives dead was badly made. **Right:** tamper-responsive devices respond to being dropped, frozen or X-rayed, and a device that erased itself in transit has done exactly its job.

21.99 Chapter summary in 20 lines

1. A card reader that arrives from the supplier already loaded with a secret turns every card number into unreadable ciphertext before the shop's own software is allowed near it.
2. There are exactly two places in the story where readable account data exists — inside the reader and inside the decrypting party's hardware security module — and neither of them is in the shop.
3. The product being bought is not the cryptography but the scope reduction: a merchant that cannot read card numbers answers a short questionnaire instead of the full PCI DSS regime.
4. The reader uses a different key for every transaction, destroys each key after use, and sends a non-secret key serial number so the far end can reconstruct the key it needs.
5. The plain version overstates the protection, because most solutions return the BIN and last four digits in clear, and often the expiry and the cardholder name as well.
6. Card data and PIN travel as two encrypted blocks under keys of different usages, so a compromise of the account-data key does not surrender PINs.
7. P2PE with a capital P is a PCI Security Standards Council programme with a standard, an assessor qualification, reporting templates and a public list, not a generic description of engineering.
8. The programme defines a solution, an application, a component and a solution provider, and the provider retains overall responsibility however much of the work it outsources.
9. Five domains carry the requirements — encryption, application, solution management, decryption and key management — with key management shared between the encrypting and decrypting sides.

10. The encryption itself is specified in SRED, which demands unique keys per device, approved algorithms, no path by which clear-text account data can leave, and resistance to side channels as well as screwdrivers.
11. What must be protected is account data, not merely the PAN, and sensitive authentication data must not be retained after authorisation at all, encrypted or otherwise.
12. DUKPT makes per-transaction keys practical without a network round trip, deriving every key of every device from a base derivation key that exists only inside the provider's HSMs.
13. The TDEA scheme uses a ten-byte KSN and yields 1,048,575 usable transactions; the AES scheme uses twelve bytes and yields 2,448,023,842, because both counters are limited by how many bits may be set rather than by their width.
14. Derivation is irreversible, so a compromised device leaks its future traffic and not its past, and the commercial consequence of a suspected extraction is a re-injection rather than an investigation.
15. Key blocks bind a key's usage, algorithm and mode to the key itself, and the phased mandate of PIN Security Requirement 18-3 in 2019, 2023 and 2025 explains a decade of integration pain.
16. Initial keys reach devices either through a physically secured injection facility under dual control and split knowledge, or by remote key distribution with mutual authentication over an untrusted network.
17. Decryption happens inside a tamper-responsive HSM that uses keys without ever revealing them, and everything the merchant no longer has to protect the solution provider protects instead, under audit and for a price.
18. Four things routinely destroy the scope reduction: an unlisted application, another route by which clear-text card numbers enter the merchant's systems, deviation from the P2PE Instruction Manual, and holding both ciphertext and the means to decrypt it.
19. Encryption and tokenisation are complementary rather than competing, and the right architecture is a listed solution for capture with an acquirer token returned for storage, so that refunds, subscriptions and analytics all run on the token.
20. None of this cryptography protects the merchant from the merchant's enemies; it removes the merchant from the problem, and a shop that cannot read card numbers cannot leak them, cannot be socially engineered out of them, and cannot be sued for them.

Sources: PCI Security Standards Council document library, including the PCI Point-to-Point Encryption Standard v3.2 (June 2025) and its summary of changes, the P2PE Program Guide v3.2, the P2PE v3.x Technical FAQs, the P2PE Solution, EMS, DMS, KMS, Application and MMS P-ROV templates, the P2PE Instruction Manual and Attestation of Validation templates, "Securing Account Data with the PCI Point-to-Point Encryption Standard v3" (July 2020), and the Assessment Guidance for Non-listed Encryption Solutions (June 2020); PCI PIN Transaction Security Point of Interaction Modular Security Requirements, Evaluation Module 4 (Secure Reading and Exchange of Data) and its section K, with version history to v7.0 (May 2025); PCI PTS Hardware Security Module Modular Security Requirements v4.0 (December 2021) and v5.0 (May 2026); PCI DSS v4.0.1 Requirements 3.3.1, 3.4.1 and 3.5.1; PCI SSC "Revisions to the Implementation Date for PCI PIN Security Requirement 18-3" (17 July 2020) and the PIN Security Requirement 18-3 Key Blocks information supplement; ANSI X9.24-1 and ANSI X9.24-3-2017, with the Supplement to ANSI X9.24-3-2017 Test Vectors published by the Accredited Standards Committee X9 for the derivation data, key usage indicators and worked key values; ANSI X9.143-2022 and ANSI X9 TR-31 for key blocks; ANSI X9.24-2 and TR-34 for asymmetric key distribution; ISO 9564-1, ISO 11568 and ISO 13491; NIST SP 800-131A Rev. 2 and the FIPS 140-3 validation programme; AWS Payment Cryptography industry terminology documentation for KSN field layouts and key type naming; IBM z/OS and Linux on Z cryptographic services documentation for the X9.24 transaction counter behaviour.

22

CHAPTER

What the Terminal Actually Sends

22.0 What this chapter gives you

1. You will be able to take any piece of a payment message and say which of the four writers — the card, the terminal, the acquirer or the issuer — put it there.
2. You will be able to explain why the terminal almost never sends the message the banks actually read, and what the acquirer's host builds from scratch on its behalf.
3. You will be able to read a field 55 tag by tag and recover the transaction from it, including the single bit that explains why it went online at all.
4. You will be able to do the byte arithmetic on field 55 and predict which tags a 255-byte or 255-character ceiling will amputate, and which of those losses are fatal.
5. You will be able to explain why a disagreement between DE 4 and tag `9F02` presents as a cryptographic failure when it is really a mapping bug in somebody's message translator.
6. You will be able to say what DE 22 claims about a transaction, and why claiming a chip read while sending no field 55 is the commonest certification failure in the industry.
7. You will be able to state what the terminal must and must not do when tag `91` is present in the response, and what it must do when it is absent.
8. You will be able to explain why a reversal has to replay the exact bytes that were sent rather than regenerate them from a database.
9. You will be able to distinguish the authorisation from the completion advice and from the clearing record, and say which of the three actually claims the money.
10. You will be able to explain why the twelve-character reference the acquirer stamps on a message today is the identifier a dispute will be argued from next summer.

At 09:45:12 on Monday 17 August 2026, in a coffee shop in Leeds, somebody buys a flat white for £4.20 and puts a card in a slot. Eleven chapters of this volume have been about what happens inside that moment: the plastic, the number, the chip, the eight bytes of mathematics it produces, the reader limits, the keypad, the encryption in the head of the terminal. This chapter is about the sentence at the end of it. The terminal has finished thinking. Something now has to leave the shop.

What leaves is about three hundred and fifty bytes. Less than a photograph of the receipt. Less than this paragraph, encoded as text. Inside those bytes are two things of quite different natures: a set of facts asserted by the machine on the counter, and a short, sealed block written by the chip, which the machine cannot read, cannot alter and does not understand. The whole of the rest of this book is the story of where those bytes go. This chapter is the story of how they are assembled, and of who wrote each one.

It is also the last chapter of Volume II, so it has a second job. Everything up to here has been about one end of the wire. From the next chapter the subject changes: four parties, a message format, a network, a settlement cycle, a dispute process. The join between the two is a single data element with a two-digit number, and getting the join wrong is the most common way for an otherwise correct chip implementation to fail certification.

The plain version

Imagine a hospital referral. Your GP writes a letter to a consultant: your name, your age, the date, what she observed, what she thinks. Stapled to that letter is a sealed envelope from the laboratory containing a test result. The GP cannot open it. She does not need to. Her job is to describe the room she was standing in; the laboratory's job is to say something the consultant can verify independently. The consultant reads both, and the value of the letter depends heavily on the fact that the envelope was sealed before it left the laboratory.

The terminal is the GP. The chip is the laboratory. The letter is the transaction: how much, when, where, which machine, how the card was read, whether anybody typed a PIN. The sealed envelope is eight bytes of cryptogram plus the exact facts the chip signed over when it made them. The bank at the other end reads the letter, opens the envelope, recomputes the mathematics with its own copy of the key, and sees whether the answer matches.

There is one more character: the courier. The card machine does not know how to reach your bank and has never heard of it. It knows exactly one address, the company the shop signed a contract with, which the trade calls the acquirer. The acquirer takes the letter, puts it in a standard envelope of a format the whole industry agrees on, writes its own reference number on the outside, adds its own return address, works out from the first few digits of the card number which bank the letter is for, and sends it.

So there are three writers, and one of the most useful habits you can build is asking, of any piece of a payment message, which of the three wrote it.

A worked example

Take the flat white. The basket is £4.20. The shop is in the United Kingdom, so the currency is sterling. The card is a chip card, inserted rather than tapped. The amount is small, so no PIN is asked for. The terminal decides, for reasons Chapter 16 set out, that it will not approve this one on its own authority, and asks the chip for the kind of cryptogram that means "ask the bank".

The chip produces its sealed block. In this transaction it is one hundred and thirty-four bytes long, and it contains twenty-one separate items. Eight of them the chip wrote. Thirteen the terminal wrote and then handed to the chip to be signed over, so that nobody in the middle can change them without the mathematics breaking. Among the thirteen are the amount, £4.20; the currency, sterling; the country, the United Kingdom; the date; the four random bytes the terminal generated to make this transaction unlike any other; and the terminal's own account of what the customer did or did not have to do to prove who they were.

Around that block, the terminal writes the letter. The parts that matter to a human read roughly as follows.

What it says	The value	Who wrote it
Amount	£4.20	The till
Currency	Pounds sterling	Terminal configuration

What it says	The value	Who wrote it
Local time and date	09:45:12 on 17 August	The terminal's clock
How the card was read	Chip, inserted, PIN pad present	The terminal
Which machine	Terminal 20194831	The acquirer, at installation
Which shop	Merchant 602948300012345	The acquirer, at contract
What kind of shop	Eating places and restaurants	The acquirer
The card number	Sixteen digits	The card
The sealed block	134 bytes	The chip, and the terminal

Then the acquirer adds its own three lines: a reference number that will follow this transaction for the next eighteen months, its own institution number so the reply knows where to come back to, and a timestamp in universal time so that two computers in different countries can agree on the order of events.

The reply arrives in under a second and is much shorter. It contains a two-character verdict, a six-character approval code that the receipt will print, and, if the bank has bothered, a second sealed envelope addressed to the chip. That second envelope is the interesting one. It lets the chip check that the approval really came from its own bank and not from someone in the middle of the wire with a convincing manner. The terminal cannot read that envelope either. It simply passes it to the chip, and the chip says yes or no.

Count the writers once more. The chip wrote eight items and sealed twenty-one. The terminal wrote thirteen of the sealed ones and about a dozen unsealed ones. The acquirer wrote the outside of the envelope. The bank wrote three lines back. Nobody wrote the whole message, and no single party in the chain could have.

Where the plain version stops being true

The terminal does not send the message the banks read

The letter analogy quietly implies that the GP's letter is the document the consultant receives. It is not. In almost every real estate, the terminal speaks a protocol of its own to the acquirer — sometimes a national standard, sometimes a vendor's own format, increasingly a web API carrying JSON — and the acquirer's host builds the industry-standard message from scratch. The terminal frequently has no idea that data elements have numbers. It sends fields with names like `amount` and `emvData`, and something in a data centre translates.

This matters more than it sounds. It means that when a specification says "the terminal sends field 55", what it actually means is that the terminal sends a blob of chip data through whatever pipe it has, and the acquirer puts that blob into field 55 without touching it. It means the person debugging a chip problem may be looking at three different representations of the same bytes. And it means that the two halves of the industry have different vocabularies for the same object: Visa's documentation calls it Field 55, Mastercard's calls it DE 55, and both are talking about the same one hundred and thirty-four bytes.

There is a stronger version of the correction. Where point-to-point encryption is in use, as in the previous chapter, the card number is not in the terminal's message at all. What the terminal sends is a ciphertext and a key serial number. The card number appears for the first time inside a hardware security module in the acquirer's data centre, and is written into the message there. The field that

everybody thinks of as the most important one in a payment message is, in those estates, authored by the acquirer.

Field 55 is not "the chip data"

It is a selection from the chip data, made by the acquirer, and about half of it was written by the terminal rather than the card.

The card gives the terminal far more than travels. Public key certificates, the issuer's public key remainder, the card's own list of acceptable ways of verifying a cardholder, its offline counters and limits, its PIN try counter, its transaction log, the templates telling the terminal how to format commands: none of that leaves the shop. What leaves is a list of tags that the acquirer's specification names, and if a tag is not on the list it does not travel however interesting it is.

The other half of the correction is the part that surprises people who have only read about EMV. Of the twenty-one objects in our one hundred and thirty-four bytes, thirteen were written by the terminal: the amount, the currency, the country, the date, the transaction type, the random number, the terminal's type and capabilities, its serial number, its account of cardholder verification. They are in the chip's sealed block not because the chip knows them but because the chip was asked to sign over them. That is the entire mechanism by which an issuer can detect that the amount was altered after the card agreed to it.

The same fact appears twice, and only one copy is protected

The amount is in the message twice: once in the ordinary amount field, and once inside the sealed block. So is the currency. So is the date, in two different formats. So is the way the card was read. So is the card's sequence number. This is not redundancy for safety; it is two systems that were designed twenty years apart and glued together.

The consequence is a rule worth memorising. Only the copy inside the sealed block is protected by the cryptogram. If the two copies disagree, the issuer's cryptogram check will validate against the inner copy and fail against the outer one, and the failure will look like a cryptographic problem when it is really a mapping bug in somebody's message translator. A great deal of expensive debugging in this industry is somebody discovering that their host was populating the outer amount from the basket total and the inner one from the pre-authorisation amount.

One message is not a transaction

The plain version tells the story as request and answer, which is how it looks from the counter. It is not how the money moves. The authorisation is a question about whether funds exist and whether the issuer consents. It moves no money and, in most schemes, is not even the record that will be settled. A second message follows, telling the acquirer that the sale actually completed and carrying the chip's final cryptogram; a third, in a batch overnight, is the one that claims the money. If the answer to the first message never arrives, a fourth kind of message has to be sent to unwind it, and that message must repeat enough of the first to name it unambiguously.

Approval is not payment. Chapter 30 is about the difference, and Chapter 49 is about what happens when somebody disputes the whole thing eleven months later and the only surviving evidence is the twelve-character reference the acquirer stamped on this message today.

The technical version

The three hops, and the protocol on each

The path from the counter to the issuer has at least three links, and each has a different standard.

The first, where the terminal is not a standalone device, is between the electronic cash register and the point of interaction. This is the link nobody certifies as a payment interface, and it carries the basket total in one direction and the receipt data in the other. In Europe it is increasingly the nexo Retailer Protocol, an ISO 20022 message set defining the exchanges between a Sale System and a POI System.

The second link is terminal to acquirer. In the United Kingdom this was standardised by APACS: Standard 60, *Message Interchange between Card Acceptor and Acquirer*, dated 2000, and its successor Standard 70, issued from 2007 in numbered books, in which Books 2 and 3 carry the message formats while the remaining books apply whichever message standard the terminal uses. Elsewhere in Europe the modern answer is the nexo Acquirer Protocol, described by nexo as a standard that "replaces ISO 8583 and its national derivatives", built from the ISO 20022 Acceptor-to-Acquirer Card Transactions catalogue. Its authorisation pair is `caaa.001.001.15`, `AcceptorAuthorisationRequestV15`, and `caaa.002`, `AcceptorAuthorisationResponse`; its completion advice is `caaa.003`. Many acquirers offer none of this and publish their own format, which in a modern gateway means a JSON document with an `emvData` string in it.

The third link is acquirer to scheme, and this is where ISO 8583 lives. The fourth, scheme to issuer, is ISO 8583 again in the scheme's own dialect, or increasingly the ISO 20022 Acquirer-to-Issuer set, whose authorisation initiation is `cain.001.001.05`.

ISO 8583, in exactly enough detail to hand over

Three facts will carry you to the next chapter.

A message begins with a four-digit **message type indicator**. `0100` is an authorisation request, `0110` its response, `0200` a financial request that both authorises and claims, `0210` its response, `0400` a reversal.

After it comes a **bitmap**, sixty-four bits, one per possible data element. A bit set to one means that element is present later in the message, in numerical order. This is why the format is compact and why it is unreadable without a specification.

Then come the **data elements** themselves, identified only by their position in the bitmap. They are numbered, not named, and the numbering is the industry's shared vocabulary: everybody in acquiring knows what DE 4 is without looking. Volume III takes this apart properly.

For now, one number matters: **55**. It is the element that carries chip data. Later editions of the standard formalise it as the integrated-circuit-card-related data element, with a three-digit length prefix and a theoretical maximum of 999 bytes; every scheme caps it far lower, and 255 bytes is the common ceiling. Crucially, the standard is explicit that its own rules for structured elements do not apply inside DE 55: the linking of sub-elements there follows ISO/IEC 7816-6. In plain terms, field 55 is a bag of EMV tag-length-value objects, parsed by EMV's rules, sitting inside a message parsed by ISO 8583's rules. It is a format inside a format, and every practitioner error in this area comes from forgetting which set of rules applies to which bytes.

What the kernel hands the terminal

Before anything is assembled, the chip's part of the transaction ends and a boundary is crossed inside the terminal itself.

In the contact world the boundary is the second GENERATE AC, or, for an online transaction, the first: the card returns the cryptogram, the counter, the cryptogram information data and its issuer

application data, in one of the two response templates of EMV Book 3. In the contactless world the boundary is explicit and named. The kernel completes and hands the terminal an outcome carrying a **Data Record** and **Discretionary Data**; EMV Contactless Book A's Annex A lists which data elements populate each. The Data Record is the set the terminal is expected to forward to the acquirer. The Discretionary Data is for the terminal's own logs and receipts and is not meant to travel.

Either way, what the terminal now holds is a pool of tags — some from the card, some of its own — and a list, supplied by the acquirer, of which ones to send.

Field 55, tag by tag, for one flat white

Here is the actual field for the £4.20 transaction. The values follow the worked cryptogram of Chapter 15 and the terminal risk management of Chapter 16, so the arithmetic joins up with what those chapters computed.

Tag	Object	Bytes	Value	Written by
9F26	Application Cryptogram	8	733084432BA24DAE	Card
9F27	Cryptogram Information Data	1	80	Card
9F10	Issuer Application Data	7	06000C03A00000	Card
9F37	Unpredictable Number	4	9A1BC3D4	Terminal
9F36	Application Transaction Counter	2	003D	Card
95	Terminal Verification Results	5	0000008000	Terminal
9A	Transaction Date	3	260817	Terminal
9C	Transaction Type	1	00	Terminal
9F02	Amount, Authorised	6	000000000420	Terminal
9F03	Amount, Other	6	000000000000	Terminal
5F2A	Transaction Currency Code	2	0826	Terminal
82	Application Interchange Profile	2	1D00	Card
9F1A	Terminal Country Code	2	0826	Terminal
9F33	Terminal Capabilities	3	E0F8C8	Terminal
9F34	CVM Results	3	1F0002	Terminal
9F35	Terminal Type	1	22	Terminal
9F1E	Interface Device Serial Number	8	3132333435363738	Terminal
84	Dedicated File Name	7	A0000000031010	Card
9F09	Application Version Number	2	008C	Terminal

Tag	Object	Bytes	Value	Written by
9F41	Transaction Sequence Counter	2	0071	Terminal
5F34	Application PAN Sequence Number	1	01	Card

Read down the values and the transaction is legible. 9F27 of 80 says the card produced an authorisation request cryptogram rather than an approval or a decline, because the top two bits of that byte carry the cryptogram type. 95 has one bit set, byte 4 bit 8, "transaction exceeds floor limit", and that single bit is the reason this transaction is online at all. 9F34 of 1F 00 02 says the terminal applied the rule "no cardholder verification required", under the condition "always", and considers it successful — a statement, as Chapter 17 laboured, not a proof. 9F10 of 06 00 0C 03A00000 tells the issuer's host which master key to load and which cryptogram version to run, and carries the card's own four-byte account of the transaction, which the schemes append to the terminal's data before recomputing the MAC. 82 has byte 1 bit 3 set, so the card is willing to be told, cryptographically, that the reply really came from its issuer. And 9F41 of 0071 is the terminal's own transaction counter: the one hundred and thirteenth sale on that machine since it was last reset.

Encoded, the whole field is:

```
9F2608733084432BA24DAE 9F270180 9F100706000C03A00000
9F37049A1BC3D4 9F3602003D 95050000008000 9A03260817 9C0100
9F0206000000000420 9F0306000000000000 5F2A020826 82021D00
9F1A020826 9F3303E0F8C8 9F34031F0002 9F350122
9F1E083132333435363738 8407A0000000031010 9F0902008C
9F41020071 5F340101
```

One hundred and thirty-four bytes. Add the tags and lengths and there is no slack in it: 9F26 costs eleven bytes to carry eight, because a two-byte tag and a one-byte length ride with every object.

The byte budget, and what truncation does

That is the arithmetic nobody does until it bites. Twenty-one objects, one hundred and thirty-four bytes, against a scheme ceiling of 255. Comfortable. Now add the tags that a contactless transaction wants — the terminal transaction qualifiers, the card transaction qualifiers, the form factor indicator — and the tags a tokenised transaction wants, and the issuer script results after a script has run, and a kernel's proprietary data objects, and the budget tightens quickly.

Two encodings make it tighter still. If the interface between the terminal and the acquirer is character-based and carries the field as hexadecimal text, one hundred and thirty-four bytes become two hundred and sixty-eight characters. Any component in that chain with a 255-character limit will silently cut the field in the middle of a tag. And a field 55 truncated at the end loses its trailing objects first, which in the list above means 5F34, 9F41 and 9F09 — the least interesting three. Reorder the same list, as BER-TLV entitles you to, and truncation eats 9F10 instead, at which point the issuer cannot select a key, cannot derive a session key, cannot verify the cryptogram, and declines with a code that says nothing about why.

Which tags an acquirer requires is published, and reading that list is the first task of any integration. One large processor's public specification for encrypted device integration, for instance, marks as mandatory 4F, 5F2A, 5F34, 82, 84, 95, 9A, 9C, 9F02, 9F03, 9F09, 9F10, 9F1A, 9F26, 9F27, 9F33, 9F34,

9F35, 9F36 and 9F37, with 9F1E, 9F41, 50, 9F5B and 9F6E optional, and in the response 8A, 91, 71 and 72. That shape — the card's cryptogram set, the terminal's signed context, the identity of the application — is common to every scheme, and the differences between acquirers are at the edges.

Read those lists sceptically. The same document describes tag 9F6E as the application transaction counter, which it is not, and tag 9F1A as a two-letter alphabetic country code, which it is not either: 9F1A is three numeric digits in two bytes, 0826 for the United Kingdom. Vendor documentation copies vendor documentation. The primary sources are EMV Book 3's Annex A for what a tag is, and your acquirer's own message specification for whether they want it.

What the terminal adds outside field 55

The rest of the terminal's contribution is the ordinary transaction, and its home in a scheme's authorisation request looks like this.

DE	Element	Value here	Where the value comes from
2	Primary account number	4111111111111111	Card, or the acquirer's decryption HSM
3	Processing code	000000	Transaction type at the till
4	Amount, transaction	000000000420	The basket
11	System trace audit number	004871	Message originator
12	Local transaction time	094512	Terminal clock, local
13	Local transaction date	0817	Terminal clock, local
14	Expiration date	2903	Card
18	Merchant type	5812	Acquirer's record of the merchant
22	Point of service entry mode	051	Terminal
23	Application PAN sequence number	001	Card, from tag 5F34
25	Point of service condition code	00	Transaction context
35	Track 2 data	From tag 57	Card
41	Card acceptor terminal identification	20194831	Acquirer, at installation
42	Card acceptor identification code	602948300012345	Acquirer, at contract
43	Card acceptor name and location	40 characters	Acquirer
49	Currency code, transaction	826	Terminal configuration
55	Integrated circuit card data	134 bytes	Card and terminal

Several rows deserve a note.

DE 22 is three digits in the 1987-format messages: two for how the account number was captured and one for the terminal's PIN entry capability. 05 is contact chip, 07 contactless chip, 91 the legacy contactless stripe mode, 80 a fallback to magnetic stripe after the chip failed. Ours is 051: chip read, PIN pad present. This field is where the commercial consequences live. Claim 05 and send no field

55 and you have failed a scheme certification test in the most common way there is. Claim `07` after performing a stripe read and the issuer looks for a cryptogram that is not there. And the third digit must be true: a terminal that claims PIN capability and never sends a PIN block invites an issuer to distrust everything else it says.

DE 23 is three digits carrying what the chip holds in one byte, right-justified and zero-filled. `5F34` of `01` becomes `001`. This is the field most often left out, and the failure it causes is specific: an issuer with two cards live on one account cannot tell which one it is being asked about, so it cannot select the right master key.

DE 35 carries the track 2 equivalent data the chip returned in tag `57` — not a stripe read, but the same layout, including a service code whose first digit of `2` or `6` announces that this is a chip card. The separator the chip encodes as hexadecimal `D` appears as `=` on character-based interfaces, and a translator that fails to swap it produces a field the issuer rejects for format.

DE 12 and DE 13 are local. **DE 7**, which the acquirer writes, is universal time. Our transaction at 09:45:12 British Summer Time is `094512` in DE 12 and `0817084512` in DE 7. Neither DE 12 nor DE 13 carries a year, which is a real problem exactly once a year.

DE 41 and DE 42 are not the terminal's opinion. They are identifiers assigned by the acquirer and loaded into the terminal at installation, which is why a terminal moved between shops without reconfiguration produces transactions that reconcile to the wrong merchant and are extremely tedious to unpick.

The PIN, when there is one

Our flat white needed no cardholder verification. Had it been a £47.50 purchase requiring online PIN, one more field would have appeared and it would not have come from the chip at all.

The PIN, captured on the approved keypad, is formatted into an ISO 9564-1 PIN block and encrypted under a key shared with the acquirer. In the 1987-format messages it occupies **DE 52**, defined as sixty-four bits — exactly one Triple DES block, which is why formats 0 to 3 are eight bytes long. It travels under a different key from the card data, with the key management information alongside it in **DE 53**, whose layout differs in every scheme's dialect. At each hop between the terminal and the issuer's own hardware security module the block is translated: decrypted and re-encrypted under the next zone's key, inside a secure cryptographic device, never in the clear on any bus. That regime is the PCI PIN Security Requirements, and it is older and stricter than anything governing the card number.

One seam is worth flagging now because it is a live migration problem. ISO 9564-1's format 4 block, introduced for AES, is sixteen bytes. A sixty-four-bit field cannot hold it. Carrying AES PIN blocks therefore requires arrangements outside the original definition of DE 52, and they are specified by each scheme rather than by ISO 8583. If you are planning an AES migration, that is a question for your acquirer's message specification and not for the standard.

Note also what the PIN is not. If the cardholder verification was offline PIN, nothing about the PIN appears in the authorisation message at all. The only trace is `9F34` saying it happened, and the card's own account inside `9F10` saying whether it agrees.

What the acquirer adds

The acquirer's host receives the terminal's message and builds the scheme message. Its additions fall into four groups.

Identity and routing. The acquiring institution identification code in **DE 32**, and a forwarding institution code in **DE 33** where an intermediary is in the path. The acquirer also does the routing itself: the leading digits of the account number are looked up in a table to decide which scheme, and therefore which network connection, this message belongs to.

Timekeeping and identifiers. DE 7, transmission date and time, in universal time, rewritten at every hop by whoever is sending. **DE 37**, the retrieval reference number, twelve alphanumeric characters, assigned by the acquirer and echoed unchanged by everybody downstream. Its internal construction is not fixed by the standard; a common convention is year, day of year, hour and sequence, which is how `622908004871` was built for this transaction: year 6, day 229 — 17 August 2026 — hour 08 universal time, sequence 004871. That string is the identifier with the longest life in the entire message. It will still be the key in a chargeback file next summer.

Merchant description. DE 18, the merchant category code — `5812`, eating places and restaurants, from ISO 18245 — and **DE 43**, forty characters of name and location, conventionally twenty-three of trading name, thirteen of town, two of region and two of country. Both are the acquirer's records rather than the terminal's, and the second is what appears on the cardholder's statement, which makes it the single most effective fraud-prevention field in the message and the one most likely to be wrong.

Cryptography and scheme dialect. The account data, if it arrived encrypted, is decrypted in a hardware security module and written into **DE 2** and **DE 35**. The PIN block, if present, is translated. A message authentication code over the whole message may be computed and placed in **DE 64**. And the scheme's own private data elements are populated: Mastercard's DE 61 point-of-service data, DE 48 additional data, DE 63 network data, and their equivalents elsewhere. Those are the fields where scheme dialects diverge most, and a field number learned on one network must never be carried across to another.

Two things the acquirer does that are not fields at all. It applies its own duplicate detection and velocity rules before the message leaves, which is why an acquirer can decline a transaction the issuer never saw. And it keeps a record of what it sent, because if the response does not arrive it is the acquirer, not the terminal, that must reverse.

The assembled message

Put it together and the `0100` for our flat white is a four-digit message type indicator, an eight-byte bitmap with seventeen bits set, and the elements in numerical order. In character encoding with a binary field 55, it is a little over three hundred and fifty bytes. Rendering field 55 as hexadecimal text pushes it near five hundred. Packing the numerics as binary-coded decimal brings it well under three hundred.

American Express's *Global Credit Authorization Guide*, written to the 1993 edition of the standard, states the chip requirement plainly: ICC system related data in data field 55 must be present, alongside track 1 in data field 45 and/or track 2 in data field 35, and a point of service data code in data field 22. Visa's *Transaction Acceptance Device Guide* puts the division of labour equally plainly from the other side: the acquirer formats the authorisation message with the chip data in Field 55. Both sentences say the same thing. The terminal produces chip data; somebody else builds the message; the two must agree about what the transaction was.

The response, and what the terminal does with it

The `0110` comes back with three things that matter and one that matters to the chip.

DE 39, the response code, two alphanumeric characters in the 1987 edition and three in the 1993 edition: `00` approved. **DE 38**, the authorisation identification response, six characters, which the receipt prints and the clearing record carries. And **DE 55** again, now populated by the issuer rather than the terminal.

In the response, field 55 carries up to three objects.

Tag	Object	Purpose
`91`	Issuer Authentication Data	Proves to the chip that the issuer answered
`71`	Issuer Script Template 1	Commands for the chip, before the final cryptogram
`72`	Issuer Script Template 2	Commands for the chip, after it

For our transaction, under ARPC Method 1 and with the two ASCII characters `0` and `0` as the authorisation response code, Chapter 15 computed the issuer's cryptogram as `735C24FD87F1BF0E`. The issuer authentication data is that cryptogram followed by the response code, so the object on the wire is `91 0A 735C24FD87F1BF0E3030`: ten bytes of value, twelve on the wire.

What the terminal does next is not optional and is where implementations diverge from the specification most often. If `91` is present, the terminal delivers it to the card, either through EXTERNAL AUTHENTICATE or inside the second GENERATE AC, and sets the issuer-authentication bit in the transaction status information. If `91` is absent, the terminal must not attempt issuer authentication, must leave that status bit at zero, must leave the corresponding failure bit in the terminal verification results clear, and must proceed on the response code alone. A downgraded authorisation is not a failed one.

Then the terminal asks the card once more, using the second data object list, for a completion cryptogram, and passes it the two-byte authorisation response code in tag `8A` — which it must pass through exactly as received and must never invent when it came from a response message. The card runs its own second round of risk management and answers with a transaction certificate if it is content or a decline cryptogram if it is not. A card that declines after an approved authorisation has almost always either failed to verify the issuer's cryptogram or verified it and found the approval bit clear.

Where scripts were sent, the terminal delivers them, records what happened in tag `9F5B`, the issuer script results, and sets the script-processing bit in the transaction status information. Kernel 3 is relaxed about how much of `9F5B` travels: acquirer documentation quoting it notes that it is acceptable to transmit only byte 1, though preferable to send all five.

For our transaction the transaction status information ends as `B8 00`: offline data authentication performed, card risk management performed, issuer authentication performed, terminal risk management performed, no scripts. That two-byte object, and the final cryptogram, belong to the next message rather than this one.

The second and third messages

The authorisation is finished and nothing has been paid.

What follows depends on the scheme and the acquirer. In a dual-message model the terminal or its host sends a completion advice — `caaa.003` in the nexo set, a scheme-specific advice or an entry in the overnight batch elsewhere — carrying the transaction certificate in the same tag `9F26`, the cryptogram information data now showing a transaction certificate rather than a request, the transaction status information, and the script results. That message is what tells the acquirer the sale actually happened. The clearing record built from it is what claims the money, and it is the record a dispute will be argued from.

In a single-message model the `0200` did both jobs at once, and the chip data requirements are the same.

And if the response never arrived, the acquirer must reverse. A reversal names its victim by repeating the original message type indicator, system trace audit number, transmission date and time and institution codes, and acquirer specifications commonly require that the original field 55 be repeated in the reversal too. This is the practical reason for a rule that sounds pedantic: keep the exact bytes you sent, not a re-derivation of them. A reversal built by regenerating field 55 from a database will differ from the original in some small way and will fail to match.

A short field guide to failures

DE 22 says chip and field 55 is absent. The commonest certification failure in the industry. The message claims a chip was used and contains no evidence of one.

Field 55 truncated. Trailing tags vanish. If `9F10` was among them the issuer cannot select a key and declines uninformatively. Check the length before suspecting the cryptogram.

Hexadecimal where binary was expected, or the reverse. Every length doubles or halves and parsing fails at the first tag. This is the single most common integration fault on a new acquirer interface.

Flat parser meets a constructed tag. `71` and `72` contain further tag-length-value objects. A parser that treats them as opaque will deliver an issuer script the card cannot read.

Amount mismatch between DE 4 and `9F02`. The cryptogram validates against `9F02`. If the two disagree the issuer sees evidence of tampering, which is exactly what the design intends, and the tampering is your own message translator.

Card sequence number omitted. Two cards on one account; the issuer loads the wrong master key; cryptogram verification fails for a card that is entirely genuine.

Response code altered before the second GENERATE AC. A host that normalises `00` into something friendlier for its own logs, and passes the friendly version to the card, breaks issuer authentication and gets a decline cryptogram after an approval.

Reversal rebuilt rather than replayed. The original and the reversal differ in one byte and never match. This is how transactions become stuck: authorised, never completed, never reversed, and visible to the cardholder as a held balance for a week.

What you now know, and what Volume III answers

Volume II has been a single argument, and it is worth stating in one place before the subject changes.

A card is a physical object carrying an account number whose structure is public and whose last digit is arithmetic. The number alone is a bearer credential and always has been, which is why the printed codes, the stripe codes and the chip codes are four different things with four different threat models.

The chip is a computer running a specified application, and its value is not that it stores a secret but that it computes one: eight bytes that depend on a key never transmitted, a counter that only rises, and the exact terms of this transaction. Whether it goes online is decided by bit arithmetic between the terminal's observations and the issuer's declared policies, both held as five-byte lists. Whether the cardholder was verified is a claim the terminal writes down, not a fact the network can check. Contactless is the same machinery with different limits and a different kernel. A phone holds not your card number but a separate credential minted for that device. The terminal is a certified object whose configuration is auditable after the fact, and in a well-built estate it can encrypt the account number so thoroughly that the shop itself never sees it. And all of it converges on the message this chapter assembled: three hundred and fifty-odd bytes, one hundred and thirty-four of them a sealed statement the shop cannot read.

Volume III begins the moment that message leaves the acquirer's building, and answers the questions this chapter has deliberately left open.

Question	Chapter
Who are the parties, and who pays whom?	23, The Four-Party Model
What is ISO 8583, precisely, and why does it survive?	24
How does a bitmap work, and why not text?	25
What does each field mean, field by field?	26
What does one whole transaction look like, end to end?	27
What is the six-character approval code actually worth?	28
What does a decline code mean, and what may you retry?	29
Where does authorisation end and payment begin?	30
Where does the £4.20 go, and how much of it arrives?	31

The terminal's job ends with a message. The rest of this book is about what a message is worth.

22.98 Common wrong ideas

Wrong: the terminal builds the message the banks read. **Right:** in almost every real estate the terminal speaks a protocol of its own — a national standard, a vendor format, increasingly a web API carrying JSON — and the acquirer's host builds the industry-standard message from scratch.

Wrong: field 55 is the chip data. **Right:** it is a selection from the chip data, named by the acquirer's specification, and about half of it was written by the terminal rather than the card.

Wrong: the card number is always the card's contribution to the message. **Right:** where point-to-point encryption is in use the terminal sends ciphertext and a key serial number, and DE 2 is authored inside a hardware security module in the acquirer's data centre.

Wrong: the amount appearing twice is harmless redundancy. **Right:** only the copy inside the sealed block is protected by the cryptogram, so if the two disagree the issuer sees evidence of tampering, and the tampering is your own message translator.

Wrong: an approval means the money is on its way. **Right:** the authorisation moves nothing and in most schemes is not even the record that will be settled; a completion advice follows, and a clearing record is what claims the money.

Wrong: a missing tag `91` means issuer authentication failed. **Right:** if `91` is absent the terminal must not attempt issuer authentication, must leave the status and failure bits clear, and must proceed on the response code alone, because a downgraded authorisation is not a failed one.

Wrong: the card sequence number is an optional nicety. **Right:** it is the field most often left out, and without it an issuer with two cards live on one account cannot select the right master key, so cryptogram verification fails for a card that is entirely genuine.

Wrong: a host may tidy the authorisation response code before handing it to the card. **Right:** tag `8A` must be passed through exactly as received and never invented, because a host that normalises `00` into something friendlier breaks issuer authentication and gets a decline cryptogram after an approval.

Wrong: tags may be reordered freely inside field 55 because BER-TLV permits it. **Right:** it does permit it, and the order decides what truncation eats first; lose `9F10` and the issuer cannot select a key, cannot derive a session key, and declines uninformatively.

Wrong: vendor tag documentation is a primary source. **Right:** vendor documentation copies vendor documentation, and published specifications in circulation describe `9F6E` as the application transaction counter and `9F1A` as a two-letter alphabetic country code, neither of which is true.

22.99 Chapter summary in 20 lines

1. About three hundred and fifty bytes leave the shop, and they contain two things of quite different natures: facts asserted by the machine on the counter, and a sealed block written by the chip.
2. Three parties write the outbound message — the chip, the terminal and the acquirer — and a fourth, the issuer, writes the reply.
3. For a £4.20 flat white the chip's sealed block is one hundred and thirty-four bytes holding twenty-one objects, eight of them written by the card.
4. Thirteen of those objects were written by the terminal and merely signed over by the chip, which is the entire mechanism by which an issuer detects that an amount was altered after the card agreed to it.
5. The terminal usually knows nothing about numbered data elements; it sends named fields down whatever pipe it has, and something in a data centre translates.
6. Field 55 is a bag of EMV tag-length-value objects parsed by EMV's rules, sitting inside a message parsed by ISO 8583's rules, and most practitioner errors here come from applying the wrong rule-book to the wrong bytes.
7. Reading the tags down the list makes the transaction legible: `9F27` of `80` says the card asked the bank, and one bit of `95` says the amount exceeded the floor limit, which is the only reason this sale went online.
8. Several facts appear twice, inside and outside the sealed block, because two systems designed twenty years apart were glued together, and only the inner copy is cryptographically protected.
9. The byte budget is real: one hundred and thirty-four bytes against a 255-byte ceiling is comfortable until contactless, tokenisation and script results are added, and hexadecimal encoding doubles everything at a stroke.
10. Truncation removes the trailing objects first, so the order in which tags are written decides whether the loss is trivial or fatal.
11. Outside field 55 the terminal contributes the ordinary transaction fields, while the terminal and merchant identifiers were assigned by the acquirer at installation and at contract.

12. DE 22 declares how the card was read and whether a PIN pad is present, and every digit of it must be true, because claiming a chip read with no field 55 is the commonest certification failure there is.
13. Where an online PIN exists it travels in DE 52 under a different key from the card data, translated inside a secure cryptographic device at every hop under the PCI PIN Security Requirements.
14. Offline PIN leaves no trace in the authorisation at all beyond the terminal's claim in `9F34` and the card's own account of it inside `9F10`.
15. The acquirer adds identity and routing, universal time, the retrieval reference number, the merchant category code and statement descriptor, and the scheme's own private data elements.
16. That twelve-character retrieval reference number has the longest life of anything in the message and will still be the key in a chargeback file next summer.
17. The response carries a two-character verdict, a six-character approval code, and optionally a second sealed object addressed to the chip that the terminal cannot read and simply passes on.
18. If tag `91` is present the terminal must deliver it to the card and set the issuer-authentication bit; if it is absent it must do neither and rely on the response code.
19. The authorisation then ends with nothing paid: a completion advice carries the final cryptogram, a clearing record claims the money, and a reversal — replayed byte for byte, never rebuilt — unwinds a request whose answer never came.
20. Volume II closes here because everything before it concerned one end of the wire, and everything after it concerns what a message is worth once it leaves the acquirer's building.

Sources: EMV Integrated Circuit Card Specifications for Payment Systems, Books 2, 3 and 4, in particular Book 2 sections 8.1 and 8.2 for cryptogram and ARPC generation, Book 3 Annex A for the data element dictionary and Annex C for the terminal verification results and cardholder verification coding, and Book 4 section 12 for online processing and issuer authentication; EMV Contactless Specifications for Payment Systems, Book A Annex A for the Data Record and Discretionary Data, and Book C-3 for Kernel 3's issuer script results requirement; EMV Payment Tokenisation Specification – Technical Framework for token carriage; ISO 8583 and its 1987, 1993, 2003 and 2023 editions for data element definitions and for the statement that DE 55 sub-element linking follows ISO/IEC 7816-6; ISO/IEC 7816-4 and 7816-6; ISO/IEC 7813 for track 2 and the service code; ISO 4217 and ISO 18245; ISO 9564-1:2017 for PIN block formats; ISO 20022 message catalogue, Acceptor to Acquirer Card Transactions (caaa.001.001.15 and its siblings) and Acquirer to Issuer Card Transactions (cain.001.001.05), with nexo standards' own descriptions of the nexo Acquirer and Retailer Protocols; APACS Standard 60:2000, Message Interchange between Card Acceptor and Acquirer, and Standard 70 Books 1 to 7 (2007); Visa Transaction Acceptance Device Guide v3.3; American Express Global Credit Authorization Guide (ISO 8583:1993); Mastercard Transaction Processing Rules; Worldpay ISO 8583 Reference Guide v2.51 and 610 Interface Reference Guide v2.56 for a published acquirer dialect; Fiserv encrypted device integration documentation for a published mandatory and optional tag list, cited also as an example of vendor documentation misdescribing tags 9F6E and 9F1A; PCI PIN Security Requirements v3.1 and PCI PTS POI v7.0 for PIN capture and translation. Tag names, formats and lengths cross-checked against EFTlab's EMV and NFC tag list, neapay's tag reference and emvlab.org.

VOLUME III

The Networks

The four-party model, ISO 8583 field by field, the several numbers all called the transaction reference, and where the money goes.

23

CHAPTER

The Four-Party Model

23.0 What this chapter gives you

1. You will be able to name the five roles in a card payment and say which of them holds a contract with which others, and which pairs have no contract at all.
2. You will be able to break a fifty pound sale into interchange, scheme fees and acquirer margin, and say who receives each slice.
3. You will be able to correct the commonest error in payments journalism, that a scheme charges shops 0.3 per cent, and explain why it is wrong twice over.
4. You will be able to explain why a merchant in a dispute has no cause of action against the issuer and must raise the matter through its own acquirer.
5. You will be able to say why a scheme with a three-party corporate structure can be classified as four-party, and what that reclassification costs it.
6. You will be able to tell an independent sales organisation from a payment facilitator by asking who holds the merchant contract and who bears the chargeback.
7. You will be able to explain why interchange is capped by statute in Europe and published by the scheme in the United States, and what Regulation II's exclusivity ban does to American debit economics.
8. You will be able to explain what IC++ pricing shows a merchant that blended pricing hides, and which of the three numbers is actually negotiable.
9. You will be able to explain why an outage at a card scheme stops new authorisations without destroying value that has already cleared.
10. You will be able to trace where the cost of a cashback reward finally falls, including on the customer paying cash.

The plain version

Imagine a very large school. Nobody carries cash, because cash gets lost. Instead every pupil has an account with one of several house banks, and every shop in the town around the school has an account with one of several other banks. The shops want to sell to the pupils. The pupils want to buy from the shops. The problem is that a shop banking with Barnaby's has no idea whether a pupil banking with Cresswell's actually has any money, and no reason to trust Cresswell's if it says so.

So the school sets up a switchboard in the middle, with a rulebook attached to it. The rules are the important part. They say what a shop is allowed to ask, what a bank must answer, how fast it must answer, what happens when the answer is wrong, and who pays when a pupil says "I never bought that." Every bank that wants to join signs the rulebook. Every shop signs a contract with its own bank, not with the switchboard and not with anybody else's bank.

Now watch what happens when a pupil buys a pair of trainers.

The pupil hands a card to the shop. The shop's till sends a message to the shop's own bank: "Someone is claiming to be a customer of another bank. This is the card number, this is the amount, this is my shop." The shop's bank looks at the first few digits of the card number, which act like a postcode, and sees that this card belongs to Cresswell's. The shop's bank does not phone Cresswell's. It passes the message to the switchboard. The switchboard passes it to Cresswell's. Cresswell's checks the balance, checks that the card is not reported stolen, decides, and sends one word back down the same chain: approved. The whole round trip takes about a second, and the shop's till prints a receipt.

That is four parties. The pupil. The shop. The shop's bank. The pupil's bank. And a fifth thing in the middle which is not a party to the purchase at all: the switchboard and its rulebook. In the real world the pupil is the *cardholder*, the shop is the *merchant*, the shop's bank is the *acquirer*, the pupil's bank is the *issuer*, and the thing in the middle is the *scheme*. Visa is a scheme. Mastercard is a scheme. Neither of them issued your card and neither of them has a contract with the shop.

Where the money actually goes

Here is the part that surprises people, so let us do it with real numbers.

Say the trainers cost fifty pounds, and the pupil pays with an ordinary British consumer credit card at a British shop. Your statement will say £50.00. The shop will not receive £50.00. It will receive something like £49.65.

The missing 35 pence splits three ways.

The largest slice, in this example fifteen pence, goes from the shop's bank to the pupil's bank. It is called *interchange*, and in the United Kingdom and the European Union it is capped by law at 0.3 per cent of the transaction for a consumer credit card and 0.2 per cent for a consumer debit card. Nought point three per cent of fifty pounds is fifteen pence. That money does not go to Visa or Mastercard. It goes to the bank that issued the card, and it is one of the reasons your bank is happy to give you a card, send you a replacement when you lose it, and refund you when someone in another country buys a television with your card number.

A smaller slice, call it five pence, goes to the scheme. These are *scheme fees*, and both banks pay them: the pupil's bank pays a fee for having its transaction switched, and the shop's bank pays one too. This is how Visa and Mastercard make money. It is a much thinner slice than most people assume.

What is left, about fifteen pence here, is the shop's bank's own margin. It is the price of the terminal, the fraud checks, the money the bank fronts to the shop tomorrow morning before it has actually been paid by the pupil's bank, and the risk that the shop goes bust next month owing refunds. The shop negotiated that margin when it signed its contract, and a big supermarket negotiates a very different number from a corner café.

Add them up: fifteen plus five plus fifteen is thirty-five pence on a fifty pound sale, which is 0.7 per cent. The whole deduction, taken together, is called the *merchant service charge*. The shop's bank quotes it, collects it, and passes the interchange and scheme fee parts onwards.

Approving is not paying

The second thing that surprises people is that nothing has moved yet. When Cresswell's said "approved", it did not send any money. It made a promise, and it set aside fifty pounds of the pupil's available balance so it could keep that promise. Actual money moves later, usually the next working day, and it moves in bulk. At the end of the day the switchboard adds up everything every bank

owes every other bank, cancels most of it out, and produces one number per bank: Cresswell's owes 4.2 million pounds today, Barnaby's is due 3.8 million. Only those net numbers are actually paid, through the ordinary banking system, bank to bank. Thousands of individual card payments become one transfer.

So a shop can be approved and still not be paid. If the pupil later says "that wasn't me", the rulebook lets the pupil's bank claw the money back from the shop's bank, and the shop's bank takes it back off the shop. That is a *chargeback*, and it travels back along exactly the same chain of contracts the payment came down.

Why the shop never phones your bank

Because it has no relationship with your bank whatsoever. There is no contract between them, no account, no phone number, no way to prove who is calling. The shop's only counterparty is its own bank. Your only counterparty is your own bank. The two banks' only shared counterparty is the scheme, and the scheme's whole job is to be the trusted stranger in the middle so that a card issued by a bank in Manila works in a bakery in Cardiff without those two institutions ever having heard of each other.

There is a practical reason as well as a legal one. Visa alone had close to five billion cards in circulation and more than 175 million merchant locations in its 2025 financial year. If every merchant had to arrange terms with every issuing bank, the number of contracts would be the number of merchants multiplied by the number of banks, and nobody would ever accept a foreign card. The scheme reduces that multiplication to addition: every bank signs one rulebook, every merchant signs one contract with one acquirer, and everything connects.

Now for the interesting variation. American Express, for most of its history, did not do this. It issued its own cards to its own customers and signed up its own shops directly, both ends, one company. There was no other bank in the picture, so there was no interchange, because there was nobody to pay it to. Amex simply decided what to charge the shop and kept all of it. That let it pay for very generous cardholder rewards, and it is why shops complained that Amex was expensive and why, for decades, plenty of them did not take it.

Where the plain version stops being true

The switchboard analogy makes the scheme sound neutral. It is not. A telephone exchange does not set the price of the call, decide who is allowed to be a customer, or fine you for connecting badly. A card scheme does all three. It writes the rulebook, and the rulebook runs to hundreds of pages of binding obligations on both banks. It sets the *default* interchange rate, which is the rate that applies whenever an issuer and an acquirer have not agreed something bilaterally, which in practice is nearly always. It operates the network, sells processing and fraud services in competition with its own members' vendors, arbitrates disputes between the two banks, and levies non-compliance assessments. Calling it a neutral intermediary is the single most misleading thing in the plain version. The European Union thought so too: Article 7 of the Interchange Fee Regulation forces schemes to separate the scheme business from the processing business, with independent accounting, organisation and decision-making, precisely because the two roles conflict.

Interchange does not go to the scheme, and scheme fees are not interchange. In the plain version I kept these separate, but almost every article you will read outside this book blurs them, and practitioners will judge you instantly for it. Interchange is defined in the Regulation as a fee paid between the issuer and the acquirer for each transaction. The scheme sets the default level but receives none

of it. Scheme fees are a completely separate charge, billed by the scheme to its licensed members, and they are not published. When a newspaper says "Visa charges shops 0.3 per cent", it is wrong twice over: Visa charges the shop nothing, and the 0.3 per cent goes to a bank Visa does not own.

"Four parties" undercounts the participants by a factor of two or three. A real card payment in 2026 touches, at minimum, a payment gateway, possibly a payment facilitator or an independent sales organisation, an acquirer processor that may not be the acquirer, the acquiring bank of record, the scheme's authorisation network, the issuer's processor, the issuing bank, and often a token service provider, a 3-D Secure directory server and access control server, and a digital wallet provider holding the credential. "Four-party" is a legal and commercial classification of who holds the licences and who bears the liability. It is not a headcount of the boxes the message passes through, and it never was.

Three-party and four-party are regulatory categories, not descriptions of plumbing, and a scheme can cross the line without changing its logo. American Express has not been purely three-party for a long time. Through Global Network Services it licenses other banks to issue Amex-branded cards, and through OptBlue it lets third-party acquirers sign small merchants for Amex acceptance and set their own price. Under Article 1(5) of the Interchange Fee Regulation, a three-party scheme that licenses other payment service providers to issue or acquire, or that issues with a co-branding partner or through an agent, "is considered to be a four party payment card scheme" for the purposes of the rules. The classification follows the licensing behaviour, not the corporate structure. That is a live commercial decision with a price attached, not a historical footnote.

Finally, the scheme does not move any money. It calculates. The instruction that actually transfers value between two banks is executed outside the card network, through settlement banks and national payment systems, and the card scheme's role ends with producing an authoritative net figure and a settlement report. A card network is a messaging and rule system that happens to determine who owes what. The payment leg is somebody else's infrastructure.

The technical version

The five roles and the contract lattice

The precise definitions are worth quoting, because they are legally operative in the European Economic Area and in the United Kingdom, where the Regulation was retained after withdrawal from the European Union. Regulation (EU) 2015/751, the Interchange Fee Regulation, defines them in Article 2.

An *acquirer*, Article 2(1), is "a payment service provider contracting with a payee to accept and process card-based payment transactions, which result in a transfer of funds to the payee". An *issuer*, Article 2(2), is "a payment service provider contracting to provide a payer with a payment instrument to initiate and process the payer's card-based payment transactions". A *payment card scheme*, Article 2(16), is "a single set of rules, practices, standards and/or implementation guidelines for the execution of card-based payment transactions and which is separated from any infrastructure or payment system that supports its operation". Note what that definition does: it makes the scheme a rulebook, not a network. The network is a separate thing which the same corporate group may or may not own.

A *four party payment card scheme*, Article 2(17), is "a payment card scheme in which card-based payment transactions are made from the payment account of a payer to the payment account of a payee through the intermediation of the scheme, an issuer (on the payer's side) and an acquirer (on the payee's side)". A *three party payment card scheme*, Article 2(18), is "a payment card scheme in which the scheme

itself provides acquiring and issuing services and card-based payment transactions are made from the payment account of a payer to the payment account of a payee within the scheme".

The commercial architecture follows from the contracts, and the contracts are worth setting out explicitly because almost every dispute in card payments is resolved by asking who has privity with whom.

Contract	Parties	What it governs
Cardholder agreement	Cardholder and issuer	Credit line or account access, liability for unauthorised use, fees, dispute rights
Merchant agreement	Merchant and acquirer	Pricing, settlement timing, chargeback liability, reserves, prohibited categories, termination
Issuer licence	Issuer and scheme	Right to issue branded credentials, BIN ranges, rulebook compliance, scheme fees
Acquirer licence	Acquirer and scheme	Right to sign merchants and submit transactions, settlement obligations, scheme fees
Scheme rules	Binding on both licensees	Message formats, timeframes, interchange defaults, dispute process, fines

There is no contract between the merchant and the issuer. There is no contract between the cardholder and the acquirer. There is no contract between the merchant and the scheme, though the scheme's rules reach the merchant indirectly because the acquirer is obliged to flow them down into the merchant agreement. This flow-down is the mechanism by which a merchant in Leeds ends up bound by a rule written in San Francisco without ever having signed anything from San Francisco.

What each party actually does

The *issuer* underwrites and funds. It performs know-your-customer checks, sets credit limits or account access, manufactures or provisions the credential, and holds the ledger against which authorisations are drawn. Operationally its critical function is the authorisation decision: within a scheme-mandated response window it must evaluate the incoming request against balance, velocity, fraud models and any Strong Customer Authentication requirement, and answer. It bears fraud losses on transactions where the liability rules place them with it, funds the interchange it receives, and handles cardholder disputes. It is also the party that answers to the cardholder in law: in the United Kingdom, section 75 of the Consumer Credit Act 1974 makes a credit card issuer jointly and severally liable with the merchant for breach of contract or misrepresentation on qualifying purchases, which is a liability the merchant's own bank does not carry.

The *acquirer* underwrites and funds in the other direction, and this is consistently underappreciated. When an acquirer signs a merchant, it is taking a credit exposure, not selling a service. If the merchant takes payment for holidays in January, collapses in March, and every customer charges back, the acquirer must return the money to the issuers and then try to recover from an insolvent estate. That is why merchant underwriting looks like lending, why acquirers impose rolling reserves and delayed settlement on high-risk categories such as travel and event ticketing, and why merchant category codes matter so much. The acquirer also allocates merchant identifiers and terminal identifiers,

registers its agents with the schemes, ensures PCI DSS compliance across its estate, and funds the merchant, typically on a T+1 basis, before it has itself received settlement.

The *scheme* is a rule-maker, a brand, a numbering authority, a specification body and, in most cases, a network operator. It allocates issuer identification numbers, publishes and versions the message specifications, operates the authorisation switch and clearing system, calculates settlement positions, publishes default interchange, arbitrates issuer-acquirer disputes, runs compliance and fraud-monitoring programmes with financial penalties, and licenses participants. Its revenue is scheme fees, which are charged on both sides and generally combine a per-transaction element with an ad valorem element and a set of behavioural fees, plus a growing book of value-added services.

The scale of that thin slice is worth making concrete. Mastercard reported gross dollar volume of $10.632 trillion for 2025 with net revenue of $32.8 billion. That is roughly thirty-one basis points of volume, and the figure is generous to the network, because net revenue includes cyber, intelligence and other value-added services that are not switching at all, and is stated after customer incentives and rebates. Visa's fiscal 2025, ended 30 September 2025, ran to $16.7 trillion of total volume and $14.2 trillion of payments volume, with 329 billion Visa-branded transactions of which 258 billion were processed by Visa itself.

The message flow

Card authorisation is carried in ISO 8583, a message format that has outlived several attempts to replace it. Every message begins with a four-digit message type indicator. Position one gives the version of the standard: 0 for the 1987 edition, 1 for 1993, 2 for 2003. Position two gives the message class: 1xx for authorisation, 2xx for financial, 3xx for file actions, 4xx for reversal and chargeback, 8xx for network management. Position three gives the function: 0 request, 1 request response, 2 advice, 3 advice response. Position four gives the origin: 0 acquirer, 2 issuer, 4 other.

That yields the message types a practitioner sees daily. An 0100 is an authorisation request. An 0200 is a financial request. An 0800 is a network management request, which is how the two ends of a link exchange sign-on, sign-off and echo tests to prove the connection is alive.

Immediately after the MTI comes the primary bitmap, sixty-four bits indicating which of data elements 1 to 64 are present. Bit 1 is not a data element; it signals the presence of a secondary bitmap covering elements 65 to 128. This is why ISO 8583 is compact: absent fields cost nothing, and a parser walks the bitmap rather than the message.

The elements that carry the commercial meaning of a four-party transaction are these.

Element	Contents	Notes
DE2	Primary account number	Variable length; the routing key
DE3	Processing code	Transaction type and account types
DE4	Amount, transaction	Minor units, no decimal point
DE7	Transmission date and time	Set by the sending institution
DE11	System trace audit number	Six digits, assigned by the originator
DE12	Local transaction time	Time at the point of service
DE13	Local transaction date	Date at the point of service
DE37	Retrieval reference number	Twelve characters, the reconciliation handle
DE38	Authorisation identification response	Six characters, the approval code

Element	Contents	Notes
DE39	Response code	Two characters; values defined by scheme rules
DE41	Card acceptor terminal identification	Eight characters
DE49	Currency code, transaction	ISO 4217 numeric
DE55	Integrated circuit card system related data	EMV chip data

DE11 and DE37 deserve a note, because they are what makes a four-party model auditable. The system trace audit number is assigned by the message originator and echoed back, which lets a party match a response to its own request across a link where messages may arrive out of order. The retrieval reference number is the handle by which the same transaction is identified later, in clearing, in a retrieval request, and in a chargeback, potentially months afterwards and by an institution that was not present at authorisation. Without a stable identifier travelling the full length of the chain, disputes between two banks that have never spoken would be unresolvable.

The critical structural point is that DE2 is the routing key. The acquirer's switch reads the leading digits of the primary account number, matches them against a table of issuer identification numbers, and determines which scheme owns the range and therefore where to send the message. Card numbering is governed by ISO/IEC 7812, which since its 2017 revision provides for eight-digit issuer identification numbers within a primary account number of up to nineteen digits, with a Luhn check digit in the final position. Visa ranges begin with 4; Mastercard uses 51 to 55 and the 2221 to 2720 range added in 2017; American Express uses 34 and 37 with a fifteen-digit account number. The acquirer therefore knows which scheme to route to and, from the fuller BIN table, which issuer and which product. What it does not have is any means of contacting that issuer directly.

Two processing models coexist. In the *dual-message* model, typical of credit, an 0100 authorisation request obtains a decision and a hold, and a separate clearing record submitted later, in a batch, actually presents the transaction for settlement. In the *single-message* model, typical of domestic debit and PIN-based networks, an 0200 financial request both authorises and clears in one exchange. The distinction matters commercially: in dual-message the authorised amount and the cleared amount can differ, which is what makes tipping, fuel pre-authorisation and partial shipment possible, and it is also what creates the reconciliation problem that DE37 exists to solve.

Both schemes are engaged in a long migration towards ISO 20022 message structures for card traffic, but ISO 8583 remains the working format in the overwhelming majority of acquiring estates, and any acquirer building today will still be writing bitmap parsers.

Clearing and settlement

Authorisation creates an obligation. Clearing records it. Settlement discharges it.

After authorisation, the acquirer submits presentments into the scheme's clearing system, historically Visa's BASE II and Mastercard's Global Clearing Management System with its Integrated Product Message format. The scheme edits each record against the rulebook, applies the correct interchange rate based on the transaction's characteristics, and produces reconciliation and settlement reports for each member. Interchange is not invoiced; it is applied as an adjustment inside the clearing figures, so the acquirer's settlement position is already net of the interchange it owes.

Settlement itself is a net obligation between each member and the scheme's settlement service, denominated in a settlement currency and discharged through nominated settlement banks over ordinary interbank payment infrastructure. The scheme is not a bank and does not hold customer funds in the general case. It computes positions and instructs. This is why an outage at a card scheme prevents new transactions from being authorised but does not, by itself, destroy value that has already cleared.

The fee stack, exactly

Three distinct charges sit between the price on the shelf and the money in the merchant's account, and confusing them is the most common analytical error in the industry.

Interchange is defined at Article 2(10) of the Regulation as "a fee paid for each transaction directly or indirectly (i.e. through a third party) between the issuer and the acquirer involved in a card-based payment transaction", and the definition explicitly adds that "the net compensation or other agreed remuneration is considered to be part of the interchange fee". It flows from acquirer to issuer.

Scheme fees flow from both licensees to the scheme. They are contractual, confidential, and revised on a published bulletin cycle.

Merchant service charge is defined at Article 2(12) as "a fee paid by the payee to the acquirer in relation to card-based payment transactions". It is the merchant's total cost and it contains the other two.

Interchange in the United States is unregulated for credit and published by the schemes. The Visa USA Interchange Reimbursement Fee schedule effective 18 April 2026 gives, among many hundreds of rates, the following.

Programme	Rate
CPS/Retail, Traditional Rewards credit	1.51% + $0.10
CPS/Card Not Present, Traditional Rewards credit	1.89% + $0.10
Visa Traditional, non-qualified consumer credit	3.15% + $0.10
CPS/Retail Debit, exempt issuer	0.80% + $0.15
CPS/Retail Debit, regulated issuer	0.05% + $0.21

The last two lines are the entire American debit interchange debate in miniature. Regulation II, at 12 CFR Part 235, caps the interchange an issuer may receive on an electronic debit transaction at the sum of 21 cents and 5 basis points of transaction value, with a further 1 cent fraud-prevention adjustment available to issuers that implement qualifying fraud policies. Section 235.5 exempts issuers that, together with affiliates, hold less than $10 billion in assets, which is why the schedule carries both an "exempt" and a "regulated" line for the same product. Section 235.7 separately prohibits network exclusivity: an issuer must enable at least two unaffiliated payment card networks on every debit card and may not restrict the merchant's routing choice. That single provision is what gives American merchants a genuine routing decision and what makes debit economics in the United States structurally different from credit. The Federal Reserve proposed in October 2023 to reduce the base component to 14.4 cents and the ad valorem component to 0.04 per cent; as of the end of 2025 the proposal had not been finalised, and banking trade associations were formally asking for its withdrawal.

In the European Economic Area and the United Kingdom the position is the opposite: interchange is capped by statute rather than published by the scheme. Article 3 caps consumer debit card interchange at 0.2 per cent of transaction value, with member state discretion for domestic debit including a per-transaction maximum of five euro cents combined with a rate not exceeding 0.2 per cent. Article 4

caps consumer credit card interchange at 0.3 per cent. Chapter II does not apply to commercial cards, to cash withdrawals, or to cards issued by three-party schemes, per Article 1(3).

Article 5 is the anti-avoidance provision and the one that catches designs that look clever on a whiteboard: any agreed remuneration with equivalent object or effect, including net compensation, is treated as part of the interchange fee. Article 2(11) defines net compensation as "the total net amount of payments, rebates or incentives received by an issuer from the payment card scheme, the acquirer or any other intermediary in relation to card-based payment transactions or related activities". An issuer incentive routed through the scheme rather than the acquirer is still interchange for the purposes of the cap.

The remaining business rules complete the picture. Article 6 prohibits territorial licensing restrictions, which is what makes cross-border acquiring within the Union lawful and is the legal basis for a merchant in one member state contracting with an acquirer in another. Article 8 protects co-badging and bars automatic mechanisms that force brand selection, while permitting the merchant to set a priority. Article 10 restricts the Honour All Cards Rule so that accepting one issuer's cards does not compel acceptance of all, save within the same brand and category of regulated consumer card. Article 11 preserves the merchant's right to steer. Article 12 requires the acquirer to give the payee itemised information showing the merchant service charge and the interchange separately, which is the legal foundation of what the market calls interchange-plus-plus, or IC++, pricing.

The practical consequence of Article 12 is a genuine change in merchant negotiating power. Under blended pricing, a merchant is quoted one rate for everything and cannot see whether a rise in its costs came from the scheme, the issuer mix, or its acquirer's margin. Under IC++, the invoice separates pass-through interchange, pass-through scheme fees, and the acquirer's own markup, and only the third number is negotiable. Larger merchants insist on IC++ for exactly this reason, and smaller merchants are usually on blended rates because they lack the volume to make the analysis worth anyone's time.

Cross-border interchange within Europe has been the live regulatory fight of the decade. Following the United Kingdom's withdrawal from the European Union, Mastercard and Visa raised interchange on UK-EEA card-not-present consumer transactions from 0.2 and 0.3 per cent to 1.15 and 1.5 per cent respectively, because the statutory caps applied to intra-EEA transactions that these had ceased to be. The Payment Systems Regulator's market review found that the two schemes were not subject to effective competitive constraint, that it could identify no documented justification for the increases, and that the cost to United Kingdom businesses ran to roughly £150 million to £200 million a year. The regulator consulted on a two-stage price cap, then in October 2025 abandoned the interim cap in the face of litigation about its powers and instead consulted on the methodology for a longer-term cap. Separately, the European Commission accepted binding commitments from both schemes in 2019 to cut inter-regional interchange for transactions in the EEA on cards issued elsewhere, reducing them by around 40 per cent on average.

Three-party schemes and what the difference buys

In a three-party scheme the issuing and acquiring functions sit inside one balance sheet. There is no interchange, because interchange is by definition a payment between two institutions and there is only one. The scheme sets the merchant discount rate unilaterally and keeps all of it.

American Express describes the resulting information position precisely in its own filings: "Wherever we manage both the card-issuing activities of the business and the acquiring relationship with merchants, there is a 'closed loop' in that we have direct access to information at both ends of the card transaction." It uses that position to underwrite risk, reduce fraud and target marketing, and it

describes its business as spend-centric, focused on driving spending on its cards rather than on lending balances.

Commercially, four consequences follow.

First, revenue concentration. A four-party scheme captures a thin switching margin, on the order of tens of basis points, while the fat part of the merchant's cost goes to thousands of issuers as interchange. A three-party scheme captures the whole discount rate. The Supreme Court, in *Ohio v. American Express Co.* in 2018, noted that Amex derives the bulk of its revenue from merchant fees, in contrast to networks whose issuers earn from lending.

Second, price setting. In a four-party model, the acquiring market is competitive and the acquirer's markup is squeezed by rivals, but the interchange floor is set collectively or by regulation and no individual merchant can negotiate it. In a three-party model there is no floor and no competitive acquiring layer at all: the merchant negotiates with the scheme itself, and large merchants therefore obtain real concessions while small ones do not.

Third, acceptance. A higher single price bought Amex a narrower merchant footprint for decades, which in turn constrained cardholder value. This is the classic two-sided platform bind, and it was the analytical heart of *Ohio v. American Express*. The Court, dividing five to four, held that Amex's anti-steering provisions, which prohibit merchants from dissuading customers from using Amex cards or promoting other cards more heavily, did not violate the Sherman Act, because the relevant market had to be defined to include both sides of the platform simultaneously. The opinion recorded 2013 shares of United States credit and charge card purchase volume of 45 per cent for Visa, 26.4 per cent for Amex, 23.3 per cent for Mastercard and 5.3 per cent for Discover.

Fourth, and most importantly for anyone designing a scheme today, the boundary is porous and crossing it has a regulatory price. Amex operates Global Network Services, under which partner institutions issue Amex-branded cards and in some markets also acquire, under independent operator arrangements, network card licence arrangements and joint ventures, with issuer rates agreed bilaterally with each partner. It operates OptBlue, under which third-party acquirers contract directly with small merchants for Amex acceptance and set the merchant's Amex rate themselves, exactly as they set the rate for the other brands. Amex's own merchant materials put the OptBlue eligibility ceiling at an estimated annual charge volume of less than $3 million, with certain categories such as charity, education, government, healthcare, insurance, residential rent and utilities exempt from the volume test.

Both of those programmes are, functionally, four-party arrangements wearing a three-party brand, and Article 1(5) of the Interchange Fee Regulation says so in terms. A scheme that licenses other payment service providers to issue or acquire, or issues through a co-branding partner or agent, is treated as a four-party scheme, which pulls it inside the interchange caps it would otherwise escape under Article 1(3)(c). Article 1(4) exempts three-party schemes from Article 7's scheme-processing separation requirement, so the classification carries structural obligations as well as pricing ones. The choice between three-party and four-party is therefore not a description of what a company is; it is a decision about how much distribution to buy and how much regulatory scope to accept in exchange.

The historical direction of travel is worth noting. Visa and Mastercard began as bank co-operatives: BankAmericard, launched in 1958 and licensed to other banks from 1966, became National BankAmericard Incorporated in 1970 and Visa in 1976; the Interbank Card Association, formed in 1966, became Master Charge and then MasterCard. Both were owned by their member banks, which is why the word "association" persists in older documents, and both demutualised in the 2000s, Mastercard

in 2006 and Visa in 2008, becoming ordinary listed companies whose customers are the banks that used to own them. The four-party model is a co-operative artefact that outlived the co-operative.

Distribution: acquirers, ISOs, payment facilitators and marketplaces

An acquirer licence is expensive to hold and slow to obtain, and acquiring banks are, as a class, poor at selling to small businesses. The industry therefore developed intermediaries, and the distinctions between them are precisely the distinctions of contract and liability set out earlier.

An *independent sales organisation*, or ISO, in Mastercard's vocabulary a member service provider, is a sales and servicing agent. It is registered with the scheme by a sponsoring acquirer, it may brand terminals and statements, and it earns residuals on the portfolio it introduces. What it does not do is hold the merchant contract or touch settlement funds. The merchant contracts with the acquirer. The ISO's role in law is agency, and the acquirer remains liable for its agent's conduct.

A *payment facilitator*, Visa's term, with Mastercard using the same concept and calling the merchant beneath it a sub-merchant, is a different animal entirely. The facilitator signs its own merchant agreement with the acquirer and then signs *sub-merchants* under that agreement, becoming, in Visa's vocabulary, the sponsor of *sponsored merchants*. It is the merchant of record. It receives settlement from the acquirer and disburses to its sub-merchants. It performs onboarding and underwriting under the scheme's standards, and it carries the liability for its sub-merchants' chargebacks and fraud, with the acquirer standing behind it. This is what allows an online platform to onboard a self-employed tradesman in ninety seconds when a bank's own onboarding would take three weeks.

Both schemes limit how far this can go by transaction volume. Mastercard raised its sub-merchant threshold from $100,000 to $1,000,000 in annual volume in 2014, and Visa followed with the same $1,000,000 threshold for sponsored merchants in the United States and Canada from 31 August 2017, granting merchants that crossed it a two-year window to complete a direct agreement with the acquiring bank. Above that level, the scheme's position is that the merchant is too large to be an anonymous line item inside somebody else's portfolio and must have a direct relationship with a licensed acquirer. The exact drafting, categories and effective dates are set out in the current editions of the Visa Core Rules and Visa Product and Service Rules and the Mastercard Rules, and any acquirer relying on them should read the edition in force rather than a summary, including this one: the schemes revise these provisions on a semi-annual bulletin cycle.

Marketplaces are a further variation, where the platform aggregates supply from many sellers but the commercial and legal position varies by jurisdiction and by scheme rule. The two questions that determine everything are who is the merchant of record and who bears the chargeback. A platform that answers "us" to both is operating as a facilitator whether or not it uses the word.

Finally, the acquirer of record is often not the entity doing the work. Much of the world's acquiring processing is done by specialist processors under contract to licensed acquiring banks, which is why a merchant may hold a contract with a bank it never speaks to, receive statements from a processor, be sold to by an ISO, and be onboarded by a facilitator, all inside what the rulebook regards as a single acquiring relationship.

Why the merchant never talks to your bank

Pull the threads together and the answer has four independent parts, any one of which would be sufficient on its own.

There is no contract. Privity runs merchant-to-acquirer and cardholder-to-issuer, and the only bridge is the pair of scheme licences. A merchant asserting a claim against an issuer has no cause of action

to assert it under; it must go through its acquirer, which raises the matter under the scheme's dispute rules, which is why chargeback representment is a process between two banks in which the merchant is a supplier of evidence rather than a party.

There is no settlement path. The merchant has a bank account at its acquirer or nominated for funding by its acquirer. It has no account at your bank and no means of receiving funds from it. Money reaches the merchant because the acquirer pays it, out of the acquirer's own funds, in advance of the acquirer being made whole through scheme settlement.

There is no addressability, and no trustworthy identity. The merchant knows the primary account number, from which it can derive the issuer identification number and therefore, via a BIN table, the issuing institution's name. That is a lookup, not a channel. There is no authenticated path from a terminal in a shop to an issuer's authorisation host that does not run through an acquirer's licensed connection and the scheme's switch, and there could not be: the whole security model depends on both ends being licensed, audited entities with cryptographic keys exchanged under scheme control.

And there is deliberate data minimisation. Under PCI DSS, the merchant is obliged to minimise what it stores, and network tokenisation is steadily removing the real account number from the merchant environment altogether, replacing it with a token that only the scheme's token service can map back to the underlying credential. The direction of travel is for the merchant to know *less* about the cardholder's bank over time, not more.

The nearest the two sides ever come is 3-D Secure, where an issuer's access control server briefly renders a challenge inside the merchant's checkout flow. Even that is mediated: the merchant's server talks to a 3DS server, which talks to the scheme's directory server, which routes to the issuer's access control server. The issuer and the merchant appear on the same screen and still never exchange a message directly.

What it costs, and who is actually paying

The final thing worth saying plainly is that the four-party model is a transfer mechanism as much as a payment mechanism, and every participant's economics depend on someone else's.

The merchant pays the merchant service charge, and prices it into the goods. The issuer receives interchange and spends most of it on rewards, fraud losses and the cost of running the credential. The scheme receives basis points from both banks. The acquirer receives a margin for taking credit risk it is often not thanked for taking. The cardholder pays nothing at the point of sale in most markets and receives a cashback or points benefit funded, ultimately, out of the merchant's margin and therefore out of the retail price, including the retail price paid by customers using cash.

That last transfer is the reason interchange is regulated in the European Union, the United Kingdom, Australia and a growing list of other jurisdictions, and the reason it remains largely unregulated for credit in the United States. It is a genuine policy disagreement about whether the price that balances a two-sided market should be set by the platform or by the state, and it is not going to be settled by an appeal to how the plumbing works. The plumbing works the same way either way. Only the numbers change.

23.98 Common wrong ideas

Wrong: interchange is what the card scheme charges. **Right:** interchange is paid by the acquirer to the issuer; the scheme sets the default rate, receives none of it, and earns its own money from separate and confidential scheme fees billed to both licensees.

Wrong: the scheme is a neutral switchboard. **Right:** it writes a binding rulebook, sets default interchange, arbitrates disputes between its own members, sells processing and fraud services in competition with its members' vendors, and levies non-compliance assessments.

Wrong: a four-party transaction involves four participants. **Right:** "four-party" is a legal and commercial classification of who holds the licences and bears the liability; a real payment also touches a gateway, processors on both sides, often a facilitator, a token service provider and 3-D Secure components.

Wrong: American Express is a three-party scheme. **Right:** through Global Network Services and OptBlue it licenses others to issue and to acquire, and Article 1(5) of the Interchange Fee Regulation therefore treats it as a four-party scheme for the purposes of the rules.

Wrong: the card scheme moves the money. **Right:** it calculates net positions and instructs; the transfer between banks is executed outside the card network through settlement banks and ordinary interbank payment systems.

Wrong: an approval means the merchant has been paid. **Right:** an approval is a promise and a hold on the cardholder's balance; settlement happens later and in bulk, and a chargeback can travel back down the same chain of contracts and take the money off the merchant again.

Wrong: a payment facilitator is just a reseller with better software. **Right:** it signs its own merchant agreement, becomes the merchant of record for its sub-merchants, receives settlement and disburses it, and carries their chargeback and fraud liability with the acquirer standing behind it.

Wrong: the acquirer is selling the merchant a service. **Right:** it is taking a credit exposure, funding the merchant on a T+1 basis before it has itself been settled, which is why merchant underwriting looks like lending and why travel and ticketing attract rolling reserves.

Wrong: 3-D Secure lets the merchant and the issuer talk to each other. **Right:** the merchant's server talks to a 3DS server, which talks to the scheme's directory server, which routes to the issuer's access control server; the two appear on the same screen and never exchange a message directly.

Wrong: the schemes have always been the listed companies they are today. **Right:** Visa and Mastercard began as bank co-operatives owned by their members and demutualised in 2008 and 2006, and the four-party model is a co-operative artefact that outlived the co-operative.

23.99 Chapter summary in 20 lines

1. A card payment connects a cardholder, a merchant, the merchant's acquirer and the cardholder's issuer, with a scheme in the middle that is not a party to the purchase at all.
2. The scheme exists so that every bank signs one rulebook and every merchant signs one contract, which turns a multiplication of relationships into an addition.
3. On a fifty pound British consumer credit card sale the shop receives about £49.65, and the missing thirty-five pence splits into interchange, scheme fees and the acquirer's own margin.
4. Interchange flows from acquirer to issuer, is capped at 0.3 per cent for consumer credit and 0.2 per cent for consumer debit in the United Kingdom and the European Union, and reaches the scheme not at all.
5. Scheme fees are a separate, confidential charge levied on both licensees, and they are a far thinner slice than most people assume.
6. The merchant service charge is the merchant's total cost and contains the other two, which is why the itemisation duty in Article 12 is the legal foundation of IC++ pricing.

7. The switchboard analogy fails because the scheme sets prices, writes binding rules, arbitrates disputes and fines its members, which is why Article 7 forces the scheme business to be separated from the processing business.
8. Privity runs cardholder-to-issuer and merchant-to-acquirer, bridged only by the two scheme licences, and almost every dispute in card payments is resolved by asking who has privity with whom.
9. The scheme's rules bind a merchant in Leeds only because the acquirer is obliged to flow them down into the merchant agreement.
10. The issuer underwrites and funds one side, makes the authorisation decision inside a scheme-mandated response window, and is the party that answers to the cardholder in law.
11. The acquirer underwrites and funds the other side, taking a credit exposure it is rarely thanked for, which is why reserves, delayed settlement and merchant category codes matter as much as they do.
12. Authorisation is carried in ISO 8583, whose four-digit message type indicator encodes version, class, function and origin, and whose bitmap makes absent fields free.
13. DE 2 is the routing key, DE 11 matches a response to its request across one link, and DE 37 is the handle by which the same transaction is found months later in a chargeback.
14. Dual-message and single-message models coexist, and the gap between authorised and cleared amounts in the dual-message model is what makes tipping, fuel pre-authorisation and partial shipment possible.
15. Authorisation creates an obligation, clearing records it, and settlement discharges it through nominated settlement banks over infrastructure the scheme does not own.
16. In the United States credit interchange is unregulated and published by the schemes, while Regulation II caps debit interchange and bans network exclusivity, which is what gives American merchants a genuine routing decision.
17. In Europe and the United Kingdom interchange is capped by statute instead, with Article 5 catching any equivalent remuneration, including issuer incentives routed through the scheme.
18. A three-party scheme captures the entire discount rate because there is no second institution to pay interchange to, and it historically paid for that with a narrower merchant footprint.
19. The boundary between three and four party is porous: licensing others to issue or acquire reclassifies a scheme, making the choice a decision about how much distribution to buy and how much regulatory scope to accept in exchange.
20. The four-party model is a transfer mechanism as much as a payment mechanism, funding cardholder rewards out of the merchant's margin and therefore out of the retail price, including the price paid by customers using cash.

Sources: Regulation (EU) 2015/751 (Interchange Fee Regulation) via EUR-Lex; 12 CFR Part 235 (Regulation II) via eCFR and the Federal Reserve; Visa USA Interchange Reimbursement Fees schedule effective 18 April 2026; Visa Core Rules and Visa Product and Service Rules; Visa Inc. Fiscal 2025 Annual Report; Mastercard Incorporated fourth quarter and full year 2025 earnings release; American Express Company Form 10-K and OptBlue merchant materials; Ohio v. American Express Co., 585 U.S. 529 (2018); UK Payment Systems Regulator market review MR22/2 into cross-border interchange fees; European Commission antitrust commitments decision IP/19/2311; ISO 8583 and ISO/IEC 7812 message and numbering standards.

24

CHAPTER

ISO 8583

24.0 What this chapter gives you

1. You will be able to explain what the bitmap does, and read the shape of a transaction off eight bytes of hexadecimal before parsing a single value.
2. You will be able to decode a four-digit message type indicator into version, class, function and origin, and say what `0100`, `0420`, `0800` and `1240` each mean.
3. You will be able to explain why the twelve digits `000000004250` mean nothing until you have read DE 49, and why the same digits are four thousand two hundred and fifty yen in Tokyo.
4. You will be able to say why "ISO 8583 compliant" on a vendor's data sheet is worth almost nothing, and what three questions to ask instead.
5. You will be able to explain why the 2023 edition reused version digit 2, and why that is a standards body formally conceding that the wire format is frozen.
6. You will be able to distinguish a request from an advice from a repeat, and say which of them may be refused and which may only be acknowledged.
7. You will be able to explain why the industry stayed on the 1987 edition even though the 1993 response codes are better in every respect.
8. You will be able to explain why a malformed bitmap produces a £4,250,000 purchase rather than a parse error at the malformed byte.
9. You will be able to say what DE 90 does, and why a reversal needs five separate sub-elements to name the message it wants undone.
10. You will be able to explain why what leaves a terminal is not an ISO 8583 message, and name the layers wrapped around it.

There is a sentence that gets repeated at payments conferences until it stops meaning anything: *ISO 8583 is the language of card payments*. It is not a language. It is closer to a filing convention — an agreement about which numbered box holds the amount, which holds the card number, and how to tell the person at the other end which boxes you bothered to fill in.

That is a smaller idea than "language", and it is the reason the format has survived. Languages evolve and fragment. Filing conventions calcify, and calcification is what you want in a system where a terminal made in 2003 has to be understood by a mainframe replaced in 2019, which has to be understood by a card issuer in a country neither has heard of. The format was first published in 1987, and the overwhelming majority of card authorisations on earth are still carried in messages whose first digit says *1987*. This chapter explains what is in one of those messages, digit by digit and field by field, and why the phrase "ISO 8583 compliant" on a vendor's data sheet tells you almost nothing you needed to know.

The plain version

Imagine you and a friend need to send each other messages by telegram, and the telegram office charges you by the letter. You will be sending the same *kind* of message hundreds of times a day: how much, whose account, which shop, what time. Writing out "Amount: forty-two pounds fifty" every time would bankrupt you.

So you agree on a form. Not a form you send — a form you both memorise. Box 2 is always the card number. Box 3 is always what kind of transaction this is. Box 4 is always the amount. Box 11 is always a serial number so you can tell one message from another. Box 39 is always the answer: yes or no.

Now you never have to write a label again. You only write the contents.

But there is a problem. Most messages do not fill in every box. A shop till sending a payment does not need Box 36, which holds data from a bank card's third magnetic track that nobody has used since the 1980s. If you leave boxes out, how does your friend know that the "000000004250" they are reading is the amount and not something else?

So you add one more thing at the very front of the message: a checklist. Sixty-four ticks in a row, one for each box, in order. A tick means "this box is in the message". A blank means "skip it". Your friend reads the checklist first, learns that boxes 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 37, 41, 42, 43 and 49 are coming, and then reads the contents of exactly those boxes, in exactly that order, and knows what each one is without a single label being sent.

That checklist is called a *bitmap*, and it is the whole trick. Everything else in ISO 8583 is bookkeeping around it.

A worked example

You buy a book. £42.50, at a bookshop in London, on 17 March 2026, at 32 minutes and 5 seconds past two in the afternoon. You tap your card. The terminal fills in the form:

Box	What it holds	What it says
2	Card number	4111111111111881
3	What kind of transaction	000000 (a purchase)
4	Amount	000000004250
7	Date and time sent	0317143205
11	Serial number	004417
12	Local time	143205
13	Local date	0317
14	Card expiry	2803
18	What kind of shop	5942 (a bookshop)
22	How the card was read	071 (tapped)
25	Circumstances	00 (nothing unusual)
32	Which bank is asking	454311
37	Reference number	607614430041
41	Which till	TERM0042
42	Which merchant	KEDBYTEBOOKS01
43	Merchant name and town	KEDBYTE BOOKS LTD LONDON GB
49	Currency	826 (pounds sterling)

Two things in that table are worth stopping on. The amount has no decimal point and no pound sign. It is twelve digits: `000000004250`. The decimal point is not sent, because Box 49 already said "826",

which means pounds, and everybody has agreed that pounds have two digits after the point. Four thousand two hundred and fifty pennies. Had the currency been Japanese yen, the same digits would have meant four thousand two hundred and fifty yen, and there would have been no decimal point to imagine. And the card expiry is `2803`, not `March 2028`: year first, then month, so the digits sort in date order if you line them up.

Somewhere between one and two seconds later, a message comes back the other way with almost nothing in it. The serial number, so the till knows which purchase this is about. A six-character code the bank made up, `A4T29B`, so that if there is ever an argument the two sides have something to point at. And Box 39, which is two characters long:

```
00
```

That is it. Two characters. `00` means yes. Anything else means some flavour of no, and the till has a printed list telling it what to say to the customer for each one. The receipt prints. You take your book.

Why it is not written out in words

Suppose you wrote that same purchase out with labels, the way a modern computer system usually would, so that every value has its name written next to it. It comes to 639 characters. Trim the labels down to bare numbers and it still comes to 289. Written in the numbered-box form with the checklist at the front, it is 181 characters, and if you use the compact way of writing digits — two squeezed into the space of one letter — it is 129. In 1987, a shop's card terminal talked to its bank down a telephone line at 1,200 bits per second, which is about 120 characters every second. So:

Form	Size	Time to send at 1,200 bits per second
Written out with full labels	639 characters	5.3 seconds
Labels shortened to numbers	289 characters	2.4 seconds
Numbered boxes, plain digits	181 characters	1.5 seconds
Numbered boxes, packed digits	129 characters	1.1 seconds

And that is one direction. Double it for the reply. The difference between the first row and the last row is the difference between a customer waiting eleven seconds at the till and a customer waiting two. The format was not designed to be elegant. It was designed to fit down a copper wire while somebody stood there holding their shopping.

Four kinds of message

There are only four things any of these messages can be. A **question** asks: may I take £42.50 from this card? An **answer** comes back the other way carrying the yes or the no. A **notice** says: I have already done something, and I am telling you about it — a shop whose terminal lost its phone line and took the payment on paper sends a notice, not a question, and the bank cannot refuse something that has already happened. A **cancellation** says: forget the last one. If the till asked for £42.50 and the answer got lost on the way back, the till does not know whether the money was set aside, so it sends a cancellation quoting the exact details of the message it wants undone, and keeps sending it until somebody confirms.

Question, answer, notice, cancellation. Everything else — sign-on messages, key exchanges, the nightly totals two banks send each other to check they agree — is one of those four wearing a different hat.

Where the plain version stops being true

It is not one form. It is an agreement to number the boxes. This is the most important correction in the chapter and the one the industry is loosest about. ISO 8583 specifies the shape of the envelope: a message type at the front, one or two checklists, then the contents in checklist order. It does not, in the current edition, even contain the list of what the boxes mean. In the third edition, published in July 2023, the message definitions, field definitions and code values were deliberately moved *out* of the standard and into a separately maintained register, so that they can be changed without republishing an International Standard. Two systems can both be entirely faithful to ISO 8583 and be unable to exchange a single message, because faithfulness to ISO 8583 does not settle what goes in box 48, or box 62, or what response code `05` means to you as opposed to me.

"Binary format" is only about a tenth true. The checklist genuinely is binary — eight raw bytes whose individual bits carry meaning. But the rest of the message is usually characters: digits and ordinary letters, written in ASCII or, on older mainframes, in EBCDIC. Some implementations pack numeric fields two digits to a byte and some do not, and the standard permits both, so the same logical message can be 129 bytes on one link and 181 on another. Sizes quoted for ISO 8583 messages are always sizes for a *particular implementation*, never for the format.

The box numbers do not mean the same thing everywhere. In the 1987 edition, data element 24 holds a network routing identifier; in the 1993 edition the same number holds a function code doing an entirely different job. Response codes are two characters long in 1987 and three in 1993. And the codes saying which kind of account to take the money from differ between published specifications that both describe themselves as ISO 8583: one major scheme's documentation uses `10` for savings and `20` for a current account, while a major acquirer's published message specification uses `01` and `02` for the same two things. The box number is agreed. The contents are negotiated.

The standard is silent about everything that makes it work. ISO 8583 says nothing about how a message gets from the terminal to the bank, how you know where one message ends and the next begins, how you prove it was not tampered with, or how the money actually moves afterwards. All of that is bolted on outside: routing headers borrowed from an older transport standard, a length prefix in front of every message on a TCP connection, a cryptographic checksum in the last field, and a separate settlement process hours or days later. Reading ISO 8583 and expecting to understand a card network is like reading the specification for a shipping container and expecting to understand global trade. It is also why `00` does not mean the money moved: it means the issuer agreed that it *would*, and the instruction that shifts funds between banks is a different message sent later, often the following day.

The technical version

What ISO 8583 is, precisely

ISO 8583 is an International Standard maintained by ISO Technical Committee 68 (*Financial services*), Subcommittee 9 (*Information exchange for financial services*). Its title is *Financial-transaction-card-originated messages — Interchange message specifications*. Its scope is one sentence: it "specifies a common interface by which financial-transaction-card-originated messages can be interchanged between acquirers and card issuers", and it "specifies message structure and format, including normalized data types". It then says explicitly that "the method by which messages are transported or settlement takes place is not within the scope of this document".

The editions are:

Edition	Designation	Published	Status as of writing
1	ISO 8583:1987	1987	Withdrawn
2	ISO 8583:1993	December 1993	Withdrawn 17 June 2003
—	ISO 8583-2:1998	1998	Withdrawn
—	ISO 8583-1:2003	June 2003	Withdrawn 18 July 2023
—	ISO 8583-3:2003	2003	Withdrawn
3	ISO 8583:2023	July 2023	Current

The third edition cancels and replaces all three parts published between 1998 and 2003 and reunifies them into a single document, and it does so by making that document much smaller: the messages, data elements and code values that used to occupy the bulk of ISO 8583-1:2003 are now published separately at `standards.iso.org/iso/8583` and maintained by an ISO 8583 maintenance agency, which as of writing is the Association française de normalisation (AFNOR). The foreword records the reason: the document "has been restructured to facilitate maintenance of the messages, data elements and code values by a new ISO 8583 maintenance agency".

The 2023 edition also carries a note that tells you what four decades did to the subject matter. It observes that "a range of tokens now exist for identifying account relationships", and warns that "the actual token numeric issued by a financial institution can be different from the associated card numeric". A standard about cards has quietly conceded that the thing in the message is frequently not a card. Its only normative references are ISO/IEC 7812-1 and ISO/IEC 7816-6.

Message structure

Clause 4.1.1 of the 2023 edition specifies that every message shall be constructed in exactly this sequence:

1. the message type;
2. one or two message bitmaps;
3. a series of data elements, in the order of the bitmap representation.

There is no framing, no terminator and no length. Those are the transport's problem, and every deployment solves them differently.

The message type identifier, digit by digit

The message type is four digits. The first is a version number; the remaining three are the message type identifier proper, which the standard describes as "a three-digit numeric field identifying the message class, message function and transaction originator".

Position 1 — version. The standard's Table 1 assigns:

Digit	Standard	Year
0	ISO 8583	1987
1	ISO 8583	1993
2	ISO 8583-1	2003
2	ISO 8583	2023
3 to 7	Reserved for ISO use	
8	Reserved for national use	
9	Reserved for private use	

The repetition of 2 is not an error, and it is the most quietly devastating line in the standard. The 2023 footnote explains that "code no. 2 is reused for this edition since no changes have been incorporated into this document that would necessitate a new version to properly construct or parse a message for processing". A new version digit is only assigned, per clause 4.1.2.2, "when sufficient changes have been made in a revision of this document such that it is necessary to know which version was used to construct a message in order to properly process the message". Thirty-six years on, ISO published a new edition and did not need a new digit: a standards body formally acknowledging that the wire format is frozen.

Position 2 — message class. What kind of business this message is about.

Digit	Class
1	Authorisation
2	Financial
3	File actions
4	Reversal and chargeback
5	Reconciliation
6	Administrative
7	Fee collection
8	Network management

The distinction between class 1 and class 2 is the dual-message versus single-message divide, and it decides the entire downstream architecture. An authorisation message asks whether funds are available and obtains an approval but posts nothing to the account; a separate clearing exchange posts it later. A financial message asks and posts in one go. Credit cards overwhelmingly run dual-message; ATM withdrawals and much domestic debit run single-message.

Position 3 — message function. What this message does within its class.

Digit	Function
0	Request
1	Request response
2	Advice
3	Advice response
4	Notification
5	Notification acknowledgement
6	Instruction
7	Instruction acknowledgement

The standard's definitions are worth quoting because the industry uses these words loosely. A *request* is a "message in which the sender informs the receiver that a transaction is in progress", and a response is required to complete the activity. An *advice* is a "message in which the sender notifies the receiver of an action that has been taken, requiring no approval but requiring a response". The difference is not politeness: a request can be refused, an advice can only be acknowledged. Functions 6 and 7 arrived in the 2003 edition and are almost never seen on live scheme links.

Position 4 — transaction originator.

Digit	Origin
0	Acquirer
1	Acquirer repeat
2	Issuer
3	Issuer repeat
4	Other
5	Other repeat

The repeat values matter more than they look. Clause 4.2 requires that "whenever a repeat message is identified, that repeat message shall be identical to its original message, with the exception of the message type identifier and, if necessary, date and time transmission and the message authentication code data elements". A repeat is a byte-for-byte retransmission with a changed origin digit, and that is how the receiver knows it may be a duplicate rather than a second purchase.

Reading them together. The four digits compose:

MTI	Reading
0100	1987, authorisation, request, from acquirer
0110	1987, authorisation, request response, from issuer
0120	1987, authorisation, advice, from acquirer
0130	1987, authorisation, advice response, from issuer
0200	1987, financial, request, from acquirer
0210	1987, financial, request response, from issuer
0220	1987, financial, advice, from acquirer
0230	1987, financial, advice response, from issuer
0400	1987, reversal, request, from acquirer
0410	1987, reversal, request response, from issuer
0420	1987, reversal, advice, from acquirer
0430	1987, reversal, advice response, from issuer
0800	1987, network management, request, from acquirer
0810	1987, network management, request response, from issuer
1240	1993, financial, notification, from other
1442	1993, chargeback, notification, from issuer

Note the last two. `1240` is Mastercard's First Presentment message and `1442` is a chargeback: both are 1993-version messages, and both travel over the same scheme as the 1987-version `0100` that authorised the purchase in the first place. One transaction, two editions of the standard, routinely.

The bitmap

The second component is "one or two message bitmaps, each consisting of 64 bits. Each bit signifies the presence (1) or the absence (0) in the message of the data element associated with that particular bit."

The primary bitmap is always present and indexes data elements 1 to 64. Bit 1 does not index a data element; it signals the presence of a secondary bitmap covering elements 65 to 128. In the 1993 and later editions, data element 65 can in turn signal a tertiary bitmap covering 129 to 192, which is exceptionally rare in production.

Take the message from the worked example, which carries data elements 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 37, 41, 42, 43 and 49, and no secondary bitmap. Writing a 1 in each of those positions and a 0 everywhere else:

```
0111 0010 0011 1100 0100 0100 1000 0001
0000 1000 1110 0000 1000 0000 0000 0000
```

Grouped into nibbles and written in hexadecimal, that is:

```
723C448108E08000
```

Eight bytes. Seventeen fields declared. No field names, no delimiters, no terminators, no ambiguity about order. Going the other way, `7010001102C04804` expands to

```
0111 0000 0001 0000 0000 0000 0001 0001
0000 0010 1100 0000 0100 1000 0000 0100
```

which declares elements 2, 3, 4, 12, 28, 32, 39, 41, 42, 50, 53 and 62 — a response message, carrying a card number, processing code, amount, local time, fee, acquirer identifier, response code, terminal, merchant, settlement currency, security control information and one private field. You can read the *shape* of a transaction off eight bytes before parsing a single value, which is why network engineers debug card traffic from hex dumps.

Data element notation

Each data element is defined by a content type and a length rule.

Notation	Content
a	Alphabetic, including spaces
n	Numeric only
s	Special characters only
an, as, ns, ans	The obvious combinations
b	Binary
z	Track 2 and track 3 code set, per ISO/IEC 7813 and ISO/IEC 4909
x+n	A sign character, `C` for credit or `D` for debit, followed by digits
p	Pad character

Length is either fixed or variable. A fixed field is written with its length: `n 6` is exactly six digits. A variable field carries its own length in a prefix, written as leading dots: `n..19` is an LLVAR field, a two-digit length followed by up to nineteen digits; `ans...999` is an LLLVAR field, a three-digit length followed by up to 999 characters. Clause 4.4.2 onwards distinguishes *primitive* elements, which have no internal structure; *constructed* elements, holding a fixed number of sub-elements all of which must be present; and, from 2003, *composite* elements, which contain datasets indexed by a second-level dataset bitmap or encoded as tag-length-value under ISO/IEC 8825.

The elements that carry a purchase

DE	Format	Name
2	n..19	Primary account number
3	n 6	Processing code
4	n 12	Amount, transaction

DE	Format	Name
7	n 10	Transmission date and time
11	n 6	System trace audit number
12	n 6	Local transaction time
13	n 4	Local transaction date
14	n 4	Expiration date
18	n 4	Merchant type
22	n 3	Point of service entry mode
25	n 2	Point of service condition code
32	n..11	Acquiring institution identification code
35	z..37	Track 2 data
37	an 12	Retrieval reference number
38	an 6	Authorisation identification response
39	an 2	Response code
41	ans 8	Card acceptor terminal identification
42	ans 15	Card acceptor identification code
43	ans 40	Card acceptor name and location
49	a or n 3	Currency code, transaction
52	b 64	Personal identification number data
55	ans...999	Integrated circuit card data
64	b 64	Message authentication code
90	n 42	Original data elements

Several of these repay a closer look.

DE3, processing code, is six digits in three pairs: transaction type, the account the money comes from, the account it goes to. `000000` is a purchase from the default account, `010000` a cash withdrawal, `200000` a refund, `300000` a balance enquiry. The account-type pairs are where implementations diverge most visibly, and the divergence is public: Mastercard's Single Message System material uses `00` for no account specified, `10` for savings, `20` for checking and `30` for a credit card account, while Worldpay's published Lync ISO 8583 message specification builds cash withdrawal from savings as `010100` and cash advance from credit as `010300`. Both describe ISO 8583. Neither is wrong. They are not the same dialect.

DE4, amount, is twelve digits with no sign and no decimal separator. The implied decimal position comes from the currency in DE49, an ISO 4217 code: `826` for pounds sterling and `840` for US dollars, both with two minor digits; `392` for Japanese yen, with none; `414` for Kuwaiti dinar, with three. A system that stores amounts as decimal strings and infers the exponent from the currency will survive; one that stores them as floating-point will not.

DE7, DE12 and DE13 are three different clocks. DE7 is the transmission date and time in MMDDhhmmss, expressed in Coordinated Universal Time by the sender of that particular message, and it changes at every hop. DE12 is the local transaction time at the point of sale, hhmmss, and DE13 the local transaction date, MMDD. Neither carries a year in the 1987 edition, so a system reconstructing a full timestamp has to supply the year from somewhere else, and the last week of December is where that assumption goes to die.

DE11, the system trace audit number, is six digits assigned by the message originator. It is not globally unique and was never meant to be: it is unique within one acquirer, within one processing day, and it wraps at 999999. Its job is to let the two ends of one link match a response to a request.

DE37, the retrieval reference number, is twelve alphanumeric characters and is the identifier that survives longest into the dispute process. Its internal construction is not fixed by ISO; a common convention is a single-digit year, a three-digit day of year, a two-digit hour and a six-digit sequence, which is how the example `607614430041` was built: year 6, day 076 (17 March), hour 14, sequence 430041.

DE38, the authorisation identification response, is the six-character approval code the issuer generates and the receipt prints. **DE39** is the response code: two alphanumeric characters in the 1987 edition, three in 1993.

DE55 is where EMV lives, and it is a format inside a format. The 2023 standard is explicit that the composite dataset structure "does not apply to the integrated circuit card (ICC)-related data element, as the linking of related sub-elements shall be accomplished in accordance with the definitions given in ISO/IEC 7816-6". DE55 is therefore a bag of EMV tag-length-value objects — the application cryptogram in tag 9F26, the cryptogram information data in 9F27, the application transaction counter in 9F36, the terminal verification results in 95, the unpredictable number in 9F37 — parsed by entirely different rules from the message carrying them, and commonly capped by schemes far below the ISO maximum of 999 bytes.

DE52 carries the PIN, always enciphered, as a 64-bit block formatted according to ISO 9564-1. Format 0 blocks, which combine the PIN with digits of the account number, dominate the installed base; format 4 exists to allow AES. **DE64** and its secondary-bitmap twin DE128 carry a message authentication code computed with a shared key, so the receiver can detect tampering in transit.

DE90, original data elements, is forty-two digits and is a constructed field in the strict sense: five sub-elements, all mandatory, no length prefixes, laid out end to end.

Sub-element	Digits	Content
Original MTI	4	The message being reversed
Original DE11	6	Its system trace audit number
Original DE7	10	Its transmission date and time
Original DE32	11	Its acquiring institution code
Original DE33	11	Its forwarding institution code

Four plus six plus ten plus eleven plus eleven is forty-two. DE90 is how a reversal names its victim, and the fact that it needs five separate fields to do so tells you something true and slightly alarming about how transactions are identified in this format.

Why a bitmap rather than a text format

The engineering argument has four parts, and only the first is about size.

Size. The worked authorisation above, encoded with ASCII digits and a hexadecimal bitmap, is 181 bytes; with packed binary-coded decimal numerics and a raw binary bitmap it is 129. The same content as tagged text with descriptive names is 639 bytes, and with the tags reduced to bare field numbers it is 289. On a 1,200 bit-per-second dial-up link with start and stop bits — ten bits per character — that is the difference between roughly eleven seconds of round trip and roughly two. On a 9,600 bit-per-second X.25 leased line shared between hundreds of terminals, it is the difference between a switch that copes at the Saturday peak and one that does not.

Determinism. A bitmap-indexed message can be parsed with a fixed loop, a lookup table and no dynamic memory allocation whatsoever. There is no tokeniser, no nesting, no escaping, no quoting

and no ambiguity about whether a value is a string or a number, because the field definition already said. A terminal with sixteen kilobytes of RAM could parse this in 1987, and a switch handling tens of thousands of messages a second can parse it now without a garbage collector ever running.

Bounded worst case. Every field has a maximum length declared in the specification, so the largest possible message is computable in advance and buffers can be statically sized. Text formats have no such property.

Honesty about what it costs. All four of those advantages come from positional encoding, and positional encoding has one enormous liability: the message is not self-describing. A malformed bitmap does not produce a parse error at the malformed byte; it produces a message in which every subsequent field is read at the wrong offset with the wrong length, and the failure surfaces somewhere else entirely, often as a nonsense amount. Anyone who has debugged card traffic has seen a £42.50 purchase arrive as £4,250,000 because one field upstream was written one digit short.

The three versions, and why 1987 will not die

1987, the first edition, defined 128 data elements, a two-character response code in DE39, and DE24 as a network international identifier used for routing.

1993 kept the message structure identical and changed the vocabulary. The most consequential change was to response codes, which became three digits arranged so that the leading digit tells you which class of message the code belongs to:

Range	Meaning
000–099	Approved
100–199	Denied, card need not be retained
200–299	Denied, card to be retained
300–399	File action result
400–499	Reversal or chargeback result
500–599	Reconciliation result
600–699	Administrative result
700–799	Fee collection result
800–899	Network management result
900–999	Could not be processed

So `000` is approved, `100` is do not honour, `101` is expired card, `116` is insufficient funds, `200` is do not honour and retain the card, `911` is issuer timed out. Within each hundred, the standard reserves blocks for ISO, national and private use, so a scheme can invent codes without colliding with the standard. It is a better design than the 1987 flat two-digit list in every respect. DE24 was redefined as a function code, and the data element count rose to a possible 192 through a tertiary bitmap.

2003 went further and broke things. Currency ceased to be a standalone data element and became a sub-element of each amount, which is arguably correct and which invalidates every parser written for the previous sixteen years. Composite data elements and datasets arrived, along with the option of encoding sub-elements as tag-length-value under ISO/IEC 8825.

2023 restructured the document, moved the definitions to a maintenance agency, folded in accumulated corrections approved by the registration and maintenance management group, and reused version digit 2.

Deployment tells a different story from publication. Visa's BASE I technical specifications, which define the authorisation messages of the V.I.P. system, are built on the 1987 edition with a large body of Visa-specific additions. Mastercard's Customer Interface Specification carries authorisations as 1987-version messages; its Integrated Product Messages clearing formats carry presentments and chargebacks as 1993-version messages such as the First Presentment/1240. The 2003 edition, twenty-three years after publication, never achieved meaningful adoption on scheme links at all.

The reason is not inertia in the pejorative sense. It is arithmetic. A card network is a mesh of bilaterally certified endpoints — acquirers, processors, gateways, issuer hosts, ATM estates, terminal fleets — numbering in the tens of thousands, and every pair that exchanges messages has been through a certification process that produced a signed test result. A change to the wire format requires every one of those pairs to re-certify, and a flag day on which both ends switch simultaneously, because a 1987 parser handed a 2003 message does not degrade gracefully; it misreads the amount. The value of the newer edition is real but marginal, and the cost of the transition is catastrophic. So the industry stayed on 1987 and expressed everything new in private-use fields, which brings us to the part that actually costs money.

What "ISO 8583 compliant" is worth

Roughly nothing, on its own. The right question is never "is it ISO 8583?" but "which specification, which version of that specification, and have you got the certification letter?"

Every network of consequence publishes its own document. A partial list of the names, which is useful because these documents are what integration projects actually work from:

Network	Specification
Visa	BASE I Technical Specifications; Single Message System ATM/POS Technical Specifications; Dual Message System Authorization Technical Specifications
Mastercard	Customer Interface Specification; Global Clearing Management System Reference; Integrated Product Messages Clearing Formats; Mastercard Debit Switch Online Specifications; Single Message System Specifications
American Express	Global Credit Authorization Guide; Global Electronic Data Capture Terminal Message Specification; Plural Interface Processing Terminal Interface
Discover	Authorization Interface Technical Specifications
JCB	J/Link Online Interface Guide
UnionPay	CUP Online Message
Diners Club	DCISC Relay Authorisation System; Relay Xpress Authorization System

They diverge in ways that go well past field contents. Visa's BASE I messages are normally carried with a 22-byte header whose values determine the format variant of certain data elements, and BASE I places data elements in three bitmapped arrays of 64 elements each, with all three bitmaps positioned together after the message type and before the message body — rather than deriving the second and third from bit 1 of the primary and bit 1 of DE65, as the standard describes. A parser written strictly to ISO will not read a BASE I message at all.

Mastercard carries the bulk of its proprietary content in private data. In authorisation messages that content lives in numbered subelements of DE48; in clearing messages it lives in private data subelements, the four-digit PDS tags such as PDS 0032 for enhanced acceptor name or PDS 0198 for device type indicator. The same business fact appears in a different container depending on whether you are looking at the authorisation or the clearing record, which is a large part of why reconciliation between the two is the work it is.

Below the schemes, national and vendor dialects abound. Australia's AS 2805 series incorporates ISO 8583 and extends it across a far wider range of payment topics; processor platforms such as ACI's BASE24 define their own external message formats, ISO 8583 in shape and proprietary in content. Data elements 60 to 63 and 120 to 127 are reserved for private use in the standard itself, which is its way of saying: put your incompatibilities here, and stop asking us.

The practical consequence is that conformance is not a property of your code. It is a property of a certification event between your endpoint and one named counterparty, on one named specification, at one named version, evidenced by a test script both sides signed. "We are ISO 8583 compliant" is a statement about a shape.

Message flow

Request and response. The acquirer sends `0100`; the issuer answers `0110`. The response echoes the fields the two sides use to match it to the request — conventionally DE7, DE11, DE32 and DE37, though the precise matching key is set by each scheme and not by ISO — and adds DE38 and DE39. The acquirer runs a timer, and if it expires the acquirer does not know whether the issuer approved, because a response can be lost after the issuer has already committed a hold. That ambiguity is the origin of most of what follows.

Repeat. On a timeout, the acquirer may retransmit with origin digit 1 instead of 0, producing `0101`. Per clause 4.2 the repeat must be byte-identical apart from the MTI, the transmission date and time and the MAC. The issuer, seeing the same STAN and retrieval reference number, is expected to return the original decision rather than authorise a second time.

Reversal. If the acquirer concludes that the transaction did not complete — timeout after retries, a customer who removed the card, a till that crashed mid-sale — it sends `0400`, quoting DE90 so the issuer can find the original, and expects `0410`. Reversals are retried persistently, sometimes for days, because an unreversed authorisation leaves the cardholder's available balance reduced against a purchase that never happened. Where the acquirer is stating a fact rather than asking permission, it uses the advice form `0420`, acknowledged with `0430`.

Advice. Advices exist for everything that happened before anyone could ask. A terminal that authorised offline because the amount fell below its floor limit sends `0220`. A transaction approved by the scheme on the issuer's behalf while its host was unreachable — stand-in processing — reaches the issuer as an advice, because by then the answer has already been given to the merchant.

Completion. A pre-authorisation taken at a hotel check-in or a fuel pump is `0100` for an estimated amount; the final amount arrives later as a financial advice `0220`, which is why a hotel hold can sit on a card for days after checkout.

Network management. `0800` and `0810` keep the link itself alive. DE70, the network management information code, distinguishes the purposes; the values almost universally implemented are `001` for sign-on, `002` for sign-off and `301` for an echo test. Echo tests run every few seconds on a live link, and a switch that stops answering them is taken out of rotation before any customer notices. When you

read that a card network was "down for eleven minutes", the detection almost certainly happened in an `0800`.

Reconciliation. Class 5 messages carry the totals each side believes it exchanged during a settlement window — counts and amounts of credits, debits, reversals and fees, in data elements 74 to 89.

One authorisation, framed for the wire

What actually leaves a terminal is not an ISO 8583 message. It is an ISO 8583 message wrapped in whatever the link demands. A typical arrangement over TCP:

Layer	Bytes	Content
Length prefix	2	Binary count of the bytes that follow
Transport header	5	Protocol identifier `60`, destination address, source address
Message type	4	`0100`
Primary bitmap	8 or 16	`723C448108E08000`
Data elements	remainder	In bitmap order

The five-byte transport header is a Transaction Protocol Data Unit, inherited from an older transport standard and kept because ISO 8583 itself contains no routing information at all. Its convention is one byte of protocol identifier, conventionally `60`, then two bytes of destination address and two of source address; the response is formed by swapping the two addresses. None of it is in ISO 8583. All of it is mandatory in practice.

Where it runs out

The pressure on ISO 8583 is not performance. Switches process these messages in tens of microseconds. It is that the world now wants to say things the format has no room for: full merchant and payee details for sanctions screening, structured remittance information, extended tokenisation data, richer fraud signals, and the growing bulk of EMV data in DE55. The industry's answer for thirty years has been to overload private fields with sub-formats — subelements inside DE48, PDS tags inside clearing records, TLV inside DE55 — until a message contains three different encoding schemes nested inside each other and the integration cost of any change is measured in quarters.

That is the case for ISO 20022, and the subject of a later chapter. ISO 8583 is not being replaced because it failed, but because it succeeded so completely that a format designed to fit a purchase down a 1,200 bit-per-second telephone line is now the constraint on what a purchase is allowed to mean.

The next chapter takes the bitmap on its own and shows, bit by bit, what those sixty-four flags do to the parser at the other end.

24.98 Common wrong ideas

Wrong: ISO 8583 is the language of card payments. **Right:** it is a filing convention specifying the shape of the envelope, and since 2023 the standard does not even contain the message, field and code definitions, which now live in a separately maintained register.

Wrong: two systems that are both faithful to ISO 8583 can exchange messages. **Right:** faithfulness settles nothing about what goes in DE 48 or DE 62, or what response code `05` means to you as opposed

to me; conformance is a property of a certification event between two named endpoints, not of your code.

Wrong: ISO 8583 is a binary format. **Right:** only the bitmap is genuinely binary; the rest is usually characters in ASCII or EBCDIC, and because the standard permits both packed and unpacked numerics the same logical message is 129 bytes on one link and 181 on another.

Wrong: a data element number means the same thing in every edition. **Right:** DE 24 is a network routing identifier in 1987 and a function code in 1993, and response codes are two characters in 1987 and three in 1993.

Wrong: a response code of `00` means the money moved. **Right:** it means the issuer agreed that it would; the instruction that shifts funds between banks is a different message sent later, often the following day.

Wrong: the amount field carries its own currency and decimal point. **Right:** DE 4 is twelve unsigned digits with no separator, and the implied decimal position comes from the ISO 4217 code in DE 49 — two places for sterling, none for yen, three for the Kuwaiti dinar.

Wrong: a repeat message is simply another attempt. **Right:** clause 4.2 requires it to be identical to its original apart from the message type identifier, the transmission date and time and the MAC, which is precisely how the receiver knows it may be a duplicate rather than a second purchase.

Wrong: DE 11 uniquely identifies a transaction. **Right:** the system trace audit number is unique only within one acquirer within one processing day and wraps at 999999; DE 37 is the identifier that survives into the dispute process.

Wrong: a parser written strictly to the standard will read any scheme's traffic. **Right:** Visa's BASE I carries a 22-byte header and places three bitmapped arrays together after the message type rather than deriving them from bit 1, so a strictly conformant parser will not read a BASE I message at all.

Wrong: the industry stays on the 1987 edition out of inertia. **Right:** it is arithmetic — tens of thousands of bilaterally certified endpoint pairs would each have to re-certify and switch on a flag day, because a 1987 parser handed a 2003 message does not fail cleanly, it misreads the amount.

24.99 Chapter summary in 20 lines

1. ISO 8583 is not a language but a filing convention: an agreement about which numbered box holds which value, and how to tell the other end which boxes you bothered to fill in.
2. The whole trick is the bitmap, sixty-four bits declaring the presence or absence of each data element, so that no field name ever has to travel.
3. That compression was necessity rather than elegance: at 1,200 bits per second the difference between labelled text and numbered boxes was the difference between a customer waiting eleven seconds and waiting two.
4. Every message is a message type, one or two bitmaps, and then the data elements in bitmap order, with no framing, no terminator and no length, all of which are the transport's problem.
5. The four-digit message type indicator encodes version, class, function and origin, and reading it digit by digit tells you what kind of message you are holding before you parse anything else.
6. There are really only four kinds of message: a question that may be refused, an answer, an advice that may only be acknowledged, and a cancellation that quotes the message it wants undone.

7. The 2023 third edition moved the messages, data elements and code values out of the standard into a register held by a maintenance agency, so that they can change without republishing an International Standard.
8. It also reused version digit 2, formally conceding that nothing had changed in a way that affects how a message is constructed or parsed.
9. The standard is silent about transport, framing, integrity and settlement, so every deployment bolts on a length prefix, a routing header inherited from an older standard, and a cryptographic checksum.
10. Data elements are defined by a content type and a length rule, fixed or variable, with the variable ones carrying two- or three-digit length prefixes of their own.
11. DE 4 carries twelve digits with no decimal separator, and the exponent comes from the ISO 4217 currency code in DE 49, which is why floating-point storage of amounts does not survive contact with this format.
12. DE 7, DE 12 and DE 13 are three different clocks — universal time rewritten at every hop, and a local time and date carrying no year at all.
13. DE 11 matches a response to a request across one link, while DE 37 is the twelve-character handle that survives into clearing and into a dispute months later.
14. DE 55 is a format inside a format: EMV tag-length-value objects linked according to ISO/IEC 7816-6, carried inside a message parsed by entirely different rules.
15. DE 90 names the victim of a reversal with five mandatory sub-elements totalling forty-two digits, which says something true and slightly alarming about how transactions are identified here.
16. The bitmap buys size, deterministic parsing, a bounded worst case and no dynamic allocation, at the single cost that the message is not self-describing.
17. A malformed bitmap therefore does not fail at the malformed byte; every later field is read at the wrong offset, and a £42.50 purchase arrives as £4,250,000.
18. The 1993 edition's three-digit response codes, classed by leading digit, are better than the 1987 flat list in every respect and still lost, because re-certifying the mesh costs more than the improvement is worth.
19. Every network of consequence publishes its own specification, and the divergences run past field contents into bitmap layout and private data, so "we are ISO 8583 compliant" is a statement about a shape.
20. The format is not being replaced because it failed but because it succeeded so completely that a design fitted to a 1,200 bit-per-second telephone line is now the constraint on what a purchase is allowed to mean.

Chapter sources: ISO 8583:2023, third edition, clauses 1, 3, 4 and 7 and Table 1; ISO catalogue entries for ISO 8583:1993, ISO 8583-1:2003 and ISO 8583:2023; the ISO register of maintenance agencies; ISO 4217; ISO 9564-1; ISO/IEC 7812-1 and ISO/IEC 7816-6; Mastercard Switch Rules, Transaction Processing Rules, Single Message System documentation and Mastercard Developers material for message types 1240 and 1442, DE48 subelements and PDS tags 0032 and 0198; Worldpay Lync ISO 8583 Message Specification v1.3 for processing codes; iso8583.info for the Visa BASE I message structure and the scheme specification names; EFTlab for the Transaction Protocol Data Unit; and the Wikipedia ISO 8583 article for the consolidated 1987 and 1993 data element and response code tables, cross-checked against the standard text where possible.

25

CHAPTER

The Bitmap

25.0 What this chapter gives you

1. You will be able to read a sixteen-character hexadecimal string such as `F23C448128E09200` and say, without a tool, which data elements the message carries.
2. You will be able to explain why bit 1 is not a field at all, and why getting it wrong shifts every subsequent byte in the message by eight or sixteen.
3. You will be able to tell the three wire forms of the same sixty-four bits apart — raw binary, ASCII hexadecimal and EBCDIC hexadecimal — by inspecting the first sixteen bytes of a link.
4. You will be able to write the receive path for an ISO 8583 message in about twenty lines, and say why it needs no tokeniser, no state machine and no backtracking.
5. You will be able to read the notation `n 6`, `n..19` and `ans...999` for what it says about length, and read `b 64` as eight bytes rather than sixty-four.
6. You will be able to diagnose a misaligned parse from its symptom — a card number of all zeroes, an amount with two extra digits, a MAC failure — because there will be no error message to work from.
7. You will be able to say why sixty-four bits is exactly optimal in the worst case, gives up about a quarter against an ideal coder on real traffic, and beats a list of field numbers from about the tenth field onward.
8. You will be able to explain why an unknown field is survivable in Protocol Buffers and fatal in ISO 8583, and why that one property makes adding a field to a card network a certification programme rather than a deployment.
9. You will be able to use a bitmap as an instrument: classify traffic before decoding any field, and name the exact difference between a request and its response with a single exclusive-or.
10. You will be able to state where the bitmap runs out — 128 slots, no versioning, no room for meaning — and why the industry answered with nested structures inside elements 48, 62, 63, 124 and 127 rather than with the tertiary bitmap.

There is a small piece of engineering at the front of every card message on earth that almost nobody outside the payments industry has heard of, and that almost nobody inside it can explain properly. It is eight bytes long. It contains no card number, no amount, no merchant, no date, and no information about the transaction at all. What it carries is information about the *message* — specifically, which of the message's sixty-four possible fields the sender bothered to fill in.

It is called the bitmap. It was designed in the mid-1980s for terminals with kilobytes of memory talking over telephone lines at 1,200 bits per second. Those constraints are gone. The bitmap is not. It is still there, at the front of authorisations approved in 2026, doing the same job in the same eight bytes, and

the reason is not inertia. The reason is that for the problem it solves, sixty-four bits is very close to the smallest number of bits that could possibly work.

This chapter takes those eight bytes apart. By the end of it you will be able to look at a hexadecimal string in a log file and say, without a tool, which fields are in the message.

The plain version

Imagine a class of sixty-four children who always sit in the same order — the same sixty-four chairs, the same sixty-four names, every day, for forty years. Nobody ever moves.

Every morning the class hands in homework, and there is a rule: nobody writes their name on it. The teacher collects the pile, the pile is in seat order, and that is the only thing identifying whose is whose. Most days most of the class has not done any, so on a typical morning maybe twenty children hand something in and the pile is twenty sheets deep, in seat order, with no names.

How does the teacher know the third sheet belongs to child eleven and not child nine?

Because at the top of the pile there is a register. Not a list of names — a row of sixty-four boxes, one per seat, in seat order. A tick means "this child handed something in". A blank means "skip this seat". The teacher reads the register first, learns that seats 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 35, 37, 41, 42, 43, 49, 52 and 55 have ticks, and deals the pile out in that order. Twenty sheets, twenty owners, no names written anywhere.

That row of ticks is the bitmap. Everything in this chapter follows from it.

Ticks are expensive; symbols are cheap

Writing out sixty-four boxes takes a lot of space, so we compress it with a trick that takes five minutes to learn. Take the ticks four at a time. Four boxes can be filled in sixteen different ways: all blank, only the last one, only the third one, and so on up to all four. Invent sixteen symbols, one per pattern, and write one symbol instead of four boxes. The symbols are `0` to `9` and then `A`, `B`, `C`, `D`, `E`, `F`. This is called hexadecimal, and the whole of it fits in a small table.

Symbol	Ticks	Symbol	Ticks
`0`	blank blank blank blank	`8`	tick blank blank blank
`1`	blank blank blank tick	`9`	tick blank blank tick
`2`	blank blank tick blank	`A`	tick blank tick blank
`3`	blank blank tick tick	`B`	tick blank tick tick
`4`	blank tick blank blank	`C`	tick tick blank blank
`5`	blank tick blank tick	`D`	tick tick blank tick
`6`	blank tick tick blank	`E`	tick tick tick blank
`7`	blank tick tick tick	`F`	tick tick tick tick

There is a pattern that makes this easy without the table. Left to right, the four boxes are worth 8, 4, 2 and 1; add up the ticked ones. Blank, tick, tick, blank is 4 plus 2, which is 6. Tick, tick, tick, blank is 8 plus 4 plus 2, which is 14, and 14 is `E`. That is the entire system.

Sixty-four boxes, four to a symbol, is sixteen symbols. So the register — the thing that tells you the shape of an entire card payment — is sixteen characters long. That is shorter than most people's email address.

One real register, decoded by hand

Here is a real one, from a chip-and-PIN purchase:

```
F23C448128E09200
```

Decode it left to right, four boxes per symbol, keeping a running count of which seat we are on.

Symbol	Ticks	Seats covered	Ticked
F	1111	1 to 4	1, 2, 3, 4
2	0010	5 to 8	7
3	0011	9 to 12	11, 12
C	1100	13 to 16	13, 14
4	0100	17 to 20	18
4	0100	21 to 24	22
8	1000	25 to 28	25
1	0001	29 to 32	32
2	0010	33 to 36	35
8	1000	37 to 40	37
E	1110	41 to 44	41, 42, 43
0	0000	45 to 48	none
9	1001	49 to 52	49, 52
2	0010	53 to 56	55
0	0000	57 to 60	none
0	0000	61 to 64	none

Twenty-one seats are ticked: 1, 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 35, 37, 41, 42, 43, 49, 52 and 55.

Now put the names back. Seat 2 is the card number, 3 is what kind of transaction, 4 is the amount, 7 is when it was sent, 11 is the serial number. Seats 12 and 13 are the shop's own clock and calendar; 14 is the card's expiry; 18 is what kind of shop; 22 is how the card was read; 25 is anything unusual; 32 is which bank is asking; 35 is the data copied off the card; 37 is a reference number. Seats 41, 42 and 43 are the till, the merchant and the merchant's name and town. Seat 49 is the currency, 52 is the encrypted PIN, and 55 is everything the chip said.

You have just read a card payment out of sixteen characters. You know it was a chip transaction with a PIN, in a shop, from a real till, with a real merchant name attached. You have not seen a single value yet.

Seat one is not a child

Look at that list again. Seat 1 is ticked, but nothing in the description above is seat 1, because seat 1 is not a child. It is a note that says *there is a second register*. Sixty-four seats was enough in 1985 and it was not enough by 1990, so the designers reserved the very first tick to mean "another sixty-four boxes follow, covering seats 65 to 128". If seat 1 is blank, the register is sixteen characters and that is all. If seat 1 is ticked, the register is thirty-two characters, and the second half describes a second class of sixty-four children who are only occasionally in the room.

In our example the second register is:

```
0000000000000001
```

Fifteen zeroes and a one. Read the same way, the final `1` is the last box of the last group of four, which is seat 128 — the cryptographic seal proving the message was not altered on the way. So the full register for this payment is thirty-two characters:

```
F23C448128E09200 0000000000000001
```

and it declares twenty-two things: bit 1 saying "look at the second half", the twenty fields of the transaction, and the seal at the end.

Some homework is one page and some is forty

There is one more problem the register does not solve. It tells you *whose* homework is in the pile. It does not tell you *how many pages* each piece is.

For most seats it does not matter, because the answer is fixed and everyone knows it. Seat 4, the amount, is always exactly twelve digits. Seat 11, the serial number, is always exactly six. Seat 37 is always exactly twelve characters. You never have to ask. You count.

But some fields cannot be fixed. A card number can be thirteen digits or nineteen. A merchant's extra notes might be four characters or four hundred. For those, the field carries its own page count on the front. A card number arrives as the two digits `16` followed by sixteen digits. Chip data arrives as the three digits `120` followed by 120 bytes. Two digits of length at the front is called LLVAR, three digits is LLLVAR, and that is the whole naming convention: one letter `L` per digit of length, then the word "variable".

So the reader's job is a loop with no cleverness in it at all. Read the register. For each ticked seat, in order: if it is fixed, take exactly that many characters; if it is variable, read two or three digits of length first, then take that many. Move on. Stop when the ticks run out.

What happens when the register is wrong

Now the thing that makes this design frightening. Suppose the register ticks seat 25, but the child in seat 25 did not hand anything in. The teacher takes the next two characters and calls them seat 25's homework — but those two characters were the first two of seat 32's. Seat 32 is now short by two and reads two characters of seat 35. Seat 35 is variable length, so the teacher now reads a *length* out of the middle of somebody else's data, and that length might say 87 when 200 characters remain, or it might say 04.

Every field after the mistake is wrong. Not missing. Wrong. And nothing in the message says so, because there is no name on any of the homework to contradict the register: the whole point was that we stopped writing names to save space.

That is the bargain. The bitmap buys the smallest possible message and the fastest possible reader, and the price is that a single wrong bit does not produce an error where the error is. It produces a plausible-looking message full of nonsense, and the nonsense surfaces somewhere else entirely — often as an amount with two extra zeroes on it.

Where the plain version stops being true

The register is not a register of children, and the numbering is self-referential. In the standard's own numbering the bitmap *is* data element 1, so bit 1 of data element 1 indicates the presence of a second instance of data element 1. That is not a curiosity: it is why you will see published field tables listing element 1 as `b 16` and others listing it as `b 64`, describing the same object. One is counting hexadecimal characters, the other bits, and neither says which. The same ambiguity infects every binary element

in the catalogue. When a specification says `b 64` it almost always means sixty-four *bits*, which is eight bytes, which is sixteen hexadecimal characters. Reading it as sixty-four bytes is a classic first-week mistake.

Sixteen characters and eight bytes are both true, and they are not the same number of bytes on the wire. The plain version treats the bitmap as "sixteen characters". That is one of at least three encodings in real use. The bitmap can travel as eight raw binary bytes, whose individual bits carry the meaning directly, or as sixteen printable characters in ASCII, or as sixteen printable characters in EBCDIC on IBM mainframe links. All three carry identical information, occupy eight, sixteen and sixteen bytes respectively, and look nothing like each other on the link. A parser configured for the wrong one is not slightly wrong; it is catastrophically wrong from the first byte.

A tick tells you a field is present. It does not tell you where it is. The bitmap is not an index. There is no offset table, no pointer, no random access. To find data element 55 you must parse elements 2 through 43 first, because several of them are variable-length and you cannot know where 55 begins until you have measured everything before it. The bitmap makes the message small and the parse deterministic. It does not make the message seekable.

The bitmap says a field exists. It never says what the field means. Bit 62 set tells a parser that data element 62 is present and, from the local field table, how long it is. It says nothing about the contents, because 62 is reserved for private use and is defined by whoever you happen to be connected to. A parser holding the wrong counterparty's field table will read a syntactically valid message and produce semantically wrong output, silently. Worse: for a private element whose length rule the parser does not have, it cannot even skip the field, because it does not know how many bytes to skip. This is the deep limitation of positional encoding, and no care in the bitmap fixes it.

The technical version

What the standard specifies

ISO 8583 defines the message as a message type identifier, followed by one or more bitmaps, followed by the data elements that the bitmaps declare, in ascending order of element number. The third edition, published in July 2023, describes the bitmap component as "one or two message bitmaps, each consisting of 64 bits", in which "each bit signifies the presence (1) or the absence (0) in the message of the data element associated with that particular bit".

Three properties follow from that sentence and they are the whole design. The bitmap is *fixed width*: sixty-four bits always, whether the message carries three data elements or fifty, so a bitmap needs no length prefix of its own. It is *positional*: bit n corresponds to data element n, with no lookup, no translation table and no negotiation, the correspondence being a property of the numbering itself. And it is *ordering*: the data elements follow in ascending numerical order, so the bitmap answers "which fields" and "in what sequence" with the same sixty-four bits. That is why no separators are needed anywhere in the message.

Bit numbering, and the convention people get wrong

Bits are numbered from 1 to 64, left to right, starting at the most significant bit of the first byte. Bit 1 is `0x80` of byte 0, bit 8 is `0x01` of byte 0, bit 9 is `0x80` of byte 1. Formally, for element n the byte index is `(n - 1) / 8` and the mask is `0x80 >> ((n - 1) % 8)`.

This is big-endian bit ordering within the byte, and it is the most common source of a first implementation that almost works. A developer whose instinct is to test `byte & (1 << bit)` — least significant bit first, the convention used almost everywhere else in computing — will produce a parser that reverses

every group of eight, and the failure is not obvious, because the reversed set is still a plausible-looking set of field numbers. Byte `0xF2` correctly declares elements 1, 2, 3, 4 and 7; read least-significant-bit-first it declares 2, 5, 6, 7 and 8. Both look like field numbers. Only one is right.

One bitmap, decoded completely

Take the primary bitmap from earlier — a `0100` authorisation request for a contact chip transaction with online PIN — and expand it fully, in binary, with the element number each bit governs written alongside:

```
byte 0  F2  1111 0010   elements  1  2  3  4 |  5  6  7  8
byte 1  3C  0011 1100   elements  9 10 11 12 | 13 14 15 16
byte 2  44  0100 0100   elements 17 18 19 20 | 21 22 23 24
byte 3  81  1000 0001   elements 25 26 27 28 | 29 30 31 32
byte 4  28  0010 1000   elements 33 34 35 36 | 37 38 39 40
byte 5  E0  1110 0000   elements 41 42 43 44 | 45 46 47 48
byte 6  92  1001 0010   elements 49 50 51 52 | 53 54 55 56
byte 7  00  0000 0000   elements 57 58 59 60 | 61 62 63 64
```

Set bits: 1, 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 35, 37, 41, 42, 43, 49, 52, 55.

Bit 1 is set, so a secondary bitmap follows immediately, before data element 2.

```
00 00 00 00 00 00 00 01
```

The only set bit is the last bit of byte 7, which is bit 64 of that bitmap, which is data element 128.

On the wire, then: MTI, primary bitmap, secondary bitmap, and then elements 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 35, 37, 41, 42, 43, 49, 52, 55 and 128. Four are variable length — the primary account number in element 2 (`n..19`), the acquiring institution identification code in 32 (`n..11`), the track 2 data in 35 (`z..37`) and the integrated circuit card data in 55. The rest are fixed: `n 12` for the amount in 4, `n 6` for the system trace audit number in 11, `an 12` for the retrieval reference number in 37, `ans 8` for the terminal identification in 41, `b 64` for the PIN block in 52 and for the message authentication code in 128.

Element 55 is worth a warning. Published tables disagree: the widely circulated 1987 catalogue lists it as `ans ...999`, while scheme-facing references describe it as binary with a three-digit length, which is what it is in every EMV deployment — a concatenation of EMV tag-length-value objects, which are binary and are not character data. Take the counterparty's specification, not the general table.

Going the other way

To construct a bitmap, start with eight zero bytes and set one bit per element: element 2 sets `0x40` in byte 0, element 39 sets `0x02` in byte 4, element 64 sets `0x01` in byte 7. Two published bitmaps make good test vectors. `7010001102C04804` expands to elements 2, 3, 4, 12, 28, 32, 39, 41, 42, 50, 53 and 62 — a response carrying a card number, a processing code, an amount, a local time, a transaction fee, an acquirer identifier, a response code, a terminal, a merchant, a settlement currency, security-related control information and one private field. `7234054128C08000` expands to elements 2, 3, 4, 7, 11, 12, 14, 22, 24, 26, 32, 35, 37, 41, 42 and 49 — a magnetic stripe authorisation from an era when element 24 still carried a network international identifier. If your parser reproduces both, your bit arithmetic is right.

The secondary bitmap

The secondary bitmap covers data elements 65 to 128. It is present if and only if bit 1 of the primary bitmap is set, and when present it sits immediately after the primary bitmap, before any data element value.

The consequence is that the secondary bitmap is not optional in the way other fields are optional. Its presence or absence changes the byte offset of every subsequent field in the message by eight or sixteen bytes. Getting bit 1 wrong is the highest-blast-radius single-bit error in the whole format.

Elements 65 to 128 hold what a purchase does not need: settlement and reconciliation totals in 74 to 89, the original data elements of a reversal in 90, the file-action elements in 91 to 94, the settlement and receiving institution identifiers in 99 and 100, and private-use space in 120 to 127. Element 96 holds a message security code and 128 a message authentication code.

A `0400` reversal request typically sets element 90, "Original data elements", a forty-two digit fixed field holding the original message's MTI, system trace audit number, transmission date and time, and acquiring and forwarding institution identification codes, concatenated. That is how a reversal names the transaction it is reversing when there is no other shared identifier. A reversal carrying element 90 and a MAC in 128 has a secondary bitmap of:

```
00 00 00 40 00 00 00 01
```

Byte 3 is `0x40`, whose second bit is bit 26 of the secondary bitmap, which is element 90. Byte 7 is `0x01`, element 128.

The tertiary bitmap

Later editions extend the element range to 192, with a tertiary bitmap covering 129 to 192 signalled by element 65. The 1987 catalogue lists element 65 as `b 1`, an extended bitmap indicator — a single bit whose whole purpose is to say that a third bitmap follows.

You will not see one. Every reference that discusses it describes it as extremely rare or effectively unused on card rails, because the schemes solved the same problem a different way, by defining private elements that contain their own internal structure. Mastercard's private data subelements inside data elements 48, 124 and 127, and Visa's proprietary subfields inside 48, 62 and 63, are nested tag-length-value structures with their own subelement identifiers. They gave the industry unbounded extension space without ever touching the top-level bitmap.

Three wire forms of the same sixty-four bits

The bitmap can be represented as eight bytes of binary data or as sixteen hexadecimal characters in ASCII or EBCDIC. This is not an implementation detail. It is the first thing you must establish about any link, and it is not carried in the message. The same primary bitmap, in all three:

Form	Bytes on the wire	Length
Raw binary	`F2 3C 44 81 28 E0 92 00`	8
ASCII hexadecimal	`46 32 33 43 34 34 38 31 32 38 45 30 39 32 30 30`	16
EBCDIC hexadecimal	`C6 F2 F3 C3 F4 F4 F8 F1 F2 F8 C5 F0 F9 F2 F0 F0`	16

The second row is the sixteen characters `F23C448128E09200` written out one byte each. The third is the same sixteen characters in the encoding IBM mainframes use, where the digit `0` is `0xF0` rather than `0x30`.

Now the failure. Suppose a parser expects raw binary and receives the ASCII hexadecimal form. It reads the first eight bytes, `46 32 33 43 34 34 38 31`, as a bitmap — in binary `01000110 00110010`

`00110011 01000011 00110100 00110100 00111000 00110001` — and declares elements 2, 6, 7, 11, 12, 15, 19, 20, 23, 24, 26, 31, 32, 35, 36, 38, 43, 44, 46, 51, 52, 53, 59, 60 and 64.

Twenty-five elements, including the MAC in element 64. Bit 1 is clear, so the parser concludes there is no secondary bitmap. Nothing about that set is implausible on its face. The parser then reads twenty-five fields out of what are in fact the remaining eight characters of the bitmap followed by the message body, and produces a card number that is not a card number, an amount that is not an amount, and a confident structured object that is entirely fiction. This is why the encoding is the first question on any new link, and why "it parses" is not evidence that it parses correctly.

Fixed and variable length data elements

Every data element has a content type — `a` alphabetic, `n` numeric, `s` special characters, `an`, `as`, `ns` and `ans` for the combinations, `b` binary, `z` for track 2 and track 3 code set per ISO/IEC 7813 and ISO/IEC 4909, `x+n` for a signed amount, `p` for pad — and a length rule.

The length rule is written after the type. A bare number means fixed: `n 6` is exactly six digits, always, and a sender with fewer must pad. Leading dots mean variable, one dot per digit of the length prefix: `n..19` is LLVAR, a two-digit length then up to nineteen digits; `ans...999` is LLLVAR, a three-digit length then up to 999 characters. A single dot, `n.6`, is LVAR, which exists in the notation and is essentially unused. For binary elements the declared number is bits, not bytes, so `b 64` is eight bytes — a constant source of confusion, made worse by tables that use the same notation for character counts.

Most of the catalogue is fixed length, and of the elements that carry a normal card purchase only 2, 32, 35 and 55 are variable. That is deliberate: fixed fields need no length prefix and no bounds arithmetic, and can be copied with a constant offset and a constant count.

LLVAR and LLLVAR on the wire

This is the part of ISO 8583 that produces the most interoperability defects, because the notation `n..19` specifies the semantics of the length prefix and says nothing about its encoding. There are four independent decisions, and each is a place two implementations can differ.

How the length digits are encoded. ASCII digits use one byte per digit, so an LLVAR prefix is two bytes and an LLLVAR prefix three. Packed binary-coded decimal uses one byte per two digits, so an LLVAR prefix is one byte and an LLLVAR prefix two, with a leading zero nibble. The length 27 is `0x32 0x37` in ASCII and `0x27` in BCD. A parser expecting the wrong one is off by one or two bytes at the front of every variable field.

Whether the length is binary or decimal. Some specifications encode the length as a plain unsigned integer rather than decimal digits, making 27 `0x1B`. The pyiso8583 library, which exists to model this variety, treats length encoding as a configuration axis separate from data encoding for this reason.

What the length counts. For character data it counts characters, which is also bytes. For packed BCD numeric data it usually counts *digits*, and the field occupies half that many bytes. For binary data it usually counts bytes. Get this wrong on a packed field and you read half the field or twice it. Serious libraries make it explicit: pyiso8583 offers a `len_count` setting whose values are `bytes` and `nibbles`, with bytes as the default, alongside left- and right-padding options for odd-length packed data. That such an option has to exist tells you how much disagreement is out there.

What happens at the maximum. `n..19` means up to nineteen. It does not mean a sender may send twenty. Scheme specifications routinely tighten the maxima below the ISO values, so a message that is valid ISO 8583 may still be rejected.

LLVAR and LLLVAR are therefore not one thing but a family of closely related things, and the specification that governs your link — not the ISO document, not a reference website — is the only authority on which member of the family you are speaking.

The parser loop

Once the encodings are settled, the entire receive path is about twenty lines.

```
read MTI (4 characters)
read primary bitmap (8 bytes, or 16 characters if hex-encoded)
if bit 1 set:
    read secondary bitmap (same size)
for n from 2 to 128:
    if bit n is not set: continue
    look up definition[n]
    if definition[n] is fixed:
        take definition[n].length units
    else:
        take definition[n].prefix_digits, decode as length L
        take L units
    store as element n
assert no bytes remain
```

There is no tokeniser, no state machine, no backtracking, no recursion, no dynamic allocation if the buffers are pre-sized, and no ambiguity about whether a value is a string or a number, because the definition already said. That is why a terminal with sixteen kilobytes of RAM could do it in 1987 and a switch handling tens of thousands of messages per second can do it now without a garbage collector running.

The last line is the only error detection the format gives you for free, and it is worth more than it looks. If the bitmap and the body agree, the parse consumes exactly the message and stops; if they do not, there are bytes left over or the parser runs off the end. Implementations that silently tolerate trailing bytes throw away the one cheap integrity check they had.

Why sixty-four bits is close to optimal

The bitmap solves a narrow problem: communicate an arbitrary subset of sixty-four known items, as compactly as possible, decodable in constant time by a machine with no memory to spare. It is worth being precise about how good the answer is.

The worst case is exactly optimal. There are 2 to the power 64 possible subsets of sixty-four elements, so distinguishing all of them requires at least 64 bits by counting alone. The bitmap uses exactly 64. No encoding that can express every subset is smaller in the worst case, and the bitmap meets the bound with equality. That is not “close to optimal”, it is optimal, for the problem as stated.

The typical case gives up about a quarter. Real authorisations carry around seventeen to twenty-one elements, not arbitrary subsets. The information content of an arbitrary seventeen-of-sixty-four subset is the base-two logarithm of sixty-four choose seventeen, which is 50.3 bits, or 6.3 bytes. The bitmap spends 8. The overhead against a cardinality-aware ideal coder is about 27 per cent, or 1.7 bytes, and you would pay for that saving with an arithmetic coder in the hot path of every transaction on earth.

The obvious alternative loses above nine fields. The intuitive design is to send a list of the field numbers present, and a field number in the range 1 to 128 fits in seven bits. Seven bits per present field beats a 64-bit bitmap only below about nine fields: at nine the list costs 63 bits and the bitmap costs 64, and from ten onward the bitmap wins and never stops winning. Add a one-byte count so the receiver knows when the list ends and the crossover moves to exactly eight. A card authorisation

carrying twenty-one elements would need 147 bits, nineteen bytes, as a field list, against eight as a bitmap.

Against a modern self-describing format the gap is about two to one. Protocol Buffers encodes each present field with a varint tag holding the field number and wire type, which is one byte for field numbers up to 15 and two bytes for 16 to 2047. The twenty-one elements of our example would cost 34 bytes of tags against 16 bytes for the two bitmaps, on framing overhead alone.

Against a fixed layout the gap is three orders of magnitude. The other way to avoid labels is to give every field a fixed position whether it is used or not. Summing the maximum lengths of data elements 2 to 64 gives 12,733 characters, before the secondary range. Even excluding the twelve elements defined as `...999` — the additional-data and reserved fields — the remainder is 745 characters. A real authorisation whose two bitmaps and twenty-one populated elements come to 351 bytes, with the bitmaps 4.6 per cent of that, makes a fixed layout not a design but a joke about telephone bills.

And sixty-four is the right number for the machine, not just the message. A 64-bit bitmap is one register on any processor built since the mid-1990s. Testing presence is a single AND. Counting fields is a single population-count instruction. Comparing two messages' field sets is a single XOR. In 1987 this meant two 32-bit words rather than one; the committee did not choose 64 because it was a machine word, but it has aged into being one, which is a large part of why nobody has managed to justify replacing it.

The comparison that actually matters

All of the above is about size, and size is the least interesting axis. The real difference between a bitmap and a tagged format is *skippability*.

In Protocol Buffers, JSON, XML or ASN.1 BER, a receiver that meets a field it has never heard of can skip it, because the wire type or the delimiter says how many bytes to step over. Unknown fields are survivable and forward compatibility is free. In ISO 8583 a receiver that meets a field it has no definition for cannot skip it, because the number of bytes to skip is in the definition it does not have. It cannot parse the rest of the message either. There is no recovery.

This single property explains an enormous amount about how the card industry behaves. It is why adding a field to a card network is a multi-year certification programme rather than a deployment, why schemes publish mandates eighteen months ahead, and why private data lives inside elements 48, 62, 63, 124 and 127 as nested tag-length-value structures — because inside a `...999` wrapper whose length the parser *does* know, tags become skippable again, and the schemes rebuilt forward compatibility one level down where the bitmap could not reach. It is also a large part of the argument for ISO 20022, which is self-describing from top to bottom and pays for it in bytes.

Parsing failure modes

These occur in production, and every one presents as a misaligned parse rather than a clean error. That is the point.

Wrong bitmap encoding. Binary read as ASCII hex or the reverse. Off by eight bytes for the whole message. Diagnosed by checking whether the first sixteen bytes all fall in `30` to `39` and `41` to `46` (ASCII) or `F0` to `F9` and `C1` to `C6` (EBCDIC); if they do, it is a hex-encoded bitmap.

Reversed bit order within bytes. Least-significant-bit-first testing. The declared field set is a plausible-looking but wrong permutation: `F2` gives 1, 2, 3, 4, 7 correctly and 2, 5, 6, 7, 8 incorrectly. Diagnosed by decoding a known-good bitmap such as `7010001102C04804` and checking you get 2, 3, 4, 12, 28, 32, 39, 41, 42, 50, 53, 62.

Bit 1 set with no secondary bitmap sent. The parser consumes the first eight bytes of data element 2 as a secondary bitmap. If element 2 begins with the ASCII characters `16411111`, it reads a secondary bitmap declaring elements 67, 68, 72, 75, 76, 78, 79, 83, 84, 86, 91, 92, 96, 99, 100, 104, 107, 108, 112, 115, 116, 120, 123, 124 and 128 — twenty-five phantom fields, several in private-use space — and tries to read them all out of the transaction body. The most destructive bitmap defect there is, and common in hand-rolled senders.

Bit 1 clear with a secondary bitmap sent. The mirror image. The parser reads the secondary bitmap's eight bytes as the value of the first declared element. Because a secondary bitmap is usually mostly zeroes, the classic symptom is a card number of all zeroes.

A bit set for a field the sender did not write. Every subsequent field shifts left by the missing field's length. Fixed fields absorb the shift silently; the first variable field afterwards reads its length prefix from the middle of somebody else's data, and the parse either overruns the buffer or ends with bytes remaining.

A field written without its bit set. The parser stops early, or reads the unannounced bytes as the next declared field. The trailing-bytes assertion catches it, if the implementation has one.

Length prefix encoding mismatch. ASCII versus BCD on LLVAR and LLLVAR. Off by one or two bytes per variable field, cumulative: four variable fields drift by four to eight bytes by the end.

Length counting mismatch. Digits versus bytes on packed numeric fields. Reads half the field or twice it. The symptom is a PAN of the right length with the wrong digits, or half the expected length.

Scheme-tightened maxima. Valid against the ISO catalogue and rejected by the counterparty. This one at least fails loudly, usually with a format error — response code `30` in the 1987 two-character response code set.

Private element with no local definition. Unrecoverable: the parser cannot determine the length and cannot proceed. The correct behaviour is to reject the whole message rather than guess.

MAC failure standing in for a parse failure. Data elements 64 and 128 carry a message authentication code computed over the message, so any change to the bitmap changes the MAC. A bitmap defect on a MAC-protected link therefore presents first as an authentication failure rather than a parse error, which is confusing until you realise the MAC is doing exactly what it should.

The through-line is that positional encoding has no redundancy. Nothing in the byte stream says "this is the amount". The claim that byte 47 is the amount lives entirely in the bitmap and the field table, and if either is wrong the message will still parse, still validate, and still be wrong. Every mature payments organisation ends up with the same three defences: assert that the parse consumes the message exactly, range-check every field against its declared type before using it, and keep a golden set of known-good messages that new builds must reproduce byte for byte.

The bitmap as a diagnostic instrument

Once you can read a bitmap you get several things for free.

You can classify traffic without parsing it. A production switch can read eight bytes and know whether a message carries chip data, a PIN block or a MAC, and whether it is a stripe transaction — routing, fraud-screening and capacity-planning information available before any field is decoded.

You can diff two messages in one instruction. Exclusive-or the two primary bitmaps and the set bits in the result are exactly the elements that differ in presence. Our `0100` request and a corresponding `0110` response, whose primary bitmap is `703800010E808200`, differ by `8204448026601000`, which expands

to elements 1, 7, 14, 18, 22, 25, 35, 38, 39, 42, 43 and 52. The response has dropped the card data and the terminal context and added elements 38 and 39, the authorisation identification response and the response code. The entire logical difference between a request and its answer, in one exclusive-or.

You can specify conformance precisely. Scheme certification is largely a statement about bitmaps: for this message type, these bits must be set, these must not, these are conditional on those. A test plan is, at bottom, a list of required and forbidden bit patterns, which is why it automates so well. And you can read a log over somebody's shoulder — an engineer who decodes the characters after the MTI on sight can tell in seconds whether a failing message is malformed at the top level or merely carrying a value the issuer did not like, which is usually the first useful question in an incident.

Where the bitmap runs out

The bitmap's limits are structural, not incidental, and they are the whole case for the format that will eventually replace ISO 8583.

There are 128 slots, most spoken for by the ISO catalogue and the rest private-use fields that different counterparties have filled with different things. There is no way to add a 129th concept without either the tertiary bitmap nobody implements or a nested structure inside an existing element. The industry chose nesting, which works, but which means the top-level bitmap no longer describes the message — it describes the outer envelope, and the interesting content is inside element 48 or 55 or 127, indexed by a second-level scheme the bitmap knows nothing about.

There is no versioning. The bitmap does not say which field table it was built against. The MTI's first digit names an edition of the standard, but the standard no longer contains the field definitions — in the third edition they were moved out into a separately maintained register. Two systems can agree on the bitmap and disagree on what element 62 is, and nothing in the message will tell them.

And there is no room for meaning. A bit can say "present". It cannot say "present but unreliable", or "present, populated by a stand-in processor rather than the issuer". Every richer statement becomes a value inside another field, which is how element 48 came to contain dozens of subelements in some scheme specifications.

None of that stops the bitmap working. As of writing, in 2026, the overwhelming majority of card authorisations on earth still begin with a four-digit MTI whose first digit says 1987 and then eight bytes of flags, and they will for years yet. It is a forty-year-old trick that is provably optimal for the problem it was given, and the problem it was given was smaller than the one the industry now has. That gap, not any defect in the bitmap, is what the rest of this volume is about. The next chapter takes the fields the bitmap declares and says what each of them actually holds.

25.98 Common wrong ideas

Wrong: `b 64` in a field table means sixty-four bytes. **Right:** it means sixty-four bits, which is eight bytes, which is sixteen hexadecimal characters, and reading it the other way is a classic first-week mistake.

Wrong: bit 1 declares data element 1. **Right:** bit 1 declares a second bitmap covering elements 65 to 128, and its presence or absence moves the byte offset of every subsequent field in the message.

Wrong: the bitmap is an index, so a parser can jump straight to data element 55. **Right:** it declares presence and order and nothing else; several earlier elements are variable length, so you must measure 2 through 43 before you know where 55 begins.

Wrong: test a bit with `byte & (1 << bit)`, as everywhere else in computing. **Right:** bits are numbered from the most significant bit of the first byte, so element *n* is `0x80 >> ((n - 1) % 8)` in byte `(n - 1) / 8`; the least-significant-first version yields a plausible-looking but wrong set of field numbers.

Wrong: the bitmap is sixteen characters on the wire. **Right:** that is one of three encodings in real use, alongside eight raw binary bytes and sixteen EBCDIC characters, and nothing in the message says which one you are receiving.

Wrong: if the message parses, the parse is correct. **Right:** positional encoding has no redundancy, so a wrong bitmap or a wrong field table produces a complete, valid-looking, entirely fictional object; only asserting that the parse consumes the message exactly catches it.

Wrong: a parser can skip a private field it does not recognise. **Right:** the number of bytes to skip lives in the definition it does not have, so it can neither skip the field nor parse the rest of the message, and the correct behaviour is to reject the whole thing.

Wrong: LLVAR means a two-byte ASCII length prefix. **Right:** `n..19` fixes the semantics of the prefix and says nothing about its encoding, which may be ASCII digits, packed binary-coded decimal or a binary integer, and which may count bytes or nibbles.

Wrong: the tertiary bitmap is how you get past 128 elements. **Right:** it is described everywhere as effectively unused on card rails, and the schemes extended instead by nesting tag-length-value structures inside elements they already had.

Wrong: a message that is valid ISO 8583 will be accepted. **Right:** scheme specifications routinely tighten the maxima below the ISO values, so a valid message can still come back with a format error.

25.99 Chapter summary in 20 lines

1. Every ISO 8583 message begins with eight bytes that say nothing about the transaction and everything about the message: which of its sixty-four possible fields the sender filled in.
2. A set bit means the corresponding data element is present, a clear bit means it is absent, and the declared elements follow in ascending numerical order.
3. Because the bitmap gives both the set of fields and their sequence, the message needs no field names, no separators and no offsets anywhere.
4. Sixty-four bits is sixteen hexadecimal characters, and reading them four bits at a time, worth 8, 4, 2 and 1, is the whole skill.
5. The primary bitmap `F23C448128E09200` declares elements 1, 2, 3, 4, 7, 11, 12, 13, 14, 18, 22, 25, 32, 35, 37, 41, 42, 43, 49, 52 and 55, which is a chip-and-PIN purchase in a shop, read without seeing a single value.
6. Bit 1 is not a data element: it announces a secondary bitmap covering elements 65 to 128, which sits immediately after the primary bitmap and before any value.
7. Getting bit 1 wrong is the highest-blast-radius single-bit error in the format, because it shifts every subsequent field by eight or sixteen bytes.
8. The bitmap says which fields are present; the field table says how long each is, and fixed lengths are simply counted while variable ones carry a two- or three-digit length prefix.
9. The receive path is therefore a loop of about twenty lines with no tokeniser, no state machine, no backtracking and no allocation, which is why a terminal with sixteen kilobytes of memory could run it in 1987.
10. Its one free integrity check is that the parse must consume the message exactly, and implementations that tolerate trailing bytes throw that check away.

11. The same sixty-four bits travel as eight raw binary bytes, sixteen ASCII characters or sixteen EBCDIC characters, and the encoding is the first thing to establish on any new link because the message does not carry it.
12. A parser configured for the wrong encoding is not slightly wrong; it produces a confident, well-formed and entirely fictional transaction from the first byte.
13. Bits are numbered from the most significant bit of the first byte, and the usual least-significant-first instinct yields a plausible but wrong field set that looks like field numbers either way.
14. For the problem as stated the bitmap is exactly optimal in the worst case, costs about 27 per cent more than an ideal cardinality-aware coder on real traffic, and beats a list of field numbers from roughly the tenth field onward.
15. Against Protocol Buffers the framing gap is about two to one, and against a fixed layout with every field always present it is three orders of magnitude.
16. The property that actually matters is not size but skippability: a tagged format can step over a field it has never heard of, and ISO 8583 cannot, because the length is in the definition the receiver lacks.
17. That single property explains multi-year certification programmes, mandates published eighteen months ahead, and the whole practice of nesting private data inside elements 48, 62, 63, 124 and 127 where lengths are known.
18. Every bitmap defect presents as a misaligned parse rather than a clean error, with symptoms such as a card number of all zeroes, an amount with extra digits, or a MAC failure standing in for a parse failure.
19. Once read fluently the bitmap becomes a diagnostic instrument: it classifies traffic before any field is decoded, and an exclusive-or of two bitmaps names exactly the elements by which a request and its response differ.
20. The bitmap's limits are structural — 128 slots, no versioning, and no way to say anything richer than "present" — and that gap, rather than any defect in the design, is the case for whatever eventually replaces it.

Chapter sources: ISO 8583:2023, third edition, July 2023, on the message bitmap component, via the publicly available standard preview; the ISO catalogue entry for ISO 8583-1:2003; the Wikipedia ISO 8583 article for the 1987 data element catalogue, the format and length notation tables, the worked bitmap 7010001102C04804 and the 1987 response code table; the ISOParser ISO 8583 reference for the field catalogue with LLVAR and LLLVAR annotations, the worked bitmap 7234054128C08000, the note that binary element lengths are expressed in bits, and Visa BASE I/BASE II and Mastercard IPM private-field practice; the pyiso8583 v4.0.1 specifications manual for length encoding, length-count in bytes versus nibbles and odd-length packed padding; the moov-io/iso8583 bitmap documentation for bitmap sizing; and the IBM App Connect ISO8583 sample documentation for the rarity of the tertiary bitmap. All bitmap expansions, byte counts, encoding tables and size, crossover and information-content calculations were computed from those definitions and verified against the published worked examples.

26

CHAPTER

The Fields That Matter

26.0 What this chapter gives you

1. You will be able to say, for each of the fifteen fields that carry a card authorisation, what it holds, who writes it and what specifically breaks when it is wrong.
2. You will be able to read a field definition such as `n..19`, `ans 8` or `ans...999` and know the character set, the length rule, and the justification and padding that go with it.
3. You will be able to explain why data element 4 carries `000000004250` with no decimal point, and why the position of that point is determined by data element 49 rather than by the amount.
4. You will be able to decompose a processing code such as `010300` into transaction type and two account types, and say why an account-type pair of `00` does not mean "current account".
5. You will be able to say which clock a scheme rule is talking about when it names DE7, DE12, DE13 or DE15, and why a purchase at 23:58 on 31 December can be reconstructed 364 days wrong.
6. You will be able to explain why sending `01` in DE22 for a card that was physically present is a commercial event rather than a technical one.
7. You will be able to distinguish the identifiers by lifetime — the STAN that lasts one request-and-response pair, the RRN that lasts into a chargeback weeks later — and say which one an acquirer will ask a merchant for.
8. You will be able to state exactly which parts of an authorisation are cryptographically protected and which are unauthenticated assertions made by whoever is sending.
9. You will be able to name which tags inside DE55 come from the card and which from the terminal, and say why the amount, currency and country have to appear twice in one message.
10. You will be able to explain why DE39 `00` does not mean the transaction completed, and what else a response can carry that changes the outcome.

The previous two chapters gave you the envelope and the index. The message type indicator says what kind of message this is; the bitmap says which numbered boxes are inside. Neither says anything about the contents, and the contents are where integrations actually fail.

This is worth stating plainly. A message that will not parse is an easy problem: it fails immediately, it fails identically every time, and the fix is usually one line. The expensive failures are the ones where every field parses perfectly, the bitmap is correct, the lengths are right, and the value in one box is simply wrong — a refund flagged as a purchase, an amount off by a factor of a hundred, a terminal identifier reused across two shops, a reference number regenerated on a retry. Those do not surface at the parser. They surface three days later in a reconciliation report, or six weeks later in a chargeback, and by then the message that caused them has been through four systems belonging to three companies.

So this chapter walks the fields one at a time. For each one: what it holds, who writes it, and what specifically goes wrong when it is wrong. There are 128 numbered elements in ISO 8583 and 192 if you count the tertiary bitmap, but a working card authorisation is built from perhaps twenty, and about fifteen of those carry the weight. Those fifteen are the subject here.

The plain version

Imagine a form that travels along a row of desks.

The form has numbered boxes and no labels — that trick you already know. What you do not yet know is the rule that governs the desks. Every desk along the row is allowed to write in *some* of the boxes, and only those. Every other box on the form it must copy across to the next desk exactly as it received it: same characters, same spaces, same everything. Copy it wrong and nobody notices at your desk. They notice at the far end, or on the way back, when the form no longer matches the one that was sent.

There are four desks. The till in the shop. The shop's bank. The card company in the middle. Your bank. The form goes left to right, your bank writes the answer into two boxes, and the form comes back right to left.

Who writes what

The till fills in almost everything factual about the sale. It knows the card number, because the card just told it. It knows the amount, because the shopkeeper just typed it. It knows what kind of sale this is — a purchase, not a refund. It knows what time its own clock says. It knows how the card was presented: tapped, inserted, swiped or typed in by hand. It knows which till it is and which shop it belongs to. And it knows what the chip said, because the chip signed a little package of data and handed it over.

The shop's bank fills in the boxes about the journey rather than the sale. What time it is sending this message. A serial number of its own so it can match the answer to the question. Its own identifying number. And a reference number that it invents now and that will follow this transaction for the rest of its life — into the shop's bank statement, into the settlement file, and into any argument that happens later.

The card company in the middle mostly reads. It looks at the first six or eight digits of the card number, works out which bank issued it, and forwards the form.

Your bank writes exactly two boxes. Box 38 gets a six-character approval code that your bank makes up on the spot — the thing printed on the receipt as "AUTH CODE". Box 39 gets two characters that mean yes or no. Everything else on the returning form is a copy.

The worked example

The same purchase as two chapters ago. A book, £42.50, at a bookshop in London, on 17 March 2026, at 32 minutes and 5 seconds past two in the afternoon. Here is the form when it leaves your bank on the way back, with the author of each box named.

Box	Contents	Who wrote it
2	4111111111111881	The till, from the card
3	000000 (a purchase)	The till
4	000000004250	The till
7	0317143205	The shop's bank
11	004417	The shop's bank

Box	Contents	Who wrote it
12	143205	The till's clock
13	0317	The till's clock
22	071 (tapped, has a PIN pad)	The till
37	607614430041	The shop's bank
38	A4T29B	Your bank
39	00 (yes)	Your bank
41	TERM0042	The till
42	KEDBYTEBOOKS01	The till
49	826 (pounds)	The till
55	(the chip's signed package)	The chip and the till together

Fifteen boxes. Two of them written by the bank that decided; thirteen of them written before the message ever left the shop.

Three ways this goes wrong

The amount with no decimal point. Box 4 is `000000004250`. There is no decimal point in it and there never will be, because box 49 says `826` and everybody has agreed that pounds have two digits after the point. Now suppose the till was set up by someone who thought box 4 held whole pounds. It sends `000000000042`. The customer is charged 42 pence, the shop is short by £42.08, and nothing anywhere reports an error, because `000000000042` is a perfectly valid amount. Or suppose the mistake runs the other way and the till sends `000000425000`. The customer is charged £4,250 for a book, and the first anyone hears about it is a telephone call.

Two tills with the same serial number. Box 11 is a serial number the shop's bank uses to match answers to questions. Suppose the shop opens a second till and the engineer copies the configuration from the first, including the counter it starts from. Now two purchases, seconds apart, both go out as number `004417`. Two answers come back. Each till takes the first answer that arrives. There is a fifty-fifty chance that till one prints the approval that belonged to till two, and the customer whose card was actually declined walks out with a book.

The box that says how the card was read. Box 22 says `071`: the card was tapped, on a terminal that has a PIN pad. Suppose the till is configured to send `011` instead, which means the number was typed in by hand. Nothing fails. The purchase is approved and the receipt prints. But "typed in by hand" is what a telephone order looks like, and the rules that decide who pays when a purchase turns out to be fraud treat a typed-in number very differently from a tapped card. The shop has just given away its protection, at a slightly higher fee, by sending one digit it did not mean.

None of those three is a software crash. All three are one box holding a value that is the wrong kind of true.

Where the plain version stops being true

The desks do not merely copy. The rule that every desk copies other people's boxes unchanged is the single most useful lie in the plain version, and it is a lie. Acquirers routinely rewrite fields on the way through. They renumber the serial number in DE11, because the number the terminal used is unique only within that terminal and the acquirer needs one unique within itself. They may replace the card number in DE2 with a network token, so the number the issuer sees is not the number embossed on the card. They translate processing codes between their own terminal interface and the scheme's

dialect. They convert amounts when the settlement currency differs from the transaction currency, adding DE5 and DE50 that the terminal never sent. The practical consequence is that the message the issuer receives is not the message the terminal sent, and when you are debugging across an acquirer boundary you are looking at two different messages that describe one purchase. Field values are not conserved along the chain. Only a handful are, and knowing exactly which is most of the skill.

One field does not have one meaning. ISO 8583 fixes the number, the name and the format of each data element. It does not fix the vocabulary inside most of them. DE3's account-type pairs, DE22's entry-mode codes, DE39's response values and DE55's tag requirements are all set by the scheme, and the schemes disagree with each other in small, specific, expensive ways. The same six digits in DE3 can mean a cash advance from a credit account on one network and something the next network does not implement at all. Worse, the field's own shape changes between editions of the standard: DE22 is three digits in the 1987 edition and a twelve-position code in 1993; DE39 is two characters in 1987 and three in 1993; DE12 is a local time in 1987 and a local date-and-time in 1993. "Which field" is never a complete question. The complete question is which field, on which scheme, in which edition, in which direction.

The reply is not two boxes. The plain version says the issuer writes DE38 and DE39 and copies the rest. In practice the response can also carry a different amount from the one requested, when the issuer approves a partial amount and the terminal must handle the shortfall. It can carry balances in DE54. It can carry issuer authentication data and issuer scripts in DE55, which the terminal must hand back to the chip, and which can cause a card to refuse a transaction that DE39 just approved. It can carry additional response data in DE44 and private data in DE48 that changes what the terminal is required to display. Treating DE39 as the whole answer is how terminals end up printing "APPROVED" for transactions that did not complete.

Almost nothing in the message is proof. The last correction is the one that reframes everything else. Of the fifteen fields in the table above, exactly one is cryptographically protected against a lying sender: DE55, which contains a cryptogram the chip computed with a key only the chip and the issuer hold. DE52 protects the PIN. DE64 protects the message as a whole between two adjacent parties. Everything else — the entry mode, the terminal identifier, the merchant identifier, the amount, the currency — is an unauthenticated assertion made by whoever is sending. The issuer cannot verify that DE22 is honest. It can only decide how much to trust the party that sent it. This is why card payments are governed by liability rules and scheme membership rather than by proof, and it is why an incorrect DE22 is a commercial event rather than a technical one.

The technical version

Reading a field definition

Every scheme specification describes a data element with the same shorthand, and it is worth being precise about it before using it fifteen times.

The letters give the character set. `n` is numeric, digits only, and is conventionally carried as packed binary-coded decimal, two digits to the byte. `a` is alphabetic, `an` alphanumeric, `ans` alphanumeric plus the special characters, `b` binary, and `z` the track-2 character set of ISO/IEC 7813, which includes the field separator. `x+n` denotes an amount preceded by a debit or credit indicator.

The number gives the length, and the notation for the length is the part people get wrong. `n 6` is fixed: exactly six digits, always present in full. `n..19` is variable with a two-digit length prefix, conventionally written LLVAR: the field begins with two digits saying how many follow, so a sixteen-digit card number

is transmitted as `16` then the sixteen digits. `ans...999` is variable with a three-digit length prefix, LLLVAR. Some specifications write `n..19` as `n-19` or `LLVAR n 19`; they mean the same thing.

Fixed numeric fields are right-justified and zero-filled. Fixed alphanumeric fields are left-justified and space-filled. That asymmetry causes real defects: a terminal identifier of `TERM42` in an eight-character field is `TERM42` plus two spaces, not `00TERM42`, and a system that trims trailing spaces before comparing will match records that a system which does not trim will treat as different.

DE2, primary account number

Format `n..19`, LLVAR, up to nineteen digits.

DE2 carries the account number as defined by ISO/IEC 7812-1: an issuer identification number, an individual account identifier, and a Luhn check digit in the final position. Real lengths run from twelve to nineteen, with sixteen dominant and fifteen normal for American Express.

It is populated by the terminal, from track 2 of the magnetic stripe, from EMV tag `5A` on a chip card, or by hand from the embossing. It is frequently *re*populated afterwards. Under network tokenisation the value the issuer sees is a device token or a merchant token rather than the funding account number, and the mapping back to the real number happens inside the scheme. Under point-to-point encryption the acquirer's decryption endpoint puts the clear PAN into DE2 for the first time at the edge of its own secure zone.

DE2 is also, quite often, absent. When the terminal sends full track 2 data in DE35, the account number is already inside that field, and many acquirer interfaces treat DE2 and DE35 as mutually exclusive rather than duplicating the number. A parser that assumes DE2 is always present will fail on perfectly valid magnetic-stripe traffic.

What goes wrong. Odd-length PANs in packed BCD need a padding nibble, and implementations differ on whether it goes at the front or the back; get it wrong and a nineteen-digit number arrives with a stray zero. Truncation at eighteen digits is a recurring defect in systems written when sixteen was universal. Storing DE2 anywhere it does not need to be is the single most common PCI DSS finding in card systems, and the field's presence in logs is the usual culprit. Finally, a mismatch between DE2 and the account number embedded in DE35 or in EMV tag `5A` is a hard decline at most issuers, because it is exactly what a tampered message looks like.

DE3, processing code

Format `n 6`, fixed, six digits in three pairs.

Positions 1 and 2 are the transaction type. Positions 3 and 4 are the account the money comes from. Positions 5 and 6 are the account it goes to. It is populated by the terminal and frequently translated by the acquirer.

The transaction types that matter in practice, as documented for Mastercard and Visa authorisation traffic:

Code	Transaction type
00	Goods or service purchase, point of sale only
01	Withdrawal or cash advance
09	Purchase with cash back
10	Account-funding transaction, incoming
11	Quasi-cash
17	Cash disbursement

Code	Transaction type
18	Card account verification
20	Purchase return or refund
21	Deposit
28	Payment transaction, incoming Mastercard OCT
26	Money transfer, incoming Visa OCT
30	Balance enquiry
40	Account transfer
50	Bill payment or payment to a third party

The account-type pairs are a much shorter list: `00` default, `09` other, `10` savings, `20` checking, `30` credit.

So `000000` is a purchase from the default account, `010100` a cash withdrawal from savings, `010300` a cash advance against credit, `200000` a refund, `300000` a balance enquiry.

The divergence between networks is public and it is not subtle. Not every network implements every transaction type, and several codes carry network-specific meaning: `17` marks the over-the-counter cash disbursement flag on Mastercard, while the equivalent condition on Visa is `01` combined with merchant category code 6010. A processing code is meaningful only against a named specification.

What goes wrong. Sending `000000` for what is commercially a refund produces a second debit rather than a credit, and the fix involves a manual adjustment and an unhappy customer. Sending a balance enquiry with a non-zero DE4 is rejected by some issuers and silently treated as a purchase attempt by others. Specifying an account type the cardholder does not have produces DE39 `52` or `53` — "no checking account", "no savings account" — which support teams reliably misdiagnose as an issuer outage, because the card works fine everywhere else. And an account-type pair of `00` does not mean "current account". It means "not specified, issuer decides", which is a different thing and behaves differently at an ATM.

DE4, amount, transaction

Format `n 12`, fixed, twelve digits, unsigned, no decimal separator.

The value is expressed in the minor units of the currency named in DE49. The decimal position is implied by the ISO 4217 exponent for that currency and is never transmitted. Two decimal digits for sterling (`826`) and US dollars (`840`); none for Japanese yen (`392`); three for Kuwaiti dinar (`414`). The maximum representable value is 999,999,999,999 minor units.

It is populated by the terminal. It is also, on occasion, populated by the issuer: a partial approval returns an amount lower than the amount requested, and the terminal must then either collect the difference by another means or reverse the whole thing.

DE4 does not travel alone. DE5 is the settlement amount, in the currency the acquirer settles in, added by the acquirer or the scheme. DE6 is the cardholder billing amount, in the currency of the cardholder's account, added by the issuer side. DE54 carries additional amounts, each with its own account type, amount type, currency code and sign, which is how ATM balances and cashback components travel. Four fields, four currencies, four opportunities for a conversion to be applied twice.

What goes wrong. Floating-point arithmetic anywhere near this field is a defect waiting for a specific input; amounts belong in integers of minor units or in decimal types with explicit scale. Hard-coding two decimal places breaks the moment the system sees yen or dinar. Estimated amounts create their own class of problem: a fuel pump authorises a nominal amount and clears the real one, a hotel

authorises a nightly estimate and increments it, and every one of those flows depends on the clearing amount being allowed to differ from the authorised amount within scheme-defined tolerances. And an account verification is DE3 transaction type `18` with DE4 set to zero, not a one-pound purchase followed by a reversal, although the industry spent a decade doing the latter.

DE7, DE12 and DE13, the three clocks

DE7 is transmission date and time, `n 10`, MMDDhhmmss, in Coordinated Universal Time, set by the sender of the individual message. It changes at every hop. It is not the time of the purchase; it is the time this particular leg was transmitted.

DE12 is local transaction time, `n 6`, hhmmss, in the local time of the point of sale, set by the terminal. DE13 is local transaction date, `n 4`, MMDD, likewise.

Together they are the reason chapter after chapter of scheme documentation insists on which clock a rule refers to. A cut-off expressed in DE7 is a different instant from one expressed in DE12 and DE13.

The edition trap is severe here. In the 1987 edition, DE12 is a time and DE13 is a date, and neither carries a year. In the 1993 edition DE12 becomes a combined local transaction date and time, twelve digits, and DE13 is redefined. A system that has to reconstruct a full timestamp from 1987-edition fields must supply the century and year from context, and the last days of December are where that inference fails: a transaction at 23:58 local on 31 December, transmitted at 00:02 UTC on 1 January, produces a DE7 whose date is in the following year and a DE13 that is not, and a naive reconstruction dates the purchase 364 days wrong.

What goes wrong, beyond the year. Terminal clocks drift, and nothing in the protocol corrects them; a terminal an hour out because nobody handled British Summer Time will put transactions in the wrong reconciliation window even though DE7 is perfect. Systems that use DE7 for duplicate detection and DE12 for reporting will produce two internally consistent reports that disagree with each other. And DE15, the settlement date, is a third date again, owned by the acquirer, and is the one that actually determines when money moves.

DE11, system trace audit number

Format `n 6`, fixed, six digits.

The STAN is assigned by the institution that originates the message. It is unique within that originator, within a processing day, and it wraps at 999999 back to 000001. It is not globally unique, it was never intended to be, and treating it as a transaction identifier is a design error that surfaces at scale.

Its purpose is narrow and important: it is part of the key that lets two adjacent systems match a response to a request. The full matching key is set by each scheme rather than by ISO, but it conventionally combines DE11 with DE7, DE32 (the acquiring institution identification code) and DE37.

The rules around it are strict in three places. A repeat message — origin digit changed from 0 to 1, producing `0101` from `0100` — must carry the same STAN as the original, because the whole point is for the receiver to recognise it as the same transaction rather than a new one. A reversal quotes the original STAN inside DE90. And an advice must not reuse a STAN that is still in the matching window for a different transaction.

What goes wrong. Two terminals sharing a STAN sequence, described in the plain version, is the classic. So is a STAN counter that resets on reboot rather than persisting, which produces collisions in a burst after every power cut. So is generating a fresh STAN on a retransmission, which converts

a timeout into a genuine double charge, because the issuer now sees two unrelated authorisation requests instead of one request and its repeat.

DE22, point of service entry mode

In the 1987 edition, `n 3`: two digits of PAN entry mode followed by one digit of PIN entry capability. It is set by the terminal, and it is the field that decides a large part of the commercial outcome of the transaction.

The PAN entry mode values, as documented against the Mastercard Customer Interface Specification and VisaNet authorisation messages:

Code	PAN entry mode
00	Entry mode unknown
01	Manual entry
02	Magnetic stripe
03	Bar code reader
04	Optical character reader
05	Integrated circuit card, contact chip
07	Contactless chip
09	Electronic commerce with a DSRP cryptogram in DE55 (Mastercard)
10	Electronic commerce, credential on file
79	Hybrid terminal failed to send a fallback transaction or failed to read the chip (Mastercard)
80	Chip failed at a chip-capable terminal, defaulted to magnetic stripe read
81	Electronic commerce with an authentication value in the UCAF (Mastercard)
82	Auto PAN entry via server
90	Magnetic stripe, full unaltered track data present
91	Contactless magnetic stripe, full unaltered track data present
95	Chip card with unreliable card verification value data (Visa only)

The third digit, PIN entry capability, is a much shorter list: `0` unspecified, `1` the terminal has PIN entry capability, `2` it does not, `3` software-based PIN entry on mobile, `8` the terminal has a PIN pad but it is not currently operative.

Note the pairs that are easy to confuse and expensive to confuse. `05` and `07` are chip transactions, contact and contactless, and both carry a cryptogram in DE55. `02`, `90` and `91` are magnetic-stripe transactions, and `91` is a contactless tap that produced stripe-equivalent data rather than a chip cryptogram — a different security posture entirely from `07` despite looking identical to the customer. `80` is chip fallback, and many issuers decline it by standing policy because it is the signature of a cloned card being presented at a terminal whose chip reader has been deliberately damaged.

The 1993 edition replaces all of this. DE22 becomes the point of service data code, twelve positions rather than three, each describing one property of the acceptance environment. American Express's Global Credit Authorization Guide, which is written to ISO 8583:1993, lays them out as: card data input

capability, cardholder authentication capability, card capture capability, operating environment, cardholder present, card present, card data input mode, cardmember authentication method, cardmember authentication entity, card data output capability, terminal output capability and PIN capture capability. Mastercard carries substantially the same information in DE61 in its 1987-format authorisation messages. Same facts, three different homes, depending on which specification you are implementing.

What goes wrong. Everything downstream of DE22 is a rule about liability and price. Sending `01` for a card that was physically present forfeits card-present interchange and card-present chargeback protection. Sending `07` when the terminal actually performed a magnetic-stripe read causes the issuer to look for a cryptogram in DE55 that is not there, producing a decline that looks like a network fault. Sending `05` while omitting DE55 entirely is the same failure in the other direction and is one of the most common findings in scheme certification testing. And the PIN entry capability digit must match reality: a terminal that claims capability `1` and then never sends a PIN block in DE52 will have its cardholder verification method results in DE55 questioned by the issuer.

DE37, retrieval reference number

Format `an 12`, fixed, twelve alphanumeric characters.

The RRN is assigned by the acquirer and is echoed unchanged by everyone downstream. It is the identifier with the longest life in the whole message: it persists from authorisation into the clearing record, into the settlement report, and into the retrieval requests and chargebacks that may follow weeks later. When a merchant telephones an acquirer about a specific transaction, the RRN is what the acquirer wants.

Its internal construction is not fixed by ISO 8583, which defines it only as a reference supplied by the system retaining the original source information. The schemes impose their own conventions. A Julian date prefix in the form YDDD — one digit of year, three digits of day of year — is widespread; Mastercard's IPM clearing format defines the first four positions of DE37 as the Julian processing date the acquirer assigned to the transaction. A common full convention extends that with a two-digit hour and a six-digit sequence, which is how the example `607614430041` decomposes: year 6, day 076 which is 17 March 2026, hour 14, sequence 430041.

What goes wrong. Regenerating the RRN on a retransmission destroys the receiver's ability to recognise the retransmission, with the same consequence as a regenerated STAN. Failing to echo it exactly — trimming trailing spaces, upper-casing, treating it as a number and dropping leading zeros — breaks reversal matching, because the reversal will not join to the original. Reusing RRNs within the retention window creates records that cannot be told apart in a dispute, and the party that cannot produce a unique reference is the party that loses the dispute.

DE38, authorisation identification response

Format `an 6`, fixed, six alphanumeric characters.

Generated by the issuer, returned in the response, printed on the receipt as the authorisation code, and required in the subsequent clearing record. It is the issuer's own handle on the approval it granted.

What goes wrong. Acquirers and gateways occasionally invent a value for DE38 when the issuer did not return one — for offline approvals, for stand-in processing, or simply as a placeholder. That value will not match anything at the issuer, and the clearing record carrying it is a candidate for a chargeback on the grounds of no valid authorisation. The reverse also happens: an approval in DE39 with DE38 blank or space-filled, which some acquirer platforms accept and then cannot present. And the field is

alphanumeric, so a system that stores it in a numeric column will corrupt any code containing a letter, which is most of them.

DE39, response code

Format `an 2` in the 1987 edition, `an 3` in 1993 and later. Set by the issuer, or by the scheme when the scheme answers on the issuer's behalf.

The 1987 vocabulary is a flat two-character list. The values that carry real operational weight are these:

Code	Meaning
00	Approved or completed successfully
01	Refer to card issuer
03	Invalid merchant
04	Pick-up
05	Do not honour
12	Invalid transaction
13	Invalid amount
14	Invalid card number, no such number
30	Format error
41	Lost card
43	Stolen card, pick-up
51	Not sufficient funds
54	Expired card
55	Incorrect personal identification number
57	Transaction not permitted to cardholder
58	Transaction not permitted to terminal
61	Exceeds withdrawal amount limit
62	Restricted card
65	Exceeds withdrawal frequency limit
91	Issuer or switch is inoperative
96	System malfunction

Codes `04`, `07`, `41` and `43` carry card-retention semantics: the terminal is being instructed to capture the card, which unattended terminals implement literally and countertop terminals implement as a message to the operator.

The 1993 edition replaces the flat list with a three-digit structure in which the leading digit identifies the class of outcome, so `000` is approved, `100` is a plain decline, `200` is a decline with card capture, `9xx` is could-not-process. The two systems coexist. A single transaction can pass through an interface using the 1987 codes and another using the 1993 codes, and the mapping between them is lossy in both directions. Response codes get a chapter of their own later in this volume, because the interesting question about them is not what they are but why so many of them are deliberately uninformative.

What goes wrong here is chiefly interpretation. DE39 `00` does not mean the transaction completed; it means the issuer approved it. The terminal can still fail, the chip can still decline in the second generate-AC after examining the issuer's response in DE55, and the customer can still walk away. A system that treats `00` as terminal state rather than as one input to terminal state will under-report reversals and over-report revenue.

DE41 and DE42, terminal and merchant identity

DE41, card acceptor terminal identification, is `ans 8`, fixed, left-justified and space-filled. DE42, card acceptor identification code, is `ans 15`, fixed, left-justified and space-filled. DE43, card acceptor name and location, is `ans 40` in the 1987 edition.

DE41 is assigned by the acquirer and configured into the terminal. It must identify one physical acceptance point uniquely within the card acceptor. DE42 is the acquirer's identifier for the merchant, and it is the field on which the acquirer's settlement, its interchange qualification and its risk monitoring all hang. DE43 is the human-readable name and town, and it is the raw material from which the descriptor on the cardholder's statement is built.

What goes wrong. Duplicate DE41 values across sites — the same terminal identifier deployed to two shops after a hardware swap — make it impossible to attribute a transaction to a location, which matters both for the merchant's own reporting and for fraud investigation. A mismatch between the DE42 in the authorisation and the DE42 in the clearing record breaks the acquirer's matching and produces unmatched clearing, which is expensive to resolve manually. And DE43 is quietly one of the highest-leverage fields in the entire message: a merchant whose descriptor reads as an unrecognisable holding-company name generates "I do not recognise this transaction" disputes at a multiple of the rate of one whose descriptor reads as the shop's trading name and town. That is a chargeback cost caused entirely by forty characters of free text.

DE49, currency code, transaction

Format `n 3` in general use, defined in the 1987 edition as three characters which may be alphabetic or numeric. In practice every major scheme requires the ISO 4217 numeric code: `826` sterling, `840` US dollar, `978` euro, `392` yen, `356` Indian rupee.

It is set by the terminal, from its own configuration rather than from the card. It says what currency DE4 is denominated in, and by implication where the decimal point sits, because the ISO 4217 minor-unit exponent is not carried in the message.

Its companions are DE50, the settlement currency, and DE51, the cardholder billing currency. A cross-border purchase can legitimately involve three different currency codes in one message chain, and dynamic currency conversion adds a fourth relationship, because the cardholder is offered a rate at the terminal and the transaction is then presented in the cardholder's own currency with the conversion already applied.

What goes wrong. Sending the alphabetic code `GBP` where the interface expects `826` is a format error, and it is a common first-day defect because the alphabetic form is what humans write. Defaulting to `840` in a terminal configuration template is worse, because it does not fail: it produces a valid message denominated in a currency the shop does not trade in. Assuming two decimal places universally corrupts every yen and every dinar amount by two orders of magnitude in one direction or three in the other. And a currency code that differs between the authorisation and the clearing record will not match, which the acquirer will see as an unmatched presentment.

DE55, integrated circuit card data

Format `ans...999` in the 2023 edition of the standard; schemes cap it far lower, commonly at 255 bytes with a three-digit length prefix. It is a format inside a format, and it is parsed by entirely different rules from the message carrying it.

The 2023 standard is explicit that its own composite-dataset structure does not apply to the ICC-related data element, because the linking of related sub-elements there follows ISO/IEC 7816-6. DE55 is there-

fore a sequence of EMV tag-length-value objects, with tags of one or two bytes, lengths in BER-TLV form, and values whose interpretation is defined in EMV Book 3 rather than in ISO 8583 at all.

The tags that carry the transaction, on the way to the issuer:

Tag	Name	Origin
9F26	Application cryptogram	Card
9F27	Cryptogram information data	Card
9F10	Issuer application data	Card
9F36	Application transaction counter	Card
82	Application interchange profile	Card
5F34	Application PAN sequence number	Card
9F09	Application version number	Card
95	Terminal verification results	Terminal
9A	Transaction date	Terminal
9C	Transaction type	Terminal
9F02	Amount, authorised	Terminal
9F03	Amount, other	Terminal
5F2A	Transaction currency code	Terminal
9F1A	Terminal country code	Terminal
9F33	Terminal capabilities	Terminal
9F34	Cardholder verification method results	Terminal
9F35	Terminal type	Terminal
9F37	Unpredictable number	Terminal
9F41	Transaction sequence counter	Terminal
84	Dedicated file name	Card

And on the way back, from the issuer:

Tag	Name	Origin
91	Issuer authentication data	Issuer
71	Issuer script template 1	Issuer
72	Issuer script template 2	Issuer

The division of labour is the point. The card supplies what only the card knows and signs it; the terminal supplies the context the card was asked to sign over; the issuer recomputes the cryptogram from the same inputs using the same key and either gets the same answer or does not. The reason 9F02, 5F2A, 9F1A and 9F37 must appear in DE55 even though the amount, the currency and the country are already elsewhere in the message is that these are the values the card signed. If DE4 says one amount and tag 9F02 says another, the cryptogram will validate against 9F02 and the issuer now knows that something between the chip and it altered the amount.

What goes wrong. Truncation at a scheme's byte limit silently removes trailing tags, and if 9F10 is one of them the issuer cannot derive the session key and cannot validate the cryptogram, producing a decline with no informative code. Sending the field as ASCII hexadecimal where the interface expects binary, or the reverse, doubles or halves every length and yields a parse failure at the first tag. Re-ordering tags is permitted by BER-TLV and is nonetheless a frequent cause of failure in implementations that parse by position rather than by tag. Constructed tags — templates whose value is itself a sequence of TLVs — need recursive parsing, and a flat parser that meets tag 71 will read an issuer

script as a single opaque blob. And omitting DE55 entirely on a transaction whose DE22 claims `05` or `07` is the certification failure named earlier: the message says a chip was used and contains no evidence that one was.

The fields in one table

DE	Name	Format	Written by	Echoed unchanged	Principal failure
2	Primary account number	n..19 LLVAR	Terminal, may be tokenised	Usually	Truncation, BCD padding, mismatch with DE35
3	Processing code	n 6	Terminal, may be translated	Yes	Refund sent as purchase; wrong account type
4	Amount, transaction	n 12	Terminal; issuer on partial approval	Usually	Wrong implied exponent; floating point
7	Transmission date and time	n 10	Each sender, UTC	No, changes per hop	Confused with transaction time
11	System trace audit number	n 6	Message originator	Within a leg	Collision; regenerated on retry
12	Local transaction time	n 6 (1987)	Terminal clock	Yes	Clock drift; no year
13	Local transaction date	n 4 (1987)	Terminal clock	Yes	Year-end rollover
22	Point of service entry mode	n 3 (1987)	Terminal	Yes	Misstated entry mode shifts liability
37	Retrieval reference number	an 12	Acquirer	Yes, exactly	Regenerated or reformatted; disputes fail
38	Authorisation identification response	an 6	Issuer	Into clearing	Invented locally; lost in numeric storage
39	Response code	an 2 (1987), an 3 (1993)	Issuer or scheme	Response only	Treated as final transaction state
41	Card acceptor terminal identification	ans 8	Acquirer, configured in terminal	Yes	Duplicated across sites
42	Card acceptor identification code	ans 15	Acquirer	Yes	Auth and clearing disagree
49	Currency code, transaction	n 3	Terminal configuration	Yes	Alphabetic code; default 840

DE	Name	Format	Written by	Echoed unchanged	Principal failure
55	Integrated circuit card data	ans…999, TLV	Card and terminal; issuer in response	Yes, byte for byte	Truncation; encoding; flat parsing

What this table leaves out

Fifteen fields is enough to move money and nowhere near enough to run an acquiring business. DE35 and DE45 carry track 2 and track 1 data. DE25 carries the point of service condition code, which says whether this is a normal sale, a recurring payment, a mail order or a repeat. DE32 and DE33 identify the acquiring and forwarding institutions and are part of the matching key. DE48 is additional data, private, and is where every scheme has quietly parked the things ISO 8583 has no room for. DE52 carries the enciphered PIN block, formatted per ISO 9564-1. DE54 carries additional amounts. DE61 and DE62 are private fields where Mastercard puts point of service data and transaction-life-cycle identifiers. DE64 and DE128 carry message authentication codes. DE90 carries the original data elements that let a reversal find what it is reversing.

Each has the same three questions attached to it: what it holds, who writes it, and what breaks when it is wrong. The habit is worth more than the list, because the list changes between schemes and the habit does not.

The next chapter takes a single authorisation and walks it end to end, byte by byte, with every field in this chapter populated and every hop accounted for. It is the same purchase, the same bookshop and the same £42.50, and by the end of it there will be nothing left in the message you cannot name.

26.98 Common wrong ideas

Wrong: every desk copies the boxes it did not write, so the message the issuer receives is the message the terminal sent. **Right:** acquirers renumber DE11, may replace DE2 with a network token, translate processing codes and add DE5 and DE50, so debugging across an acquirer boundary means comparing two different messages that describe one purchase.

Wrong: the field number determines the meaning. **Right:** ISO 8583 fixes the number, the name and the format but not the vocabulary inside; the complete question is which field, on which scheme, in which edition, in which direction.

Wrong: the issuer writes DE38 and DE39 and copies the rest. **Right:** the response can also carry a reduced amount for a partial approval, balances in DE54, issuer authentication data and scripts in DE55, and private data in DE48 that changes what the terminal must display.

Wrong: DE11 identifies a transaction. **Right:** the STAN is unique only within one originator within a processing day and wraps at 999999; it is one part of a matching key that conventionally also uses DE7, DE32 and DE37.

Wrong: a retransmission should carry a fresh STAN and a fresh retrieval reference number. **Right:** a repeat carries the same values with the MTI's origin digit changed from 0 to 1; regenerating them converts a timeout into a genuine double charge.

Wrong: amounts have two decimal places. **Right:** the exponent comes from the ISO 4217 entry for the currency in DE49 — none for the yen, three for the Kuwaiti dinar — and hard-coding two corrupts those amounts by orders of magnitude.

Wrong: `07` and `91` in DE22 are much the same, because the customer tapped the card either way. **Right:** `07` is a contactless chip transaction carrying a cryptogram in DE55 and `91` is a contactless read that produced stripe-equivalent data, which is a different security posture entirely.

Wrong: DE38 can be filled in locally when the issuer did not return one. **Right:** an invented authorisation code matches nothing at the issuer, and the clearing record carrying it is a candidate for a chargeback on the grounds of no valid authorisation.

Wrong: DE43 is cosmetic free text. **Right:** it is the raw material for the statement descriptor, and a descriptor reading as an unrecognisable holding company generates disputes at a multiple of the rate of one reading as the shop's trading name and town.

Wrong: DE55 can be parsed by position like the rest of the message. **Right:** it is BER-TLV under ISO/IEC 7816-6, where tags may legitimately be re-ordered and templates nest, so a flat positional parser will read an issuer script as a single opaque blob.

26.99 Chapter summary in 20 lines

1. A message that will not parse is a cheap problem; the expensive failures are the ones where every field parses perfectly and one value is simply the wrong kind of true.
2. There are 128 numbered data elements, a working card authorisation is built from about twenty of them, and about fifteen carry the weight.
3. The plain rule is that each desk along the chain writes its own boxes and copies the rest, with the till writing the facts of the sale, the acquirer the facts of the journey, and the issuer exactly two boxes.
4. That rule is the most useful lie in the chapter, because acquirers renumber the STAN, tokenise the account number, translate processing codes and add currency fields the terminal never sent.
5. A field definition is a character set and a length rule, where a bare number means fixed, two dots mean a two-digit length prefix and three dots mean a three-digit one.
6. Fixed numeric fields are right-justified and zero-filled while fixed alphanumeric fields are left-justified and space-filled, and that asymmetry causes real matching defects.
7. DE2 carries the account number, is frequently repopulated by tokenisation or point-to-point decryption, and is quite often absent altogether when full track 2 data travels in DE35 instead.
8. DE3 is three pairs — transaction type, account from, account to — and the transaction types diverge between networks publicly and expensively.
9. DE4 is twelve unsigned digits of minor units with no decimal separator, and the decimal position is implied by the ISO 4217 exponent of the currency named in DE49.
10. DE7, DE12 and DE13 are three different clocks — the sender's transmission time, the terminal's local time and the terminal's local date — and DE15, the settlement date, is a fourth date that determines when money moves.
11. The 1987-edition date fields carry no year, so a purchase at 23:58 local on 31 December transmitted at 00:02 UTC can be reconstructed 364 days wrong.
12. DE11 is a correlation key unique only within one originator for one processing day, and treating it as a transaction identifier is a design error that surfaces at scale.
13. DE22 states how the card was read, and because liability rules and interchange hang on it, one wrong digit is a commercial event rather than a technical one.
14. DE37 has the longest life in the message, running from authorisation into clearing, settlement and any dispute, and it must be echoed exactly, leading zeros and trailing spaces intact.
15. DE38 is the issuer's own handle on the approval it granted, it is alphanumeric so a numeric column will corrupt it, and it must never be invented downstream.

16. DE39 is two characters in the 1987 edition and three in 1993, the two vocabularies coexist, and the mapping between them is lossy in both directions.
17. DE41, DE42 and DE43 identify the terminal, the merchant and the statement descriptor, and forty characters of free text in DE43 drive a measurable share of disputes.
18. DE49 fixes the currency and by implication the decimal position, and a configuration template defaulting to `840` produces a valid message denominated in a currency the shop does not trade in.
19. DE55 is a format inside a format, a sequence of EMV tag-length-value objects whose amount, currency and country are the values the card actually signed, so a disagreement with DE4, DE49 or the country code tells the issuer that something altered the transaction.
20. Everything else in an authorisation is an unauthenticated assertion by whoever sent it, which is why card payments are governed by liability rules and scheme membership rather than by proof.

Chapter sources: ISO 8583:1987 and ISO 8583:1993 data element and response code tables as consolidated in the Wikipedia ISO 8583 article and cross-checked against ISO catalogue entries for ISO 8583:1993 and ISO 8583-1:2003, including clause 6.4.9 and Annex A.16 on the point of service data code; ISO 8583:2023 on the ICC data element and ISO/IEC 7816-6; SoFi Tech Solutions (Galileo) DE003 and DE022 code references, stated to be derived from the Mastercard Customer Interface Specification of 6 February 2024 and VisaNet Authorization-Only Online Messages of 15 October 2023; Mastercard IPM Clearing Formats on the Julian processing date in DE37; American Express Global Credit Authorization Guide, ISO 8583:1993 Version 1, October 2019, for the twelve positions of the point of service data code; Worldpay ISO 8583 Reference Guide v2.50 for acquirer-interface field attributes; EMV Integrated Circuit Card Specifications Book 3 tag definitions as tabulated by EMV Lab and EFTlab; ISO/IEC 7812-1, ISO/IEC 7813, ISO 4217 and ISO 9564-1.

27

CHAPTER

A Transaction, Narrated

27.0 What this chapter gives you

1. You will be able to narrate a contactless card authorisation from card presentation to receipt, naming every hop and what each party does in it.
2. You will be able to say where the 1.8 seconds actually goes, and why the printer is a larger component than the network, the acquirer, the scheme and the issuer put together.
3. You will be able to explain why the beep at 320 milliseconds is not an approval, and why nothing has left the shop when it sounds.
4. You will be able to say why the issuer's response cryptogram in tag `91` is discarded on a contactless tap and delivered to the card on a contact chip transaction, and what that costs in security and in time.
5. You will be able to expand a primary bitmap such as `723C648108E08201` by hand and list the twenty elements it declares.
6. You will be able to name which fields the acquirer adds, which it rewrites and which it passes through unchanged, and why the ones it passes through are the ones every downstream reconciliation depends on.
7. You will be able to describe how an issuer validates a cryptogram — master key, card key, session key, recomputation — and state the three things a match proves that nothing else in the message can.
8. You will be able to explain why an approval reduces an available balance and touches no ledger anywhere, and why no money has moved at 1,800 milliseconds.
9. You will be able to say what a terminal must do when no response arrives, and why retrying with a fresh trace number is the wrong answer.
10. You will be able to reason about the tail rather than the median — mobile data, cold DNS, PIN entry, stand-in processing — because that is where merchant contracts and scheme rules are written.

Everything in this volume so far has been apparatus. The four-party model gave you the cast. ISO 8583 gave you the envelope. The bitmap gave you the index. The fields gave you the contents. None of it has yet moved.

This chapter moves it. One purchase, one card, one terminal, one acquirer, one scheme, one issuer, and a stopwatch running from the moment the card enters the reader's field to the moment the receipt is torn off. Every message is written out. Every field is named and populated. Every hop is timed, and at the end the timings are added up and compared against the number everyone quotes and nobody audits, which is that a card payment takes about two seconds.

The purchase is the same one used in the previous two chapters, so that nothing has to be relearned: a book costing £42.50, bought at a small bookshop in London on Tuesday 17 March 2026, at eight minutes past half past two in the afternoon, paid for by tapping a contactless Visa debit card on a countertop terminal. The card number, `4111111111111881`, is from the test range every payments engineer recognises, and it is used here deliberately so that no real account appears in print.

Two warnings. The millisecond figures are a representative budget for a well-built United Kingdom deployment on a fixed broadband line, not a measurement of a particular system; the shape of the budget is the point, and the shape is stable even when the numbers move. And several values in the message are marked illustrative, because inventing an issuer's proprietary bytes and presenting them as fact is exactly what this book refuses to do.

The plain version

Imagine a shopkeeper who cannot accept your promise to pay, only your bank's.

You want a book. The shopkeeper wants £42.50. He does not know you, and he cannot look inside your bank account. So before he will let you leave with the book, he wants a sentence from your bank that says: *we have seen this, we have put £42.50 aside, and we will pay the shop*. Getting that one sentence, from a bank that might be four hundred miles away, is the entire event.

The trouble is that the shopkeeper cannot telephone your bank. There are thousands of banks and millions of shops, and nobody could keep that many phone numbers. So the question travels along a fixed chain of four parties, each of whom only ever talks to the party next to them.

The **card** is the first. It is not a piece of plastic with a number on it; it is a tiny computer that can perform a calculation nobody else can perform, because it holds a secret only it and your bank know.

The **till** — properly, the card terminal — is the second. It knows the price, the date, the shop, and how the card was presented. It cannot decide anything.

The **shop's bank**, the acquirer, is the third. It has an account for the shopkeeper, a direct line into the card networks, and a filing cabinet in which this transaction will live for years.

The **card company** in the middle, Visa in this case, is the fourth. It is a switchboard. Its job is to look at the front of the card number, work out which bank issued it, and pass the question on.

And **your bank**, the issuer, is the fifth and the only one that can say yes.

The question goes card, till, shop's bank, card company, your bank. The answer comes back your bank, card company, shop's bank, till. Nine steps in all, and it is over before you have finished putting your card away.

Where the time actually goes

Here is the same purchase with a stopwatch on it. Start the clock the instant the card comes within range of the reader.

What is happening	Time it takes	Clock reads
The card and the till talk to each other	0.32 s	0.32 s
The till writes out the question and sends it	0.08 s	0.40 s

What is happening	Time it takes	Clock reads
The shop's bank reads it, stamps it, forwards it	0.07 s	0.47 s
The card company works out which bank, forwards it	0.06 s	0.53 s
Your bank checks the card is genuine and the money is there	0.25 s	0.78 s
The answer travels all the way back to the till	0.12 s	0.90 s
The till reads the answer and puts APPROVED on the screen	0.03 s	0.93 s
The till prints the receipt	0.87 s	1.80 s

Read that table twice, because it says something nobody expects.

The part everyone imagines is slow — the message flying across the country to your bank and back — is the smallest number in it. All four outward journeys and all four return journeys, added together, come to about a tenth of a second. The computers at the other end are not slow either; your bank's whole decision takes a quarter of a second.

The two big numbers are the ones nobody thinks about. Nearly a fifth of the time is the card and the till talking to each other over a few centimetres of air before anything leaves the shop at all. And almost half of it is the printer.

That is worth saying plainly. In a modern card payment, the single slowest component is a strip of paper being pushed past a hot line of dots.

Three things that are not what they look like

The beep is not the approval. The terminal beeps and shows a green light at about 0.32 seconds, when the card has finished its part and can be taken away. At that moment nothing has left the shop. No bank has seen anything. The beep means "I have got what I need from your card", and people read it as "you have paid", which is why customers walk off and why terminals then have to display a decline to somebody's back.

Your bank is not looking up your balance and subtracting. It does two quite different things. It checks that the card is genuine, by redoing the calculation the chip performed and seeing whether it gets the same answer. And it checks the money is available, by looking at your balance and at everything else it has already promised out of it today. Then it *earmarks* £42.50.

No money has moved. At the end of all 1.8 seconds, the shopkeeper has a promise and you have a hold on your account. The £42.50 has not left your bank and it has not arrived at the shopkeeper's. That happens later, in a batch, overnight, and it is the subject of a later chapter. All this chapter's machinery exists to produce one six-character code and a two-character yes.

Where the plain version stops being true

There is no such thing as "the message". The plain version speaks of a question travelling along a chain, as though one document were being carried from hand to hand. Nothing of the sort happens. The terminal produces a message in one format; the acquirer parses it, discards some of it, adds fields the terminal never sent, rewrites fields the terminal did send, and produces a different message in a different format; the scheme does the same again on the way to the issuer. In the United Kingdom

this is unusually visible, because the terminal-to-acquirer link has historically not used ISO 8583 at all. It uses Standard 70, the Card Acceptor to Acquirer Interface Standards maintained on the industry's behalf by UK Payments Administration and its predecessors, whose Book 2 covers real-time systems and whose Book 3 covers post-event systems. A British terminal typically speaks Standard 70 to its acquirer, and the acquirer speaks ISO 8583 to the scheme. Two formats, two vocabularies, one purchase. When you debug across that boundary you are comparing two different documents, and the correct question is never "what did the message say" but "which message, at which hop".

The card is gone before the answer exists. In the plain version the chain is symmetrical: the question goes up and the answer comes back down the same path. For a contact chip transaction that is nearly true, because the card stays in the slot and receives the issuer's reply. For a contactless tap it is false in a way that matters. The card finishes its work at around 320 milliseconds and is back in your pocket by 500. The issuer's reply, which contains a cryptogram the card could in principle have verified, arrives at the terminal at around 900 milliseconds with nothing to deliver it to. The terminal discards it. This is not a defect; contactless was designed this way on purpose, trading one protection for the speed that makes tapping tolerable at a ticket barrier. But it means the security model of a tap and the security model of an inserted chip are genuinely different, and the difference is invisible to everyone in the shop.

These are not constants; they are distributions with a long tail. Every number in the plain version's table is roughly a median. The engineering discipline of payments is not about medians. An acquirer's contract with a merchant, and a scheme's rules for an issuer, are written about the far end of the distribution: how often does this exceed one second, exceed three, time out altogether. The interesting behaviour is all in that tail, and it is where the protocol's timeouts live. Mastercard's published documentation for one of its processing products states that the issuer must respond within four seconds before that processor stands in on the issuer's behalf, and that the processor in turn has a maximum of seven seconds to respond to Mastercard before Mastercard stands in. Those numbers exist because the tail exists. A system built to the median will meet them comfortably in testing and breach them on the first Saturday before Christmas.

Nothing in the narrative is the transaction. The receipt feels like the end. It is the end of the authorisation and the beginning of everything else. The clearing record has yet to be created, the interchange has yet to be assessed, the funds have yet to settle, the hold has yet to be released or converted, and the transaction remains disputable for months. What has been narrated here is the fastest and most reliable part of the life of a payment, and it is the part on which the industry's public reputation rests, which is precisely why the slow and expensive parts that follow are so poorly understood.

The technical version

The setting

A countertop terminal in a bookshop in London. It is EMV Level 1 and Level 2 approved, contact and contactless, PCI PTS approved, with a thermal printer and a wired Ethernet connection to a business broadband line. It is configured for a single United Kingdom acquirer.

The card is a consumer Visa debit card, issued in the United Kingdom, dual interface, supporting Visa's contactless application. Its application identifier is `A0000000031010`.

The date is Tuesday 17 March 2026, day 076 of the year. British Summer Time does not begin until Sunday 29 March 2026, so on this date the United Kingdom is on Greenwich Mean Time and local

time equals Coordinated Universal Time exactly. This is a convenience that disappears twelve days later, and the disappearance breaks a great many reconciliation jobs each spring.

The card is presented at 14:32:04.800 local. That instant is T+0 for everything that follows.

The transaction is £42.50. The relevant United Kingdom contactless limits, as of writing in 2026, are a single-transaction ceiling of £100, in force from 15 October 2021 until 18 March 2026, after which the Financial Conduct Authority permitted firms to remove the mandatory cap; and a cumulative threshold of £300, or five consecutive contactless transactions, since the last application of strong customer authentication, the cumulative figure having been raised from £130 by the FCA in policy statement PS21/2. On 17 March 2026 the £100 ceiling still applies, and £42.50 is inside both, so no cardholder verification will be requested.

Phase one, T+0 to T+320 ms: the card in the field

The reader has been polling continuously. The card's antenna couples to the reader's field at 13.56 MHz under ISO/IEC 14443, draws its power from that field, and answers.

Elapsed	Step
0 to 35 ms	Field detection, ISO/IEC 14443-3 anticollision, answer to select
35 to 65 ms	SELECT of the proximity payment system environment, `2PAY.SYS.DDF01`
65 to 95 ms	SELECT of the application identifier `A0000000031010`
95 to 255 ms	GET PROCESSING OPTIONS, carrying the PDOL; card computes and returns the cryptogram
255 to 300 ms	READ RECORD for the remaining card data
300 to 320 ms	Kernel completes, terminal beeps and shows the green light

Three details of that sequence do real work.

The terminal does not ask for the payment application by name. It selects the proximity payment system environment, `2PAY.SYS.DDF01`, reads back the list of applications the card supports, and picks one by its own configured priority. This is why a card carrying two applications can behave differently at two terminals that are otherwise identical.

The GET PROCESSING OPTIONS command is where the transaction is actually committed. The card publishes a processing options data object list, the PDOL, naming the values it wants from the terminal. For Visa's contactless kernel this list includes the terminal transaction qualifiers in tag `9F66`, the authorised amount in `9F02`, the other amount in `9F03`, the transaction currency in `5F2A`, the terminal country in `9F1A`, the unpredictable number in `9F37` and the transaction type in `9C`. The terminal concatenates those values in the order the card asked for them and sends them. The card signs over them. That is the whole basis of the cryptogram's meaning: it attests to *these values*, in this order, at this moment.

And in Visa's contactless EMV mode the cryptogram comes back in the response to GET PROCESSING OPTIONS. There is no separate GENERATE AC exchange, and therefore no second exchange in which the issuer's reply could be handed to the card. This is the single design decision that makes contactless fast and that makes the issuer's response cryptogram undeliverable.

At 320 milliseconds the terminal beeps and the customer takes the card away. The terminal now holds an authorisation request cryptogram, tag `9F26`, eight bytes, computed by the chip using a key derived from a master key held by the issuer, together with the transaction counter in `9F36` that makes this cryptogram different from every other one this card has produced.

Phase two, T+320 to T+400 ms: the terminal builds the request

The terminal now performs terminal action analysis. It examines the terminal verification results in tag `95`, five bytes of flags, and the cryptogram information data in tag `9F27`, one byte, whose value indicates an authorisation request cryptogram. It applies its floor limit, which for a United Kingdom contactless acceptance point is effectively zero, and concludes what it already knew: this transaction must go online.

It stamps the local transaction time and date. Its clock reads 14:32:05.120, so data element 12 becomes `143205` and element 13 becomes `0317`. It assigns its own system trace audit number, `000982`, from a counter persisted to non-volatile storage. It composes the message, applies whatever point-to-point encryption its acquirer requires, and hands it to the network stack.

At 375 milliseconds the request goes out over an already-established TLS session. A terminal that keeps a persistent, authenticated connection to its acquirer pays the handshake cost once, at sign-on. A terminal that opens a fresh connection per transaction pays it every time: a TLS 1.3 handshake costs one round trip before application data can flow, TLS 1.2 costs two, and both cost certificate validation on a processor that is not fast.

The one-way transit to the acquirer's front end, over business fibre to a London data centre, takes about 25 milliseconds including the shop's own router. Arrival at 400 milliseconds.

The message as it leaves the terminal

Presented here in ISO 8583 terms for comparability with what follows, although a British terminal would more usually present it in Standard 70 form.

MTI `0100`, authorisation request: version 0 for the 1987 edition, class 1 for authorisation, function 0 for request, origin 0 for acquirer.

Primary bitmap `723C448000E08200`, declaring sixteen elements present.

DE	Value	Note
2	`16 4111111111111881`	LLVAR, two-digit length then the PAN
3	`000000`	Purchase, from default account, to default account
4	`000000004250`	4,250 minor units
7	`0317143205`	MMDDhhmmss, UTC, stamped by the terminal
11	`000982`	The terminal's own trace number
12	`143205`	Local transaction time
13	`0317`	Local transaction date
14	`2809`	Expiry, YYMM
18	`5942`	Merchant category code, book stores, per ISO 18245
22	`071`	Contactless chip, terminal has PIN entry capability
25	`00`	Point of service condition code, normal transaction
41	`TERM0042`	Eight characters, left justified, space filled
42	`KEDBYTEBOOKS01`	Fifteen characters, one trailing space
43	`KEDBYTE BOOKS  LONDON  GB`	Forty characters
49	`826`	ISO 4217 numeric for sterling

DE	Value	Note
55	160 bytes of BER-TLV	See below

Data element 43 deserves a note. Its forty characters are subdivided into name, town and country, but the subdivision is scheme-defined and the schemes do not agree on the boundaries; the value above is padded on the convention of twenty-five characters of name, thirteen of town and two of country code.

Two coincidences in the message are not coincidences. Data element 49 carries `826` for sterling and data element 19, which the acquirer will add shortly, carries `826` for the United Kingdom. ISO 4217's numeric currency codes were assigned to match ISO 3166-1's numeric country codes wherever a country has its own currency, so the sterling code and the British country code are the same three digits by construction. Systems that validate one against the other by equality will work perfectly in Britain and fail in the eurozone.

Data element 55, in full

The acquirer interface used here encodes data element 55 as a three-character ASCII total length followed by binary tag-length-value data, a convention documented in Worldpay's Lync ISO 8583 message specification and typical of the industry. The tag-length-value structure is BER-TLV as defined for EMV, not the composite-dataset structure that ISO 8583:2023 defines for its own fields; the 2023 standard explicitly exempts the integrated-circuit-card element and defers to ISO/IEC 7816-6.

Twenty-four objects, 160 bytes:

Tag	Bytes of value	Content	Exact here?
`9F26`	8	Application cryptogram, the ARQC	Illustrative
`9F27`	1	Cryptogram information data, indicating an ARQC	Exact
`9F10`	7	Issuer application data	Illustrative, card profile dependent
`9F37`	4	Unpredictable number	Illustrative
`9F36`	2	Application transaction counter	Illustrative
`95`	5	Terminal verification results	Illustrative
`9A`	3	Transaction date, `260317`	Exact
`9C`	1	Transaction type, purchase	Exact
`9F02`	6	Amount authorised, `000000004250`	Exact
`9F03`	6	Amount other, `000000000000`	Exact
`5F2A`	2	Transaction currency code, `0826`	Exact
`82`	2	Application interchange profile	Illustrative
`9F1A`	2	Terminal country code, `0826`	Exact

Tag	Bytes of value	Content	Exact here?
9F33	3	Terminal capabilities	Illustrative
9F34	3	Cardholder verification method results	Illustrative
9F35	1	Terminal type	Illustrative
84	7	Dedicated file name, A0000000031010	Exact
9F09	2	Application version number	Illustrative
9F41	2	Transaction sequence counter	Illustrative
5F34	1	Application PAN sequence number	Illustrative
9F6C	2	Card transaction qualifiers	Illustrative
9F66	4	Terminal transaction qualifiers	Illustrative
9B	2	Transaction status information	Illustrative
57	19	Track 2 equivalent data	Illustrative

Note tag 57. On a chip transaction the track 2 equivalent data lives inside data element 55, and the acquirer interface used here specifies that where tag 57 is present it is the source of track data. That is why the message carries no data element 35: duplicating it would create two sources of truth for the same account number.

Note also 9F02, 5F2A and 9F1A. The amount, the currency and the country each appear twice in this message: once in the ISO 8583 elements 4, 49 and 19, and once inside the cryptogram's input. Only the second copy is protected. If those two copies disagree, the issuer knows that something between the chip and itself altered the transaction, and it will decline. This is the only integrity guarantee in the entire message that does not depend on trusting the sender.

The same acquirer specification states that data element 55 is required on 0100, 0200, 0220 and 0400 messages and on stand-in messages, and must not be sent to the network for manual or fallback transactions. That last clause is the certification failure named in the previous chapter, in its formal wording.

Total message length as it leaves the terminal, in the ASCII-and-hexadecimal representation used here: 484 characters, of which 323 are data element 55.

Phase three, T+400 to T+470 ms: the acquirer

The acquirer's front end does more in seventy milliseconds than any other party in the chain.

It terminates the TLS session and parses the message, defensively: length prefixes are the standard attack surface of this protocol and a hostile or broken length is the first thing a front end must survive. It performs duplicate detection against the terminal's trace number and terminal identifier over a short window, because the commonest cause of a double charge is not fraud but a terminal that retried.

It decrypts. If the account data arrived under a point-to-point encryption scheme, the ciphertext goes to a hardware security module — a tamper-responsive device holding keys that cannot be read out of it in the clear — and comes back as a primary account number inside the acquirer's secure zone. That round trip is a few milliseconds and it is the moment at which the acquirer's PCI DSS scope is defined.

Had this been a PIN transaction, the same module would also have translated the PIN block from the key shared with the terminal into the key shared with the scheme, without the PIN ever existing in plaintext outside a tamper boundary.

It applies its own risk rules: velocity by terminal, by card, by merchant, and amount thresholds. A small proportion of traffic dies here and never reaches a scheme.

Then it writes its fields.

- Data element 7 is restamped with the acquirer's own transmission time.
- Data element 11 is replaced. The terminal's `000982` is meaningful only within that terminal; the acquirer substitutes `004417` from a counter unique within the acquirer for the processing day.
- Data element 19 is set to `826`, the acquiring institution's country.
- Data element 32, the acquiring institution identification code, is populated with the acquirer's scheme-assigned identifier. This is `060000` in the worked example and is illustrative.
- Data element 37 is created: `607614430041`, decomposing as year digit 6, day 076, hour 14 and a sequence. This value will now outlive everything else in the message.
- Data element 15, the settlement date, may be set here, determining which business day this transaction belongs to regardless of what the clocks say.
- Data element 64, an eight-byte message authentication code, is computed over the assembled message under a key shared with the scheme.

At 455 milliseconds the acquirer's routing logic reads the leading digits of the account number, identifies a Visa product, and hands the message to its scheme endpoint. Twenty digits of routing decision, made on a table that is refreshed daily and that every acquirer in the world maintains separately.

Fifteen milliseconds of transit. Arrival at the scheme, 470 milliseconds.

The message as it leaves the acquirer

MTI `0100`. Primary bitmap `723C648108E08201`.

That hexadecimal string is worth expanding by hand, because it is the whole argument of the bitmap chapter reduced to eight bytes.

Byte	Hex	Bits	Elements present
1	72	01110010	2, 3, 4, 7
2	3C	00111100	11, 12, 13, 14
3	64	01100100	18, 19, 22
4	81	10000001	25, 32
5	08	00001000	37
6	E0	11100000	41, 42, 43
7	82	10000010	49, 55
8	01	00000001	64

Twenty elements declared in sixty-four bits. Bit 1 is zero, so there is no secondary bitmap and nothing above element 64 is present.

The message now runs to 523 characters. Compared with what the terminal sent, four elements have been added (19, 32, 37, 64) and two have been rewritten (7, 11). Sixteen have been passed through unchanged, and it is on those sixteen that every downstream reconciliation depends.

Phase four, T+470 to T+525 ms: the scheme routes it

The scheme is a switch, and its work is narrow, fast and unglamorous.

It validates the message authentication code in data element 64 against the key it shares with this acquirer. It reads the issuer identification number from the leading digits of data element 2 — under ISO/IEC 7812-1 as revised in 2017 these are eight digits, though six-digit ranges remain in wide use and both must be supported — and looks up the destination in a routing table.

If the account number in data element 2 is a network token rather than a funding account number, the scheme's token vault resolves it here, and the payment account reference, a value that stays constant across every token issued against the same underlying account, travels alongside so that the merchant and the issuer can still tell that two different tokens are the same customer. In the acquirer interface used above this reference occupies data element 56.

The scheme scores the transaction against its own risk models, starts a response timer, and records enough to build a clearing record later and to answer on the issuer's behalf if the issuer does not reply. Then it sends. VisaNet, by Visa's own published figures, is engineered to handle more than 65,000 transaction messages per second; this is one of them, and it occupies the switch for perhaps thirty-five milliseconds.

Twenty milliseconds of transit to the issuer's processor. Arrival at 525 milliseconds.

Phase five, T+525 to T+775 ms: the issuer decides

This is the largest single block of computation in the transaction, and it is a quarter of a second.

Elapsed	Step
525 to 540 ms	Parse, validate, check for a duplicate against DE11, DE37 and DE32
540 to 570 ms	Validate the cryptogram in a hardware security module
570 to 615 ms	Read the account, the balance and the outstanding holds
615 to 700 ms	Score the transaction against the fraud model
700 to 740 ms	Decide, place the hold, memo-post, generate DE38 and DE39
740 to 760 ms	Generate the response cryptogram in the security module
760 to 775 ms	Build and transmit the `0110`

The cryptogram validation is the part worth understanding in detail, because it is the only step in the whole chain that proves anything.

The issuer holds a master key for the card portfolio inside a hardware security module. From that master key, together with the account number and the application PAN sequence number in tag `5F34`, the module derives the unique key belonging to this individual card. From that card key, together with the application transaction counter in tag `9F36`, it derives a session key that is used for this transaction and no other. It then recomputes the cryptogram over the data the card signed — `9F02`, `9F03`, `5F2A`, `9F1A`, `95`, `9A`, `9C`, `9F37`, `82`, `9F36` and `9F10` — and compares its answer with the eight bytes in `9F26`.

If those eight bytes match, the issuer knows three things it cannot know any other way: that a genuine chip carrying its key was present; that the chip saw this amount and this currency; and that this is not

a replay, because the counter has advanced. If they do not match, everything else in the message is unsupported assertion.

The balance check is not a subtraction from a single number. The issuer holds a ledger balance, which is what has actually posted, and computes an available balance by subtracting authorisations already granted and not yet cleared. The £42.50 becomes another such deduction: an entry in a shadow-balance table with an expiry, not a movement in the ledger.

The fraud model is where the variance lives. A real-time scoring service consulting device, merchant, geography and behavioural features is the likeliest reason an issuer's median response time is 250 milliseconds rather than 90, and its failure mode is a timeout rather than an error.

The decision is yes. The issuer generates the authorisation identification response, six alphanumeric characters, `A4T29B`, and sets the response code to `00`. It generates an authorisation response cryptogram in the security module and packages it, with a response code the card would understand, into tag `91` inside data element 55.

The response

MTI `0110`, authorisation response: same version, same class, function 1 for request response, origin 0 because it retains the acquirer origin of the request.

Primary bitmap `723860010EC08201`.

Byte	Hex	Bits	Elements present
1	`72`	01110010	2, 3, 4, 7
2	`38`	00111000	11, 12, 13
3	`60`	01100000	18, 19
4	`01`	00000001	32
5	`0E`	00001110	37, 38, 39
6	`C0`	11000000	41, 42
7	`82`	10000010	49, 55
8	`01`	00000001	64

Eighteen elements, 186 characters. Of those eighteen, exactly two are new:

DE	Value	Meaning
38	`A4T29B`	Authorisation identification response, the code printed on the receipt
39	`00`	Approved or completed successfully

Data element 55 in the response contains a single object: tag `91`, issuer authentication data, eight or ten bytes depending on which response cryptogram method the card profile uses. Everything else is echo.

Data element 4 is echoed at `000000004250`. Had the issuer approved a lesser sum, this field would have come back smaller, and the terminal would have been required to notice, tell the operator, and collect the difference or reverse the whole transaction. A terminal that reads only data element 39 and ignores data element 4 will approve a partial authorisation as though it were full and leave the merchant short.

Phase six, T+775 to T+895 ms: back down the chain

Elapsed	Step
775 to 795 ms	Issuer to scheme, transit
795 to 815 ms	Scheme stops its timer, logs, seeds the clearing record, revalidates the MAC
815 to 830 ms	Scheme to acquirer, transit
830 to 870 ms	Acquirer records the authorisation, maps DE39 to its terminal vocabulary, rebuilds the response
870 to 895 ms	Acquirer to terminal, transit

The acquirer's forty milliseconds are not idle. It writes an authorisation record keyed on data element 37 that will have to be matched against a clearing record arriving days later, it reserves the merchant's exposure, and it translates its terminal-facing response vocabulary — a mapping that is lossy in both directions for every value except approval.

Phase seven, T+895 to T+1,800 ms: the terminal finishes

The terminal parses the response, confirms it matches the request it is waiting on, and updates its state machine to approved. At 935 milliseconds the screen reads APPROVED and the operator can start bagging the book.

It also discards tag 91. The card left the field six hundred milliseconds ago, and there is no second exchange in a contactless transaction to deliver the issuer's cryptogram to. Had this been a contact chip transaction the terminal would now issue a second GENERATE AC command carrying that data, and the card would verify the issuer's cryptogram and return either a transaction certificate or an application authentication cryptogram — retaining the right to decline a transaction the issuer has just approved. That exchange costs a further two to four hundred milliseconds and requires the card to still be in the slot, which is the entire reason chip-and-PIN feels slower than tapping.

Then it prints. A thermal printer laying down a merchant copy of roughly 120 millimetres at a realistic line speed takes about 660 milliseconds, and cutting, clearing and prompting for the customer copy takes another 200. Receipt out at 1,800 milliseconds.

The printed record carries data element 38 as the authorisation code, elements 12 and 13 as the date and time, element 41 as the terminal identifier, a masked form of element 2, and the entry mode derived from element 22. It does not carry element 37, which is the one number the acquirer will ask the merchant for when something goes wrong. That omission is a small, permanent, industry-wide own goal.

Where the two seconds actually goes

Component	Milliseconds	Share
Card and terminal, over the air	320	18%
Terminal software, both directions	95	5%
Network transit, all six one-way hops	120	7%
Acquirer processing, both directions	95	5%
Scheme processing, both directions	55	3%
Issuer decision	250	14%
Printer and paper	865	48%
Total	**1,800**	**100%**

Three conclusions follow, and all three are unwelcome to somebody.

The network is not the problem. Six one-way hops across two commercial boundaries, including a leg into a global switch, come to 120 milliseconds. That is under seven per cent, and it is the component with the least headroom, because most of it is the speed of light in glass.

The issuer is not the problem either, though it is the largest computational block. A quarter of a second to validate a cryptogram in a hardware security module, read an account, score it against a fraud model and commit a hold is not slow; it is the cost of doing the work honestly.

The problem, if there is one, is physical. The card conversation and the printer together are two thirds of the elapsed time, and neither is a computer being slow. One is a passive device drawing its power from a radio field. The other is paper. This is why the most effective latency work in retail payments over the past decade has been suppressing the customer receipt and printing the merchant copy asynchronously, which removes up to 865 milliseconds from the queue without touching a line of protocol code.

The tail

Everything above is a median. Here is the same transaction going wrong in the ordinary ways, and what each costs.

Condition	Added time
Terminal on a mobile data connection instead of fixed broadband	400 to 1,200 ms
Fresh TLS handshake per transaction rather than a persistent session	60 to 200 ms
DNS resolution with a cold or expired cache	20 to 500 ms
Contact chip rather than contactless, second GENERATE AC exchange	200 to 400 ms
Cardholder verification required, PIN entered	2,000 to 6,000 ms, human-limited
Issuer fraud model cold path or garbage collection pause	200 to 800 ms
Issuer times out and the scheme stands in	Up to the scheme's timer

The last row is the one that changes the shape of the outcome rather than the duration. Mastercard's published documentation for one of its processing products puts the issuer response window at four seconds, and the processor's own window at seven seconds, after which Mastercard authorises on the issuer's behalf. When that happens, the approval the merchant receives was granted by the scheme against limits the issuer configured in advance, the issuer learns about it afterwards by advice message, and the liability position is different from a normal approval in ways set out in the scheme's rules rather than in any technical specification.

Note also the cardholder verification row. The moment a human is asked to do something, every millisecond above becomes rounding error, and no amount of protocol optimisation touches it.

When the answer never comes

The narration above has a response arriving. Consider the case where it does not: the acquirer's reply is lost between 870 and 895 milliseconds, after the issuer has already placed a hold.

The terminal times out. It does not know whether the transaction was approved, declined, or never seen. What it must not do is simply retry with a fresh trace number, because the issuer would then see two unrelated requests and place two holds.

The correct behaviour is one of two things. Either it retransmits the identical message with the MTI's origin digit changed from 0 to 1, producing `0101`, with the same trace number and retrieval reference number, so the issuer recognises a repeat of something it may already have answered. Or it sends a reversal, MTI `0400`, quoting the original message's identifying values inside data element 90 — the original MTI, trace number, transmission date and time, and acquiring and forwarding institution identification codes — so the issuer can find the hold and release it.

Reversals are the least glamorous and most consequential messages in the protocol. A terminal that fails to send them leaves customers with money held against purchases that never happened, for as long as the issuer's hold expiry allows. Every complaint that begins "the payment failed but the money has gone" is a missing `0400`.

What has and has not happened

At 1,800 milliseconds the customer has a book, the operator has a receipt, and the following is true.

The issuer has reduced the cardholder's available balance by £42.50 and has not touched the ledger balance. The acquirer holds an authorisation record keyed on `607614430041` with an approval code of `A4T29B`. The scheme holds a routing and timing record. The terminal holds a batch entry awaiting the day's close.

No money has moved. No entry has been posted to any account at the Bank of England. The merchant will not be paid until the transaction is submitted for clearing, matched, assessed for interchange — at, as of writing, a United Kingdom domestic cap of 0.2 per cent for consumer debit where merchant, acquirer and issuer are all in the United Kingdom, which on £42.50 is 8.5 pence — and settled between the parties' banks. On this transaction the merchant service charge, the scheme fees and the acquirer's margin will all be assessed against a payment that took 1.8 seconds to authorise and will take days to complete.

The next chapter deals with the identifiers left behind — the trace number, the retrieval reference number, the acquirer reference number and the authorisation code — because those four values are the only handles anybody has on this transaction for the rest of its life, they have four different owners and four different lifetimes, and asking the wrong party for the wrong one is how a simple query becomes a three-week correspondence.

27.98 Common wrong ideas

Wrong: one message travels along the chain from the till to the issuer. **Right:** each hop parses, discards, rewrites and re-emits, and in Britain the terminal typically speaks Standard 70 to its acquirer while the acquirer speaks ISO 8583 to the scheme, so the correct question is never "what did the message say" but "which message, at which hop".

Wrong: the network is the slow part of a card payment. **Right:** all six one-way hops come to about 120 milliseconds, under seven per cent of the elapsed time, while the printer is 865 milliseconds and nearly half.

Wrong: the beep means the payment is approved. **Right:** at 320 milliseconds the card has merely finished its part and nothing has left the shop, which is why customers walk away before a decline can be displayed.

Wrong: a tap and an inserted chip have the same security model. **Right:** in Visa's contactless mode the cryptogram arrives in the response to GET PROCESSING OPTIONS, there is no second GENERATE AC exchange, and the issuer's reply reaches a terminal with nothing left to deliver it to.

Wrong: the till's trace number identifies the transaction all the way through. **Right:** the acquirer replaces DE11 with a value unique within itself and creates DE37, which is the number that outlives everything else in the message.

Wrong: DE39 `00` is the whole answer, so the terminal can print APPROVED. **Right:** the response can echo a smaller DE4 for a partial approval, and a terminal that reads only DE39 will treat it as a full authorisation and leave the merchant short.

Wrong: the amount, currency and country the issuer checks are elements 4, 49 and 19. **Right:** only the copies inside DE55 — tags `9F02`, `5F2A` and `9F1A` — are covered by the cryptogram, and a disagreement between the two copies is exactly what an altered transaction looks like.

Wrong: the approval means money has moved. **Right:** the issuer has written an entry with an expiry into a shadow-balance table and has not touched the ledger balance, and nothing posts until clearing and settlement days later.

Wrong: when the response times out the terminal should send the transaction again. **Right:** it must either retransmit the identical message as a `0101` with the same trace and retrieval reference numbers, or send a `0400` reversal quoting the original values in DE90, because a fresh trace number produces two holds.

Wrong: the median timings are the engineering target. **Right:** contracts and scheme rules are written about the tail, where Mastercard's published four-second issuer window and seven-second processor window live, and a system built to the median will meet them in testing and breach them on the first Saturday before Christmas.

27.99 Chapter summary in 20 lines

1. One purchase — a £42.50 book tapped on a countertop terminal in London on 17 March 2026 — takes 1,800 milliseconds from card presentation to receipt.
2. The card is a computer holding a key only it and the issuer know, the terminal knows the context and can decide nothing, the acquirer owns the merchant relationship, the scheme is a switchboard, and only the issuer can say yes.
3. The first 320 milliseconds are the card and the reader talking over a few centimetres of air: anti-collision, selection of the proximity payment system environment, application selection, and GET PROCESSING OPTIONS.
4. The card publishes a processing options data object list naming the values it wants, the terminal supplies them in that order, and the cryptogram in tag `9F26` attests to exactly those values at that moment.
5. In Visa's contactless mode the cryptogram comes back in the GET PROCESSING OPTIONS response, so there is no second exchange and no way to deliver the issuer's reply to the card.
6. The terminal performs terminal action analysis, stamps its local time and date, assigns its own trace number, and sends the request at 375 milliseconds over an already-established TLS session.
7. Its message declares sixteen elements and runs to 484 characters, of which 323 are data element 55.
8. The acquirer's seventy milliseconds are the busiest in the chain: parse defensively, detect duplicates, decrypt in a hardware security module, apply risk rules, then write its own fields.
9. It adds elements 19, 32, 37 and 64, rewrites 7 and 11, and passes sixteen elements through unchanged, and it is on those sixteen that every downstream reconciliation depends.
10. The scheme validates the message authentication code, reads the issuer identification number from the leading digits of the account number, resolves a token if there is one, scores the transaction, starts a response timer and forwards it.

11. The issuer's quarter of a second is the largest computational block: parse, validate the cryptogram, read the account and its holds, score for fraud, decide, place the hold and build the `0110`.
12. Cryptogram validation derives a card key from a portfolio master key and a session key from the transaction counter, recomputes the cryptogram and compares it, proving that a genuine chip was present, that it saw this amount and currency, and that this is not a replay.
13. The balance check subtracts authorisations already granted from a ledger balance, and the £42.50 becomes another such deduction with an expiry rather than a movement in any ledger.
14. The response carries exactly two new fields, the authorisation code `A4T29B` in element 38 and the response code `00` in element 39, plus tag `91` inside element 55 that a contactless terminal will discard.
15. Adding the components up: the card and terminal 320 milliseconds, all network transit 120, the acquirer 95, the scheme 55, the issuer 250, terminal software 95, and the printer 865.
16. The two slowest components are not computers being slow but a passive device drawing power from a radio field and a strip of paper, which is why suppressing the customer receipt is the most effective latency work of the past decade.
17. The tail is where the engineering is: mobile data, per-transaction TLS handshakes, cold DNS, the contact chip's second GENERATE AC, and above all a human being asked to enter a PIN.
18. When the issuer does not answer within its window the scheme stands in, and the approval the merchant receives was granted against limits the issuer configured in advance, with a liability position set by scheme rules rather than by any technical specification.
19. When no response reaches the terminal the correct behaviour is a `0101` repeat carrying the same identifiers or a `0400` reversal quoting the original data elements in DE90, and every complaint that begins "the payment failed but the money has gone" is a missing `0400`.
20. At 1,800 milliseconds the customer has a book and the merchant has a promise, but no money has moved, no entry has been posted at the Bank of England, and the transaction remains disputable for months.

Chapter sources: ISO 8583:1987 data element and MTI tables as consolidated in the Wikipedia ISO 8583 article; Worldpay Lync ISO 8583 Message Specification, data element 55, for the DE55 subelement encoding scheme, the tag and length table including 9B, 9F66 and 9F6C, the treatment of tag 57 as the source of track data, and the message types requiring DE55; Visa Developer Request and Response Codes for point of service condition codes 00, 08 and 51; UK Payments Administration Standard 70, Books 2 and 3, as catalogued at iso8583.info; Mastercard Processing Debit API documentation for the four-second issuer and seven-second processor windows and stand-in processing; Visa's published figure of more than 65,000 transaction messages per second; FCA policy statement PS21/2 for the £300 cumulative contactless threshold, and reporting of the £100 limit in force from 15 October 2021 to 18 March 2026; Payment Systems Regulator on the UK Interchange Fee Regulation caps; ISO 18245 code 5942; EMVCo contactless entry point documentation and the arXiv systematisation "SoK: Security of EMV Contactless Payment Systems" for the PPSE, application selection and GET PROCESSING OPTIONS flows; ISO/IEC 14443, 7812-1, 7816-6, ISO 3166-1 and ISO 4217. All bitmap expansions, message lengths, timing totals and percentages were computed from the field sets stated in the text.

28

CHAPTER

The Number You Were Given

28.0 What this chapter gives you

1. You will be able to name the four identifiers a card payment leaves behind, say who created each one, where it appears and how long it lives.
2. You will be able to explain why searching on a system trace audit number alone returns every terminal's `004821` for every day in the retention window, and what the composite key should be instead.
3. You will be able to say why the approval code on a customer's receipt is corroboration and never a lookup key.
4. You will be able to tell a customer whether they were charged twice or are looking at one charge and one hold, and why refunding an uncleared authorisation leaves them net up.
5. You will be able to say which number to quote to your acquirer, which to the scheme, which to the issuer, and which never to ask a customer for.
6. You will be able to store each identifier at the width and type its specification demands, and say why twenty-three digits will not fit in a 64-bit integer.
7. You will be able to explain why an acquirer reference number exists if and only if the transaction cleared, and what that means for the commonest class of complaint.
8. You will be able to say why both major schemes had to invent identifiers of their own, and why Mastercard's has to be assembled from three fields.
9. You will be able to contrast a card payment's identifiers with a UETR, and say why a Swift payment can be tracked end to end and a card payment cannot.
10. You will be able to read a signature from a pair of records — sequential trace numbers seconds apart, two terminal identifiers, an incremental authorisation, a tip adjustment, a late reversal — and name the cause.

The plain version

Imagine a school trip. Forty children, one coach, one museum, one long day.

Before the coach leaves, the teacher walks down the aisle handing out numbers. "You're fourteen today." It is written on a sticker. It is used all day for one purpose only: counting heads. Every time the coach stops, the teacher calls the numbers and waits for forty voices. Number fourteen is not who you are. It is where you sit in the count. Next week, on the next trip, somebody else will be fourteen. On the coach going home, you might even be given a different number, because a different teacher is counting.

At the museum door you are given a wristband. It has a code printed on it, something like `M-3040-118`. It goes on at nine in the morning and stays on until you cut it off at home. It is yours for the whole

day, and no other child in the building has the same one. If you get lost, the wristband is what the staff read out over the radio.

Inside, at the gift shop, you buy a keyring. The person at the till presses a button, a machine thinks for a second, and a small slip prints with a code on it: `07B4C1`. That code means the shop's bank said yes. It does not say what you bought, or how much, or who you are. It only says: at this moment, a machine said yes to something.

Then, a week later, the school office does the accounts. Somebody sits down with a stack of receipts and writes each one into a ledger, giving each line its own long reference. That reference did not exist on the day. It was born a week later, in a different building, made by a different person, for a different purpose: settling up. But if you later ring the school and say the gift shop charged you twice, that ledger reference is the only number that gets you anywhere, because it is the one attached to actual money that actually moved.

Four numbers. The coach number, the wristband, the till slip code, the ledger reference. They are not four copies of the same thing. They have four different owners, four different lifetimes, and four different jobs, and if you ask the wrong person for the wrong one you get a shrug.

A card payment works exactly like this.

Say you buy lunch at a café in Leeds on Tuesday 3 March. The bill is £42.60. You tap your card. Between the tap and the receipt printing, four numbers come into existence, or will do shortly.

The first is the card machine's own counter. The terminal has been sitting there all morning ticking upward: 004819, 004820, and now yours, **004821**. It puts that number in the message it sends off, and when the answer comes back, that answer carries 004821 too. That is the entire point of it. It is how the terminal knows the reply that just arrived is the reply to *your* lunch, and not to the person behind you buying a coffee. It is the coach number. It counts up to 999999 and then rolls back round to the beginning and starts again. Yesterday's 004821 was somebody else's sandwich.

The second is a longer reference, twelve characters, something like **506201304821**. The café's bank makes this one. Unlike the counter, it is meant to last. It is the wristband. It goes onto the request, it comes back on the answer, and it stays attached to the record of your lunch when that record is filed away at the end of the day. If, six weeks later, somebody has to go and find the exact transaction in the café's bank's systems, this is the number they search on.

The third is the approval code: **07B4C1**. Six characters. Your own bank makes this one, and only makes it if the answer was yes. It appears on the paper receipt, which is why it is the number a customer is most likely to be holding when they ring you up. It is the till slip code from the gift shop. It proves that a yes happened. On its own it does not prove what the yes was *for*.

The fourth does not exist yet. At the end of the day the café's card machine closes its batch and sends the day's takings off to be paid. Somewhere in that process, a twenty-three digit reference is created for your £42.60 specifically: something in the shape of **74530045062001234567890**. That is the ledger line. It is the number the card networks use to talk about your lunch as *money that moved*, rather than as *a question that was asked*. If you dispute the charge, this is the number the dispute is filed under.

Now the useful part. Suppose you look at your banking app on Wednesday and see £42.60 twice.

If you ring the café and read out the counter number, they can do nothing with it, because every card machine they own has been through 004821 many times. If you read out the approval code, they can look, but approval codes are short and get reused, so they may find several. What actually settles the question is the twenty-three digit ledger reference, one per movement of money. Two of those means

two payments and one needs refunding. One of those, plus one leftover approval that never became a ledger line, means you were charged once and are looking at a hold that will fall off your statement in a few days.

Same event. Four numbers. Only one of them answers the question you actually asked.

Where the plain version stops being true

The wristband is not unique for ever, and it is not necessarily a number. The twelve-character reference is defined in the card message standard as twelve *alphanumeric* characters, not twelve digits. It can contain letters. It can begin with zeros. Any system that stores it as an integer will silently destroy it, and a great many systems do. It is also only unique within the organisation that issued it and within a time window that organisation chooses. Two different acquiring banks can and do produce the same twelve characters on the same day for two completely unrelated transactions. It is a good search key inside one estate. It is not a global identifier, and treating it as a primary key across parties is how reconciliation breaks.

The till slip code does not name the purchase. The six-character approval code says that an approval occurred. It does not encode the amount, the card, the merchant or the time. Issuers generate them cheaply and reuse them freely. Matching two records on approval code alone will produce false matches at any real volume, and it will produce them most often exactly when you are under pressure, because the customer with a problem is the one who has the receipt in front of them and nothing else. The approval code is a corroborating field. It is never a lookup key on its own.

The ledger reference is not the same journey. The twenty-three digit reference is created during clearing, which is a separate, later, batch-shaped process from the authorisation that happened at the till. An authorisation that is never cleared never gets one. A declined transaction never gets one. A hold that is left to expire never gets one. So there is a whole class of customer complaint, arguably the most common one, where the correct answer is "there is no ledger reference, because there is no charge", and the analogy of one wristband per child breaks down entirely: some children never went into the museum.

The numbers do not travel unchanged. The coach analogy implies one teacher counting. In reality the message crosses a terminal, a gateway, an acquirer's switch, the scheme's network, perhaps an issuer processor, and finally the issuer, and several of those nodes are entitled to replace the counter with one of their own before passing the message on, keeping a private map from the number they received to the number they sent. Your 004821 may be somebody else's 771903 two hops later. This is not a bug. It is how a switch keeps its own outstanding requests straight. It does mean there is no single number that is guaranteed to be present, unchanged, at every point in the chain, which is precisely the gap the card schemes later tried to close with identifiers of their own.

The technical version

The message these numbers live in

Card authorisation messages are ISO 8583 structures. The message type indicator is four digits: position one is the standard version (0 for the 1987 edition, 1 for 1993, 2 for 2003), position two the message class (1xx authorisation, 2xx financial, 3xx file actions, 4xx reversal and chargeback, 8xx network management), position three the function (0 request, 1 request response, 2 advice, 3 advice response) and position four the origin (0 acquirer, 2 issuer, 4 other). An authorisation request is therefore 0100, its

response 0110, a financial request 0200, a reversal request 0400 and a network management request 0800.

After the MTI comes the primary bitmap, sixty-four bits indicating whether data elements 1 to 64 are present. Bit 1, when set, signals a secondary bitmap covering elements 65 to 128. Everything discussed below is a data element addressed by that bitmap.

Four of them matter for tracing, and they are not interchangeable.

Element	Name	Format	Created by	Lives for
DE11	System trace audit number	n 6	The message originator	One request-response pair
DE37	Retrieval reference number	an 12	Usually the acquirer	Authorisation through to research
DE38	Authorisation identification response	an 6	The issuer	The life of the approval
DE31 (clearing)	Acquirer reference data / ARN	n 23	The acquirer, at clearing	The life of the money, and of any dispute

Two supporting elements do most of the disambiguating work and are frequently forgotten: DE41, card acceptor terminal identification, ans 8, and DE42, card acceptor identification code, ans 15. Alongside them sit DE7 transmission date and time, DE12 local transaction time (n 6, hhmmss), DE13 local transaction date (n 4, MMDD), DE4 amount, DE49 currency code as an ISO 4217 numeric code, DE39 response code (an 2) and DE55 for the EMV data.

Note DE13 carefully. In the 1987 catalogue the local transaction date is four digits, MMDD. There is no year in it. Every date-range query you write against a store of raw ISO 8583 must derive the year from somewhere else, normally DE7 or the acquirer's own receipt timestamp, and every year-end produces a fresh crop of reconciliation breaks in systems that did not.

DE11: the system trace audit number

DE11 is six numeric digits. It is assigned by whichever node originates the message, and it must be returned unchanged in the corresponding response so that the originator can pair the two. That is its whole function. It is a correlation key.

The consequences of that follow directly from the size of the field. Six digits gives you one million values before the counter wraps to zero and starts again. A busy terminal estate exhausts that in a day. A switch handling scheme traffic exhausts it in minutes. So a STAN is unique only within one originator, within one wrap cycle, and the sequence is very often per terminal rather than per estate, which means STAN 004821 exists simultaneously on every terminal you own.

This produces the single most common analytical error in card operations: searching for a STAN. A bare STAN search across an estate returns every terminal's 004821 for every day in the retention window. The correct key is composite. In practice you need the STAN together with the acquiring institution or terminal identity (DE41 or DE42), the local date (DE13, plus a year from DE7), and usually the amount (DE4) and currency (DE49) as a sanity check. Only then does the tuple identify one transaction, and even then only within one processing estate.

DE11 has one hard operational obligation attached to it. A reversal — a 0400 — must carry the STAN of the original message it is reversing, so that the issuer can find and release the hold it placed. Systems that generate a fresh STAN for the reversal and put the original nowhere in the message are the direct cause of authorisation holds that sit on customers' statements for a week. If you build a terminal application, store the original STAN in a durable local record before you send the request, not after you get the response, because the case in which you most need to reverse is the case in which no response arrived.

Mastercard's single-message system uses DE11 in exactly this way and names it accordingly. In Mastercard's fraud submission interface the identifier `TRC` is defined as "Trace Id (DE 11 in Single Message Authorization Message)", with a minimum and maximum length of 6 and a numeric data type. That is the STAN, promoted to an externally quotable identifier, and it is never quoted alone: Mastercard requires it in combination with `SER`, the Switch Serial Number, DE63 subfield 3, nine numeric digits. Two fields, because one is not enough. That is the standard telling you, in its own documentation, that a STAN is not a transaction identifier.

DE37: the retrieval reference number

DE37 is twelve alphanumeric characters, `an 12`. The name is the specification: it is the number you quote when you *retrieve* a transaction later, which historically meant asking the acquirer to pull the paper draft out of a filing cabinet.

Two things about it are routinely got wrong.

First, it is alphanumeric. Worldpay's published ISO 8583 reference guide records a change history entry to the effect that field 37 was corrected from numeric to alphanumeric, which tells you how easy the mistake is to make even inside a large processor. Store it as a fixed-width character field of exactly twelve. Do not trim it. Do not cast it. Leading zeros in a retrieval reference number are load-bearing.

Second, ISO 8583 does not specify any internal structure for DE37. It says twelve characters and stops. Everything you have read about what the characters mean is convention, and the convention belongs to whoever generated the value. Industry material describes one layout more often than any other, in which the leading characters carry a date and an hour, commonly a single digit of year, three digits of day-of-year and two digits of hour, with the remaining six characters a sequence that is often just the STAN. Acquirers do generate values of exactly that shape. But no published specification requires it, none of the public sources that set it out is a primary one, and so you must not write a parser that assumes it for traffic you did not generate. Some acquirers pack a terminal identifier in there instead. Some emit an opaque sequence. Some processors' own integration guides tell the merchant it may put whatever it likes in the field so long as the value comes back unchanged. A parser that assumes structure will one day interpret a letter as a month.

What DE37 buys you that DE11 does not is persistence. The retrieval reference number is generated once and carried forward, so the same value appears on the authorisation, on the advice, and typically on the acquirer's own clearing record and in the merchant's settlement report. That makes it the natural join key between the operational world of authorisations and the financial world of settlement, from the merchant's and acquirer's point of view. It is the best single number a merchant can hold, and it is the one your gateway should be surfacing to you in its API responses.

It is not, however, an identifier the issuer will necessarily accept as authoritative, and it is not unique across acquirers. Twelve characters and no registry of who may use which range means collisions between institutions are not merely possible but arithmetically inevitable at scheme volumes. Within your acquirer it identifies a transaction. Outside it, it is a strong hint.

India is the clearest illustration of DE37's second life. The retrieval reference number that appears in an IMPS or UPI transaction — the "UPI reference number" that Indian consumers are told to quote — is a twelve-digit RRN in the ISO 8583 lineage. Oracle's India IMPS payments documentation states plainly that the system "auto generates the RRN (Retrieval Reference Number) on authorization of the payment transaction" and that "this is a 12-digit number". The identifier a customer in Bengaluru reads off a UPI app screen and the identifier in DE37 of a card authorisation in Manchester are the same field, from the same standard, doing the same job.

DE38: the authorisation identification response

DE38 is six alphanumeric characters. It is generated by the issuer, or by the scheme acting on the issuer's behalf during stand-in processing, and it is present only when the transaction was approved. It travels back in the 0110 response alongside DE39, the response code, which in the 1987 catalogue is two alphanumeric characters, with `00` conventionally denoting approval. (The 2003 edition of the standard renames the field altogether: ISO's own published contents for ISO 8583-1:2003 list bit 39 as the action code, and dialects built on the later editions generally carry three-digit code values rather than two.)

DE38 is the number printed on the customer's receipt, labelled "auth code" or "approval code", which makes it the number you will most often be handed at the start of an investigation. It is also the weakest of the four.

Six alphanumeric characters is a small space. Issuers do not coordinate their generation, and there is no requirement that a given issuer's codes be unique across time, across cards or across merchants — only that the code be usable by the issuer to identify the approval it granted. The practical rule that follows is that an authorisation code is only meaningful in the presence of the card, the amount and the date. A support agent searching a transaction database on approval code alone, with no other predicate, is running a query that becomes less reliable as the business grows.

There are two further seams worth knowing. When an issuer is unavailable and the scheme approves on its behalf under stand-in processing, the approval code in DE38 was generated by the scheme, not the issuer, and the issuer may have no record of it until the advice reaches them. And in a partial approval, the amount approved is not the amount requested, so an approval code paired with the requested amount rather than the approved amount will not reconcile. DE38 tells you that a yes happened. DE4, DE49 and DE39 tell you what the yes was.

The acquirer reference number

The acquirer reference number is the one identifier in this chapter that is not created at the point of sale. It comes into existence during clearing, when the acquirer assembles the day's approved transactions into a settlement file and submits them to the scheme.

On Visa's side, the BASE II clearing system's own glossary defines it exactly: "Acquirer Reference Number: A 23-digit identification number associated with every draft and voucher. It consists of a Format Code, BASE Identification Number (BIN), Capture Date, Film Locator, and Check Digit." Five components, twenty-three digits, and the vocabulary tells you the history — a "film locator" is where the microfilm image of the paper sales draft was filed. The capture date is expressed as a Julian date, which Visa's glossary defines as "a date expressed as the day's position in a year rather than in a particular month. The format is YDDD or YYDDD."

On Mastercard's side the field is called Acquirer Reference Data and lives in DE31 of the clearing message. Mastercard's own developer documentation for fraud submission specifies the constraint

precisely: the identifier `ARN` is "Acquirer Reference Number (DE31 in Clearing Message)" and for it "Min Length is 23, Max Length is 23 and Data Type is N". Exactly twenty-three, numeric, no exceptions.

The check digit at the end is the practically useful part. It is computed from the preceding digits by a defined formula, which means a malformed ARN can be rejected before it costs anyone a research request. It also means that if you are validating ARNs you did not generate, you should validate the check digit rather than the semantics: do not try to parse the BIN and the date out of an ARN and reason about them, because the format code in the first position exists precisely to signal that the remaining layout may differ.

Two properties of the ARN follow from its birth at clearing rather than at authorisation, and both matter enormously in customer support.

The ARN exists if and only if the transaction cleared. No clearing, no ARN. Authorisations that expired, holds that were reversed, declines, and transactions still inside the acquirer's capture window all have no ARN and never will.

The ARN is the key the dispute process runs on. When an issuer initiates a chargeback, when an acquirer represents it, when a refund is traced across the network, the ARN is what identifies the item. It is also the number a merchant can give a customer that the customer's own bank can actually act on, which is why "please ask the merchant for the ARN" is the standard advice issuers give. It travels, in a way that a STAN never does.

The scheme's own key, and why it had to be invented

By this point the shape of the problem should be obvious. The acquirer owns a number. The issuer owns a number. The terminal owns a number. The clearing file owns a number. Nobody owns a number that is present, unchanged, at every hop, in both the authorisation and the clearing worlds. Card networks worked around this for decades with fuzzy matching on card number, amount, date and merchant.

Both major schemes eventually created an identifier of their own to close the gap.

Visa assigns a Transaction Identifier to each authorisation. As the Merchant Risk Council summarises it, it is "a numeric value up to 15 digits that uniquely identifies the original authorization", and the industry commonly calls it the Network Transaction Identifier or NTI. Its purpose is to let a later transaction point back at an earlier one — the classic case being a merchant-initiated recurring payment that must reference the original cardholder-authenticated authorisation.

Mastercard's equivalent is built from the Banknet reference number. In Mastercard's own field definitions, `BRN` is the "Banknet Reference Number (DE63, subfield 2 in Dual Message Authorization Message)", with a minimum length of 6 and maximum of 9, alphanumeric, and the documentation warns explicitly that "DE63, subfield 1 (Financial Network Code) is NOT to be appended". The composite identifier Mastercard asks merchants to store and echo — the Trace ID — is fifteen positions: positions 1 to 3 the Financial Network Code, positions 4 to 9 the Banknet Reference Number, positions 10 to 13 the settlement date as MMDD, and positions 14 to 15 blank.

Look at what that construction admits. To get an identifier that is stable enough to quote, Mastercard has to combine a network code, a reference number and a date. Three fields, because the reference number alone is six characters and six characters is not enough to be unique. This is the same lesson as the STAN, at a higher level of the stack: identity in payments is almost always composite, and any system that stores a single "transaction ID" column and believes it has captured identity is storing a correlation key and calling it a name.

The complete picture for one card payment

Putting it together for the £42.60 lunch in Leeds on Tuesday 3 March. These values are illustrative, not real.

What	Value	Who made it	Where it appears	Who can search on it
STAN (DE11)	`004821`	The terminal	0100 and 0110 only	Terminal estate operator, gateway, acquirer switch, each within its own hop
RRN (DE37)	`506201304821`	The acquirer	Authorisation, advice, clearing, merchant settlement report	Acquirer, gateway, merchant
Auth code (DE38)	`07B4C1`	The issuer	0110 response, customer receipt	Issuer, weakly
ARN (DE31 clearing)	`74530045062001234567890`	The acquirer, at clearing	Clearing file, dispute records, issuer research	Acquirer, scheme, issuer
Merchant order ID	`BC-2026-03-03-1187`	The merchant	The merchant's own database, sometimes the descriptor	The merchant only

That last row is the one merchants forget to design. Your own order identifier is the only number in the table you control, the only one guaranteed to be unique in your own system, and the only one you can attach to a customer record. Every investigation that goes well starts there and moves outward. Every investigation that goes badly starts with a number somebody read off a bank statement.

A customer says they were charged twice

Here is the order to work in, and why.

Step zero: establish whether there are two charges or one charge and one hold. This resolves the majority of cases and costs nothing. An authorisation that has not cleared appears in a banking app as a pending item and has no ARN. Ask the customer whether both lines are pending, or whether one is pending and one has posted, or whether both have posted. If both are pending and the amounts are identical, the likely cause is a double authorisation at the terminal or an unreversed retry, and the remedy is a reversal, not a refund. Refunding an uncleared authorisation creates a credit that settles while the hold quietly expires, and the customer ends up net up. This mistake is common and expensive.

Step one: collect what the customer actually has. Date, amount, currency, last four digits of the card, the descriptor as it appears on their statement, and the approval code if they have a receipt. Do not ask them for an ARN. Most issuers do not show it in the app, and asking is how a support conversation becomes a complaint.

Step two: find the order in your own system. Search on your merchant order identifier, or on amount plus date plus card token or last four. You are looking for how many order records exist and how many authorisation attempts each one made. This is where genuine double-submits, retried API calls without idempotency, and repeated tap events reveal themselves.

Step three: get the gateway or acquirer record for each attempt. For each attempt you now want DE37, the retrieval reference number, DE38, the approval code, DE39, the response code, and the local date and time. Two approved attempts with different retrieval reference numbers is two authorisations. Two approved attempts with different approval codes and different STANs one minute apart is almost always a terminal double-tap or a customer who tapped again after a slow response.

Step four: check clearing. For each authorisation, does an ARN exist? One ARN means one charge and the second authorisation is a hold. Two ARNs means two charges and one of them needs refunding. Zero ARNs on a same-day query means nothing has cleared yet and you should not conclude anything until the batch closes.

Step five: escalate with the right number to the right party. To your acquirer, quote the ARN if the item cleared and the retrieval reference number if it did not. To the scheme, via your acquirer, the ARN. To the issuer, if the customer is doing the asking, the ARN plus the date and amount. The approval code goes along as corroboration in all three cases and as the primary key in none of them.

What not to do at any step: do not search on STAN alone; do not match on approval code alone; do not assume the gateway's transaction identifier means anything to the acquirer, because it is the gateway's private key and typically does not appear in any ISO 8583 field at all; and do not tell the customer a number is "the transaction ID" when it is a correlation key that will be reused by teatime.

The underlying causes are a short list, and each has a signature. Two authorisations seconds apart with sequential STANs: terminal double-submit. Two authorisations minutes apart with different terminal identifiers: the customer tried a second till. An authorisation followed by a larger authorisation with the same retrieval reference lineage: an incremental authorisation, common in hotels and car hire, entirely legitimate. An original amount posting alongside a slightly larger cleared amount: a tip adjustment, where the cleared figure exceeds the authorised one within scheme tolerance. A charge and a matching credit landing days apart: a reversal that arrived after clearing rather than before, which looks like a double charge for as long as it takes the credit to post.

Bank transfers: the reference is not an identifier

Everything above concerns card rails. Account-to-account transfers have a parallel set of identifiers, and one important structural difference: the field the customer types is not the field the system traces on.

In the United Kingdom, a Bacs Direct Credit or Direct Debit is carried in a Standard 18 file. Open Banking Limited's mapping of its payment API onto the scheme formats, which cites the Bacs Electronic Funds Transfer file structures document PN5011 and its section on credit and debit payment instructions for bank grade users, lists the Standard 18 fields: field 1 destination sorting code, six characters; field 2 destination account number, eight; field 5 originating sorting code, six; field 6 originating account number, eight; field 8 amount in pence, eleven; field 10 service user's reference, eighteen; field 11 destination account name, eighteen.

That eighteen-character service user's reference is the entire space you have to say what a Bacs payment is for. It is what a building society roll number goes in. It is what an invoice number goes in. It is not unique, not validated, not guaranteed to reach any particular system at the far end, and if the sending customer types it wrong nothing anywhere will notice. Eighteen characters is also less than the thirty-five that ISO 20022 allows in a structured remittance reference, so a payment initiated through a modern API with a full-length reference must be either truncated or rejected at the boundary. Open Banking's specification says exactly that: it is for the account servicing institution "to decide whether to reject the payment-order consent or truncate the field".

Faster Payments is, perhaps surprisingly to anyone who has only worked on cards, also an ISO 8583 system — a different dialect, using the private-use data elements for its own purposes. Open Banking's mapping gives the field numbers: field 62 carries the end-to-end reference at 31 characters, field 120 the reference information at 18, field 121 the remittance information at 140, fields 42 and 43 the originating credit institution and customer account number, fields 95 and 35 the beneficiary equivalents, field 118 the beneficiary customer account name at 40, field 61.1 the payment sub-type and field 122 regulatory reporting. Pay.UK publishes an ISO 20022 standards library for FPS describing the recommended conversion of those ISO 8583 messages into ISO 20022.

Note what this does to your mental model of data element numbers. Data element 62 in a card authorisation is a private reserved field that Visa uses for its own purposes. Data element 62 in a Faster Payments message is the end-to-end reference. The number is not the meaning. The number plus the dialect is the meaning, and any library that decodes ISO 8583 without being told which specification it is decoding against is guessing.

CHAPS took the other road. It carried MT103 messages, in which the beneficiary reference sits in field 70 across up to four lines of thirty-five characters, with the code `/ROC/` introducing the ordering customer's reference and `/RFB/` introducing a reference for the beneficiary, the latter limited to sixteen characters in practice. In June 2023 the CHAPS high-value payment system migrated to ISO 20022, as the Bank of England records in its own policy statement on mandating enhanced data.

That migration matters here because ISO 20022 finally provides what card rails still lack. A pacs.008 customer credit transfer carries a payment identification block containing several distinct identifiers: an instruction identification, which is point-to-point between one pair of institutions; an end-to-end identification, which the debtor supplies and which must be passed unchanged all the way to the creditor; a transaction identification; and the UETR, the unique end-to-end transaction reference. The UETR is a thirty-six character UUID version 4 as defined by IETF RFC 4122, now superseded by RFC 9562. It is carried in field 121 of the user header block of an MT message and in the payment identification block of a pacs.008, and it has been mandatory on customer credit transfers over Swift's FIN network since 18 November 2018.

A UETR is what the card world does not have: one value, generated once, unchanged at every hop, unique by construction rather than by convention, and long enough that no two payments will ever collide. If you have wondered why a Swift payment can be tracked end to end and a card payment cannot, the answer is a hundred and twenty-eight bits of randomness against six digits of a wrapping counter.

India: UTR, RRN and what a customer can actually trace

India runs several rails and each stamps its own identifier, which is why the question "what is my UTR" has more than one correct answer.

For RTGS, the Reserve Bank of India's own FAQ is unambiguous: "Unique Transaction Reference (UTR) number is a 22-character code used to uniquely identify a transaction in RTGS system." Twenty-two characters, alphanumeric, generated by the sending bank. The RBI is equally clear about who may use it, noting that customers do not have the facility to track the transaction themselves — the UTR is what the bank quotes to the central bank when it asks about status on the customer's behalf.

Indian payments material consistently describes a decomposition in which the first four characters are the sending bank's IFSC prefix, followed by a character denoting the RTGS system, a channel or message-type character, the date, and a sequence. That description is convention rather than published rule: the RBI's FAQ gives the length and says nothing about the layout, no primary specification of

the internal structure is public, and you should not parse it. NEFT references are commonly quoted as sixteen characters with a similar bank-code-then-date-then-sequence shape, again by convention rather than by published rule.

For IMPS and UPI the identifier consumers are told to quote is the twelve-digit retrieval reference number. It is the same DE37 discussed above, because the National Payments Corporation of India's switch is ISO 8583 derived. A UPI payment additionally carries a transaction identifier assigned by the payment service provider, which is the identifier your app uses internally and which the receiving bank generally cannot resolve.

The practical guidance for anyone building support processes for Indian payments is therefore the mirror image of the card guidance. The twelve-digit RRN is the number both banks can see. The PSP's transaction identifier is the number only your own systems can see. The UTR is the number that exists for RTGS and NEFT and not for UPI. And in every case the RBI's structural point holds: the identifier is an input to a request made by a bank, not a self-service tracking code. When a customer says "I have the UTR, why can't you find it", the honest answer is that the number identifies the payment inside a system they do not have access to, and so do you unless you are a bank.

Design rules that follow

Store every identifier you are given, and never overwrite one with another. The STAN, the retrieval reference number, the approval code, the ARN when it arrives, the scheme transaction identifier, the gateway's own key and your own order identifier are seven different columns, not one. The cost of seven columns is nothing. The cost of having only the one you thought you needed is a research request per customer.

Store them as text, fixed width, with the specification's width. `an 12` means twelve characters, letters permitted, leading zeros significant. `n 23` means twenty-three digits, and twenty-three digits will not fit in a 64-bit integer.

Make every lookup key composite by default. STAN plus terminal plus date plus amount. Retrieval reference number plus acquirer. Approval code plus card plus amount plus date. If a query in your support tooling takes a single identifier and returns a single row, ask which identifier it is, because for three of the four in this chapter that query is lying to you.

Generate your own identifier at the earliest possible moment, before the first network call, and put it everywhere you are allowed to. Where the rail gives you a durable customer-visible reference field — the eighteen characters of a Bacs service user's reference, the thirty-five of an ISO 20022 end-to-end identification — spend it on something you can look up, not on a description of the goods.

And be careful what you tell customers a number means. The one on their receipt is an approval code and it proves that a bank said yes. The one that proves money moved is created hours later by a different institution, and until it exists there is nothing to trace. Both of those statements are true simultaneously, and explaining the difference clearly is most of the job.

28.98 Common wrong ideas

Wrong: the retrieval reference number is twelve digits. **Right:** it is `an 12`, twelve alphanumeric characters whose leading zeros are load-bearing, and any system that stores it as an integer silently destroys it.

Wrong: the leading characters of a retrieval reference number encode a year, a day and an hour, so you can parse them. **Right:** ISO 8583 specifies no internal structure at all; the common layout is a

convention belonging to whoever generated the value, and some acquirers pack a terminal identifier or an opaque sequence in there instead.

Wrong: the approval code identifies the purchase. **Right:** six alphanumeric characters generated without coordination between issuers say only that a yes happened, and matching on them alone produces false matches at any real volume.

Wrong: the system trace audit number is the transaction identifier. **Right:** six digits wrap at a million, the sequence is very often per terminal, and Mastercard's own documentation requires the trace identifier to be quoted alongside the switch serial number because one is not enough.

Wrong: a reversal can carry a fresh trace number. **Right:** a `0400` must quote the original message's trace number so the issuer can find and release the hold, and failing to do so is the direct cause of authorisation holds sitting on customers' statements for a week.

Wrong: every transaction has an acquirer reference number. **Right:** the ARN is created during clearing, so declines, expired authorisations, reversed holds and anything still inside the capture window have none and never will, and "there is no reference because there is no charge" is often the correct answer.

Wrong: you can parse the bank identification number and the capture date out of an ARN and reason about them. **Right:** the format code in the first position exists precisely to signal that the remaining layout may differ, so validate the check digit rather than the semantics.

Wrong: data element 62 means the same thing in every ISO 8583 message. **Right:** it is a private reserved field on card rails and the end-to-end reference in Faster Payments; the number plus the dialect is the meaning, and a decoder not told which specification it is decoding against is guessing.

Wrong: the reference a customer types into a bank transfer is an identifier the system traces on. **Right:** the eighteen characters of a Bacs service user's reference are unvalidated free text, shorter than the thirty-five ISO 20022 allows, and the receiving institution may truncate or reject them.

Wrong: a UTR is a self-service tracking code and every Indian payment has one. **Right:** RTGS has a twenty-two character UTR that a bank quotes on the customer's behalf, IMPS and UPI use the twelve-digit retrieval reference number, and a UPI payment's own service-provider identifier is visible only inside that provider.

28.99 Chapter summary in 20 lines

1. A card payment leaves behind four numbers with four owners, four lifetimes and four jobs, and asking the wrong party for the wrong one gets you a shrug.
2. The system trace audit number in DE11 is six digits assigned by whichever node originates a message and returned unchanged so that the originator can pair the answer with the question.
3. Six digits wrap at a million and the counter is often per terminal, so the same trace number exists simultaneously on every terminal in an estate and was somebody else's transaction yesterday.
4. Searching on a trace number alone is the commonest analytical error in card operations; the key must be composite, adding terminal or acquirer identity, the local date, and usually the amount and currency.
5. A reversal must quote the original trace number, and because the case in which you most need to reverse is the case in which no response arrived, the original must be stored durably before the request is sent.

6. The retrieval reference number in DE37 is twelve alphanumeric characters created by the acquirer and carried forward into the advice, the clearing record and the merchant's settlement report.
7. Its leading characters often look like a year, a day of the year and an hour, but no published specification requires that layout, and a parser assuming it will one day interpret a letter as a month.
8. DE37 is the best single number a merchant can hold, but it is unique only within one acquirer, and collisions between institutions are arithmetically inevitable at scheme volumes.
9. The same field has a second life in India, where the twelve-digit retrieval reference number is what an IMPS or UPI customer is told to quote.
10. The authorisation identification response in DE38 is six alphanumeric characters generated by the issuer, or by the scheme standing in for it, and only when the answer was yes.
11. It is the number printed on the receipt and therefore the one a customer hands you first, and it is also the weakest of the four, encoding nothing about the amount, the card, the merchant or the time.
12. The acquirer reference number is not created at the point of sale at all: it is born during clearing, twenty-three digits made of a format code, a bank identification number, a capture date, a film locator and a check digit.
13. Validate that check digit rather than the semantics, because the format code exists to warn that the rest of the layout may differ.
14. The ARN exists if and only if the transaction cleared, and it is the key on which the entire dispute process runs, which is why issuers tell customers to ask the merchant for it.
15. Nobody owns a number that is present, unchanged, at every hop, which is why Visa created a Transaction Identifier and Mastercard assembled a fifteen-position Trace ID from a network code, a Banknet reference and a settlement date.
16. That construction admits the general rule: identity in payments is almost always composite, and a system with one "transaction ID" column is storing a correlation key and calling it a name.
17. When a customer reports being charged twice, establish first whether there are two charges or one charge and one hold, because refunding an uncleared authorisation creates a credit that settles while the hold quietly expires.
18. Then work outward from your own order identifier, the one number you control, through the gateway or acquirer record for each attempt, through clearing, and only then escalate with the right number to the right party.
19. Account-to-account rails have the same problem in different clothes, from an eighteen-character Bacs reference that nothing validates to a Faster Payments dialect in which data element 62 means something entirely different from what it means on card rails.
20. ISO 20022 and Swift's UETR are what the card world lacks — one value, generated once, unchanged at every hop, unique by construction — which is the whole reason a Swift payment can be tracked end to end and a card payment cannot.

Sources: ISO 8583 data element definitions; Visa BASE II Clearing System Overview glossary; Mastercard Developers field definitions for ARN (DE31), Banknet Reference Number (DE63.2), Trace ID (DE11) and Switch Serial Number (DE63.3); Merchant Risk Council guidance on Visa Transaction Identifier and Mastercard Trace ID; Worldpay ISO 8583 Reference Guide; Open Banking Limited Domestic Payment Message Formats (FPS ISO 8583, Bacs Standard 18 per Bacs PN5011, CHAPS MT103); Pay.UK Faster Payment System ISO 20022 standards library; Bank of England policy statements on ISO 20022 in CHAPS; Reserve Bank of India RTGS FAQ; Oracle India IMPS Payments User Guide; IETF RFC 9562.

29

CHAPTER

Response Codes

29.0 What this chapter gives you

1. You will be able to point to the two-character field that carries the entire verdict of a card authorisation, name it as DE39, and say why a database column typed as an integer is a defect waiting for a Tuesday.
2. You will be able to explain why an issuer refuses to tell a merchant the reason for a decline, and show with arithmetic that a fully specific vocabulary would let an attacker take a bare card number to a working expiry, security code and liveness check in about a thousand probes.
3. You will be able to take a batch of failed subscription charges and split it into the group that deserves a retry timed around payday, the group that needs a new card number, and the group that must never be attempted again.
4. You will be able to name the four separate mechanisms by which blind retrying damages a business, only one of which shows up in the conversion number.
5. You will be able to identify the values that are approvals despite not being `00` — partial approval, VIP approval, honour with identification, no reason to decline — and say what a merchant throws away by treating them as failures.
6. You will be able to read a Mastercard decline as a pair, the response code plus the merchant advice code, and follow it to the correct next action rather than to a queue.
7. You will be able to place a Visa decline in one of the four decline categories and explain why the most common code of all sits in the generic bucket.
8. You will be able to explain why the same two characters mean a PIN verification problem on one network, a fraud decline on another and a token restriction on a third, and why storing the code without the scheme destroys the information.
9. You will be able to distinguish a timeout from a decline and send a reversal carrying the original trace number instead of a second identical request.
10. You will be able to recognise a strong customer authentication soft decline and route it through EMV 3DS rather than retrying it or losing the sale.

The plain version

Think about a door with a bouncer on it.

You want to get into a club. The bouncer looks at you, looks at your ID, thinks for a second or two, and gives you one of two kinds of answer. The first kind is “in you go”. There is only one way to say that. The second kind is “not tonight” — and behind those two words sit fifty different reasons, and the bouncer will not tell you which one applies to you.

He knows perfectly well. Maybe you are too young. Maybe your name is on a list behind the counter. Maybe the fire officer capped the building at three hundred people and you are number three hundred and one. Maybe he does not like the look of the four people standing behind you and has decided to break up the group by refusing the one at the front. But what he says is "not tonight", in the same flat voice, every time.

This is not laziness and it is not bad manners. It is the job. If the bouncer said "your ID is fake, the hologram is wrong", the next person along would bring a better hologram. If he said "you are on the barred list", the barred person would learn to send a friend in first to test the water. Every extra word he gives away is a free lesson for the next person who wants in and should not get in. So the club has a rule: one way to say yes, one flat phrase for no, and the real reason stays inside the building.

Now notice something else about "not tonight". It comes in two flavours that sound identical.

One flavour is temporary. We are full. Come back at midnight and there will be room. The door has not shut on you personally; it has shut on this moment.

The other flavour is permanent. You are barred. You were barred in January and you will be barred in December. Coming back at midnight achieves nothing, and the only thing that will change your position is a conversation with the manager during the day.

And here is the thing that catches people out. If you cannot tell the two apart, the sensible-looking strategy is to keep trying. So you come back at ten past, and twenty past, and half past, and by eleven o'clock the bouncer has stopped seeing a customer and started seeing a problem. You are no longer being refused because the club is full. You are being refused because you are the man who keeps coming back. Your own persistence has converted a temporary no into a permanent one.

A card payment works exactly like this, and the whole thing turns on a box in the reply message that holds two characters. When a card is tapped, a message goes out asking whether this card may be charged nine pounds ninety-nine. A message comes back, and near the front of that reply there is a small fixed slot, two characters wide, carrying the entire answer. If it contains `00`, the answer is yes. If it contains anything else at all, the answer is no. There are about sixty values that mean no, and the difference between a payments operation that works and one that quietly bleeds money is almost entirely in what you do with those two characters.

Take a real example. Ravinder runs a gym in Coventry. He has 1,000 members paying £24.99 a month, which is £24,990 a month if everybody pays, and he collects it all on the first of the month by charging the card each member has left on file. On 1 March he sends 1,000 requests. Nine hundred and forty-seven come back `00`. Fifty-three do not, and this is roughly how they break down:

Code	What it means, roughly	How many	Money
51	Not enough money in the account	31	£774.69
05	Do not honour — no reason given	11	£274.89
54	The card has expired	6	£149.94
41 or 43	Card reported lost or stolen	3	£74.97
91	The member's bank did not answer	2	£49.98
		53	£1,324.47

Now, the wrong thing to do, and the thing an awful lot of software does, is to treat all fifty-three the same. Put them in a queue. Try each of them again four times a day for a week. That is 53 x 4 x 7 = 1,484 additional attempts. It will recover something — a handful of the empty accounts will fill

up when wages land — but it will recover perhaps twenty of them. The other 1,464 attempts achieve nothing at all except to make Ravinder's gym look, to a computer at each member's bank, exactly like somebody testing stolen card numbers. Some of those attempts carry a fee. Some of them make the *next* attempt more likely to be refused.

The right thing to do is to read the two characters and treat the five groups as five different problems.

The thirty-one members with code 51 have a working card and an empty account. Nothing is wrong with the card, the merchant or the amount. What is wrong is the date. Retrying on the second of the month is close to pointless; retrying on the fourth, after most people have been paid, is not. Two attempts, spaced out, will typically bring back most of them. Call it twenty-four of thirty-one, £599.76 recovered, at a cost of sixty-two attempts rather than 868.

The six members with code 54 have a card that expired. There is no number of retries that unexpires a card. The only two things that work are asking the member for a new one, or using the card networks' own updating service, which quietly hands merchants the replacement card number when a bank reissues. Retrying is not merely useless here, it delays the one action that would have worked.

The three members with codes 41 and 43 have had their cards reported lost or stolen. That card is dead. It will be dead tomorrow and dead in a year, and both card networks will charge Ravinder for continuing to ask.

The two members with code 91 did not get an answer because their bank's computer was not responding. This one genuinely does deserve an immediate retry. It is the only code in the list where "just try again" is the whole correct answer.

And then there are the eleven with code 05, do not honour, which is the bouncer's flat voice. Do not honour means the member's bank has decided not to pay and is not going to say why. It might be a closed account, a fraud score, or the member having told their bank to stop this subscription and the bank doing as it was asked. Ravinder cannot tell, and — this is the important part — he is not supposed to be able to tell. The correct handling is one further attempt after several days, and if it comes back 05 again, an email to the member. Not a queue.

Total: about ninety extra attempts instead of 1,484, and about £800 of the £1,324 recovered instead of about £500. Same fifty-three customers. The only difference is that somebody read the two characters.

That is the plain version. One value means yes, everything else means no, and the no is deliberately vague because a specific no is a free lesson for a criminal. There are two kinds of no, temporary and permanent, and they look identical from the outside, which is precisely why blind retrying is so tempting and so expensive.

Where the plain version stops being true

The bouncer is not always the bank, and often nobody's bank was asked. The analogy has one door and one man on it. A real card transaction can be refused at five different points, and the two-character code you eventually see does not say which. The chip in the card can decline the transaction by itself, offline, before any message goes anywhere. The terminal or the point-of-sale application can refuse it on a floor limit or a configuration rule. The merchant's own fraud screening can block it before it ever reaches the acquirer. The acquirer or its gateway can reject it for a malformed field, an unsupported currency, or a merchant-level velocity rule. The card scheme itself can answer on the issuer's behalf when the issuer's systems are unreachable, using parameters the issuer lodged in advance — Visa calls this stand-in processing. Only after all of those does the actual issuer get a vote. So "your bank

declined this" is a sentence customer service staff say many times a day and which is frequently untrue. Worse, several of those layers write into the same field using the same vocabulary, so the shape of the answer tells you nothing about its author.

"00 is yes and everything else is no" is a good rule with real exceptions, and you must know them. Code 10 is a partial approval: the issuer has approved a smaller amount than you asked for, and if your software treats that as a decline you have just thrown away a sale you had already won, and possibly left a hold on the customer's account for money you will never take. Code 11 is an approval. Code 08, honour with identification, is an approval conditional on the merchant checking something. Code 85 is not a decline at all — it is the answer to a question you asked without wanting money, such as "is this card real and does this postcode match", and it means "no reason to decline". And `00` in the response to a *reversal* means the reversal was accepted, which tells you nothing whatever about whether the original transaction was approved. Meanwhile `00` on an authorisation does not mean money has moved. It means a bank has set some of its customer's money aside and promised to pay if you ask properly. Those are different facts and they live in different chapters.

The vagueness is being unwound on purpose, but only in one direction. The plain version says the reason stays inside the building. That was true for thirty years and it is becoming less true, because the card schemes worked out that total opacity was costing everybody money in failed recurring payments. Mastercard now requires issuers to split their generic declines into three buckets and to attach a separate advice field telling the merchant what to do next. Visa now classifies every decline code into one of four categories and penalises merchants who ignore the classification. But look carefully at what the schemes added. They added *what to do*, not *why*. "Do not try again" is a much more useful message than "do not honour", and it still does not tell the merchant that the account was closed on 3 March because the customer died. The direction of the disclosure is deliberate: the merchant learns enough to stop wasting attempts and not one thing more about the cardholder. The reason itself is still available, but only to the cardholder, from their own bank, over a channel where the bank has already checked who it is talking to.

Soft and hard are not properties of the code. Nowhere in the international standard that defines this field is there any concept of a soft decline or a hard decline. Those are operational categories invented afterwards by the card schemes, published in rulebooks that are revised on an annual cycle, and they differ between schemes, differ between regions, and move. A code that sits in the never-retry bucket this year may be moved to the retry-later bucket next year by a bulletin. More disconcertingly, the same two characters mean genuinely different things on different networks: code 79 means "already reversed by the switch" on Visa and "life cycle" on Mastercard, which are not remotely the same event. Any system that stores a decline as a normalised boolean, or that stores the code without also storing which scheme produced it, has destroyed the information it will need later.

The technical version

Where the field sits

Card authorisation messages are ISO 8583 structures. The message type indicator is four digits: position one gives the version of the standard (0 for the 1987 edition, 1 for 1993, 2 for 2003), position two the message class (1xx authorisation, 2xx financial, 3xx file actions, 4xx reversal and chargeback, 8xx network management), position three the function (0 request, 1 request response, 2 advice, 3 advice response), and position four the origin (0 acquirer, 2 issuer, 4 other). An authorisation request is 0100 and its response 0110; a financial request is 0200 and its response 0210; a reversal request is 0400 and its response 0410; a network management request is 0800 and its response 0810.

After the MTI comes the primary bitmap, sixty-four bits marking the presence of data elements 1 to 64, with bit 1 signalling a secondary bitmap covering 65 to 128. Data element 39, the response code, is an 2 in the 1987 catalogue: exactly two alphanumeric characters, fixed length, no more and no less. It is present in response messages — 0110, 0210, 0410, 0430, 0810 — and absent from requests.

The elements it must be read alongside are these.

Element	Format	Name	Why it matters here
DE4	n 12	Amount, transaction	On a partial approval the response amount is not the request amount
DE11	n 6	System trace audit number	Pairs the response to the request
DE38	an 6	Authorisation identification response	Present on approval only
DE39	an 2	Response code	The answer
DE44	an ..25	Additional response data	Where address and card-verification results are commonly carried
DE48	an ...999	Additional data, private	Where Mastercard carries the merchant advice code
DE49	n 3	Currency code, transaction, per ISO 4217	An approval is an approval of an amount in a currency
DE54	an ...120	Additional amounts	Balances and remaining amounts on some approvals
DE55	ans ...999	Integrated circuit card data	The chip's own verdict travels here

Two implementation notes follow immediately from that table and are worth stating flatly. DE38 is populated only when the transaction is approved, so an implementation that decides "approved" by checking whether an authorisation code is present is doing the right thing for the wrong reason and will eventually meet an issuer that populates DE38 with `000000` on a decline. DE39 is the authoritative field; everything else is corroboration. And DE39 is two characters, not two digits: values such as `1A`, `N7`, `R0`, `Q1`, `Z3`, `5C` and `9G` are all in live use, so a column typed as an integer is a defect waiting for a Tuesday.

The 1987 catalogue

Almost all card scheme traffic still uses the two-character response codes of the 1987 edition, and the base catalogue is worth having in front of you. It is grouped by intent rather than numerically, which is why the numbering looks arbitrary.

Code	ISO 8583:1987 meaning
00	Approved or completed successfully
01	Refer to card issuer
03	Invalid merchant
04	Pick-up

Code	ISO 8583:1987 meaning
05	Do not honor
06	Error
07	Pick-up card, special condition
08	Honour with identification
10	Approved for partial amount
11	Approved (VIP)
12	Invalid transaction
13	Invalid amount
14	Invalid card number (no such number)
15	No such issuer
19	Re-enter transaction
30	Format error
33	Expired card
34	Suspected fraud
38	Allowable PIN tries exceeded
41	Lost card
43	Stolen card, pick-up
51	Not sufficient funds
54	Expired card
55	Incorrect personal identification number
57	Transaction not permitted to cardholder
59	Suspected fraud
61	Exceeds withdrawal amount limit
62	Restricted card
65	Exceeds withdrawal frequency limit
75	Allowable number of PIN tries exceeded
91	Issuer or switch is inoperative
93	Transaction cannot be completed. Violation of law
94	Duplicate transmission
96	System malfunction

The catalogue reserves 45 to 50 and 69 to 74 for ISO use, 97 to 99 for national use, and the ranges beginning with a letter for national and private use, which is where every scheme's proprietary values live. Note the duplication the standard itself contains: 33 and 54 both say expired card, 34 and 59 both say suspected fraud, 38 and 75 both say PIN tries exceeded. The difference historically was whether the card was to be captured. That distinction is dead and the duplication survives, which is one small demonstration that this vocabulary is an archaeological site rather than a design.

The 1993 renumbering, and why you have probably never seen it

The 1993 revision of ISO 8583 replaced the two-character response code with a three-digit action code, still in DE39, and gave it a genuinely rational structure: the range says what happened, the value says why.

Range	Meaning
000-099	Approved
100-199	Denied, card not to be captured
200-299	Denied, card to be captured

Range	Meaning
300-399	Result of a file action
400-499	Result of a reversal or chargeback
500-599	Result of a reconciliation
600-699	Result of an administrative or retrieval request
700-799	Fee collection results
800-899	Network management results
900-999	Could not be processed, or liability transfer rejected

Inside those ranges the values are systematic. 000 is approved, 002 approved for partial amount, 007 approved and update the chip. 100 is do not honour, 101 expired card, 116 not sufficient funds, 117 incorrect PIN. The 200 range is the same list again but with the card to be captured: 200 do not honour, 208 lost card, 209 stolen card. And 907 is card issuer or switch inoperative, 911 card issuer timed out, 912 card issuer unavailable — three distinct failures that the 1987 catalogue collapses into 91.

This is a better design in every respect and it is not what the card networks run. Visa, Mastercard, American Express and Discover all froze their dialects on the 1987 shape and then extended it privately, so the 1993 action codes appear mainly in domestic switches, ATM networks and newer national schemes. A practitioner must know which one is on the wire before writing any comparison, because `10` and `100` are an approval and a decline respectively and are distinguishable only by knowing the version.

The published Visa set

Visa publishes its action code list openly on Visa Developer, and it is the closest thing to an authoritative reference for what a merchant will actually receive on Visa rails. The list includes the ISO base plus a substantial private extension. Some entries that matter:

Code	Visa's published description
00	Approved and completed successfully
04	Pick up card (no fraud)
05	Do not honor
07	Pick up card, special condition (fraud account)
10	Partial approval
14	Invalid account number (no such number)
41	Lost card, pick up (fraud account)
46	Closed account
54	Expired card or expiration date is missing
57	Transaction not permitted to cardholder
61	Exceeds approval amount limit
62	Restricted card (card invalid in this region or country)
78	"Blocked, first used" — Transaction from new cardholder, and card not properly unblocked
79	Already reversed (by Switch)
80	No financial impact
82	Negative CAM, dCVV, iCVV, or CVV results
85	No reason to decline a request for address verification, CVV2 verification, or a credit voucher or merchandise return

Code	Visa's published description
91	Issuer or switch inoperative and STIP not applicable or not available for this transaction; Time-out when no stand-in
94	Request is identified as a duplicate
1A	Additional customer authentication required
N0	Force STIP
N7	Decline for CVV2 failure
R0	Stop this Payment
R1	Stop all future Payments
R3	Stop all Merchants
Z3	Unable to go online; offline-declined
5C	Transaction not supported/blocked by issuer
9G	Blocked by cardholder/contact cardholder

Several of these repay attention. `R0`, `R1` and `R3` are the cardholder speaking through their bank: the customer has instructed the issuer to stop this payment, all future payments to this merchant, or all merchant-initiated payments respectively. A merchant that retries an `R1` is not fighting a machine, it is overriding an instruction its customer gave to their own bank, and both the scheme rules and consumer protection law take a dim view of it. `85` is the affirmative answer to a zero-amount verification, which is how a merchant checks that a card is real and that a billing address matches without taking any money. `80`, "no financial impact", is what a reversal response carries when the original was declined anyway.

Why the issuer is deliberately vague

The vagueness of `05` is not a documentation gap that somebody will fix. It is a control, and the argument for it is arithmetic.

Consider an attacker holding a list of 10,000 candidate card numbers of unknown quality, obtained from a breach, generated against a known issuer range, or scraped. They want to convert that list into a smaller list of numbers that can actually be charged. Now suppose issuers returned a fully specific decline for every case. Watch what the attacker learns from a single one-pound probe against a single card:

A response of 14, invalid account number, tells them the number does not exist. Cross it off.

A response of 54, expired card, tells them something far more valuable: the number *does* exist and their expiry date is wrong. An expiry is a month and a year, and cards are typically issued for three to five years, so a complete search of the plausible space is at most sixty combinations.

A response of N7 or 82, a card verification failure, tells them the number and expiry are both correct and only the three-digit security code is wrong. That is a thousand probes, with the response code confirming the moment they hit it.

A response of 51, not sufficient funds, tells them everything is correct: number, expiry, security code, and a live account that is merely empty at this instant.

So a fully specific decline vocabulary converts an unstructured guess into a staged search with a feedback signal at every stage: roughly 1,061 probes to take a bare card number and establish its expiry, its security code and its liveness. That is the definition of an oracle, and building one into the response field of the world's payment network would be an extraordinary thing to do.

Collapse all of it to `05` and the attacker learns exactly one bit per probe: not this one, for some reason, at this moment. The staged search collapses. This is why issuers' risk engines map dozens of distinct internal outcomes — negative file hit, velocity breach, device reputation, machine-learning score above threshold, account under investigation, merchant category blocked by the cardholder, sanctions screening alert, a mismatch the issuer would rather not name — onto a single unhelpful value.

There is a second reason, and it is legal rather than technical. Some declines cannot be explained to a merchant without breaking the law. If a payment is stopped because a sanctions or anti-money-laundering screen fired, telling the merchant why can constitute tipping off. Code 93, "transaction cannot be completed, violation of law", is about as much as anyone will say, and many issuers will not even say that much and will send `05` instead. Similarly, where the decline is about the *merchant* rather than the cardholder — an issuer that has decided to block a whole category, or a specific acceptor, because of chargeback history — an explicit code would amount to publishing a commercial judgement about a business to that business.

The disclosure is therefore directional rather than absent. The cardholder can find out. They ring the number on the back of the card, the bank authenticates them, and the bank tells them exactly what happened, because it is their money and their account. The merchant cannot, because the merchant is not the customer and is, statistically, sometimes the attacker. The right thing for a merchant to say to a customer holding a declined card is not "your bank rejected it" but "your bank has not told us why; if you ring them they will tell you".

Soft and hard, in the schemes' own words

Because "do not honour" wrecked recurring billing, both major schemes built classification systems on top of the code set. They are not the same system.

Visa assigns every decline code to one of four decline categories:

Category	Meaning	Merchant action
1	The issuer will never approve	Never reattempt with the same account details
2	The issuer cannot approve at this time	Wait, then reattempt within limits
3	The issuer cannot approve based on the details provided	Correct the data, then reattempt within limits
4	Generic response	Treated like category 2 for retry purposes

Category 1 is populated with the codes where the account or the card is simply gone: 04, 07, 12, 14, 15, 41, 43, 46, 57, and the stop-payment family R0, R1 and R3. Category 2 holds the temporary conditions: 03, 19, 51, 59, 61, 62, 65, 75, 78, 86, 91, 93, 96, N3, N4, 5C, 9G. Category 3 holds the data problems, which is where the retry actually has to change something before it is sent: 54, 55, 70, 82, 1A, 6P, N7. Category 4 is everything else, and — this is the sharp part — `05`, do not honour, sits in category 4. The generic code is in the generic category, and Visa has been pressing issuers to reduce their use of it in favour of a code from categories 1 to 3, because the whole scheme is worth less when the biggest single bucket is unclassifiable.

Mastercard went the other way. Rather than classifying the existing codes, it introduced three replacement values in DE39 and a separate advice field. Mastercard's own developer documentation defines

them plainly, noting that the response code "is passed to Mastercard in Data Element (DE) 39 of the issuer's network response message":

Code	Mastercard meaning
05	Do not honor
79	Life cycle (Mastercard use only)
82	Policy (Mastercard use only)
83	Fraud/Security (Mastercard use only)

So a Mastercard decline that would once have arrived as 14 or 54 now arrives as 79, life cycle, meaning something about the credential itself has changed. A decline that reflects an issuer rule arrives as 82, policy. A decline that reflects a fraud judgement arrives as 83. And alongside it, in DE 48 subelement 84, comes the merchant advice code, which is where the actual instruction lives.

MAC	Meaning
01	New account information available
02	Cannot approve at this time, try again later
03	Do not try again
04	Token requirements not fulfilled for this token type
05	Negotiated value not approved
21	Payment cancellation
24	Retry after 1 hour
25	Retry after 24 hours
26	Retry after 2 days
27	Retry after 4 days
28	Retry after 6 days
29	Retry after 8 days
30	Retry after 10 days
40	Consumer non-reloadable prepaid card
41	Consumer single-use virtual card number

Codes 24 through 30 are the most quietly remarkable thing here: the issuer telling the merchant, in the decline message itself, when the account is likely to have money in it. That is the answer to code 51 delivered as data instead of guesswork.

The Mastercard Switch Rules set out the intended pairings directly:

DE39	DE 48 subelement 84	Merchant advice	Merchant action
79 or 82	01	Updated information needed	Updated information found to be available in Mastercard ABU database; secure new information before reattempting
79 or 82	03	Do not try again	Updated credentials were not found to be available in Mastercard ABU database; do not retry

DE39	DE 48 subelement 84	Merchant advice	Merchant action
83	01	Additional information needed	Ensure card information is correct; authentication may improve likelihood of an approval — try using authentication such as EMV 3DS
83	03	Do not try again	Suspected fraud — do not retry
79, 82 or 83	02	Try again later	Retry transaction later

Read that table as a decision tree and the design becomes obvious. `79` or `82` with MAC `01` means go to the Account Billing Updater and get the new card number. `83` with MAC `01` means the credential is probably fine but the transaction was not authenticated, so run it through EMV 3DS and come back. Anything with MAC `03` means stop, permanently. MAC `21`, payment cancellation, means the cardholder has cancelled the recurring arrangement and the merchant's next move is a conversation, not a request.

Retry logic, and the four ways blind retries make it worse

Practitioners talk about retries as though the only question were how many attempts convert. There are four separate mechanisms by which unstructured retrying damages a business, and only one of them is visible in the conversion number.

The schemes charge for it. Visa operates an Excessive Reattempts rule under which a reattempt of a transaction that returned a category 1 decline is chargeable from the very first reattempt, and reattempts beyond a threshold within a rolling thirty-day window are chargeable for categories 2 and 3. The commonly published threshold is fifteen reattempts in thirty days, on the same account number at the same merchant, with the fee applying from the sixteenth; some acquirers publish the limit as twenty, and both figures appear in current merchant-facing guidance, which tells you that the rule has been revised and that regional variants exist. Mastercard runs an equivalent programme within Transaction Processing Excellence, assessing a fee for authorisation attempts beyond a threshold of previously declined attempts on the same account number, at the same card acceptor, and for the same amount, with published thresholds around ten declined attempts in twenty-four hours or thirty-five in a rolling thirty days, again with explicit regional variation. Mastercard additionally charges where a merchant retries a transaction that was declined with MAC `03` or MAC `21` and receives another `03` or `21`. The per-attempt amounts sit in the schemes' own fee schedules, which are not published, so no primary document will confirm a figure; what can be checked is what the acquirers and payment service providers who pass the charge on tell their merchants, and they quote it consistently. Adyen's published scheme fee documentation, to take one, puts the Mastercard excessive authorisations fee at USD 0.30 in the United States rising to USD 0.50 from the first of January 2025, EUR 0.55 in Europe, and USD 0.50 across Asia-Pacific, Canada and Latin America, and notes that the permitted retry counts themselves vary by region. The engineering conclusion does not depend on the figure: cap attempts below the lowest published threshold in any region you operate in, count them per account number per merchant per thirty days, and treat category 1 and MAC 03 as absolute.

The issuer scores you. Every authorisation request a merchant sends is a data point in the issuer's model of that merchant. A retry pattern of two attempts spaced four days apart looks like a subscription business. Four attempts a day for a week against cards that keep saying no looks like card

testing, because that is precisely what card testing looks like. Issuers respond by declining more of that merchant's traffic, including the good traffic, and the merchant discovers that its approval rate on healthy cards has fallen for reasons nobody can point at. The retry strategy that recovered an extra one per cent of failed subscriptions has cost two per cent of successful ones.

A timeout is not a decline, and treating it as one creates duplicates. This is the failure that produces the angriest customers. If a request goes out and no response comes back, the correct action is a reversal — a 0400 carrying the original message's trace number — followed by a fresh attempt. It is not a retry of the same request, because the first request may well have been approved by an issuer whose response was lost on the way home. Retry without reversal and the customer has two holds on their account for one purchase. This is why every terminal and gateway implementation must write a durable local record of the request, including the trace number, *before* the request goes on the wire: the case in which you most need to reverse is exactly the case in which nothing came back to tell you what to reverse.

Every attempt costs money even when it fails. Authorisation requests carry scheme fees, gateway fees and, in some acquiring arrangements, per-transaction charges regardless of outcome. At any volume the difference between ninety attempts and 1,484 is not a rounding error.

The retry logic that follows from all of this is short enough to state in full. Read the merchant advice code first if there is one, because it is an instruction and it overrides your own heuristics. If there is none, read the scheme's category. Never retry a Visa category 1 code, a Mastercard MAC 03 or 21, or any of R0, R1, R3. For data-quality declines, do not retry until the data has changed — a new expiry from the account updater, a corrected security code, a completed 3DS authentication. For temporary declines, retry on a schedule shaped by the customer's payday rather than by your batch window, and prefer the interval the issuer told you in MAC 24 to 30 if it gave you one. Cap total attempts well under the scheme threshold. Store the raw code, the scheme that produced it, the advice code and the timestamp for every attempt, and never collapse the outcome to a boolean, because the day you need to explain a fall in approval rates is the day you will need the codes.

Scheme differences in the code sets

The single most dangerous assumption in this area is that a two-character code means the same thing everywhere. It does not, and the collisions are not obscure.

Code	Visa	Mastercard	Discover
51	Not sufficient funds	Insufficient funds / over credit limit	Decline
70	PIN data required	Contact card issuer	not published in the same set
79	Already reversed (by Switch)	Life cycle	not published in the same set
82	Negative CAM, dCVV, iCVV, CVV, CAVV, dCVV2, TAVV or DTVV results	Policy	not published in the same set
83	Cannot verify PIN	Fraud/Security	Domain Restriction Controls Failure
85	No reason to decline	Not declined; valid for all zero amount transactions	No reason to decline

Code	Visa	Mastercard	Discover
1A	Additional customer authentication required	not used in this sense	Customer authentication required

Code 83 is the cleanest illustration: on Visa a PIN verification problem, on Mastercard a fraud decline, on Discover a token domain restriction failure. Three networks, one value, three unrelated events. A normalisation layer that maps `83` to a single internal meaning is wrong on at least two of them.

American Express does not use the two-character scheme at all in its own dialect. Its published set is three digits wide: `00` approved, `100` deny, `101` expired card or invalid expiration date, `109` invalid merchant, `110` invalid amount, `111` invalid account, `116` not sufficient funds, `117` invalid PIN, `122` invalid card security code, `130` additional customer identification required, `181` format error, `200` deny, pick up card, `912` issuer not available. An integration that assumes two characters will truncate every one of them.

The strong customer authentication soft decline deserves its own note because it is the newest common case and it is not handled correctly in a great deal of live code. Under the second Payment Services Directive an issuer may refuse a card-not-present transaction not because there is anything wrong with it but because the cardholder has not been authenticated. Visa signals this with `1A`, "additional customer authentication required". Mastercard signals it with `65`. American Express's `130`, "additional customer identification required", occupies the same position in its set. The correct response is neither a retry nor a failure message to the customer: it is to run the transaction through EMV 3DS, obtain the authentication data, and submit a new authorisation carrying it. A merchant that treats `1A` as a hard decline is turning a completable sale into a lost one, and a merchant that retries the identical message without authentication will receive `1A` again, indefinitely, while accumulating reattempt fees.

Note also that Mastercard's `65` is doing double duty — the ISO meaning is a withdrawal frequency limit and the European meaning in card-not-present contexts is an authentication requirement. Disambiguating it requires the context of the transaction rather than the code alone, which is a good final reminder that DE39 is two characters and two characters cannot carry a complete explanation of anything.

Codes the network writes rather than the bank

A final category worth separating out, because it changes what the right action is.

When an issuer's systems are unreachable, Visa's V.I.P. system can authorise on the issuer's behalf using parameters the issuer has lodged in advance. This is stand-in processing, and its fingerprints are all over the code set. Visa's own description of `91` is not "the issuer declined" but "issuer or switch inoperative and STIP not applicable or not available for this transaction; time-out when no stand-in". That is a statement about network availability, and it is the one decline where an immediate retry is genuinely the right move, because the condition it describes is measured in minutes. `N0`, "force STIP", is an issuer deliberately routing a transaction into stand-in rather than answering it itself. When stand-in approves, the approval code in DE38 was minted by the scheme and the issuer may not know about it until the advice message arrives, which is a small but real source of reconciliation surprise.

At the other end, `Z3`, "unable to go online; offline-declined", is written by the terminal after the chip demanded an online authorisation and the terminal could not get one. No bank saw the transaction at all. A chip that declines outright produces an application authentication cryptogram rather than an authorisation request cryptogram, and the terminal reports the refusal without any message leaving

the shop. In both cases the customer is told their card was declined while their bank has no record of anything having happened, which is exactly the conversation that ends with a customer insisting there is nothing wrong with their card and being entirely correct.

The practical instruction that closes the chapter is a small one. Log DE39 verbatim, as characters, alongside the scheme, the acquirer, DE38, DE44, the merchant advice code from DE48 where present, DE4, DE49, the MTI and the time. Do not normalise on the way in. Do not map to a boolean. The two characters are the most compressed piece of information in the entire payment system, and every layer that helpfully simplifies them destroys the only evidence you will have when somebody asks, six months later, why the approval rate moved.

29.98 Common wrong ideas

Wrong: A decline means the cardholder's bank refused the payment. **Right:** A transaction can be refused at five points — the chip, the terminal, the merchant's own fraud screening, the acquirer or its gateway, and the scheme standing in for the issuer — and the two characters do not say which, so "your bank declined this" is frequently untrue.

Wrong: `00` means yes and every other value means no. **Right:** Code 10 is a partial approval, 11 is an approval, 08 is an approval conditional on identification and 85 is the affirmative answer to a zero-amount verification, so software that treats anything other than `00` as a failure discards sales it had already won.

Wrong: An approval means the money has moved. **Right:** It means a bank has set some of its customer's money aside and promised to pay if you ask properly, and `00` in a reversal response says only that the reversal was accepted, which tells you nothing about the original.

Wrong: Soft and hard declines are categories defined by the standard. **Right:** Nowhere in ISO 8583 is there any such concept; the categories were invented afterwards by the schemes, published in rulebooks revised on an annual cycle, and they differ by scheme, differ by region and move.

Wrong: The schemes are gradually opening up the reasons behind declines. **Right:** What Visa's categories and Mastercard's merchant advice codes added is what the merchant should do next, not why the issuer said no; the reason itself remains available only to the cardholder, from their own bank, over an authenticated channel.

Wrong: A two-character code means the same thing on every network. **Right:** Code 83 is a PIN verification problem on Visa, a fraud decline on Mastercard and a token domain restriction failure on Discover, and American Express does not use two characters at all, so a normalisation layer that maps a code to one internal meaning is wrong on most of them.

Wrong: If no response comes back, send the request again. **Right:** The first request may well have been approved by an issuer whose response was lost on the way home, so the correct action is a reversal carrying the original trace number followed by a fresh attempt; retrying without reversing gives the customer two holds for one purchase.

Wrong: Retrying a decline costs nothing but a little effort. **Right:** The schemes charge for reattempts beyond published thresholds and from the very first reattempt on the never-retry category, issuers read the pattern as card testing and decline more of the merchant's healthy traffic, and every attempt carries scheme and gateway fees whatever its outcome.

Wrong: A stop-payment code is just another decline to work around. **Right:** R0, R1 and R3 are the cardholder speaking through their bank to stop this payment, all future payments to this merchant, or

all merchant-initiated payments, and a merchant that retries one is overriding an instruction its own customer gave to their own bank.

Wrong: Recording the outcome as approved or declined is enough. **Right:** Collapsing the response to a boolean, or storing the code without the scheme that produced it, destroys the only evidence available on the day somebody asks why the approval rate moved six months ago.

29.99 Chapter summary in 20 lines

1. A card authorisation reply carries its entire verdict in a single two-character field, DE39, and everything else in the message is corroboration.
2. `00` means approved, about sixty other values mean refused, and the gap between a payments operation that works and one that quietly bleeds money is almost entirely in what is done with those two characters.
3. The refusal is deliberately vague, because a specific reason is a free lesson for the next person who should not get in.
4. The arithmetic is the argument: a fully specific decline vocabulary turns an unstructured guess into a staged search, taking a bare card number to its expiry, its security code and its liveness in roughly a thousand probes.
5. Collapsing everything to `05`, do not honour, reduces the attacker to one bit per probe, and some declines — sanctions and money-laundering stops — cannot lawfully be explained to a merchant at all.
6. The disclosure is therefore directional rather than absent: the cardholder can ring their bank and be told exactly what happened, and the merchant cannot, because the merchant is statistically sometimes the attacker.
7. Refusals come in two flavours that look identical from the outside, temporary and permanent, which is precisely why blind retrying is so tempting and so expensive.
8. Ravinder's gym puts numbers on it: fifty-three failures treated identically generate 1,484 attempts and recover about £500, while treating them as five different problems generates about ninety attempts and recovers about £800.
9. Insufficient funds is a date problem that answers to a retry timed around payday, while an expired card, a lost card and a stolen card answer to nothing a retry can do.
10. Code 91 is the one entry in the list where "just try again" is the whole correct answer, because it describes a network outage measured in minutes.
11. Several values that are not `00` are nevertheless approvals — partial approval, VIP approval, honour with identification, no reason to decline — and treating them as declines gives away sales already won.
12. An approval is not a payment: it is a promise to pay if the merchant presents properly, and the authorisation identification response is populated only on approval, so deciding "approved" from its presence is the right answer for the wrong reason.
13. The 1993 revision replaced the two characters with a rationally structured three-digit action code, the card networks never adopted it, and as a result `10` and `100` are an approval and a decline distinguishable only by knowing which version is on the wire.
14. Soft and hard are not properties of the code but operational categories layered on afterwards, published in annually revised rulebooks and differing by scheme and by region.
15. Visa sorts every decline into four categories — never reattempt, cannot approve now, cannot approve on this data, and generic — with `05` sitting in the generic bucket the scheme is pressing issuers to shrink.

16. Mastercard went the other way, replacing the code with life cycle, policy and fraud values and putting the actual instruction in the merchant advice code, whose retry-after values tell the merchant how many days to wait.
17. Blind retrying damages a business in four ways: scheme reattempt fees, issuers scoring the traffic as card testing, duplicate holds when timeouts are mistaken for declines, and a per-attempt cost on every message.
18. A timeout demands a reversal carrying the original trace number, which is why that trace number must be written to durable local storage before the request ever goes on the wire.
19. The strong customer authentication soft decline is neither a retry nor a lost sale but an instruction to authenticate through EMV 3DS and submit a new authorisation carrying the result.
20. Log the response code verbatim as characters alongside the scheme, the acquirer, the authorisation code, the additional response data, the merchant advice code, the amount, the currency, the message type and the time, and never normalise on the way in, because every layer that helpfully simplifies those two characters destroys the evidence.

Sources: ISO 8583:1987 and ISO 8583:1993 data element and response code catalogues as tabulated in the ISO 8583 reference literature; Visa Developer published Request and Response Codes (action code, address verification result, CVV2 result and CAVV result tables); Visa BASE I / V.I.P. System technical specification descriptions of stand-in processing and offline-decline codes; Mastercard Switch Rules (Mastercard, 2016-2025) on DE 39 and DE 48 subelement 84 pairings; Mastercard Developers network response code and merchant advice code definitions; Worldpay Access published scheme response code tables for Visa, Mastercard, American Express and Discover, including Visa decline category assignments; Adyen raw acquirer response documentation and 3-D Secure soft-decline guidance; CardPointe and Payway merchant guidance on the Visa Excessive Reattempts rule; PayPal Braintree and TabaPay published merchant advice code tables.

30

CHAPTER

Authorisation, Clearing, Settlement

30.0 What this chapter gives you

1. You will be able to explain why a customer's available balance falls in under two seconds while not one unit of currency has moved anywhere.
2. You will be able to separate authorisation, capture, clearing, settlement and merchant funding as five events with five different units of work, and say which of them actually move money.
3. You will be able to answer the question every support queue receives — why a refund takes days when the payment took seconds — without blaming the merchant, who in most cases did the work on day zero.
4. You will be able to explain to a driver at a motorway pump why £99.00 became £68.40 and why nobody has refunded them £30.60.
5. You will be able to say why a hotel that has cleared and reversed correctly can still leave a hold sitting on a customer's card, and why the customer's own bank is the only party that can lift it.
6. You will be able to code an estimated authorisation and its increments so that the indicators and the linking identifiers survive into clearing, which is the difference between a clean transaction and dispute liability.
7. You will be able to say why a reversal carries the original total in the amount field with the corrected figure in the replacement amount field, and what happens to the hold when those two are swapped.
8. You will be able to explain why storing a card for later use should be done with a zero-amount account verification rather than a one-pound test, and what the schemes now charge for the latter.
9. You will be able to model an authorisation as three joined records with three lifecycles rather than one row with a status column, and explain why a partly captured, partly reversed authorisation cannot fit in one row.
10. You will be able to recognise when you are in a single message system and stop applying dual-message vocabulary such as capture, void and batch cut-off, several of which do not exist there.

The plain version

On Amira's street the pocket money does not live in the children's pockets. It lives in a biscuit tin on Mrs Bello's kitchen shelf.

Mrs Bello has kept the tin for thirty years. Every child on the street has a page in a notebook beside it, and the page says how much of the tin is theirs. Amira's page says £62.30. There is no envelope in the tin with Amira's name on it. There is no separate £62.30. There is one tin with a great deal of money in it and a notebook that says who it belongs to. This matters later, so hold onto it.

On Thursday the ice cream van comes round. Amira wants a lolly. She does not have £62.30 in her hand; her money is in the tin. So the driver leans out of the window and shouts down the street to Mrs Bello: "Amira, up to six pounds, is that all right?"

Mrs Bello looks at the notebook, sees £62.30, and says yes. Then she does something very specific. She does not open the tin. She does not take out any money at all. She writes on a slip of paper "up to £6.00 promised to the ice cream van", clips it to Amira's page, and calls back "yes, that's fine".

Now look at what has and has not happened. The tin still contains exactly as much money as it did a minute ago. Not one coin has moved. But Amira's page now reads £62.30 with a £6.00 slip clipped to it, which means the most she can spend on anything else today is £56.30. Ask her how much money she has and the honest answer is two different numbers at once. She *has* £62.30. She can *spend* £56.30.

That is the whole trick, and it is the thing almost nobody knows. When your banking app says your money is gone, your money is usually not gone. A promise has been made about it. The promise is instant. The money is slow.

Why did the driver ask for six pounds when a lolly is four? Because he did not know yet. Amira might want a flake in it. She might buy one for her brother. He asked for enough to cover the likely worst case, because if he hands over the ice cream and then finds there is no money, the ice cream is not coming back. This is exactly why a hotel puts a hold on your card that is bigger than the room rate, and exactly why a petrol station holds an amount you never spent. Nobody knows what the final number is at the moment they have to commit.

That evening, at six o'clock, the driver goes back to the depot and writes out the day's notes. He does not write one note per ice cream and post them one by one. He writes the whole day at once, all forty-one sales, in one bundle, and puts the bundle in the post. Amira's line in the bundle says: **£4.20**.

The next morning the bundle arrives. Mrs Bello reads Amira's line, takes the slip off the page, opens the tin, and takes out £4.20. Only now has money moved. Amira's page reads £58.10, and she can spend £58.10, because the two numbers have finally become one number again.

Notice three separate events, at three completely different speeds.

The shout down the street took two seconds. That is **authorisation**. It reserves.

The bundle of notes took a day. That is **clearing**. It presents the real amount and says what actually happened.

And Mrs Bello opening the tin is **settlement**. That is the only step at which money actually moves.

There is one more layer, and it is the one that explains why nothing in this business is ever as fast as it looks. Mrs Bello does not walk £4.20 down to the ice cream depot. The depot has its own tin, kept by Mr Doyle, and on Friday afternoon Mrs Bello and Mr Doyle meet on the corner and square up the entire week. Amira's £4.20 has by then been added to four hundred other people's ice creams, and

subtracted from the eleven that got returned, and the whole street's week has become one figure and one handshake. Amira's £4.20 never made a journey of its own. It was absorbed into a number.

Now some things that go wrong, because they all follow from the same picture.

Amira changes her mind before the driver hands anything over. He shouts "cancel that one" and Mrs Bello unclips the slip immediately. Her page goes back to £62.30 within the minute. That is a **void**, or in the trade, a reversal, and it is the only fast way to undo a hold.

The driver forgets to shout. The slip stays clipped. There is no note in the bundle that evening, or the next, or the next. Amira's page still says she can only spend £56.30 and she cannot understand why. Mrs Bello has a private rule about this: if no note arrives within a week, she takes the slip off anyway. That is an **authorisation expiring**. It is worth being precise about whose rule that is. It is Mrs Bello's rule. Not the driver's, not the depot's. The person holding the money decides when to stop holding it, and everybody else can only ask.

Amira buys her lolly at five past six, just after the driver has sealed the bundle. Her line goes in tomorrow's bundle, which arrives the day after. That is a **batch cut-off**, and it is why something you bought on Friday evening appears on your statement on Tuesday.

And now the question everyone actually asks. Amira brings the lolly back unopened. The driver agrees to refund her. Why does it take four days when buying it took two seconds?

Because there is no shouting down the street in the other direction. The whole reason the payment felt instant is that a *promise* was made instantly, and a promise costs nothing. Giving money back is not a promise. It is the money itself, and the money has only ever had one speed. The driver writes a line in tomorrow's bundle saying "minus £4.20". The bundle goes in the post. It arrives. It gets read, checked and totalled against everything else. Then Mrs Bello puts £4.20 back in Amira's page.

Buying was fast because it was fake. Refunding is slow because it is real.

Here is the whole thing with real numbers. Tunde, Amira's father, fills up at a motorway service station on Thursday. The pump reads **£68.40**. He looks at his phone in the car park and it says **£99.00 pending**. He has not bought ninety-nine pounds of anything. The pump had to commit to a number before it knew what he would take, so it committed to a ceiling. On Friday the £99.00 disappears and **£68.40** appears in its place, and he assumes he has been refunded £30.60, which he has not, because the £99.00 was never taken.

The following Monday he checks into a hotel for two nights at £120 a night. Reception takes his card and authorises **£300.00**, being the £240 of room plus what the hotel genuinely expects him to spend on dinner. On Tuesday night he spends more than that, and at 21:10 the hotel quietly asks for another **£75.00** on top. He is now holding £375.00 of promises against a card he has used exactly once. On Wednesday morning the bill comes to **£341.20**. The hotel puts £341.20 into that night's bundle, and within twenty-four hours it is obliged to send a message releasing the £33.80 difference. If it forgets, the £33.80 sits on his card for a fortnight and he blames the hotel, which is fair, and phones his bank, which cannot help him, which is not obvious but is true.

Three events. Three clocks. One tin.

Where the plain version stops being true

The slip of paper is not a ring fence, and there is no tin. The picture of a hold as money set aside is comforting and wrong. An authorisation hold is not a transfer between two ledgers and it does not create a fund earmarked for the merchant. It is an adjustment to a derived figure — your *available* balance — held in the issuer's authorisation system, sitting on top of a posted balance that has not changed at all. The merchant has no claim on it, no security interest in it, and no ability to reach it. If your issuer becomes insolvent overnight, the hold protects the merchant not at all. And on a credit card there is no money involved at any point: the hold reduces your open-to-buy, which is a permission to borrow, not a sum you own. Two different things wear the same word.

Nothing in the picture obliges the person holding the money to let go of it. This is the correction that matters most, because it is the one that misdirects a million customer complaints a year. The card schemes publish rules about when a *merchant* must clear or reverse an authorisation. They do not, in the general case, publish rules obliging an *issuer* to drop a hold at a given hour. Those are different obligations pointing in different directions. So the merchant can do everything correctly — clear on the day, reverse the excess within twenty-four hours — and the hold can still sit on your account for days, because the issuer's matching logic did not connect the two records, or because its release policy runs on a timer. When a hotel tells you "we've released it, it's with your bank now", that is very often the literal truth and it helps you not at all.

"Three events" describes one architecture, not all of them. The separation of authorisation from clearing is what the industry calls a dual message system, and it is the architecture of most credit card traffic. There is another architecture, the single message system, in which one message both asks permission and moves the money, and there is no separate presentment at all. ATM withdrawals work that way. So does a great deal of debit traffic in various markets. Once you are in single-message territory, "capture", "void" and "batch cut-off" are not merely different — several of them do not exist as concepts, and a reversal is a different animal with a different deadline. Any statement in this chapter about three timescales is a statement about dual message processing.

Settlement between the banks is not the merchant being paid. These are two distinct events, days apart, and the industry uses the same word for both, which is a permanent source of confusion in merchant support queues. The scheme calculates a net position and moves funds between the issuer's and the acquirer's settlement banks. Then, separately, under a commercial contract that the scheme is not party to, the acquirer pays the merchant — net of interchange, scheme fees and its own margin, possibly minus a rolling reserve, on a schedule that might be the next working day or might be five days, and which travels over Bacs or Faster Payments like any other bank payment. A merchant asking "has it settled?" and an acquirer answering "yes" are frequently talking about different days.

The technical version

Three events, three clocks

The three events differ not only in when they happen but in what their unit of work is. Authorisation is per transaction. Clearing is per file. Settlement is per institution. Losing track of which unit you are in is the root of most reconciliation failures.

Event	Typical vehicle	Latency	Unit of work	Moves money
Authorisation	ISO 8583 `0100` / `0110`	300 ms to 2 s	One transaction	No

Event	Typical vehicle	Latency	Unit of work	Moves money
Capture	Terminal batch close, acquirer capture file	Minutes to 24 hours	One merchant's day	No
Clearing	Visa BASE II `TC05`, Mastercard IPM `1240`	1 to 3 days	One institution's file	No
Settlement	Net position, funds transfer	1 to 3 business days	One institution's net position	Yes
Merchant funding	Bacs or Faster Payments credit	T+1 to T+5 by contract	One merchant's contract	Yes

Only the last two rows move money, and only the last row moves it to the party the customer thinks they paid.

Authorisation: what the message does and does not do

An authorisation request is an ISO 8583 `0100`, answered by a `0110`. Its four-digit message type indicator decomposes as version, class, function, origin: `0` for the 1987 edition, `1` for authorisation class, `0` for request, `0` for acquirer origin. In single message systems the equivalent is a `0200` financial request answered by a `0210`, and the distinction is exactly the one drawn above — a `0200` is understood to post.

The fields that carry the commercial substance of the request are the ones catalogued earlier in this volume. DE2 carries the primary account number. DE3 carries the processing code, six digits of transaction type plus from-account plus to-account. DE4 carries the amount in minor units of the currency named in DE49, itself an ISO 4217 numeric code, `826` for sterling, `978` for euro, `840` for US dollars. DE7 carries transmission date and time, DE11 the six-digit system trace audit number, DE12 and DE13 the terminal's own local time and date. DE37 carries the twelve-character retrieval reference number and DE41 the eight-character terminal identifier. DE55 carries the EMV data. On the way back, DE38 carries the six-character authorisation identification response, DE39 the two-character response code.

What the `0110` does is create an obligation with an expiry. It does not create a payment instruction. Nothing in the authorisation pair causes a single unit of currency to change hands, and an acquirer that never follows it with a presentment will never be paid, however emphatically the `0110` said approved.

ISO 8583 has an explicit field for the lifetime of that obligation. DE57, the authorisation life cycle, is three characters: position one is a time code, where `1` denotes calendar days, `2` hours and `3` minutes, and positions two and three carry an interval of `01` to `99`. So `102` is two calendar days and `224` is twenty-four hours. The standard's action code list likewise carries `1033` (authorization lifecycle unacceptable) and `1034` (authorization lifecycle has expired), which tells you that the drafters expected disagreement about lifetimes to be a normal, coded outcome rather than an operational incident. In practice the schemes largely govern this through their own rulebooks rather than through DE57, but the field exists and some domestic switches use it.

What the issuer actually does to the balance

An issuer holds at least two figures against an account and returns different ones to different channels.

The *posted* or *ledger* balance is the sum of transactions that have actually been applied to the account. It changes when clearing arrives, not when authorisation happens.

The *available* balance is a derived figure: posted balance, plus or minus any credit line, minus the total of outstanding authorisation holds, minus any uncleared-funds or fraud-related restraints. This is the number the authorisation system tests a `0100` against, and it is the number your app shows you.

The pending items you see listed in a banking app are outstanding authorisations, not transactions. They can change amount. They can vanish without ever posting. They can post at an amount that never appeared in the pending list at all. Support scripts that tell customers a pending item "will be taken tomorrow" are asserting something the issuer does not actually know.

When the presentment eventually arrives, the issuer must decide whether it corresponds to an existing hold. If it matches, the hold is released as the posting is applied and the customer sees one clean line. If the matching fails, the issuer applies the posting *and* leaves the hold in place until it ages out, and the customer sees the same amount twice. That is the mechanism behind the overwhelming majority of "I've been charged twice" calls, and it is why the scheme processing rules are so preoccupied with identifiers.

Estimated and incremental authorisation

Where the final amount is genuinely unknown at the point of commitment, Visa's rules permit an estimated authorisation followed by any number of incremental ones. The important word is genuinely: Visa's own merchant guidance states that an estimated authorisation "must be a genuine estimate and must not be an arbitrary amount", must not include incidental amounts such as tips or a damage buffer, and must not be used to check the status of a credential — that is what account verification exists for.

The coding requirements are specific, and they are the difference between a clean transaction and a hold the issuer cannot match. In Visa's message structure the fields are private-use subelements:

Purpose	Visa field	Value
Flag an estimated authorisation	Field 60.10, Additional Authorization Indicator	`2` estimated amount, or `3` estimated amount and terminal accepts partial approvals
Flag an incremental authorisation	Field 63.3, Message Reason Code	`3900`
Flag an incremental authorisation (US alternative)	Field 62.1, Authorization Characteristics Indicator	`I`
Link every message to the original	Field 62.2, Transaction Identifier	The TID Visa returned on the original approval
Link every message to the original	DE37, Retrieval Reference Number	The value from the original request

The amount semantics differ between message types and this trips people up constantly. In an incremental authorisation, the amount field carries the *additional* amount being requested, not the new total. In an authorisation reversal, the amount field carries the *original* total authorised — the sum of the estimate and all increments — and the corrected figure goes in the Replacement Amount field.

Failure to code these correctly is not merely untidy. Visa states that acquirers and merchants may carry dispute liability under Dispute Condition 11.3, No Authorization / Late Presentment, where there is no estimated indicator on the first message, no incremental indicator on subsequent messages, a mismatched Transaction Identifier, or processing outside the permitted timeframe.

Those timeframes are, per Visa's 2024 merchant guidance, the maximum time from a valid estimated authorisation to processing:

Segment	Maximum
Lodging, vehicle rental, cruise line	30 days from estimated authorisation approval
Rental merchant categories	10 days
Cardholder-initiated card-absent transactions	10 days
Card-present transactions	5 days

Two cautions on that table. First, Visa notes country-specific approval response validity timeframes exist and the rulebook governs. Second, and instructively, the 2023 edition of the same Visa document published a materially different set: 31 days for lodging, vehicle rental and cruise lines, 7 days for other rental categories and other card-absent transactions, 3 or 7 days for commuter transport, and same-day for other card-present transactions. Both documents are Visa's. If you are hard-coding validity windows, hard-code them against the rulebook edition in force and version them, because they move.

Incremental authorisations do not extend the window. A stay or rental that outlives the validity period must be closed within it and reauthorised, and the new authorisation gets a fresh window of the same length.

Fuel, and why the number is never the number

Automated fuel dispensers do not use the estimated and incremental framework. They use an older construct, the initial authorisation, in which the terminal sends a fixed amount before the final figure is known, the amount is capped, no incremental is permitted, and the pump must stop if the fill reaches the authorised ceiling. Visa's guidance notes that from April 2025 only AFDs may use initial authorisations, with laundries, quick copy, car wash, video rental and vending encouraged to migrate to estimates and increments.

The best known variant is the one-dollar status check, in which the terminal authorises a nominal amount and the issuer is expected to place a hold reflecting the segment ceiling rather than the nominal amount. Visa's own business news of that change, effective 17 October 2020, sets the US AFD status check authorisation limits at **USD 125 for non-Fleet cards and USD 350 for Fleet cards**, applying only where the transaction is both chip-on-chip and partial-authorisation participating, and leaving them at **USD 100 and USD 150** otherwise. Issuers remain liable up to those limits and may dispute the excess under Reason Code 11.3.

Those ceilings have since moved again, and the figure to know now is USD 175. It circulates very widely in acquirer-facing and merchant-facing material as "the" AFD limit, and although it appears nowhere in the 2020 notice it is not folklore. Visa's own fuel segment update, presented by a Visa senior business leader to the National Petroleum Energy Credit Association in May 2025, gives the maximum US status check amount as USD 175 for non-Fleet cards and USD 1,000 for Fleet cards, and warns in the same deck that issuers who hold the nominal one dollar rather than that maximum are being exploited by fraudsters, which is the whole mechanism stated in reverse. The general point survives the revision: these ceilings are rewritten every few years, so version them against the edition of the rules in force rather than against whatever number the internet is quoting this season.

The same notice describes the mechanism that actually shortens fuel holds, and it is worth knowing because it is the only widely deployed example of a message whose sole purpose is to shrink a hold in real time. Visa asks issuers to activate their BINs for enhanced AFD and real-time clearing confirmation advices, which carry the final amount ahead of settlement, and notes that settlement "can take up

to two days following completion of the AFD transaction". Issuers that apply the advice immediately release the hold in minutes. Issuers that do not, wait for clearing, and their cardholders are the ones writing to the newspapers.

Partial approval

If the account cannot cover the requested amount, an issuer that supports partial approval may approve a smaller one instead of declining. Under the 1987 response code list DE39 carries `10`, approved for partial amount; the third-edition action code list carries `0002` with the same meaning. The approved amount is returned in the amount field, and the terminal is expected to prompt for the balance by another tender.

The terminal has to have declared that it can cope with this, which is what the value `3` in Visa's Additional Authorization Indicator does: estimated amount *and* terminal accepts partial authorisation responses. A terminal that has not declared the capability and receives a partial approval anyway will typically treat it as a full approval for the requested amount, which is a genuine and repeatable way to give away goods.

Reversals, voids and cancellations

A reversal is the only fast instrument in the entire chapter. It is an ISO 8583 `0400` reversal request, or a `0420` reversal advice with its `0430` response where the recipient has no discretion to refuse.

Two fields carry the work. DE90, original data elements, is a forty-two digit block that identifies the message being reversed: original MTI (4), original STAN (6), original transmission date and time (10), original acquiring institution identification code (11) and original forwarding institution identification code (11). DE95, replacement amounts, forty-two characters, carries the corrected amounts where the reversal is partial rather than full; Visa's equivalent in its own message structure is the Replacement Amount field. Sources differ on whether DE95 is specified as numeric or alphanumeric, and scheme implementations vary in how they subdivide it, so treat the subfield layout as scheme-specific rather than universal.

The industry word "void" is a merchant-facing abstraction, not a message. Depending on where the transaction has got to, a void is implemented either as a reversal of an authorisation that has not yet been captured, or as the removal of a record from a batch that has not yet been submitted. Once the batch has gone, there is no void; there is only a refund, with everything that implies about timing.

Visa's reversal obligations are expressed in hours, not days:

- Where the transaction will not be completed at all, the entire authorised amount must be reversed within 24 hours of the earlier of the merchant becoming aware of that fact or the end of the authorisation validity period.
- Where the transaction completes and the sum of the estimate and any increments exceeds the final amount, the difference must be reversed within 24 hours of completion.

Again, the 2023 edition of the same guidance expressed the second obligation with tolerances — a 15 per cent threshold for lodging, vehicle rental and cruise lines, 20 per cent including tips for taxis — before the 2024 edition stated it flatly. Read the rulebook edition you are actually operating under.

Capture and the batch cut-off

Between the `0110` and the clearing file sits capture, which is where a merchant declares that the authorisation should become money. In an attended retail environment the terminal accumulates approved transactions locally and closes its batch, either on a schedule or on an operator action. In an

integrated or gateway environment the merchant sends an explicit capture instruction, which is why an e-commerce platform can authorise at checkout and capture at despatch three days later.

On the acquirer side, the file that carries this into Visa's clearing system is the **TC 33.A Capture File**, whose records are laid out in *BASE II Clearing: Interchange Formats, TC 01 to TC 49*. It carries purchase and refund transactions that already have approved authorisations, decomposed into transaction code records: `CP01` transaction, additional, billing and shipping, merchant, instalment, gateway and supplemental data; `CP02` EMV data; `CP03` lodging summary; `CP04` Level II and Level III purchasing data; `CP05` air passenger itinerary; `CP10` car rental; and so on. The Edit Package, Visa's own validation software that acquirers install, wraps the file with a `TC90` file header, `TC91` batch record and `TC92` file trailer, and validates that the `TC05` (Sales Draft) and `TC06` (Credit) transactions built from the capture file are correct before they go to Visa for clearing and settlement.

The cut-off is a property of every one of these hops, and they are not aligned. A terminal that closes its batch at 23:00 local, an acquirer that assembles its submission at 01:00, and a scheme clearing window that closes at a fixed time in a different time zone together produce the familiar phenomenon whereby a Friday evening purchase is presented on Monday and funded on Wednesday. None of those systems is slow. They are each waiting for the next one's door to open.

Clearing

Clearing is the exchange of transaction detail between acquirer and issuer, the assessment of interchange and fees, and the production of the figures that settlement will act on. It is a batch process, it is file-shaped, and it is where the transaction acquires the identifiers that disputes will later be filed under.

Visa clears through **BASE II**, whose record vocabulary is the transaction code series already described: `TC05` for a sales draft, `TC06` for a credit. Mastercard clears through the **Global Clearing Management System**, using the **Integrated Product Messages** format. In IPM the first presentment is message type `1240` with DE24 function code `200`; a financial detail addendum is `1644` with function code `696`; the first chargeback is `1442`. DE25 carries the message reason code, and Mastercard's switch rules use reason code `1404`, "previously approved authorization — partial amount, final clearing", to signal that a presentment is the last one against a given authorisation and that the authorisation is now closed, after which no further clearing messages may be submitted against it. Mastercard's clearing dates are expressed as the Central Site Business Date, which is the date the message cleared into GCMS and not the date anything happened in a shop.

Presentment is not optional and it is not open-ended. Mastercard's European rules require an ATM presentment within seven calendar days of the transaction date, with late presentment exposed to chargeback under message reason codes `4842`, `4880` for Maestro ATM, or `4811` for stale transactions, and transactions presented between forty-six calendar days and one year attracting a fee that is passed in full to the issuer. The same shape of rule, with different windows, applies across the card products.

Two asymmetries in clearing deserve stating plainly, because both break the tidy mental model.

A clearing record can exist with no authorisation behind it. Offline-approved transactions, forced posts and certain fallback flows all produce presentments the issuer never saw a `0100` for. The issuer must post them and argue afterwards.

And a clearing record's amount is not obliged to equal the authorised amount. Tolerances vary by segment, and where the clearing amount exceeds what the tolerance allows, the issuer's remedy is a dispute rather than a rejection at the door.

Settlement

Settlement is the discharge of the obligations that clearing recorded, and it is the only step in this chapter at which value moves between institutions.

Mastercard describes its own architecture in three named systems: the Authorization Platform, the Global Clearing Management System for clearing, and the **Settlement Account Management** system for settlement, which calculates the net position of each acquirer and issuer arising from clearing and performs two functions, sending advisements and transferring funds. Visa's equivalent function is performed by the VisaNet Settlement Service, fed by the figures BASE II produces.

Three properties follow, and each of them surprises somebody.

Settlement is *net*, not gross. An issuer that owes on ten million transactions and is owed on two million settles one figure. This is the netting described in Volume I, applied at institutional scale, and it is why a card scheme's daily settlement flows are a small fraction of its daily transaction value.

The scheme is not a bank. It computes positions and instructs; the actual transfer happens between nominated settlement banks over ordinary payment infrastructure, in a settlement currency chosen per region and per member. This is the structural reason a scheme outage stops new authorisations but does not destroy value that has already cleared.

And interchange is not invoiced. It is applied as an adjustment inside the clearing figures, so by the time a settlement position exists it is already net of the interchange the acquirer owes. There is no separate interchange bill to pay or dispute.

Then, entirely separately, the acquirer pays the merchant. That payment is governed by the merchant agreement, not the scheme rulebook. It is net of the merchant service charge, it may be net of a rolling reserve, and it travels over whatever domestic rail the acquirer uses — in the United Kingdom, Bacs or Faster Payments, with the timings Volume IV sets out. T+1 and T+3 are both common; new or higher-risk merchants are routinely funded more slowly. A merchant seeing money on Wednesday for a Monday sale is seeing the end of a chain in which the scheme settlement happened on Tuesday and had nothing to do with them.

Why the refund takes days

Put the two directions side by side and the asymmetry is obvious.

A purchase gets a real-time promise and a slow money movement. The customer experiences the promise. Their available balance drops in under two seconds and they conclude, reasonably, that the payment system is fast.

A refund has historically had no promise at all. The refund was a clearing-side credit — a `TC06` in BASE II terms — which meant it travelled at the speed of files: the merchant's batch, the acquirer's submission, the scheme's clearing cycle, the issuer's posting run. Nothing was reserved, because there is nothing to reserve; the issuer is not going to promise you money it has not received. Worldpay's own documentation puts the historical position exactly: issuers "accepted and honoured refund transactions when they received them later in the daily clearing files sent to them by processors", and "typically, this would take 24 hours for a refund transaction to be registered in cardholders' accounts", before the rest of the posting cycle ran.

The customer therefore experiences the purchase as instant and the refund as a five-day wait, and concludes that the merchant is stalling. In most cases the merchant did the work on day zero.

Both schemes have addressed this, and it is worth being precise about what they addressed, because it is a common and expensive misreading. Visa Purchase Return Authorization and Mastercard's equivalent return authorisation mandate require the merchant to obtain an *online authorisation for the refund*. Verifi, a Visa company, answered the obvious question about it in terms that leave no room: the mandate "is intended to increase consumer visibility to refunds in process by making pending refunds visible via online and mobile banking channels", and the time between refund issuance and funds appearing "can vary based on the processing timelines for each party".

In other words, the refund authorisation creates the missing promise. It does not create a faster money movement. Your bank can now show you a pending credit within seconds. The money still arrives when clearing and settlement bring it.

The compliance timeline, as of writing, is worth recording because the deadlines differ by region and by scheme:

Requirement	Region	Date
Refund without valid authorisation exposed to issuer dispute under reason code 11.3	US, Visa	From April 2020
Return authorisation without matching settlement record included in Visa's Misuse of Authorization programme	US, Visa	From July 2020
Online authorisation required for all refunds, credit vouchers and purchase returns	Europe: UK and Ireland, Visa and Mastercard	From April 2025
Mastercard non-compliance fees commence	UK and Ireland	1 May 2025
Visa non-compliance fees commence, at USD 0.05 per unauthorised refund	UK and Ireland	19 October 2025

Two operational consequences follow that catch integrations out. A refund authorisation can be *declined*, which was previously impossible, so a point-of-sale application must be able to render a declined refund without falling over. And an approved online refund typically returns a six-digit authorisation code, while an offline-accepted refund may return a five-digit one, which is a useful diagnostic when reconciling a merchant estate mid-migration. Visa publishes suggested merchant actions per decline code for returns, of which the useful ones are 14 invalid account number, meaning request an alternative card; 54 card expired, meaning ask whether a replacement exists; and 55 invalid PIN, meaning retry PIN normally.

The integrity fees, and what they are really for

The schemes charge for authorisations that never become money. The framing is data integrity, and it is honest framing: an unmatched authorisation is a hold on a customer's account for which no corresponding value will ever exist.

Visa's **Misuse of Authorization System Fee** applies in the US region to approved and partially approved authorisations that cannot be matched to a clearing transaction or an authorisation reversal. It was long assessed at USD 0.09 and was raised to **USD 0.15** with effect from January 2025 according to acquirer pass-through bulletins. Visa lists the two most common causes as failure to process estimated authorisations, incrementals and reversals to the technical requirements, and use of one-dollar

test transactions where the Account Verification Service should have been used. Visa also operates a separate unmatched clearing fee for the mirror-image failure. The same January 2025 schedule increased the Visa BASE II transmission fee from USD 0.0018 to USD 0.0025 per transaction, a figure worth knowing chiefly because it demonstrates how thin the per-item economics of clearing actually are.

Mastercard's **Processing Integrity Fee** turns on authorisation type, and the type is declared by the merchant:

Authorisation type	Requirement
Pre-authorisation	Fully reversed or cleared within 30 calendar days of the authorisation date
Final authorisation	Fully reversed or cleared within 7 calendar days; clearing amount must equal the authorised amount; clearing currency must match the authorised currency
Undefined authorisation	Cleared within 7 calendar days; Mastercard has been progressively restricting this category

The fee rates are region-specific and revised regularly. Fiserv's published Canadian pass-through schedule, as one concrete example of the shape of them, lists all three processing integrity non-compliance categories at 0.452 per cent with a USD 0.113 minimum. Do not carry a rate across regions; look it up in the schedule that applies to your acquiring entity.

The design lesson embedded in the final-authorisation rule is the sharpest one in this chapter. Mastercard will charge you if the amount you clear is not the amount you authorised, *when you told it the amount was final*. The rulebook is not asking you to be accurate. It is asking you to be honest about whether you know.

Account verification, which is not an authorisation

The correct way to check that a card exists and is in good standing is a zero-amount account verification: a `0100` with the amount field at zero, which Visa exposes as its Account Verification Service. It returns an approval and, on most implementations, address and security code verification results, without creating a hold.

The incorrect way, still widely deployed, is a small-value authorisation — the one-dollar or one-pound test — which creates a real hold on a real customer's real available balance, and which the schemes now charge for when it is never cleared. If you are storing a card for later use, verify at zero.

Design rules that follow

Model the three events as three separate records with three separate lifecycles, joined by identifiers, and never as one row with a status column. An authorisation that is partially captured and partially reversed has produced four facts, and a single row cannot hold four facts.

Store the linking identifiers on every message. For an estimated-and-incremental flow this means the scheme transaction identifier, the retrieval reference number and the system trace audit number carried forward from the original approval into every increment, every reversal and the clearing record. This is not good practice; per Visa's rules it is the difference between a clean transaction and dispute liability under condition 11.3.

Reverse promptly and reverse for the right amount. The obligation is measured in hours from the moment you knew, not days from the moment you got round to it, and the reversal message must

carry the original total in the amount field with the corrected figure in the replacement amount field. Getting those two the wrong way round produces a hold that grows rather than shrinks.

Declare your authorisation type honestly. If you know the final amount, say final and then clear that exact amount in that exact currency. If you do not know, say estimated and use increments. The undefined category exists because implementations were vague, and it is being closed.

Never tell a customer that a pending item will be taken. Tell them it is a reservation, that it may change or disappear, and that the amount that will finally post is the amount on the receipt. Every one of those statements is true, and the sentence "it'll come off tomorrow" is not.

And when a customer asks why the refund is slow, the honest answer is the one this chapter began with. The payment was not fast. The promise was fast. The money has always taken days, in both directions, and the refund is simply the first time anybody notices.

30.98 Common wrong ideas

Wrong: A hold sets your money aside for the merchant. **Right:** It adjusts a derived figure, the available balance, inside the issuer's authorisation system; it creates no fund, gives the merchant no claim and no security interest, and on a credit card no money is involved at any point, only a reduction in a permission to borrow.

Wrong: The merchant is holding the money and the merchant can release it. **Right:** The scheme rules say when a merchant must clear or reverse; they do not in the general case oblige an issuer to drop a hold at a given hour, so "we've released it, it's with your bank now" is very often the literal truth and helps the customer not at all.

Wrong: Every card transaction has three separate events. **Right:** That describes dual message processing; in single message systems one message both asks and moves, there is no separate presentment, and capture, void and batch cut-off are not merely different but absent as concepts.

Wrong: Settlement means the merchant has been paid. **Right:** Scheme settlement moves a net position between the issuer's and the acquirer's settlement banks, and the acquirer then pays the merchant days later under a commercial contract the scheme is not party to, net of charges and possibly a rolling reserve.

Wrong: A pending item will be taken tomorrow at the amount shown. **Right:** Pending items are outstanding authorisations, not transactions: they can change amount, vanish without ever posting, or post at a figure that never appeared in the pending list.

Wrong: "I've been charged twice" means the merchant sent the transaction twice. **Right:** The overwhelmingly common mechanism is that the presentment failed to match the hold, so the issuer applied the posting and left the hold in place until it aged out.

Wrong: The refund authorisation mandates have made refunds faster. **Right:** They create the promise that refunds never had, so a pending credit becomes visible within seconds; the money still arrives at the speed of clearing and settlement, which has not changed.

Wrong: An incremental authorisation carries the new total. **Right:** It carries only the additional amount; it is the reversal that carries the original total, with the corrected figure in the replacement amount field, and getting those the wrong way round produces a hold that grows instead of shrinking.

Wrong: An estimated authorisation can be whatever figure is convenient. **Right:** It must be a genuine estimate, must not include incidentals such as tips or a damage buffer, and must not be used to check the status of a credential.

Wrong: A one-pound test is a harmless way to check that a card is real. **Right:** It creates a real hold on a real customer's real available balance and is chargeable when it never clears; the correct instrument is a zero-amount account verification.

30.99 Chapter summary in 20 lines

1. A card payment feels instant because a promise is made instantly, and a promise costs nothing to make.
2. Authorisation reserves, clearing presents the real amount, and settlement is the only step at which value moves between institutions.
3. A hold is not money set aside but an adjustment to a derived available balance sitting on top of a posted balance that has not changed.
4. That is why a cardholder honestly has two numbers at once: what they have, and what they can spend.
5. Merchants ask for more than they need whenever the final amount is unknown at the moment of commitment, which is why a hotel holds more than the room rate and a pump holds more than the fill.
6. Capture is the merchant declaring that an approval should become money, and it sits between the authorisation response and the clearing file.
7. Clearing is file-shaped and batch-driven, carried by Visa's BASE II transaction codes and Mastercard's IPM messages, and it is where a transaction acquires the identifiers that disputes will later be filed under.
8. Settlement is net rather than gross, is computed by the scheme and executed between nominated settlement banks, and already has interchange applied inside it, so there is no interchange bill to pay.
9. The acquirer then pays the merchant separately, under a merchant agreement rather than the scheme rulebook, on a schedule that may be anywhere from the next working day to five days later.
10. Cut-offs exist at every hop and are not aligned, which is why a Friday evening purchase is presented on Monday and funded on Wednesday, and why none of the systems involved is actually slow.
11. A reversal is the only fast instrument in the whole chapter, and the obligation is measured in hours from the moment the merchant knew, not days from the moment it got round to it.
12. Where the final amount is genuinely unknown, an estimated authorisation followed by increments is the sanctioned pattern, on condition that the estimate is genuine and every message carries the linking identifiers forward.
13. Failing to code those indicators and identifiers is not untidiness; it moves dispute liability onto the acquirer and the merchant.
14. Fuel dispensers use an older construct with a segment ceiling instead, and the published ceilings are rewritten every few years, so they must be versioned against the rulebook edition in force rather than against whatever figure is circulating.
15. Partial approval lets an issuer approve a smaller amount than was asked, and a terminal that never declared the capability will treat the smaller approval as a full one, which is a repeatable way to give goods away.
16. Refunds historically had no promise attached at all, because the issuer will not promise money it has not received, so they travelled at the speed of files.

17. Refund authorisation mandates supply the missing promise and make a pending credit visible immediately; they do not make the money arrive any sooner, and a refund can now be declined, which point-of-sale software must handle.
18. The schemes charge for authorisations that never become money, because an unmatched authorisation is a hold on a customer's account for which no value will ever exist.
19. Mastercard's final-authorisation rule carries the sharpest lesson in the chapter: the rulebook is not asking a merchant to be accurate, it is asking it to be honest about whether it knows.
20. So the honest answer to a customer asking why the refund is slow is that the payment was never fast — the promise was fast, the money has always taken days in both directions, and the refund is simply the first time anybody notices.

Sources: Visa, Estimated and Incremental Authorization and Reversal Processing Requirements for Visa Merchants (2024 and 2023 editions); Visa Business News, U.S. Automated Fuel Dispenser Authorization Limits Will Be Increased, effective 17 October 2020; Visa Merchant Resource Library; Verifi (a Visa company), Visa Purchase Return Authorization FAQ; Visa Acceptance Solutions Payments Acquirer Implementation Guide, TC 33.A Capture File and Edit Package (BASE II Clearing: Interchange Formats, TC 01 to TC 49); Mastercard, Switching explained (Authorization Platform, GCMS, Settlement Account Management); Mastercard Switch Rules manual; Mastercard Chargeback Guide, clearing and processing cycles; Mastercard IPM Clearing Formats message type and function code tables; ISO 8583 third-edition Annex D code listings as circulated by Accredited Standards Committee X9 (action codes, authorization life cycle codes); ISO 8583 data element definitions for DE90 and DE95; Worldpay Total Hospitality documentation, Online Refund and the online authorisation of refunds mandate; Fiserv Merchant Services published pass-through fee schedule; acquirer pass-through bulletins for January 2025 Visa and Mastercard fee changes.

31

CHAPTER

Where the Money Goes

31.0 What this chapter gives you

1. You will be able to say who pays interchange and who receives it, and explain why the merchant is neither of those parties despite being the one out of pocket.
2. You will be able to break a merchant service charge into interchange, scheme fees and acquirer net revenue, and name the single component that is negotiable.
3. You will be able to explain why the issuer's slice on a £480 corporate card ticket is over a thousand times the slice on a £4.20 debit card lolly when the ticket costs only 114 times as much.
4. You will be able to list the inputs a scheme's rating engine uses to derive interchange, and say why a transaction's rate is not knowable with certainty at the point of sale.
5. You will be able to state exactly how far the UK interchange caps reach, and name the exclusions — commercial cards, cash withdrawals, three-party schemes and any cross-border pair — that put a great deal of traffic outside them.
6. You will be able to explain why two identical £100 online sales can generate 30 pence and £1.50 of interchange, and why the merchant can neither detect, price for, nor decline the difference.
7. You will be able to name the four pricing models the regulator recognises and explain why over 95 per cent of merchants are on the one that shows them nothing.
8. You will be able to show, with arithmetic, why a merchant taking £2,000 a month through a single terminal can be paying 2.75 per cent before a single transaction has been rated.
9. You will be able to cite the article of the Interchange Fee Regulation that entitles a merchant to unblended pricing, and the article that entitles it to transaction-level interchange data.
10. You will be able to explain why a coffee shop's effective rate can exceed an airline's, and why neither merchant controls the thing that actually decides it.

The plain version

The ice cream driver from the last chapter sold Amira a lolly for £4.20. On Friday afternoon, when Mr Doyle at the depot squares up the week and pays the drivers, that lolly appears on the driver's payslip as **£4.05**.

The driver did not agree to a discount. Amira did not pay less. Fifteen pence went somewhere, and the driver has never met the person who took it.

Here is where it went, and the answer is stranger than anyone expects.

Start with Mrs Bello, who keeps the biscuit tin and the notebook. When the driver shouted down the street on Thursday, it was Mrs Bello who had to stop what she was doing, find Amira's page, check the number, decide whether to say yes, write the slip and clip it on. It was Mrs Bello who took the risk

that Amira's page was wrong. It is Mrs Bello who has to argue with Amira in three weeks' time when Amira insists she never bought the lolly at all. And it is Mrs Bello who has to give Amira her money back out of the tin if the lolly turns out to have been off, whether or not the driver ever gives it back to Mrs Bello.

Mrs Bello does all that work for the driver's benefit. So Mrs Bello keeps a slice.

That slice is the thing the industry calls **interchange**, and it is the single most misunderstood number in the whole of payments, because it runs the wrong way. Ask anyone who pays it and they will say the shop pays it to the card company. Both halves of that are wrong. The shop's bank pays it, and it goes to *your* bank. It is a payment from the seller's side of the street to the buyer's side of the street, and the seller has no relationship with the person receiving it and no ability to argue about the price.

Now the second slice. Somebody printed the notebooks. Somebody wrote the rules that say Mrs Bello must answer within two seconds, that the driver may not add a surcharge, that the Friday meeting happens at four o'clock on the corner and not whenever anyone feels like it. Somebody decides what happens when Amira and the driver disagree. That is Miss Vaughan, who owns the rulebook. She was never in the room when the lolly was sold and she never touches the money. She takes a small slice from Mrs Bello and a small slice from Mr Doyle, every week, for keeping the street running.

That slice is **scheme fees**, and Miss Vaughan is Visa or Mastercard.

And now the third slice, which is the only one that goes to somebody the driver has actually met. Mr Doyle at the depot collected the bundle, checked it, chased the missing lines, argued with Mrs Bello about the disputed lolly, and — this is the important bit — paid the driver on Monday even though the money from Mrs Bello did not arrive until Wednesday. If the driver takes a hundred pounds of orders and then vanishes, Mr Doyle is the one left holding it. For that, Mr Doyle keeps a slice too.

That slice is the **acquirer's margin**.

Add all three together and you get the number the driver actually cares about, which is the gap between £4.20 and £4.05. In the trade the whole gap has a name: the **merchant service charge**, or MSC. It is not one fee. It is three fees wearing one coat, and only one of the three is negotiable with the person who sends the invoice.

Now the real numbers, and this is where it gets properly interesting.

The lolly. Amira paid £4.20 with an ordinary British debit card, tapped on the driver's reader. The rate Mrs Bello is allowed to keep is 0.20 per cent. Twenty pence in every hundred pounds. On £4.20 that is **0.84 pence**. Less than a penny. Mrs Bello did all that work — the check, the promise, the argument in three weeks, the refund risk — for less than one penny.

The aeroplane ticket. On the same afternoon, in an office two miles away, someone books a £480 flight on their company's corporate card, typing the number into a website. The rate the issuer keeps on a British corporate Visa card used online is 2.00 per cent. On £480 that is **£9.60**.

Look at those two numbers side by side, because the arithmetic is the whole lesson. The ticket costs 114 times as much as the lolly. But the issuer's slice on the ticket is 1,143 times as big. Not 114 times. Eleven hundred. The rate itself went up tenfold, from 0.20 per cent to 2.00 per cent, purely because of what kind of card it was and how it was used, and then that tenfold difference was multiplied by the hundredfold difference in price.

Two things caused that, and neither of them is anything the airline or the coffee shop did.

The first is **what card it was**. A consumer debit card and a corporate credit card are different products with different rules, and the law that holds the consumer card down to 0.20 per cent does not apply to the corporate one at all. Same plastic, same terminal, same two-second shout down the street, ten times the fee.

The second is **how it was used**. Tapped in a shop, with the card physically there, is the cheapest way a transaction can happen, because it is the hardest to fake. Typed into a website is more expensive, because it is easier to fake and somebody has to carry the loss when it is.

And there is a third thing that never shows up in a percentage at all. Almost every merchant's contract has a fixed lump in it — call it five pence a transaction, for the message, the phone line, the reconciliation, the paperwork. Five pence on a £480 ticket is nothing, one hundredth of one per cent. Five pence on a £4.20 lolly is more than the entire interchange fee. The coffee shop is not paying more because coffee is a suspicious business. It is paying more because it sells cheap things, and every cheap thing carries the same fixed lump as an expensive one.

Now the last piece, and it is the piece that decides whether the driver can ever find any of this out.

Mr Doyle can send the driver a bill in one of two shapes. He can write: *forty-one sales, one hundred and seventy-two pounds sixty, our rate one point four per cent plus five pence, total charge seven pounds four*. That is one number, and it is called **blended pricing**, and the driver cannot tell from it whether Mrs Bello took a penny or a pound.

Or Mr Doyle can write: *interchange £1.24, rulebook fees £0.61, our margin £5.19*. Three numbers. That is called **interchange plus plus**, usually written IC++, and it is the only shape of bill in which the driver can see that Mr Doyle's own slice is the biggest of the three.

Which is, of course, exactly why the second shape took a law to make available.

Where the plain version stops being true

The merchant never pays interchange, and the moment of payment is not the moment of the sale. In the picture above, three people each take a slice out of one £4.20 as it travels. That is not what happens. Interchange is never invoiced, never transferred and never paid as a discrete event. It is a *deduction*: when the issuer settles with the scheme for the day's clearing, it remits the transaction value less the applicable interchange, and the acquirer receives less than it presented. Nobody sends anybody a bill. And it is not calculated when the card is tapped — it is calculated at clearing, from the contents of the clearing record, a day or more later. This has a consequence the analogy hides completely: a transaction's interchange rate is not knowable with certainty at the point of sale. It depends on data the merchant supplied, on whether the acquirer presented within the scheme's time limits, and on how the scheme's own rating engine classified the record. A transaction can be re-rated after the fact, and a merchant can be debited weeks later for the difference.

"Scheme fees" is not a fee. It is a catalogue. The single slice taken by Miss Vaughan is, in reality, several hundred separately named line items across two schemes, and they do not share a shape. Some are ad valorem, charged as basis points on transaction value. Some are a fixed amount per authorisation message, charged whether the authorisation was approved or declined. Some are charged per settled transaction, some per chargeback, some per BIN per year, some per licence. Some are charged only when the merchant behaves in a way the scheme wishes to discourage — presenting late, using a terminal that cannot do EMV, exceeding a fraud ratio. Some are charged on refunds. And a large fraction of them are charged not to the acquirer at all but to the *issuer*, which means the merchant never sees them and never pays them, but they still shape the economics of the card in the merchant's

terminal. Any sentence of the form "scheme fees are about X per cent" is a statistical summary of a mess, not a description of a price.

The caps are far narrower than the phrase "regulated interchange" suggests. The 0.2 per cent and 0.3 per cent figures are real, statutory, and — in the UK — apply only where the transaction is a consumer card transaction, on a four-party scheme, with the payer's payment service provider and the payee's payment service provider both located in the United Kingdom. Commercial cards are excluded by the legislation itself. Cash withdrawals are excluded. Three-party schemes are excluded unless they license others. And a transaction where either side of the pair sits outside the UK falls outside the caps entirely, which is not an oversight but the mechanism by which cross-border interchange rose by a factor of five after 2021. "Regulated" describes a subset of British card traffic, not British card traffic.

Blended versus IC++ is not the choice most merchants face, because most merchants have neither. The Payment Systems Regulator found that over 95 per cent of acquirers' merchants are on what it calls *standard* pricing, which is defined by exclusion: it is not IC+, not IC++, and not fixed. Standard pricing is a handful of headline rates attached to broad transaction categories, plus a tail of additional fees triggered by events. It resembles blended pricing in that the merchant cannot see the components, but it is not one blended rate either — it is several, and which one applies to a given sale depends on classifications the merchant was never shown. A merchant asking "am I on blended or IC++?" is usually asking a question with a third answer.

The technical version

The four amounts, and the direction each one travels

There are exactly four money-flows in the fee structure of a four-party card transaction, and the entire subject becomes tractable once their directions are fixed.

Flow	Paid by	Paid to	Mechanism
Interchange fee	Acquirer	Issuer	Deducted at scheme settlement; never invoiced
Scheme fees (acquirer side)	Acquirer	Scheme operator	Periodic invoice, itemised by fee code
Scheme fees (issuer side)	Issuer	Scheme operator	Periodic invoice; invisible to the merchant
Merchant service charge	Merchant	Acquirer	Deducted from funding, or invoiced monthly

The Payment Systems Regulator's formulation, from its card-acquiring market review, is the one to memorise: the MSC "is the total amount it pays for card-acquiring services to its acquirer", and it comprises interchange fees paid by the acquirer to the issuer, scheme fees paid by the acquirer to the scheme operator, and the remainder, which the PSR calls **acquirer net revenue**.

That remainder is the only component the merchant can negotiate, and it is the only component the acquirer keeps. Everything else is pass-through, whether or not the merchant's contract admits it.

The word "scheme fees" in that formulation is doing more work than it appears to. Mastercard and Visa charge acquirers for two logically distinct things: *scheme services*, meaning membership, licensing, brand, rules and dispute machinery; and *processing services*, meaning authorisation switching, clearing

and settlement. The PSR treated these separately in its market review precisely because they behave differently — it concluded that fees for scheme services paid by UK acquirers approximately doubled per pound transacted between 2014 and 2018, while it found insufficient evidence to conclude that processing fees had risen over the same period. A merchant statement that lumps both into one "scheme fee" line is discarding the distinction that matters most.

How interchange is actually determined

Interchange is not negotiated, not quoted and not chosen. It is *derived*, by the scheme's rating engine, from the contents of the transaction record at clearing. The merchant's only influence is over what data reaches that record.

The inputs are, in rough order of importance:

The **issuer's product and country**, taken from the primary account number carried in DE2 and resolved through the BIN to a specific card product — consumer debit, consumer credit, consumer prepaid, business debit, corporate, purchasing, and their platinum and infinite variants — and to an issuing country. A single-digit difference in the BIN changes the applicable fee by an order of magnitude, and the merchant cannot see it.

The **acquirer's country**, which together with the issuer's country determines whether the transaction is domestic, intra-regional or inter-regional. This is the axis on which the regulatory caps live or die.

The **channel and authentication**, expressed in the clearing record through the point-of-service entry mode, the terminal's capability indicators and the presence of EMV data in DE55 for chip transactions, or of an authentication value for e-commerce transactions authenticated through 3-D Secure. Visa's schedules name these tiers *Secure* and *Non-Secure*; Mastercard's name them *Merchant UCAF*, *Full UCAF*, *Enhanced Electronic* and *Base*. Falling out of the top tier is what practitioners call a downgrade, and Visa's own commercial schedules label the bottom tier *Non-Qualified*.

The **merchant category code**, carried in DE18, four numeric digits, the field ISO 8583 names *Merchant type*. The code set is standardised in ISO 18245, *Retail financial services — Merchant category codes*, whose current edition is ISO 18245:2023, superseding the 2003 first edition. MCCs are not decorative. Visa's published UK schedule conditions several programmes on specific codes: the Me-to-Me Program applies to MCC 6012 (financial institutions), MCC 6211 (security brokers and dealers) and MCC 9399 (government services, not elsewhere classified); the tax-payment variant applies to MCC 9311 (tax payments). In Visa's intra-EEA schedule, MCC 8398 (charitable and social service organisations) attracts its own capped commercial rates. A merchant boarded under the wrong MCC pays the wrong interchange indefinitely and will never be told.

The **amount and currency**, from DE4 and DE49 respectively, which matter because a great many rates are percentages subject to a fixed cap, or fixed amounts subject to a percentage floor, and because Visa operates explicit large-ticket programmes with value thresholds.

Finally, **timeliness and data completeness**. Interchange schedules for commercial and purchasing cards pay explicit incentives for enhanced data. Visa's intra-EEA schedule states that "Visa Purchasing acquirers may receive an additional interchange incentive of €0.50 if they submit the transaction with additional data: 'Visa Global Invoicing System and Line Item Detail' ", and that fleet acquirers may receive an additional 0.30 per cent for submitting Level 2 and Level 3 data at fuel-related MCCs. In the UK schedule the equivalents appear as deductions from the headline rate: Visa Purchasing card-not-present is 2.00 per cent, or 2.00 per cent less £0.32 with Line Item Detail, or 2.00 per cent less £0.16 with Summary Tax.

Note what none of these inputs is. None of them is the merchant's size, the merchant's bargaining power, or the merchant's contract. Interchange is identical for the corner shop and the supermarket on identical transactions. Everything that differs between them differs in the other two components.

The published rates, as of August 2026

The schemes publish their interchange schedules. The following are drawn from those published documents; each carries the effective date of the schedule it came from, because these documents are revised on a rolling basis and a figure without a date is a liability.

Mastercard, UK intra-country, consumer, schedule valid from 23 April 2022:

Fee tier	Consumer debit	Consumer credit
Contactless	0.20% (max GBP 1.00)	0.30%
Chip and PIN	0.20% (max GBP 1.00)	0.30%
Merchant UCAF	0.20% (max GBP 1.00)	0.30%
Full UCAF	0.20% (max GBP 1.00)	0.30%
Enhanced Electronic	0.20% (max GBP 1.00)	0.30%
Base	0.20% (max GBP 1.00)	0.30%

The flatness of that table is the regulation showing through. Every consumer tier sits exactly at the statutory ceiling, and the tier structure — which in an unregulated market would spread the rates by a factor of two or three — does nothing at all. The tiers still exist because the same schedule shape is used everywhere; in the UK consumer segment they have been squeezed flat.

Mastercard, UK intra-country, commercial, schedule valid from 22 January 2021, where the same tier structure does its normal work: commercial debit at 0.70 per cent contactless and chip-and-PIN, rising to 1.10 per cent for Enhanced Electronic and Base, with a cap of GBP 1.50 on certain programmes; commercial credit at 1.50 per cent contactless rising to 1.90 per cent Base.

Visa, UK domestic, commercial, schedule effective February 2026:

Product	Card present, EMV incl. contactless	Card not present	Non-qualified
Visa Business Debit	0.75%	1.20%	1.70%
Visa Platinum Business Debit	1.50%	1.80%	2.05%
Visa Infinite Business Debit	1.60%	1.90%	2.05%
Visa Business Credit / Deferred Debit	1.40%	1.70%	2.05%
Visa Platinum Business Credit	1.60%	1.90%	2.05%
Visa Infinite Business Credit	1.70%	2.00%	2.05%
Visa Corporate	1.75%	2.00%	2.05%
Visa Purchasing	1.75%	2.00%	2.05%

Two further rows in that schedule are worth knowing because they are counter-intuitive. *Visa Business Credit — Small Market Expense* is a flat 0.30 per cent, and *Visa Corporate — Large Market Enterprise* is

likewise 0.30 per cent. There exist commercial card programmes that interchange at the consumer credit rate. The generalisation "commercial cards are expensive" is a good default and a bad certainty.

Visa, intra-EEA consumer, schedule effective April 2025, applicable to EEA-issued cards at EEA merchants in a different country: Visa Consumer Debit, Visa Consumer Prepaid, V PAY Debit and V PAY Prepaid at 0.20 per cent across contactless, secure and non-secure; Visa Consumer Credit and Visa Consumer Deferred Debit at 0.30 per cent across the same tiers.

Visa, inter-EEA consumer, effective 19 October 2019, applicable to consumer cards issued outside the EEA transacting at EEA merchants:

Product	Card present	Card not present
Visa Consumer Debit and Prepaid	0.20%	1.15%
Visa Consumer Credit and Deferred Debit	0.30%	1.50%

That last table is the one to hold onto, because those four numbers — 0.20, 1.15, 0.30, 1.50 — recur throughout the regulatory history of the last seven years, and the same four numbers will reappear below in an entirely different context.

The caps: what the law says, and exactly how far it reaches

The instrument is Regulation (EU) 2015/751 of the European Parliament and of the Council of 29 April 2015 on interchange fees for card-based payment transactions, universally called the Interchange Fee Regulation or IFR. Its operative caps took effect on 9 December 2015.

Article 3 provides that "Payment service providers shall not offer or request a per transaction interchange fee of more than 0,2 % of the value of the transaction for any debit card transaction." Article 4 provides the same for credit cards at 0,3 %. Article 3 also permitted Member States to allow a flat per-transaction alternative of no more than EUR 0.05, alone or combined with the percentage, and permitted — until 9 December 2020 only — a domestic weighted-average approach for debit. That transitional route has now expired everywhere; the per-transaction cap is the live rule.

The methodology behind the two numbers is stated in the recitals rather than the articles. Recital 20 records that the caps are based on the Merchant Indifference Test, sometimes called the tourist test: the fee level at which a merchant would be indifferent between being paid by card and being paid in cash, having compared the costs of each. Whatever one thinks of the answer, it is worth knowing that the answer was computed rather than bargained.

The scope limits are in Article 1. Chapter II — which is where the caps live — does not apply to transactions with commercial cards, to cash withdrawals at ATMs or at the counter of a payment service provider, or to transactions with payment cards issued by three-party payment card schemes. Article 1(5) then closes the obvious loophole: where a three-party scheme licenses other payment service providers to issue or acquire, or issues with co-branding partners or through agents, it is considered to be a four-party scheme and the caps bite.

In the United Kingdom the instrument in force is the retained version of that Regulation as amended by the Interchange Fee (Amendment) (EU Exit) Regulations 2019 (SI 2019/284). The territorial scope was rewritten on withdrawal: the retained Regulation applies to card-based payment transactions in the United Kingdom where both the payer's and the payee's payment service providers are located in the United Kingdom. The Payment Systems Regulator is the lead competent authority, designated by HM Treasury, with enforcement powers under the Payment Card Interchange Fee Regulations 2015.

The PSR states the position plainly: consumer cross-border card payments between the UK and the EU, or any other third country, where either the acquirer or the issuer is based outside the UK's jurisdiction, are no longer subject to the interchange fee caps.

Three further articles of the IFR shape the merchant's bill and are routinely forgotten.

Article 5 provides that any agreed remuneration, including net compensation, with an equivalent object or effect to an interchange fee shall be treated as part of the interchange fee. This is the anti-circumvention rule, and it is why a scheme cannot simply route a rebate to issuers by another name.

Article 9 is the unblending rule, and it is the legal foundation of interchange-plus-plus pricing. An acquirer must charge its merchant "merchant service charges individually specified for different categories and different brands", unless the merchant requests blended charges in writing, and the agreement must contain individually specified information on the amount of the merchant service charges, interchange fees and scheme fees for each category and brand. IC++ is not a product the industry invented out of generosity. It is a statutory entitlement that most merchants have contracted out of without noticing.

Article 12 requires that, after execution, the acquirer provides the merchant with a reference enabling the transaction to be identified, the amount, and the charges, "indicating separately the merchant service charge and the amount of the interchange fee". Every merchant in the UK is entitled to transaction-level interchange data. Most have never asked for it.

Article 10 abolishes the honour-all-cards rule across categories — a merchant that accepts a scheme's consumer debit cards is not thereby obliged to accept its commercial cards — and Article 11 prohibits scheme rules that stop merchants steering customers towards a cheaper instrument or telling them what the fees are. Between them, these are the merchant's only real weapons against expensive card types, and both are underused.

All statements in this subsection describe the position as at August 2026. The UK's regulatory architecture is itself in motion: HM Treasury's consultation response of 21 April 2026 confirmed the government's intention to consolidate the Payment Systems Regulator entirely within the Financial Conduct Authority, transferring the PSR's functions including those under the Payment Card Interchange Fee Regulations 2015, subject to primary legislation.

The cross-border hole, which is where the money went

If the caps apply only when issuer and acquirer are both in the UK, then a very large volume of British card traffic sits outside them, and after 2021 the schemes repriced it.

For **UK–EEA** consumer transactions, the PSR's market review of UK-EEA consumer cross-border interchange fees, final report MR22/2.7 published 13 December 2024, records the change precisely. Fees on card-not-present transactions moved from 0.2 per cent to 1.15 per cent for debit and from 0.3 per cent to 1.5 per cent for credit — a fivefold increase on both. Visa announced in March 2021 and implemented in October 2021; Mastercard announced in the third quarter of 2021 and implemented in April 2022. Card-present rates were left at 0.2 and 0.3. The increases applied to what the PSR calls outbound interchange fees, being those on EEA-issued cards used at UK merchants, which is a cost falling on UK merchants and their acquirers. The PSR estimated the additional cost to UK service users at approximately £150 million to £200 million per year.

The PSR found that Mastercard and Visa were not subject to effective competitive constraints and had raised these fees to an unduly high level, and consulted on a two-stage remedy: an interim cap restoring 0.2 per cent for debit and 0.3 per cent for credit on UK-EEA consumer card-not-present transactions,

followed by a lasting cap set by a fuller methodology. As of the review page's October 2025 update, the PSR had decided not to proceed with the interim cap, citing litigation concerning its powers to impose price caps, and was instead consulting on the methodology for a longer-term cap. As of August 2026, therefore, the 1.15 and 1.5 rates remain in force and no UK statutory cap constrains them.

For **inter-regional** transactions — a card issued outside the EEA presented at an EEA merchant — the constraint is not a statute but a competition settlement. On 29 April 2019 the European Commission accepted commitments from Mastercard and Visa, making them legally binding, that cut inter-regional interchange by around 40 per cent on average to the four rates in the table above: 0.2 and 1.15 for debit card-present and card-not-present, 0.3 and 1.5 for credit. Those commitments ran to November 2024, at which point both schemes voluntarily extended the same levels to November 2029, an extension of which the Commission took note. This is a materially weaker form of protection than a cap in a regulation, and it has a stated end date.

The consequence for a merchant is worth stating in one sentence. Two identical £100 online sales, one to a customer holding a British consumer credit card and one to a customer holding a French consumer credit card, generate interchange of 30 pence and £1.50 respectively — five times the fee, for a difference the merchant cannot detect, cannot price for, and cannot decline without breaching its own acquiring agreement.

Scheme fees, named

Scheme fees are opaque by construction, but they are not unknowable, because the PSR's market review of card scheme and processing fees put a number of them on the public record.

On the Visa side, the review documents flat acquiring fees introduced in 2017 — Acquirer Service Fees for card-present and card-not-present — set at 1 basis point for debit transactions and 1.4 basis points for credit. It records the International Acquiring Fee rising from 30 to 45 basis points, and the International Service Assessment Fee rising from 10 to 55 basis points across 2018 to 2020. It records that when UK-EEA traffic became cross-border in 2021, Visa set the International Acquiring Fee on it at 10 basis points for card-present and 25 basis points for card-not-present — a scheme fee increase layered on top of the interchange increase described above, on the same transactions.

On the Mastercard side, the review documents a Switch Pricing Strategy revision of mandatory fees in 2017; a specific card-not-present fee charged to both acquirer and issuer from 2018; the extension of credits and returns scheme fees to credit transactions in 2018, which is the answer to the perennial question of whether refunds are free; a revision of MOTO fee drivers in 2020; a revision of the Acquiring Volume Fee in 2021 that eliminated the minimum volume fee; and mandatory Mastercard Cyber Secure fees in 2021. It also documents behavioural pricing on both sides — Mastercard's non-EMV non-contactless fee from 2019, charged on terminals not meeting the standard, and Visa's Negative Response Fee, removed in 2021 and replaced on the issuing side by a Minimum Approval Rate Integrity Fee. And it documents a structural trick that recurs: services converted from opt-in to opt-out, so that the fee begins to apply unless the participant acts. Mastercard's AAV validation went that way in 2019; Visa's analytics platform likewise.

The aggregate findings are the ones to quote. The PSR's final report MR22/1.10, published 6 March 2025, concluded that Mastercard and Visa had increased their core scheme and processing fees to acquirers by at least 25 per cent since 2017, amounting to at least £170 million extra per year for UK businesses, and that the two schemes were not subject to effective competitive constraints. In its consultation paper CP25/3 of December 2025 the PSR proposed two remedies by direction on Mastercard and Visa: an information, transparency and complexity remedy requiring the schemes to give acquirers clear and accurate information about existing, new and modified fees, with twelve months

to comply after final directions; and a pricing governance remedy requiring pricing decisions to be documented with evidence of how the interests of service users were considered, with four months to comply. Responses were due by 13 February 2026, with a third remedy on regulatory financial reporting deferred to a consultation by 31 March 2026.

Note what is being remedied. Not the level of the fees. The *legibility* of the fees.

The acquirer's share, and everything else on the invoice

Acquirer net revenue is defined by subtraction: MSC less interchange less scheme fees. It is the price of the acquirer's own service, and it covers the settlement risk the acquirer carries between funding the merchant and being made whole, the chargeback exposure, the underwriting, the reconciliation, the support, and the capital.

For the five largest UK acquirers — Barclaycard, Elavon, Global Payments, Lloyds Bank Cardnet and Worldpay — the PSR found that acquirer net revenues for card-acquiring services accounted for 62 per cent of total revenues, with card acceptance devices and payment gateways contributing 15 per cent and value-added services the remaining 23 per cent. Slightly more than a third of what a merchant pays its acquirer, in aggregate, is not for acquiring at all.

That surrounding third arrives as a tail of additional fees, and it is where the effective rate of a small merchant is actually decided. The PSR's description of standard pricing is worth reproducing in substance: several headline rates applied to different types of purchase transaction, each of which can be a pence-per-transaction fee, an ad valorem fee or a combination; plus one or more additional fees triggered either by specific events such as chargebacks, refunds and PCI DSS non-compliance, or by specific transaction types such as e-commerce. Add to this the minimum monthly service charge, which some acquirers apply when a merchant's monthly spend falls below a threshold; the monthly fee for PCI DSS compliance services; and terminal hire, which the PSR put at typically £10 to £40 per month per device, against card readers sold outright by the largest payment facilitators for between £15 and £45.

A merchant taking £2,000 a month through a single terminal, paying £25 for the terminal, £10 for PCI compliance and a £20 minimum service charge, is paying £55 in non-transactional fees. On £2,000 of turnover that is 2.75 per cent before a single transaction is rated. This is the arithmetic that explains why the very smallest merchants overwhelmingly use payment facilitators with a single headline rate and no monthly fees, even though that headline rate is higher: the PSR found the largest payment facilitators serve nearly 80 per cent of merchants with annual card turnover up to £15,000, and fewer than 5 per cent of small and medium merchants with turnover above £60,000. The crossover is real and it is arithmetic, not marketing.

The four pricing models, stated exactly

The PSR's taxonomy is the authoritative one and the four definitions are worth having verbatim in substance.

Standard pricing is pricing where, for any given transaction, the acquirer does not automatically pass through at cost the interchange fee applicable to that transaction, and which does not satisfy the criteria for IC+, IC++ or fixed pricing.

Interchange fee plus (IC+) is pricing where, for any given transaction, the acquirer automatically passes through at cost the interchange fee applicable to that transaction. Scheme fees remain inside the acquirer's margin.

Interchange fee plus plus (IC++) is pricing where, for any given transaction, the acquirer automatically passes through at cost both the interchange fee and the scheme fees applicable to that transaction.

Fixed pricing is where the merchant pays a fixed periodic fee for card-acquiring services whose amount does not depend on the volume or value of transactions accepted or their characteristics, within specified limits; the acquirer may separately pass through at cost any scheme fees not directly attributable to transactions.

Over 95 per cent of acquirers' merchants are on standard pricing. IC++ is concentrated at the top of the market, and the PSR estimated that for merchants with annual card turnover above £50 million on IC++ pricing, the benefit of the IFR savings was around £600 million in 2018 — because on IC++ a cut in interchange reaches the merchant automatically. For small and medium merchants the same review found, on average, little or no pass-through of the IFR savings, while acquirers may have passed on nearly all of the scheme fee increases. The pricing model is not a presentational choice. It determines who keeps the benefit of regulation and who absorbs the cost of scheme repricing.

Why the coffee shop and the airline pay different effective rates

Assemble the pieces and the answer has five independent components, none of which is "the airline negotiated better", although the airline did.

Ticket size. Fixed per-transaction elements — authorisation fees, per-item scheme fees, the pence component of a headline rate — are constant in pence and therefore hyperbolic in percentage terms. A 5p fixed element is 1.19 per cent of a £4.20 coffee and 0.01 per cent of a £480 fare. Nothing else in the stack produces a hundredfold difference this cheaply.

Card mix. The coffee shop's traffic is overwhelmingly UK consumer debit, capped at 0.20 per cent. The airline's traffic is heavy in commercial and corporate cards — the very products the IFR excludes — where Visa's published UK card-not-present rate for Visa Corporate is 2.00 per cent. A merchant whose customers pay with company cards is structurally, permanently more expensive to serve, and can do nothing about it except invoke Article 10 and refuse the category, which no airline will do.

Channel. The coffee shop is card-present, contactless, EMV, in the top interchange tier and the lowest fraud-liability position. The airline is card-not-present, which raises interchange on cross-border and commercial traffic, attracts card-not-present scheme fees on both sides, carries higher chargeback exposure, and — for cross-border consumer cards — is precisely the channel where UK-EEA interchange went to 1.15 and 1.5 per cent.

Region. The coffee shop's customers are almost all UK-issued. The airline's are, by the nature of the business, spread across the EEA and the rest of the world, which means a material fraction of its interchange is not capped at all, plus international acquiring and service assessment fees on top.

Merchant category code. The airline's MCC signals a delayed-delivery business: the customer pays in March and flies in August, and if the airline fails in June the acquirer is exposed to every unflown ticket. That risk is priced in acquirer net revenue and in collateral, not in interchange, and it is why the PSR observed that Elavon specialises in serving airlines, "which carry a higher credit risk than many other merchants". The airline's low headline rate frequently coexists with a rolling reserve that costs it more in working capital than the whole MSC.

Size, last and least interesting. The PSR's segmentation shows why bargaining power concentrates where it does: merchants with annual card turnover above £50 million are 0.1 per cent of merchants and 76 per cent of transaction value, while merchants below £380,000 are 93.7 per cent of merchants

and 6.5 per cent of value. The PSR found the market working adequately for the first group and not working well for everyone below £50 million.

One transaction, priced twice

The following puts the whole chapter into two columns. The interchange figures are published scheme rates as cited above. The scheme fee and acquirer margin figures are illustrative of a plausible contract and are marked as such — they are not published rates, and no merchant should treat them as a benchmark.

	Coffee shop	Airline
Transaction	£4.20	£480.00
Card	UK consumer debit	UK Visa Corporate
Channel	Contactless, card present	E-commerce, card not present
Interchange rate (published)	0.20%	2.00%
Interchange amount	£0.0084	£9.60
Scheme fees (illustrative)	£0.009	£0.53
Acquirer net revenue (illustrative)	£0.092	£0.54
Total MSC	£0.109	£10.67
Effective rate	2.60%	2.22%

And now the twist that makes the table worth printing. On these assumptions the airline's effective rate is *lower* than the coffee shop's, but barely — and the reason is not that the airline is being overcharged. It is that the airline's customer used a corporate card. Rerun the same £480 sale on a UK consumer credit card at 0.30 per cent and the interchange falls from £9.60 to £1.44, the MSC falls to about £2.51, and the effective rate falls to 0.52 per cent. Same merchant, same channel, same acquirer, same day, five times the difference, decided entirely by which piece of plastic came out of the wallet.

The coffee shop cannot reach 0.52 per cent by any means available to it, because at £4.20 a transaction the fixed components dominate everything else. The airline reaches it or does not reach it depending on its customer mix, which it does not control either.

Which is the honest ending to this chapter. Almost none of what a merchant pays is set by the merchant, or by the party that invoices the merchant, or in the country the merchant trades in. It is set in a rate schedule published by a company the merchant has no contract with, applied to a card issued by a bank the merchant has never dealt with, through a rating engine the merchant cannot see, a day after the sale. The only two levers a merchant genuinely holds are the acquirer's own margin, which is typically the smallest of the three components on a large ticket and the largest on a small one, and the shape of the bill — because a merchant who cannot see the three components separately cannot tell which lever to pull.

That is what Article 9 was for. Ask for it in writing.

31.98 Common wrong ideas

Wrong: The merchant pays interchange to Visa or Mastercard. **Right:** The acquirer pays it to the card-issuing bank; the scheme is not the recipient and the merchant is not the payer, and the merchant has no relationship with the party that receives it.

Wrong: Interchange is a fee that somebody invoices. **Right:** It is a deduction taken when the issuer settles with the scheme for the day's clearing, so no bill for it is sent to anybody and no discrete transfer of it ever happens.

Wrong: The interchange rate is fixed at the moment the card is tapped. **Right:** It is derived at clearing from the contents of the clearing record, so a transaction can be re-rated afterwards and a merchant debited weeks later for the difference.

Wrong: Scheme fees are a percentage you can quote. **Right:** They are several hundred separately named line items across two schemes, some ad valorem, some per authorisation message whether approved or declined, some per chargeback, per BIN or per licence, some charged only on behaviour the scheme wishes to discourage, and many charged to the issuer where the merchant never sees them.

Wrong: UK interchange is regulated, so the caps apply to UK card payments. **Right:** The caps apply only to consumer card transactions on four-party schemes where the payer's and the payee's payment service providers are both located in the United Kingdom, which leaves commercial cards, cash withdrawals and every cross-border pair outside them.

Wrong: Commercial cards are always expensive. **Right:** Visa Business Credit Small Market Expense and Visa Corporate Large Market Enterprise both interchange at 0.30 per cent, the consumer credit rate, so the generalisation is a good default and a bad certainty.

Wrong: A merchant chooses between blended pricing and IC++. **Right:** Over 95 per cent of merchants are on standard pricing, which is neither: several headline rates attached to broad categories, plus a tail of event-triggered fees, with the classifications that decide which rate applies never shown to the merchant.

Wrong: Large merchants get better interchange than small ones. **Right:** Interchange is identical for the corner shop and the supermarket on identical transactions, and everything that differs between them differs in the acquirer's margin and the scheme fees.

Wrong: IC++ pricing is a concession an acquirer may or may not grant. **Right:** Article 9 makes individually specified charges the default entitlement, and blending requires the merchant to ask for it in writing, which is a statutory right most merchants have contracted out of without noticing.

Wrong: A low headline rate means a cheap merchant account. **Right:** Terminal hire, a PCI compliance fee and a minimum monthly service charge can amount to 2.75 per cent of a small merchant's turnover before a single transaction has been rated, which is the arithmetic that sends the smallest merchants to payment facilitators.

31.99 Chapter summary in 20 lines

1. The gap between what a customer pays and what a merchant receives is called the merchant service charge, and it is three fees wearing one coat.
2. Interchange runs from the acquirer to the issuing bank, which is the opposite direction from the one almost everybody assumes.
3. Scheme fees go from both the acquirer and the issuer to Visa or Mastercard, for the rulebook, the brand, the dispute machinery and the switching.
4. Acquirer net revenue is what is left after those two, and it is the only component the merchant can negotiate.
5. Interchange is never invoiced: the issuer remits the transaction value less the applicable fee, and the acquirer simply receives less than it presented.
6. It is also derived rather than quoted, computed by the scheme's rating engine at clearing, which means it is not certain at the point of sale and can be re-rated later.

7. The inputs are the issuer's product and country, the acquirer's country, the channel and authentication tier, the merchant category code, the amount and currency, and the completeness of the data submitted.
8. None of those inputs is the merchant's size, its bargaining power or its contract, so interchange is the same for the corner shop and the supermarket.
9. The statutory caps of 0.2 and 0.3 per cent come from the Interchange Fee Regulation and were computed by the Merchant Indifference Test rather than bargained.
10. Those caps reach only consumer cards on four-party schemes with both payment service providers in the United Kingdom, excluding commercial cards, cash withdrawals and all cross-border traffic.
11. That exclusion is where the money went: UK-EEA consumer card-not-present interchange moved from 0.2 to 1.15 per cent for debit and 0.3 to 1.5 per cent for credit, at an estimated cost to UK service users of £150 million to £200 million a year.
12. Inter-regional rates sit at the same four numbers, but by a competition settlement with a stated end date rather than by statute, which is materially weaker protection.
13. Scheme fees are not one price but a catalogue of several hundred items of different shapes, some charged on declines, some on refunds, and many charged to issuers where the merchant never sees them but still feels the economics.
14. The regulator found scheme and processing fees to acquirers up by at least 25 per cent since 2017, worth at least £170 million a year, and proposed remedies aimed at the legibility of those fees rather than their level.
15. Over 95 per cent of merchants are on standard pricing, in which the components are invisible and the applicable headline rate depends on classifications the merchant was never shown.
16. Article 9 makes unblended charges a statutory entitlement and Article 12 entitles every merchant to transaction-level interchange data, and most merchants have never asked for either.
17. Slightly more than a third of what merchants pay their acquirers in aggregate is not for acquiring at all, but for devices, gateways and value-added services.
18. Fixed per-transaction elements are constant in pence and therefore hyperbolic in percentage terms, which is why the coffee shop's effective rate can exceed the airline's.
19. Rerun the same £480 sale on a consumer credit card rather than a corporate card and the effective rate falls from about 2.22 per cent to about 0.52 per cent, decided entirely by which piece of plastic came out of the wallet.
20. Almost nothing a merchant pays is set by the merchant, by the party that invoices the merchant, or in the country the merchant trades in, so the only genuine levers are the acquirer's own margin and the shape of the bill.

Sources: Regulation (EU) 2015/751 (Articles 1, 3, 4, 5, 9, 10, 11, 12 and Recital 20) as retained in UK law and amended by the Interchange Fee (Amendment) (EU Exit) Regulations 2019; Payment Systems Regulator MR18/1.8, MR22/1.6, MR22/1.10, CP25/3 and MR22/2.7 and its IFR guidance pages; European Commission press release IP/19/2311; published interchange schedules of Visa (UK, intra-EEA and inter-EEA) and Mastercard (UK intra-country); ISO 8583 and ISO 18245:2023; HM Treasury, "A Streamlined Approach to Payment Systems Regulation" consultation response; UK Finance card spending statistics.

32

CHAPTER

ISO 20022

32.0 What this chapter gives you

1. You will be able to explain why ISO 20022 is not a format but a method for producing formats, a dictionary of business concepts and a committee structure for deciding what goes into it.
2. You will be able to say what actually retired on 22 November 2025, and why the authorisation message leaving a shop terminal was untouched by it.
3. You will be able to read a message identifier such as `pacs.008.001.08`, name each of its four parts, and say why a version increment is a release rather than a configuration change.
4. You will be able to distinguish a `pain` message from a `pacs` message and say which party sends which.
5. You will be able to explain why "richer data" describes a capacity rather than a delivery, given that the unstructured remittance field is the same one hundred and forty characters in both the old message and the new one.
6. You will be able to say what a structured postal address changes for sanctions screening, and why the address mandate slipped repeatedly for reasons that were about stored data rather than software.
7. You will be able to explain why translating a rich message into a legacy format is lossy by construction, and why a round-trip diff is a more useful document than any readiness report.
8. You will be able to say why validating a message against its schema does not validate its codes, and how often the external code sets have to be refreshed.
9. You will be able to explain why a domestic real-time gross settlement system can cut over in a weekend while a correspondent network of eleven thousand institutions needed two and a half years of coexistence.
10. You will be able to state the single design rule the chapter turns on: keep the ISO 20022 message as the system of record, and translate only at the last mile.

The name is a trap. Everything else in this volume has been a format: ISO 8583 is a format, EMV tag-length-value is a format, a Bacs Standard 18 record is a format. ISO 20022 is not a format. It is a method for producing formats, plus a dictionary of business concepts that those formats draw on, plus a committee structure for deciding what goes into the dictionary. When somebody says "we've gone ISO 20022", they have not adopted a message. They have adopted a family of several hundred messages, of which they probably use four.

The second trap is the word "replaces". ISO 20022 did not replace ISO 8583 in the way that HTTP/2 replaced HTTP/1.1. As of August 2026, when you tap a card in a shop, the authorisation that leaves the terminal is still an ISO 8583 `0100` and the reply is still an `0110`, and there is no serious industry programme to change that. What ISO 20022 has actually replaced, and replaced almost completely, is

the *other* legacy format: the SWIFT MT messages that carried cross-border and high-value payments between banks. That migration finished its main phase on 22 November 2025. This chapter is about what changed, what it bought, what it cost, and why a change that everybody agreed on in 2004 was still being argued about in 2027.

The plain version

In the ISO 8583 chapter you learned the numbered form: everybody memorises the code book, box 4 is always the amount, and the message is short because nothing in it says what anything is. That works beautifully until the day you need to put something in the message that the code book never anticipated.

So think about a freight depot instead.

Harkness Joinery in Leeds sends pallets of timber all over Europe. For forty years every pallet went out with a docket: a small card with numbered boxes on it, printed in the depot's own house style. Box 3 is the weight. Box 8 is the destination depot code. Box 14 is *notes*, and box 14 is one hundred and forty characters long, no more, because that is how wide the card is.

The docket works. It is small, it is cheap, it goes through the sorting machine at speed, and every depot from Leeds to Lisbon has the code book pinned to the wall. The trouble is box 14.

Because what actually needs to travel with the pallet is not a note. It is a list. This pallet covers three orders — order 4471 for £7,200, order 4488 for £6,150 and order 4502 for £5,100 — and the customer's warehouse clerk needs to know which planks belong to which order, or she cannot sign anything off. One hundred and forty characters is not enough for three orders with their dates and amounts, so the Leeds despatcher does what despatchers everywhere do. He writes:

```
ORD 4471 4488 4502 PART DEL SEE EMAIL
```

and the pallet leaves. At the other end a human being reads that, opens a spreadsheet, and spends twenty minutes working out what arrived. Multiply by four hundred pallets a day.

There is a second problem, and it is quieter. Box 9 on the docket is the consignee: four lines of text, thirty-five characters each. The Leeds despatcher types:

```
VAN DOORN HOUT BV
KANAALDIJK 14
5613 EA EINDHOVEN
NETHERLANDS
```

That is perfectly clear to you. It is completely opaque to a machine. Nothing on the docket says which of those four lines is the town. A computer at a border checkpoint that has been told to stop every consignment going to one particular address in one particular city has to guess, by pattern-matching lumps of text against a list, and it will guess wrong in both directions: it will stop pallets it should let through, and let through pallets it should stop.

Now the new sheet.

The new sheet is A4, not a card. Every line on it says what it is. Instead of box 9 there is a block that reads:

```
STREET NAME: Kanaaldijk
BUILDING NUMBER: 14
POST CODE: 5613 EA
```

```
TOWN NAME: Eindhoven
COUNTRY: NL
```

and instead of box 14 there is a whole labelled section headed *what this pallet settles*, which repeats as many times as it needs to:

```
ORDER NUMBER: 4471   DATED: 2026-06-02   AMOUNT: 7,200.00
ORDER NUMBER: 4488   DATED: 2026-06-09   AMOUNT: 6,150.00
ORDER NUMBER: 4502   DATED: 2026-06-15   AMOUNT: 5,100.00
```

Three things follow from that, and they are the whole point of the standard.

The sheet is fatter. It is several times the size of the docket, because every value now drags a label along behind it. In 1985 that would have been disqualifying. In 2026, when the depot's network cost is not measured in characters, it is a rounding error.

The sheet is self-describing. A depot that has lost the code book can still read it. More usefully, a piece of software written by someone who has never met the Leeds despatcher can read it, because "TOWN NAME" means town name everywhere the sheet is used, in every kind of sheet the depot prints — deliveries, returns, invoices, insurance claims. That consistency is not a happy accident; it is the actual product. Somebody sat down and wrote a dictionary that says what a town name is and what a consignee is, and then every form gets built out of the dictionary rather than being invented from scratch.

The clerk at the far end stops doing twenty minutes of work. This is what "richer data" means, stripped of the brochure language. It does not mean more characters. It means the same facts arrive already separated into their component parts, so that a machine can act on each part without guessing.

Now translate the whole thing back into money, with real numbers.

Harkness Joinery pays Van Doorn Hout BV **€18,450.00**, settling those three invoices. Under the old system that payment travelled as an MT 103, and the MT 103 has a field for remittance information called field 70, and field 70 is four lines of thirty-five characters — **one hundred and forty characters** — and that is all there is. The beneficiary's name and address travel in field 59, four lines of thirty-five characters, undifferentiated. Van Doorn's finance team gets a bank statement showing €18,450.00 from a name they half recognise and a reference that reads `INV 4471 4488 4502`. Somebody matches it by hand.

Under the new system the same payment travels as a message called `pacs.008`. The three invoices go in as three separate structured entries, each with its own number, its own date and its own amount. The address goes in with the town in the box marked town and the country in the box marked country. Van Doorn's ledger software reads the message, matches all three invoices automatically, closes them, and nobody touches it.

That is the promise. Here is why it took twenty-two years to deliver.

The new sheet is worthless unless the depot at the other end can read it. If Leeds prints the A4 sheet and Eindhoven only has a slot for the small card, the sheet has to be cut down to fit — and cutting it down means throwing away exactly the three invoice lines that made it worth printing. So for years every depot has to be able to handle both, and the network only moves at the speed of its worst reader. Nobody can go first, because going first costs money and buys nothing. Everybody has to go at once, and "everybody" is eleven thousand institutions in two hundred countries, each with thirty years of software behind the counter that was written on the assumption that a reference is never longer than thirty-five characters.

The way that gets solved, in the end, is not persuasion. It is a date. On **22 November 2025** the old dockets stopped being accepted for cross-border payment instructions on the SWIFT network, and everybody who had been putting it off found out what their real deadline had been all along.

Where the plain version stops being true

ISO 20022 is not a message format, and calling it "the replacement for ISO 8583" is wrong in two directions at once. It is a modelling standard: a metamodel, a repository of business concepts, and a set of rules for generating machine-readable schemas from that repository. The catalogue currently holds around eight hundred registered message definitions spanning payments, securities, foreign exchange, trade finance, collateral and cards. What was retired on 22 November 2025 was not ISO 8583 but SWIFT MT — the FIN messages of the ISO 15022 era — for cross-border payment instructions and reporting. Card authorisation traffic is untouched. ISO 20022 does contain card message sets (the business areas `caaa`, `cain`, `catp` and `casp` cover acceptor-to-acquirer, acquirer-to-issuer, ATM and sale-to-POI messaging, and the nexo standards build on them), and they have made almost no impression on global acquiring volume. The `0100` you met earlier in this volume will outlive everyone reading this.

"Richer data" describes a capacity, not a delivery. The unstructured remittance field in `pacs.008` is still one hundred and forty characters, exactly the same as MT 103 field 70; the CBPR+ market practice for cross-border payments permits one occurrence of it. The gain is structure, not length, and structure only pays when the sender populates it and the receiver consumes it. The clearest evidence that this is harder than it sounds is the Bank of England: having proposed to mandate structured remittance data in CHAPS from November 2027, it announced in **July 2026** that it would not, and would instead work on international alignment first. A field that exists, is schema-valid and is uniformly empty has bought nobody anything.

XML versus packed binary is the least interesting difference between the two standards, and ISO 20022 is not intrinsically XML. ISO 20022-4:2013 specifies XML Schema generation; ISO 20022-8:2013 specifies ASN.1 generation; the Registration Management Group approved a JSON Schema generation recommendation on 10 June 2025 targeting JSON Schema Draft 2020-12; and ISO 20022-9, published in April 2026, generalises syntax generation rules so that the model is not bound to any one encoding. Card messages under nexo are routinely carried as ASN.1, not XML. The substantive difference is not angle brackets versus bitmaps. It is that ISO 8583 defines a message and leaves the meaning of its contents to bilateral agreement, whereas ISO 20022 defines the meaning centrally and derives the message from it. That is why two ISO 8583 implementations can both be "compliant" and unable to talk to each other, and why the same concept has the same name in an ISO 20022 payment, statement and investigation.

Coexistence and translation preserve delivery, not meaning, and the loss is arithmetic rather than sloppiness. During coexistence a rich `pacs.008` bound for a bank that could only receive MT was translated in flight. MT 103 field 70 holds 140 characters and field 59 holds four lines of thirty-five; a `RemittanceInformation` block with three structured invoice entries and a fifteen-element postal address does not fit, so it is truncated or dropped. The translation is not reversible: converting back the other way produces a schema-valid message containing less than the original. This matters legally as well as operationally, because the originator and beneficiary information that must accompany a wire transfer under FATF Recommendation 16 is precisely the information that translation squeezes. An institution that keeps a translated MT as its record of the payment has quietly destroyed the evidence the standard existed to carry.

The technical version

What the standard actually is

ISO 20022 is maintained under ISO Technical Committee 68, Subcommittee 9. The core standard was published in eight parts in May 2013 and extended in 2026:

Part	Title	Published
ISO 20022-1	Metamodel	May 2013
ISO 20022-2	UML profile	May 2013
ISO 20022-3	Modelling	May 2013
ISO 20022-4	XML Schema generation	May 2013
ISO 20022-5	Reverse engineering	May 2013
ISO 20022-6	Message transport characteristics	May 2013
ISO 20022-7	Registration	May 2013
ISO 20022-8	ASN.1 generation	May 2013
ISO 20022-9	Syntax generation requirements and rules	April 2026

The architecture has three layers and confusing them is the source of most arguments about the standard. At the top is the **business model**: business processes, business roles and business concepts, expressed in UML and independent of any message. In the middle is the **logical message**: a message definition assembled from components in the Data Dictionary. At the bottom is the **physical syntax**: an XML Schema, an ASN.1 module or a JSON Schema, generated mechanically from the logical message by the rules in Parts 4, 8 or 9. The generated XSDs carry a comment recording the Standards Editor build and the ISO 20022 version they were generated from, which tells you they are outputs, not hand-written artefacts.

Governance is a registry, not a standards body in the usual sense. SWIFT acts as **Registration Authority**, maintaining the repository and checking that submissions comply with the technical specifications. The **Registration Management Group** oversees registration and reports to TC 68/SC 9. **Standards Evaluation Groups** — for Payments, Securities, Cards and Related Retail Financial Services, Foreign Exchange, Trade Finance and API Resources — validate new message definitions from a business perspective and approve changes to existing ones. A **Cross-SEG Harmonisation Group** resolves conflicts where the same concept appears in more than one domain, and a **Technical Support Group** handles the syntax mechanics. A new message begins with a business justification accepted by the RMG, is developed by a submitting organisation, is validated by the relevant SEG, and is then published by the RA into the repository.

Message definitions are grouped into four-letter business areas. The payments and cards ones matter here:

Code	Business area
`pain`	Payments initiation
`pacs`	Payments clearing and settlement
`camt`	Cash management
`remt`	Payments remittance advice
`acmt`	Account management
`admi`	Administration
`head`	Business application header
`auth`	Authorities

Code	Business area
reda	Reference data
caaa	Acceptor to acquirer card transactions
cain	Acquirer to issuer card transactions
catp	ATM card transactions
casp	Sale to POI card transactions
catm	POI management
tokm	Payment token management

The distinction between `pain` and `pacs` is the one that catches newcomers. A `pain` message is customer to bank: a corporate instructing its bank to make payments. A `pacs` message is bank to bank: the interbank leg. A corporate never sends a `pacs.008` and a clearing system never receives a `pain.001`.

The message identifier and the namespace

A message definition identifier has four parts:

```
pacs . 008 . 001 . 08
 |      |     |     |
 |      |     |     +-- version (two digits)
 |      |     +-------- variant (three digits; 001 is the base variant)
 |      +-------------- message functionality (three digits)
 +--------------------- business area (four letters)
```

`pacs.008.001.08` is version 8 of the base variant of message 008 in the payments clearing and settlement area — `FIToFICustomerCreditTransferV08`. The XML target namespace embeds the whole identifier:

```
urn:iso:std:iso:20022:tech:xsd:pacs.008.001.08
```

That last point has an operational consequence that is regularly underestimated. A version increment changes the namespace, which changes every schema binding, every XPath expression and every generated class in every application that touches the message. Upgrading from `pacs.008.001.08` to `pacs.008.001.13` is not a configuration change. It is a release. This is why market infrastructures pin versions and hold them for years: the SEPA 2025 scheme implementation guidelines use the 2019 message versions throughout — `pacs.008.001.08`, `pacs.004.001.09`, `pacs.002.001.10`, `pacs.028.001.03`, `camt.056.001.08`, `camt.029.001.09`, `camt.027.001.07`, `camt.087.001.06` — and CBPR+ pinned the same generation for cross-border.

The MT messages that were retired map across roughly as follows. "Roughly" is load-bearing: the mappings are many-to-one in places and none of them is lossless.

Retired MT	ISO 20022 equivalent
MT 103, MT 102	`pacs.008` FI to FI customer credit transfer
MT 200, 201, 202, 203, 205	`pacs.009` financial institution credit transfer
MT 202 COV	`pacs.009` COV variant
MT 900, MT 910	`camt.054` debit/credit notification
MT 940, MT 950	`camt.053` bank to customer statement
MT 941, MT 942	`camt.052` bank to customer account report
MT 192, MT 292	`camt.056` payment cancellation request
MT 196, MT 296	`camt.029` resolution of investigation
MT 101	`pain.001` customer credit transfer initiation

MT 199 and MT 299 were not retired in November 2025; they remain in use for transaction tracking, with their exceptions-and-investigations and payment-initiation use cases scheduled to migrate later.

Anatomy of a pacs.008

A CBPR+ payment on the wire is two documents in one envelope. The first is the **Business Application Header**, `head.001.001.02`, which carries routing and identification: `Fr` and `To` (the sending and receiving BICs), `BizMsgIdr` (the business message identifier, which CBPR+ requires to equal the group header's `MsgId`), `MsgDefIdr` (`pacs.008.001.08`), `BizSvc` (the business service, for example `swift.cbprplus.02`, or `swift.cbprplus.stp.02` for the straight-through-processing variant) and `CreDt`.

The second is the message document itself, rooted at `FIToFICstmrCdtTrf`, with a `GrpHdr` and one or more `CdtTrfTxInf`. Under CBPR+, `NbOfTxs` is fixed at 1 — one payment per message — which is a market practice restriction, not a schema restriction, and is the single most common cause of a schema-valid message being rejected by the network.

The datatypes are worth stating exactly, because they are where the contrast with ISO 8583 becomes concrete. Taking the definitions from the `pacs.008.001.09` XML Schema:

- `PmtId/EndToEndId` is `Max35Text`, mandatory, and must be passed unchanged along the whole chain.
- `PmtId/UETR` is `UUIDv4Identifier`, constrained by the pattern `[a-f0-9]{8}-[a-f0-9]{4}-4[a-f0-9]{3}-[89ab][a-f0-9]{3}-[a-f0-9]{12}` — thirty-six characters, a version-4 UUID, unique for the life of the payment.
- `IntrBkSttlmAmt` is `ActiveCurrencyAndAmount`: an `xs:decimal` restricted to `totalDigits` 18, `fractionDigits` 5 and `minInclusive` 0, carrying a mandatory `Ccy` attribute matching `[A-Z]{3,3}` — the ISO 4217 alphabetic code.
- `ChrgBr` is `ChargeBearerType1Code`, an enumeration of exactly `DEBT`, `CRED`, `SHAR` and `SLEV`.
- `SttlmInf/SttlmMtd` is `SettlementMethod1Code`: `INDA`, `INGA`, `COVE`, `CLRG`, `TDSO`, `TDSA`.
- Agent identification uses `BICFI`, typed `BICFIDec2014Identifier` with the pattern `[A-Z0-9]{4,4}[A-Z]{2,2}[A-Z0-9]{2,2}([A-Z0-9]{3,3}){0,1}` — the ISO 9362 BIC, eight or eleven characters.
- Accounts use `IBAN`, typed `IBAN2007Identifier`, pattern `[A-Z]{2,2}[0-9]{2,2}[a-zA-Z0-9]{1,30}` — ISO 13616.
- Legal entities use `LEI`, typed `LEIIdentifier`, pattern `[A-Z0-9]{18,18}[0-9]{2,2}` — twenty characters, ISO 17442.
- `Ctry` is `CountryCode`, pattern `[A-Z]{2,2}` — ISO 3166-1 alpha-2.
- `CreDtTm` is `ISODateTime`, which is `xs:dateTime`; CBPR+ requires the UTC offset to be present.

The same facts in both formats

This is the table to keep. It is the honest comparison between what the terminal sends and what the interbank leg sends, for the concepts that appear in both.

Concept	ISO 8583	ISO 20022 (`pacs.008`)
Message type	MTI, four digits: version, class, function, origin (`0100`, `0200`, `0800`)	Message definition identifier plus namespace (`pacs.008.001.08`)
Which fields are present	64-bit primary bitmap; bit 1 signals a secondary bitmap for elements 65 to 128	Element presence in the XML instance; cardinality in the schema
Account identifier	DE2 PAN	`DbtrAcct/Id/IBAN` or `Othr/Id`

Concept	ISO 8583	ISO 20022 (`pacs.008`)
Transaction nature	DE3 processing code	`PmtTpInf`, plus `Purp/Cd` from an external code set
Amount	DE4, fixed 12 digits, implied decimal from the currency's minor units	`IntrBkSttlmAmt`, decimal, 18 total digits, 5 fractional
Currency	DE49, ISO 4217 numeric, separate field	`Ccy` attribute on the amount, ISO 4217 alphabetic
Date and time	DE7 transmission date and time, `MMDDhhmmss`, no year, no zone; DE12 local time, DE13 local date	`CreDtTm`, `xs:dateTime` with offset; `IntrBkSttlmDt`, `xs:date`
Trace number	DE11 STAN, six digits, wraps within the day	`PmtId/UETR`, UUIDv4, unique end to end
Reference	DE37 retrieval reference number, twelve characters	`PmtId/EndToEndId`, `Max35Text`; `PmtId/TxId`; `PmtId/InstrId`
Approval identifier	DE38 authorisation identification response, six characters	No equivalent; the model is not authorise-then-clear
Outcome	DE39 response code, two characters	`pacs.002` `TxSts`, four characters, plus `StsRsnInf/Rsn/Cd`
Terminal	DE41 terminal ID, eight characters	No equivalent
Chip data	DE55, EMV tag-length-value	No equivalent

Two rows in that table are worth dwelling on.

The **amount** row explains a class of migration bug. DE4 is a fixed-width integer whose decimal point is implied by the currency in DE49; `18450.00` in euros is the twelve characters `000001845000`. `IntrBkSttlmAmt` is a decimal with the currency as an attribute: `<IntrBkSttlmAmt Ccy="EUR">18450.00</IntrBkSttlmAmt>`. Anything that parses the second into a binary floating-point type has introduced a defect that will surface as a one-penny break in a reconciliation six months later. Five fractional digits also means the schema will happily accept `18450.00001`, which is a valid decimal and not a valid euro amount; the constraint that the value must respect the currency's minor units lives in the usage guideline, not in the XSD.

The **outcome** row explains why anyone bothered. In ISO 8583, everything the issuer wants to say about a decline is compressed into DE39, two characters, of which the most commonly used means "do not honour" and is deliberately uninformative. In ISO 20022 the outcome is split: a `pacs.002` carries `TxSts` from a four-character external code set — `RCVD`, `ACTC`, `ACCP`, `ACSP`, `ACWC`, `ACSC`, `PDNG`, `RJCT` — and where the status is `RJCT` a separate `StsRsnInf/Rsn/Cd` is mandatory, itself from a further external code set, with `AddtlInf` available for free text. Status and reason are different facts and get different fields. Whether the reason is any more candid than "do not honour" is a policy question, not a format question, and in the sanctions cases it usually is not.

The address problem, exactly

Structured addresses are the least glamorous part of this migration and by a distance the most expensive. Here is the actual structure, from the `pacs.008.001.09` schema (the complex type is `PostalAddress24`):

Element	Tag	Type
Address type	`AdrTp`	choice
Department	`Dept`	`Max70Text`
Sub department	`SubDept`	`Max70Text`
Street name	`StrtNm`	`Max70Text`
Building number	`BldgNb`	`Max16Text`
Building name	`BldgNm`	`Max35Text`
Floor	`Flr`	`Max70Text`
Post box	`PstBx`	`Max16Text`
Room	`Room`	`Max70Text`
Post code	`PstCd`	`Max16Text`
Town name	`TwnNm`	`Max35Text`
Town location name	`TwnLctnNm`	`Max35Text`
District name	`DstrctNm`	`Max35Text`
Country subdivision	`CtrySubDvsn`	`Max35Text`
Country	`Ctry`	`CountryCode`, `[A-Z]{2,2}`
Address line	`AdrLine`	`Max70Text`, up to 7 occurrences

Three population patterns are possible and the industry has given them names. **Unstructured** uses `AdrLine` only. **Fully structured** uses the named elements and no `AdrLine` at all. **Hybrid** uses `AdrLine` for the street part but populates `TwnNm` and `Ctry` as discrete elements. CBPR+ narrows `AdrLine` to three occurrences; the Fedwire Funds Service hybrid profile requires at least country code and town name plus up to two free-format lines of seventy characters.

The deadline is firm and dated. Swift is removing the unstructured option with Standards Release 2026: from **November 2026**, a CBPR+ payment instruction carrying a purely unstructured postal address is rejected at the network, and only hybrid or fully structured addresses are accepted, with `TwnNm` and `Ctry` mandatory.

Compare what this replaced. MT 103 field 59 is `[/34x] 4*35x`: an optional account line followed by four lines of thirty-five characters holding name and address together, with no separator between the name and the address and no indication of which line is which. Field 50a option K is the same shape for the ordering customer. Every sanctions screening engine in the world has spent thirty years fuzzy-matching those blobs, and the false positive rate that follows is the reason a bank's payment investigations team is the size it is. Moving the town and the country into their own elements does not make screening perfect; it changes it from a text-similarity problem into a lookup, which is a different order of problem.

The reason this deadline slipped, repeatedly, has nothing to do with software. Banks cannot populate `TwnNm` for a customer whose stored address is four lines of free text typed into a branch terminal in 1998. Address remediation across tens of millions of customer records, with no reliable way to distinguish a street from a district in a language nobody in the remediation team reads, is a multi-year data project. Swift's own response has been to offer a machine learning service that infers structure from legacy address strings, which tells you how tractable the manual approach was judged to be.

Remittance information, exactly

`RmtInf` is typed `RemittanceInformation16` and has exactly two children:

- `Ustrd`, `Max140Text`, unbounded in the base schema.
- `Strd`, `StructuredRemittanceInformation16`, unbounded in the base schema.

`StructuredRemittanceInformation16` contains, in order: `RfrdDocInf` (referred document information, unbounded, each with `Tp`, `Nb` as `Max35Text`, `RltdDt` as `ISODate` and `LineDtls` for line-item detail), `RfrdDocAmt`, `CdtrRefInf` (with `Tp` and `Ref` as `Max35Text`), `Invcr`, `Invcee`, `TaxRmt`, `GrnshmtRmt`, and up to three occurrences of `AddtlRmtInf` as `Max140Text`.

That is where the three invoices in the worked example live: three `Strd` occurrences, or one `Strd` with three `RfrdDocInf` occurrences, each carrying its own number, date and amount. There is also `RltdRmtInf`, typed `RemittanceLocation7`, up to ten occurrences, which does not carry the remittance data at all but a pointer to it: a `Mtd` from `{FAXI, EDIC, URID, EMAL, POST, SMSM}` and an `ElctrncAdr` of up to 2,048 characters. That is the escape valve for genuinely large remittance sets, alongside the standalone `remt.001` remittance advice message.

Now the honest part, which vendor material tends to omit. CBPR+ restricts `Ustrd` to a single occurrence of 140 characters. MT 103 field 70 is `4*35x` — also 140 characters. For an institution that migrated by mapping field 70 into `Ustrd` and declaring victory, the cross-border remittance capacity did not increase by one byte. Every gain in this area comes from `Strd`, and `Strd` requires the originating corporate to supply invoice-level data, the originating bank to carry it, every intermediary to pass it unchanged, and the beneficiary's bank to present it on the statement. That is four consenting parties, and the Bank of England's decision in July 2026 to abandon its November 2027 structured remittance mandate for CHAPS is the clearest available evidence of how many of them were ready.

Codes that live outside the schema

A detail with disproportionate operational weight: many of the code sets are not in the schema. `ExternalPurpose1Code` is declared in the XSD merely as a string of one to four characters. `ExternalCategoryPurpose1Code` and `ExternalServiceLevel1Code` are the same; `ExternalLocalInstrument1Code` allows up to thirty-five. The permitted values live in the **External Code Sets**, published by the Registration Authority on a quarterly cycle at the end of February, May, August and November, in XLSX, XSD and JSON. The point of externalising them is that a new code can be added without incrementing a message version.

The consequence for an implementer is that validating a message against the XSD does not validate its codes, and that any implementation which compiles the code list into an enumeration in source will be wrong within three months. Treat the external code sets as reference data with a refresh job, exactly as you would treat a BIN table.

Purpose codes are the clearest example of what this buys. The Bank of England's recommended UK purpose code list includes, among others:

Code	Meaning
`PCOM`	Final payment to complete the purchase of a property
`PLDS`	Payment of funds from a lender as part of the issuance of a property loan
`PLRF`	Transfer or extension of a property financing arrangement to a new deal
`HLRP`	Property loan repayment
`HLST`	Property loan settlement
`INBC`	Settlement of interbank charges not being recharged to the debtor
`BKFE`	Bank loan fees, including agent, amendment and commitment fees

Code	Meaning
CORT	Trade settlement payment

Since **May 2025** the Bank has mandated purpose codes on all CHAPS payments between financial institutions and on property transactions, and Legal Entity Identifiers on all CHAPS payments between financial institutions. From **November 2027** purpose codes extend to all CHAPS payments, and LEIs to `pacs.004` returns and `pacs.009` COV payments between financial institutions. A four-character code declaring that a given £340,000 movement is a property completion rather than an unexplained transfer is worth more to a financial crime team than any amount of free text, and it costs eleven bytes.

Migration state, as of August 2026

Infrastructure	ISO 20022 date	Approach
SEPA (EPC schemes)	Native since scheme launch	Not applicable
T2, Eurosystem RTGS, replacing TARGET2	Live 20 March 2023, migration weekend 17 to 20 March	Big bang
EURO1, EBA Clearing	March 2023, aligned with T2	Big bang
Swift CBPR+, cross-border	Coexistence opened March 2023, closed 22 November 2025	Coexistence
CHAPS, Bank of England	June 2023, enhanced data mandated in phases from May 2025	Like-for-like, then enrichment
Lynx, Payments Canada	Introduced March 2023, MT retired 22 November 2025	Coexistence
CHIPS, The Clearing House	8 April 2024	Big bang
Fedwire Funds Service	14 July 2025	Big bang
Bacs	Not migrated; Standard 18 remains	Translation guide published

The pattern is not accidental. A domestic real-time gross settlement system has a closed participant list, a single operator, a mandate and a testing regime, so it can cut over in a weekend. T2 settled around 400,000 transactions on its first day, in line with the pre-migration average for a system that had been handling roughly €2.2 trillion a day. CHIPS processed 555,345 payments worth $1.81 trillion on its first day, 8 April 2024, figures its operator described as in line with a typical day. Fedwire, which averages more than $4.7 trillion a day, went in a single cutover on 14 July 2025. An open correspondent network with eleven thousand institutions cannot do that, which is why Swift ran two and a half years of coexistence and still needed contingency arrangements at the end.

Adoption figures track the deadline rather than the benefit. In June 2025 Swift reported that over 40% of daily traffic was in ISO 20022, more than 1.8 million payment messages a day across 180-plus sending and 220-plus receiving countries, and over 75% of its payments market infrastructure traffic. By September 2025 average daily ISO 20022 payment instructions had passed 60%; by October 2025 thirty-two RTGS market infrastructures had adopted the standard, representing 84.6% of Swift's global PMI traffic volume, and Lynx reported that over 98% of its messages were MX a month before the deadline. Read that curve honestly: the industry moved when the date got close, not when the business case was explained.

The forward calendar, again as of August 2026:

- **1 January 2026.** Swift's contingency conversion of MT messages sent by institutions that had not migrated, and its in-flow translation of inbound ISO 20022 into MT for receivers, both became chargeable.
- **November 2026, Standards Release 2026.** The unstructured postal address option is removed and rejected when used. MT 101 (single) moves to `pain.001` with contingency conversion available; MT 101 (multiple) reaches end of life. Receipt of `camt.110` via Case Management becomes mandatory.
- **November 2027.** Mandatory send and receive of `camt.110` and `camt.111` for exceptions and investigations, replacing in-flow translation. Payment cancellation moves exclusively to Case Management over FINplus. MT 199 and MT 299 for exceptions, investigations and payment initiation migrate. CHAPS extends purpose codes to all payments and LEIs to `pacs.004` and `pacs.009` COV.
- **November 2028.** Target for direct debits, charges and statement messages, with enforcement dependent on community adoption. Cheque messages continue indefinitely.

Bacs, and what a 100-character record cannot hold

The United Kingdom's bulk clearing has not migrated and is a useful antidote to the idea that ISO 20022 is universal. Bacs Standard 18 remains a fixed-length record:

Field	Position	Length	Content
Destination sorting code	1 to 6	6	Numeric
Destination account number	7 to 14	8	Numeric
Destination account type	15	1	Zero filled
Transaction code	16 to 17	2	`99`, `Z4`, `Z5` credits; `01`, `17`, `18`, `19` debits; `0N`, `0C`, `0S` AUDDIS
Originating sorting code	18 to 23	6	Numeric
Originating account number	24 to 31	8	Numeric
Free format	32 to 35	4	Blank, or `/` plus three characters
Amount in pence	36 to 46	11	Right justified, zero filled
Service user's name	47 to 64	18	Left justified
Service user's reference	65 to 82	18	Left justified
Destination account name	83 to 100	18	Left justified
Bacs processing day	101 to 106	6	`bYYDDD`, multi-day files only

One hundred characters, or one hundred and six for a multi-processing-day file. Bacs publishes an ISO 20022 to Standard 18 translation guide mapping `pain.001` to Direct Credits, `pain.008` to Direct Debits, `pacs.008` to cleared credits, `pacs.003` to cleared debits, `pacs.004` to returns, `pacs.007` to reversals, `pain.009` to AUDDIS, `pain.010` and `pain.011` to ADDACS, and `acmt.022` to AWACS, with dates converted from `YYYY-MM-DD` to the Bacs Julian form `bYYDDD` and lowercase folded to uppercase.

Look at the arithmetic and the limits of translation become obvious. The service user's reference is eighteen characters. A UETR is thirty-six. An `EndToEndId` is up to thirty-five. A `RemittanceInforma-`

`tion` block with three structured invoices is several hundred bytes. None of them fits, and no amount of implementation quality changes that. Translating a `pain.001` into Standard 18 is lossy by construction, and anyone building a translation layer should be able to enumerate, in writing, exactly which elements are dropped and hand that list to whoever signs off the migration.

Why a migration of this kind takes a decade

It is worth setting out the reasons as engineering rather than as complaint, because they generalise to every large format migration.

The value is bilateral but the cost is unilateral. Structured remittance data only pays when the sender populates it *and* the receiver consumes it. The first mover pays full cost for zero benefit. In that situation nothing happens until somebody imposes a date, which is why every milestone in the table above is a regulatory or operator mandate rather than a market outcome.

The message is the smallest part of the change. Behind a `pacs.008` sit sanctions screening, fraud scoring, liquidity and intraday position management, statement generation, reconciliation, archive, regulatory reporting and dispute handling — most of it written against thirty-five-character references and four-line addresses, some of it in COBOL, and much of it maintained by people who have retired. Changing the format at the edge is a quarter's work. Changing what the institution can *do* with the data is a decade's.

You cannot populate a field with data you have never collected. The address mandate slipped twice for this reason and no other.

Coexistence is not a bridge, it is a tax. For the duration, every institution runs two message stacks, two test suites, two sets of operational runbooks and two support rotas, and reconciles between them. That cost is why coexistence must have an end date, and why the end date must eventually stop moving. Swift's decision to make contingency conversion and in-flow translation chargeable from 1 January 2026 is the same lever, applied with a price rather than a prohibition.

The target moves during the project. Standards Releases are annual. A programme that began against `pacs.008.001.08` and runs for four years will finish inside a repository that has moved several versions on, which is why market infrastructures pin versions and why the pinned version becomes the real standard.

The forcing function is legal, not technical. The reason structured party data is mandatory is that FATF Recommendation 16 and its national implementations require identifiable originator and beneficiary information to travel with a transfer. Format follows obligation.

Coexistence, translation, and what to build

The mechanics of coexistence deserve stating precisely, because "we translate at the boundary" hides an important design decision. Swift's approach centred on Transaction Manager, which holds a central copy of the transaction so that the rich data survives even when an individual leg is delivered in a reduced format, plus in-flow translation from ISO 20022 to MT for receivers who had not migrated and, after November 2025, contingency conversion in the other direction for senders who had not. The important property is that the central copy, not the delivered message, is the record.

That generalises into the single rule worth taking away from this chapter. **Keep the ISO 20022 message as the system of record and translate only at the last mile, never in the middle.** A translated message can be revalidated; it cannot be reconstituted. Once a fifteen-element postal address has been flattened into four lines of thirty-five characters, the information required to rebuild it does not exist anywhere

in the message, and an institution that stores the flattened version has destroyed the very thing the migration was for.

Four other things follow, for anyone building against this.

Model, do not map. Store the message structure natively — nested parties, typed amounts, repeating remittance — and derive legacy formats on the way out. Every implementation that flattens ISO 20022 into an existing internal record on receipt has to be rebuilt later at greater cost, and the rebuild always arrives attached to a regulatory deadline.

Never treat the XSD as the specification. The schema permits far more than any network will accept. CBPR+ fixes `NbOfTxs` to 1; the schema does not. The schema allows seven address lines; CBPR+ allows three and Fedwire's hybrid profile allows two. The usage guideline published on MyStandards is the contract, and the schema is only the outer bound of it. A message that passes `xmllint` and fails the network is the normal case, not an anomaly.

Refresh the external code sets on a schedule. Quarterly, at the end of February, May, August and November. If a purpose code or a status reason code is an enumeration in your source tree, you have a defect with a known expiry date.

And test the round trip. Take a fully populated `pacs.008` with three structured invoices and a fully structured address, translate it to MT, translate it back, and diff the two. The diff is the true measure of what your coexistence arrangement costs, expressed in the only units that matter: the fields that arrive empty at the other end. Print it. Show it to whoever is signing the migration off. It is a more useful document than any readiness report, and it is the one nobody produces.

32.98 Common wrong ideas

Wrong: ISO 20022 is a message format that replaces ISO 8583. **Right:** It is a modelling standard with a repository and a registry behind it, and what it replaced was SWIFT MT for cross-border payment instructions and reporting, not card authorisation traffic.

Wrong: Card messages are about to move across too. **Right:** The `caaa`, `cain`, `catp` and `casp` business areas exist and have made almost no impression on global acquiring volume; the `0100` will outlive everyone reading this.

Wrong: ISO 20022 means XML. **Right:** Part 4 generates XML Schema, Part 8 generates ASN.1, a JSON Schema generation recommendation was approved in June 2025 and Part 9 generalises syntax generation altogether, and nexo card messages are routinely carried as ASN.1.

Wrong: Migrating gives you longer remittance fields. **Right:** The cross-border market practice permits one occurrence of the unstructured field at one hundred and forty characters, exactly the size of the field it replaced, and every real gain comes from the structured block, which needs four consenting parties to populate, carry, pass and present it.

Wrong: A message that validates against the schema will be accepted by the network. **Right:** The schema is only the outer bound; the market practice fixes the number of transactions at one and narrows the permitted address lines, and the usage guideline is the contract.

Wrong: The schema defines the permitted code values. **Right:** Purpose, category purpose, service level and local instrument codes are declared as plain strings whose values live in the External Code Sets, republished quarterly, so a code list compiled into source is a defect with a known expiry date.

Wrong: Translating at the boundary is a safe way to coexist. **Right:** Translation preserves delivery and not meaning, it is not reversible, and an institution that keeps the flattened message as its record has destroyed exactly the originator and beneficiary information the standard existed to carry.

Wrong: A settlement amount can be parsed into a floating-point type like any other number. **Right:** It is a decimal with up to eighteen total digits and five fractional digits, and parsing it as binary floating point produces a one-penny reconciliation break six months later.

Wrong: Five fractional digits mean the schema will catch an impossible currency amount. **Right:** A value with five decimal places is a valid decimal and an invalid euro amount, because the constraint that the value must respect the currency's minor units lives in the usage guideline and not in the schema.

Wrong: The industry migrated because the business case was compelling. **Right:** Adoption tracked the deadline rather than the benefit, because the value of structured data is bilateral while the cost of producing it is unilateral, so every milestone was a regulatory or operator mandate.

32.99 Chapter summary in 20 lines

1. ISO 20022 is not a format but a method for producing formats, plus a dictionary of business concepts and a committee structure for deciding what goes into the dictionary.
2. What retired on 22 November 2025 was SWIFT MT for cross-border payment instructions and reporting, not ISO 8583, and the card authorisation message is untouched.
3. The old numbered form is small and fast precisely because nothing in it says what anything is, which works until the day a message must carry something the code book never anticipated.
4. The new form labels every value, which makes it fatter, makes it self-describing, and makes the same concept mean the same thing in every kind of message the standard produces.
5. That consistency is the actual product: every form is built out of a central dictionary rather than invented from scratch.
6. Translated into money, three invoices that once travelled as a one-hundred-and-forty-character note now travel as three structured entries that a ledger can match and close without a human.
7. The architecture has three layers — a business model in UML, a logical message assembled from the Data Dictionary, and a physical syntax generated mechanically — and confusing them is the source of most arguments about the standard.
8. Governance is a registry rather than a standards body in the usual sense, with a Registration Authority maintaining the repository, a Registration Management Group overseeing it and Standards Evaluation Groups validating new definitions.
9. A message identifier carries business area, functionality, variant and version, and because the version is embedded in the namespace, upgrading is a release rather than a configuration change.
10. Market infrastructures therefore pin versions and hold them for years, and the pinned version becomes the real standard.
11. Set beside ISO 8583 the differences become concrete: a six-digit trace number that wraps within the day becomes a UUID unique end to end, and a two-character outcome becomes a status plus a separate mandatory reason.
12. Structured addresses are the least glamorous part of the migration and by a distance the most expensive, because they turn sanctions screening from a text-similarity problem into a lookup.
13. That deadline slipped for a reason that has nothing to do with software: a bank cannot populate a town name for a customer whose stored address is four lines of free text typed into a branch terminal in 1998.

14. Richer data is a capacity rather than a delivery, and the Bank of England's decision in July 2026 to abandon its structured remittance mandate for CHAPS is the clearest available evidence of how few parties were ready.
15. Many code sets live outside the schema and are republished quarterly, so schema validation does not validate codes and compiled enumerations rot within three months.
16. Purpose codes show what the structure actually buys: four characters declaring that a movement is a property completion rather than an unexplained transfer are worth more to a financial crime team than any amount of free text.
17. Domestic real-time gross settlement systems cut over in a single weekend because they have a closed participant list, one operator and a mandate; an open correspondent network of eleven thousand institutions cannot.
18. Adoption curves therefore track deadlines and not benefits, because the value is bilateral while the cost is unilateral and nobody can afford to go first.
19. Coexistence is not a bridge but a tax — two message stacks, two test suites, two runbooks, two support rotas — which is why it must have an end date and why the end date eventually had a price attached to it.
20. Keep the ISO 20022 message as the system of record, translate only at the last mile, never treat the schema as the specification, and test the round trip, because the fields that arrive empty at the other end are the only honest measure of what the arrangement costs.

Sources: ISO 20022 Registration Authority — ISO 20022 standard parts 1 to 8 (2013) and ISO 20022-9:2026; message definition catalogue; business area code list (RA, June 2025); external code sets publication policy; RMG governance pages; ISO 20022 Technical Support Group, Generation of JSON Schema Draft 2020-12 for ISO 20022 (v2.0, approved 10 June 2025). Message schemas: pacs.008.001.09 and pacs.008.001.07 XML Schemas as generated by the ISO 20022 Standards Editor. Swift: ISO 20022 implementation FAQs; ISO 20022 for financial institutions; ISO 20022 in bytes, "One month to go"; "Momentum builds as industry advances ISO 20022 adoption" (23 June 2025); CBPR+ roadmap beyond November 2025 (v5, June 2026); Swift AI address structuring model FAQ; CBPR+ pacs.008.001.08 usage guideline. Bank of England: Mandating ISO 20022 enhanced data in CHAPS (policy statement, 2024) and Expanding mandatory ISO 20022 enhanced data in CHAPS from 2027 (policy statement, 2025, including the July 2026 structured remittance decision); UK recommended purpose code list. European Central Bank: "Successful launch of new T2 wholesale payment system" (21 March 2023). EBA Clearing: EURO1 ISO 20022 migration. The Clearing House: CHIPS network migrates to ISO 20022 (10 April 2024). Federal Reserve Financial Services: Fedwire Funds Service ISO 20022 migration announcement (15 July 2025), ISO 20022 format FAQs, and Fedwire Funds Service ISO 20022 Quick Reference Guide. Payments Canada: Lynx ISO 20022 adoption and RTR pacs.002.001.10 usage guideline. European Payments Council: SEPA Credit Transfer Inter-PSP Implementation Guidelines 2025 v1.0. Bacs: ISO 20022 to Standard 18 Translation Guide and the Standard 18 message implementation guide. Swift Standards MT November 2021 Message Reference Guide for MT 103 field formats.

33

CHAPTER

The Physical Network

33.0 What this chapter gives you

1. You will be able to name the five roles a card authorisation passes through, and say which two of them are scheme members and therefore the only parties that can be fined or hold a settlement position.
2. You will be able to account for the 790 milliseconds of a contactless payment segment by segment, and explain why the card network is about five per cent of it while the phone and the reader are almost two thirds.
3. You will be able to separate the parts of a latency budget that are physics from the parts that are software, and say why an issuer processor in Virginia costs a London transaction about seventy-five milliseconds that no amount of money buys back.
4. You will be able to explain why a payment HSM is a computer that refuses rather than a safe that hides, and why zeroising one makes every key the institution holds permanently unreadable.
5. You will be able to trace a PIN block through translation at the acquirer, the scheme and the issuer, and say how many times the clear PIN exists and where.
6. You will be able to explain why a timeout is a financial event rather than a performance event, and what the 0400 reversal is for.
7. You will be able to say why partial failure is more dangerous than total failure, using Visa Europe's ten hours and ten minutes on Friday 1 June 2018 as the worked example.
8. You will be able to answer a due diligence question about HSM certification by naming the validated module and its certificate rather than the appliance.
9. You will be able to state the three phases of PCI PIN Security Requirement 18-3 and their dates, and say whether an estate still shipping bare encrypted keys to terminals is compliant.
10. You will be able to explain why moving scheme endpoints into the public cloud improves one firm's resilience while concentrating everybody's risk, and name the regime that now oversees that concentration.

At twenty to nine on a Tuesday morning a woman buys a sausage roll and a coffee in a bakery on Kirkstall Road in Leeds. Four pounds twenty. She holds her phone against the reader, the reader beeps, a green tick appears, and she is out of the door before the receipt has finished printing. The whole thing takes about a second.

Everything in this volume so far has described what was in that second as *messages*. A 0100 authorisation request with a bitmap and a set of data elements; a 0110 response carrying an approval in DE39 and a six-character code in DE38. That description is true and it is the one you need in order to reason about the system. It is also, in a specific sense, weightless. Messages do not travel. Optical and

electrical signals travel, along glass, through switches, into racks in buildings with concrete walls and diesel tanks.

This chapter is about the buildings. It is about which company owns which machine, what happens inside the box that no engineer is permitted to open, where the second actually goes, and what a merchant sees when the whole apparatus stops working on a Friday afternoon.

The plain version

A chain of desks

Imagine that the bakery cannot approve the payment itself, so it has to ask. The question travels down a chain of desks, and each desk does one job and hands it on.

At the first desk is the card reader on the counter. Its job is to read the phone, work out that the shop wants £4.20, write all of that down on a slip of paper in a very compressed shorthand, and hand it on.

The second desk belongs to the company the bakery pays to handle its card money: a building full of computers somewhere in England. It checks that the slip is properly filled in, works out which card network the payment belongs to, translates the shorthand into the shorthand *that network* prefers, and passes it on.

The third desk is the card network itself. It has no opinion about whether this payment is a good idea. Its job is to look at the front of the card number, work out which bank in the world issued it, and route the slip there. It is a sorting office, not a decision maker.

The fourth desk belongs to the bank's card system, and this is where the decision happens. Is this really her phone? Does the cryptographic signature the phone produced check out? Has she got £4.20 available? Has anything about this purchase looked strange in the last five minutes?

The fifth desk is the bank's actual accounts system: the ledger with her name on it and a balance next to it. Then the answer comes back down the chain in reverse, and the reader beeps.

The stopwatch

Now put real numbers on it, because the numbers are the point. Every figure below is in thousandths of a second — milliseconds.

Step	Time
Phone held against the reader, chip does its cryptographic sum	500
Reader to the payments company's building	25
Payments company checks and translates the slip	10
Payments company to the card network	5
Card network sorts and routes	20
Card network to the bank's card system	10
Bank decides	120
Bank's card system back to the card network	10
Card network back	20
Card network to the payments company	5
Payments company to the reader	25
Reader shows the tick and beeps	40
Total	**790**

Just under eight tenths of a second, and look at how it is spent. Almost two thirds of it is the phone and the reader talking to each other before anything leaves the shop at all. The card network — the thing everybody pictures when they imagine "the payment system" — accounts for forty of those eight hundred milliseconds, or five per cent. The travelling itself, all six legs of it, is seventy milliseconds. The single biggest slice after the phone is the bank thinking. So when a shopkeeper says the card machine is slow, the card network is almost never what they are complaining about.

Why there is spare time in the budget

People in the industry talk about a payment needing to complete in about two seconds. Eight hundred milliseconds is well inside that. The remaining twelve hundred are not waste; they are margin, and the margin exists because of specific, nameable things that go wrong. The bank's computers might be busy. The shop's broadband might be having a bad minute and a packet might have to be sent twice. The bank might not be in Britain.

That last one matters more than you would think, and it is the one thing in the entire chain that money cannot fix. Light in a glass fibre travels at about two hundred thousand kilometres per second, slower than light in a vacuum because glass slows it down. Leeds to London and back is roughly three milliseconds of pure travel. London to Virginia and back, which is where a great many card systems physically live, is about seventy-five. If the bank's computers are in Virginia, seventy-five milliseconds of your budget is gone before anybody has done any work, and no amount of money buys it back, because you cannot buy a faster universe.

The locked box

There is one desk in the chain where the person does not do the work themselves.

When the bank checks the customer's PIN, or checks the cryptographic signature the chip produced, it needs secret keys. Those keys are the crown jewels: anyone holding them could forge cards for millions of customers. So the bank does not keep them in an ordinary computer. It keeps them in a metal box bolted into a rack in a locked cage inside a locked room.

The box works like this. You post a question through a slot: "here is an encrypted PIN and here is the customer's account number — is the PIN right?" The box does the sum inside itself and posts out a single word: yes or no. What it will never do, no matter how you ask, is post out the PIN. There is no command for that. The engineers who built it deliberately left it out.

And if anybody tries to open the box, drill it, freeze it, or lift it off the rack, it does not resist. It erases itself, and what the thief carries away is an expensive paperweight. That is the opposite of how people imagine security. The box is not strong. The box is *suicidal*.

Two of everything

Because all of this is a chain, and because a chain stops if any link stops, everybody in it builds two of everything. Two card readers behind the counter. Two internet connections into the payments company. Two data centres, in two towns, each big enough to carry the whole load on its own.

And here is the thing a twelve-year-old works out faster than an adult: the frightening failure is not one building burning down. If a building burns down, everyone knows, and the other building takes over. The frightening failure is one building going *half* wrong. Still answering the phone. Still saying it is fine. Still sending confused messages to the other building. Nobody is sure whether to switch over, because switching over is itself dangerous, and while everybody argues, the queue at the bakery gets longer.

That is not hypothetical. It is what happened to Visa in Britain on Friday 1 June 2018, and we will walk through it.

Where the plain version stops being true

The first correction is that the five desks are five jobs, not five companies, and often not five buildings. The chain above is a chain of *roles*, and roles do not map one-to-one onto companies. Adyen and Stripe are the gateway, the licensed acquirer and the acquiring processor in one legal and technical stack, so three of my five hops become function calls between processes in the same estate. Conversely, one apparent hop routinely conceals six: what I drew as "the payments company checks the slip" may be a gateway, a tokenisation vault, a risk engine, a switch, a scheme endpoint and a message-repository write, each with a separate failure mode. Do not count boxes. Count trust boundaries — the places where data crosses from one organisation's liability into another's — and count clocks, because every boundary has a timer attached to it.

The second correction is that "two seconds" is a convention, not a specification. No card scheme publishes a rule saying an authorisation must complete within two seconds. What exists instead is a nest of separate timers with separate owners expiring at different moments: the terminal's timer waiting on its acquirer, the acquirer's on the scheme, the scheme's on the issuer, and the entirely unregulated timer running behind the customer's eyes. These disagree on purpose, and the consequences of each expiring differ. When the innermost one expires you do not get a slow approval; you get a *different transaction* — a reversal, a stand-in decision made by somebody who is not your bank, or a decline. Timeouts in payments are financial events, not performance events.

The third correction is that the locked box is a computer that refuses, not a safe that hides. The security of a hardware security module is not principally about concealment; it is about a deliberately impoverished instruction set. The device will take an encrypted PIN block and re-encrypt it under a different key, because that operation is necessary; it will not decrypt one and hand you the result, because that command was never implemented. Related, and routinely misunderstood: most keys are not stored *inside* the box. A large issuer's millions of card keys live in an ordinary database, each encrypted under a master key, and only that master key lives inside the tamper boundary. The box is not a vault holding your keys; it is the only place where your keys can be turned back into usable form, and only during an operation whose output you are permitted to see. Erasing it does not destroy your keys. It makes every copy of them, everywhere, permanently unreadable, which is worse.

The fourth correction is that "we have two data centres" is a claim about capacity, and the thing that fails is the switching. Building a second site that can carry the full load is expensive but conceptually simple. Detecting that the first site is unwell, distinguishing "unwell" from "briefly busy", and committing to a cutover that will itself cause disruption, is neither. The industry's own six-nines language conceals this. Availability of 99.9999 per cent sounds like a promise; it is 31.6 seconds of downtime in a year. The Visa Europe outage described later ran for ten hours and ten minutes: roughly one thousand one hundred and fifty years of a six-nines budget consumed in one afternoon. Such a number states design intent about steady-state operation. It says nothing about the tail, and the tail is what puts queues in bakeries.

The technical version

The five roles, stated exactly

The card industry's role names are used loosely in conversation and precisely in contracts. Here they are precisely.

Role	What it is	Licensed by a scheme?	Physical artefact
Terminal estate	POI devices on merchant counters, unattended units, softPOS on phones	No, but devices are PCI PTS approved	The PIN entry device
Payment gateway	Accepts the merchant's transaction, formats it for an acquirer	No	Servers, usually cloud
Acquirer	Holds the merchant agreement, is the scheme member, bears settlement risk	Yes, principal member	A licence and a balance sheet
Acquiring processor	Runs the switch that speaks to the scheme	No, acts for the acquirer	Data centres, switch, HSM farm
Scheme network	Routes, applies scheme rules, calculates interchange, clears and settles	It *is* the scheme	Global data centres
Issuer processor	Authorises on the issuer's behalf: cryptogram, PIN, risk, limits	No, acts for the issuer	Data centres, HSM farm
Issuer	Holds the cardholder account, owns the credit decision	Yes, principal member	A licence and a balance sheet
Core banking	The ledger of record for the customer's account	No	Mainframe or modern equivalent

Two of these are legal facts and the rest are engineering facts. Only the acquirer and the issuer are scheme members; only they can be fined, sanctioned or expelled, and only they can hold settlement positions. Everyone else in the chain, however large, is somebody's agent. This matters when things break, because the party a merchant shouts at is usually the processor and the party that owes the merchant money is always the acquirer. In Britain the acquiring names are Worldpay, Barclaycard, Global Payments, Elavon, Adyen and Stripe; the issuing-processor names include Fiserv, FIS, Thredd, Marqeta and Enfuce; the core banking names include Temenos, Finastra, Oracle, TCS BaNCS, Infosys Finacle, Thought Machine and Mambu. Ownership of these brands changes every few years. Their positions in the chain do not.

Inside an acquiring processor

An acquiring processor's estate is more than a switch, and a due diligence questionnaire will ask about each part of it by name.

A terminal management system, which knows every deployed device by serial number, pushes configuration and application updates, and injects or rotates terminal keys. An estate of tens of thousands of terminals is an estate of tens of thousands of key holders.

The switch itself: the process that receives an inbound transaction, validates it, applies merchant-level rules, routes it to the right scheme or domestic network, converts between message dialects, and applies the acquirer's own timers. Chapter 24 established that ISO 8583 is a family rather than a format, and the switch is where that family becomes a problem: the acquirer's internal dialect, Visa's, Mastercard's and a domestic scheme's are all different, and the switch owns the translation table.

A store-and-forward facility for transactions the terminal completed offline, which must be delivered later without being duplicated. A capture and clearing subsystem, which accumulates approved authorisations into the day's batch and generates the clearing files described in Chapter 30 — historically Visa BASE II and Mastercard IPM formats. An HSM farm, discussed below.

And finally the endpoint: the connection to each scheme. This is not a single wire. A processor typically holds multiple endpoints per scheme across multiple sites, each with its own institution identifier, its own key set and its own independent sequence of system trace audit numbers. DE11 is only unique within an endpoint within a day, which is precisely why Chapter 28 had to introduce four different identifiers rather than one.

The scheme switch

Visa's authorisation system began as BASE I, built by Bank of America and introduced in 1973 as the first real-time electronic card authorisation system; the batch clearing and settlement counterpart was BASE II. Both were later subsumed into V.I.P., the VisaNet Integrated Payment system. Mastercard's equivalents were INAS for authorisation and INET for clearing, carried over Banknet, and the modern architecture retains the split between a dual-message system, where authorisation and clearing are separate messages, and a single-message system, where one message does both. That split determines whether a refund can be a reversal or must be a fresh credit, and it is why debit and credit behave differently on the same terminal.

The routing decision itself is unglamorous. The scheme maintains an account range table mapping leading digits of the PAN to an issuer endpoint, the switch looks up the range, and the message goes there. Everything interesting is bolted around that lookup: inline fraud scoring, tokenisation lookups, currency conversion, and stand-in processing when the issuer does not answer.

The scale is worth stating with sources, because it is routinely exaggerated. Visa's stated figures as of October 2025 are 322 billion transactions a year, more than $16 trillion of payments volume, more than 150 million merchant locations, 4.8 billion credentials, seven independent data centres, availability of 99.9999 per cent, and a network processing up to 83,000 transaction messages per second worldwide. Its network operations centre stress-test capacity exceeds 65,000 messages per second, the figure Visa's public fact sheets have carried since 2017.

Divide 322 billion by the number of seconds in a year and the average is about 10,200 transactions per second. Set that against a peak of 83,000 messages per second and you have the number that actually governs the engineering: a peak-to-mean ratio of roughly eight to one. Card networks are not sized for the average. They are sized for the Friday before Christmas, and they spend most of the year at an eighth of capacity, which is why redundancy in this industry is cheaper than it is elsewhere.

Visa's Ashburn, Virginia site, which the company describes publicly, occupies 44 acres, is entered by only about 75 cleared employees, has 18-inch concrete walls rated to 170 mph winds, holds 500 petabytes of stored data, and is backed by 28 megawatts of standby power with 100,000 gallons of fuel for five days of autonomy. Those numbers are the answer to "what does a payment network physically look like".

The issuer side, and why authorisation does not touch the ledger

The issuer processor performs, in order: message validation; cryptogram verification, meaning it recomputes the ARQC the chip produced in DE55 and compares; cardholder verification, meaning online PIN verification if a PIN block is present; risk and fraud scoring; and an availability check.

That last step is where most people's mental model is wrong. The authorisation does not debit the customer's account. It writes to an authorisation record — variously the shadow balance, the open-to-buy or the memo post — that reduces available funds without altering the ledger balance. The ledger moves later, when the clearing record arrives, which is the mechanism behind every "pending transaction" a customer has ever queried and behind the whole of Chapter 30.

Behind the issuer processor sits core banking, which in a large bank is very often exactly what its reputation suggests: COBOL on IBM z/OS, batch-oriented, with a nightly cycle during which the ledger is closed for updating. The newer platforms are built around continuous processing precisely to remove that window. Whether the window exists is the single biggest determinant of what a bank can promise about timing, and it is a property of the ledger, not of the card system in front of it.

Why key operations happen inside tamper-responsive hardware

The threat being defended against is specific. ISO 9564 requires that a PIN never exist in cleartext outside a secure cryptographic device, because a PIN is four digits: an attacker who obtains encrypted PIN blocks and any oracle that will decrypt them can brute-force the space trivially. Equally, an issuer's card master keys can generate valid cryptograms for every card in a portfolio. Neither secret can be permitted to exist in the addressable memory of a general-purpose computer, however well patched, because a general-purpose computer's whole design goal is that its memory is addressable.

Three terms are used loosely and mean different things. **Tamper-evident** means an attack leaves visible traces. **Tamper-resistant** means an attack is made difficult by physical construction. **Tamper-responsive** means the device actively detects an attack in progress and reacts, and in a payment HSM the reaction is zeroisation: erasure of the local master key, after which every key the institution holds becomes undecryptable ciphertext. The sensors typically cover case opening, drilling, temperature excursions in both directions, voltage manipulation and radiation.

The market is small: Thales's payShield line and Utimaco's Atalla and CryptoSec families cover most of it, alongside a handful of others. Take the current Thales device as a concrete specimen, since generalities are useless here. The payShield 10K, in its PS10-S, PS10-D and PS10-F variants, appears on the PCI Security Standards Council's approved devices list under approval number 4-40266, evaluated against version 3.x of the PCI HSM Security Requirements, with an approval expiry of 30 April 2028. Its listed approved usage is *Restricted*, meaning the approval holds only when the device is deployed in an environment meeting at least the security requirements of a Controlled Environment as defined in the PCI key management requirements and the device's own security policy. It supports ISO Format 4 (AES) PIN blocks and is listed as supporting Remote Administration.

Separately, and this is the distinction practitioners get wrong, its FIPS validation is not of the whole appliance. NIST certificate #3610 covers the Thales Advanced Security Platform, a multi-chip embedded module *inside* the payShield 10000 family, validated to FIPS 140-2 overall Level 3 by UL Verification Services in January 2020 and now moved to the historical list as part of the general sunsetting of FIPS 140-2. If a supplier questionnaire asks you whether your HSM is "FIPS 140-2 Level 3 certified", the honest answer names the module and the certificate, not the box.

Published specifications for the same device: Triple DES at 112 and 168 bits, AES at 128, 192 and 256, RSA up to 4096 bits, and the FIPS 186-3 curves P-256, P-384 and P-521; host connectivity over

TCP/IP and UDP on dual 1 Gbps or 10 Gbps Ethernet ports, with an optional single FICON port for mainframe attachment; and conformance claims against ISO 9564, 10118, 11568, 13491 and 16609, ANSI X3.92, X9.8, X9.9 and X9.17, ASC X9 TR-31 and X9.143, TR-34 and X9.139, and APACS 40 and 70. The presence of FICON on a 2020s product tells you what these devices are still plugged into.

The key hierarchy, named

Key	Full name	Where it lives	What it protects
LMK	Local master key	Inside the HSM tamper boundary only	Every other key the institution holds
ZMK / KEK	Zone master key, key encrypting key	Exchanged between two institutions	The transport of working keys
ZPK	Zone PIN key	Under LMK at rest, under ZMK in transit	PIN blocks between two parties
BDK	Base derivation key	Under LMK; never leaves the HSM	Derives per-terminal, per-transaction keys
PVK	PIN verification key	Under LMK	PIN verification values
CVK	Card verification key	Under LMK	CVV, CVC, iCVV, dCVV values
IMK	Issuer master key	Under LMK	Derives per-card keys for EMV
UDK	Unique DEA key	Derived per card	Generates and verifies the ARQC

Only the LMK is physically inside the device. Everything else is a database column holding ciphertext, meaningful only when handed back through the HSM's slot. That is what makes the architecture affordable, and what makes zeroisation catastrophic.

DUKPT, Derived Unique Key Per Transaction, standardised in ANSI X9.24, solves a specific problem: a terminal on a counter is physically accessible to the attacker. So the terminal is never given a long-lived key. It is injected with an initial key derived from the acquirer's base derivation key and its own device identifier, and derives a fresh key for every transaction, discarding the previous one irreversibly. Compromising a terminal therefore yields the transactions after the compromise and none before it, and nothing about any other terminal.

Key blocks, and a compliance date you must know

For decades, keys were shared between institutions as bare encrypted values with the key's *purpose* recorded separately, in documentation, by convention. This is an obvious flaw: an attacker who can persuade a system to use a PIN key as if it were a data key can extract secrets the design never intended to expose. Key blocks fix this by cryptographically binding a key's permitted usage to the key itself, so that the binding cannot be altered without invalidating the block. The relevant standards are ASC X9 TR-31 and its successor ANSI X9.143.

PCI PIN Security Requirement 18-3 mandates key blocks in three phases, and the dates are exact. Phase 1, internal connections and key storage within service provider environments including all applications and databases connected to an HSM, took effect on 1 June 2019. Phase 2, external connections to associations and networks, took effect on 1 January 2023. Phase 3, extending key blocks to all merchant hosts, point-of-sale devices and ATMs, took effect on 1 January 2025. Anyone still transporting bare encrypted keys to a terminal estate as of this writing is out of compliance and has been for over a year and a half.

PIN translation, hop by hop

This is the single most instructive walk in the chapter, because it shows the tamper boundary doing real work.

A customer enters a PIN on a PIN entry device. The device constructs a PIN block — a fixed-length structure combining the PIN with part of the PAN so that identical PINs on different cards do not produce identical blocks — and encrypts it under a key derived by DUKPT for that transaction only. The clear PIN exists in the PED's secure processor for a few milliseconds and is then gone.

The encrypted block travels to the acquiring processor. Its HSM is given the block, the key serial number and the target key, and performs a *translation*: internally it derives the terminal's transaction key, decrypts the block, re-encrypts it under the zone PIN key shared with the scheme, and outputs the new block. The clear PIN existed for microseconds, inside the tamper boundary, and never entered the memory of the switch. At the scheme the same operation happens again: decrypt under the acquirer's ZPK, re-encrypt under the issuer's ZPK. At the issuer processor the block is finally verified rather than translated, decrypted inside the HSM and checked against a PIN verification value computed with the PVK, or against an offset, with only a yes or no coming back out.

The PIN therefore crosses three organisations and exists in the clear four times, each time for microseconds, each time inside a tamper-responsive device that has no command capable of revealing it.

The block formats are specified in ISO 9564-1. Formats 0 to 3 are built for Triple DES's 64-bit block. Format 4, standardised in 2015, is built for AES's 128-bit block and, unlike its predecessors, includes randomness so that the same PIN, PAN and key produce a different block every time. PCI's device requirements now force the migration: POI devices at version 5 and above supporting online PIN, and HSMs at version 4 and above supporting PIN processing, are required to support ISO Format 4.

The latency budget, restated with owners

The plain version's table was honest but anonymous. Here it is again with the two questions a performance engineer asks: who owns this slice, and is it physics or software?

Segment	Typical	Owner	Nature
Card or phone in the contactless field	up to ~500 ms	EMVCo kernel, card, reader	Protocol and silicon
Terminal to acquirer over broadband or 4G	20–40 ms	Merchant's network, acquirer edge	Mostly software and access network
Acquirer switch processing	5–20 ms	Acquiring processor	Software
Acquirer to scheme endpoint	2–10 ms	Private circuit or Edge connection	Physics plus equipment
Scheme routing and inline scoring	10–40 ms	Scheme	Software
Scheme to issuer processor	2–80 ms	Depends entirely on geography	Physics
Issuer decision, including HSM calls	50–250 ms	Issuer processor and core banking	Software and database
Return path	mirrors outbound	as above	as above
Terminal display and receipt	30–80 ms	Terminal	Software and printer

The contactless figure is the one hard number in the table: the EMV contactless specifications set the requirement that a card need not remain in the reader's field for more than about 500 milliseconds. Everything else is a range observed in production, not a published limit, and it moves with load.

Two conclusions follow. In a domestic transaction the network is a rounding error and the issuer's database is the story, so performance work belongs in the issuer processor rather than the wires. In a cross-border transaction the network is not a rounding error at all, and no amount of software engineering will help, because the constraint is geometry.

What physics charges you

Light in a vacuum travels 299,792,458 metres per second. In single-mode fibre, with a refractive index of about 1.468, it travels roughly 204,200 kilometres per second. The rule of thumb used by optical engineers is 4.9 microseconds per kilometre, and it is accurate enough for any budget you will ever build.

Route	Approximate fibre path	Round-trip floor	Typically observed
Leeds to London	~330 km	~3.2 ms	8–15 ms
London to Frankfurt	~750 km	~7.4 ms	12–20 ms
London to Ashburn, Virginia	~6,500 km	~64 ms	70–80 ms
London to Singapore	~11,000 km	~108 ms	160–200 ms

The gap between the floor and the observed figure is equipment: optical amplifiers, regenerators, routers, firewalls and the serialisation delay of putting bits onto a wire. It is bounded, and it is the part you can buy your way out of. The floor is not.

There is a second physics-adjacent cost engineers forget. A TLS 1.2 full handshake costs two round trips before any application data flows; TLS 1.3 reduces that to one. On a London-to-Virginia link that is 150 milliseconds of handshake for TLS 1.2 and 75 for TLS 1.3, on top of the TCP handshake. This is why payment links are never opened per transaction. They are long-lived, pooled, persistent sessions, kept warm and monitored, and a transaction that has to open a fresh connection has already lost more time than the entire issuer decision would have taken.

The heartbeat, and what it is for

Keeping a session warm requires knowing whether it is alive, which is the purpose of the 8xx message class introduced in Chapter 24. A 0800 network management request with the appropriate function code is an echo test; the peer answers 0810. The same class carries sign-on, sign-off and key change. Endpoints exchange echoes on a fixed interval, and after a configured number of consecutive failures the link is marked down and traffic redistributed to the surviving endpoints.

That is why partial failure is so much more dangerous than total failure. A dead link fails its echoes and is removed automatically in seconds. A link that answers every echo correctly while mishandling the transactions behind it will never be removed by any automatic process, because by the only test the system applies, it is healthy.

Timers, reversals and stand-in

When the issuer does not answer within the scheme's response time parameters, something must happen, and what happens is defined rather than improvised.

The scheme may substitute its own decision. Both major schemes operate stand-in processing, in which the network authorises on the issuer's behalf against limits the issuer has pre-registered. Visa's published action code 91 covers the case where stand-in is *not* available: "Issuer or switch inoperative and STIP not applicable or not available for this transaction; time-out when no stand-in". Issuer processors document a related behaviour in which Visa forces single-request stand-in and converts a 91 or 96 into an N0. Chapter 29 covered what these codes mean to a merchant; the point here is that they are emitted by network conditions rather than by any bank's opinion of the cardholder.

When the acquirer's own timer expires, it must assume the issuer may have approved and reserved funds it does not know about, and must therefore send a reversal — the 0400 reversal request of Mastercard's Transaction Processing Rules and its equivalents. The reversal is what stops a network hiccup from silently holding a customer's money for a week, and unwinding un-reversed timeouts is a large share of the reconciliation work described in Volume V.

Redundancy in practice

Serious payment estates use one of two patterns. **Active-active** runs both sites live, splitting traffic, so a site loss is a capacity loss rather than an outage and the failover has been exercised continuously. **Active-passive** runs one site live and one warm, which is cheaper and has the fatal property that the failover path is exercised only in the emergency.

Underneath sits the replication decision. **Synchronous** replication does not acknowledge a write until the second site has it, giving a recovery point objective of zero at the cost of adding the inter-site round trip to every write. That is why paired sites sit tens of kilometres apart rather than hundreds: at 4.9 microseconds per kilometre, a 50 km separation costs about half a millisecond per write and a 500 km separation costs five. **Asynchronous** replication removes that cost and accepts a non-zero recovery point objective. In payments that window is not an abstraction; it is a set of authorisations that happened, that customers were told about, and that the surviving site has never heard of.

The genuinely hard problem is neither of these. It is the decision. A cutover is disruptive in itself, so nobody wants to trigger one on a false positive; but the detection signal in a partial failure is ambiguous by definition. Firms therefore write runbooks with explicit trigger thresholds and rehearse them. Since 31 March 2025, when the transition period for the FCA's PS21/3 operational resilience regime ended, UK firms have been required to have identified their important business services, set impact tolerances for each, mapped the resources supporting them, and demonstrated by testing that they can stay within those tolerances in severe but plausible scenarios.

Friday 1 June 2018, narrated

The best public account of a scheme outage in Britain is Visa Europe's own, given to the House of Commons Treasury Committee by its chief executive Charlotte Hogg in a letter dated 15 June 2018. What follows is from that account and the contemporaneous reporting of it.

Visa Europe operated two data centres in the United Kingdom, either of which could independently handle 100 per cent of Visa's European transactions. In normal operation the systems were synchronised and either centre could take over from the other immediately.

At 14:35 on Friday 1 June 2018, a component within a switch in the primary data centre suffered what Hogg described as a very rare, partial failure. Because the failure was partial, the backup switch did not activate.

This is the exact scenario the seam warned about. The malfunctioning system at the primary site did not stop. It continued attempting to synchronise messages with the secondary site, building a backlog there which in turn degraded the secondary's own ability to process incoming transactions. The redundant capacity was not merely unused; it was being poisoned by the failing primary. Isolating the primary took nearly five hours, and full correct processing was not restored until 00:45 the following morning: ten hours and ten minutes end to end.

The numbers Visa gave Parliament are worth recording precisely. Across Europe, 51.2 million transactions were attempted during the affected period; about 10 per cent were affected, and 5.2 million failed to process. In the United Kingdom 2.4 million failed, affecting 1.7 million credit and debit cards; a further 2.8 million failed elsewhere in Europe. Disruption was not constant. There were two peaks — roughly ten minutes just after 15:00, and fifty minutes between 17:40 and 18:30 — during which 35 per cent of transactions failed. Outside those peaks the UK failure rate was closer to 7 per cent.

That last detail is the one merchants remember, because it describes what an outage actually feels like on a shop floor. Not "the card machines are down". Instead: one payment in fourteen fails for no visible reason, then briefly one in three, then one in fourteen again, with no announcement, no error message worth reading, and a customer at the front of the queue re-presenting the same card three times until it works. Staff cannot tell a scheme outage from a bad card, because on the counter they look identical. Visa engaged EY to conduct an independent review and said it would migrate European processing onto its global VisaNet platform, which it described as more resilient in detecting and recovering from partial malfunctions of exactly that type.

Three lessons generalise. Partial failure is worse than total failure. A hot standby coupled to a sick primary is a liability rather than an asset, because the coupling is the failure path. And the recovery clock is dominated not by repair but by diagnosis and by the decision to isolate.

From leased lines to TLS over the public internet

The connectivity story runs in four eras and the last one is still arriving.

Dedicated circuits, 1973 to the mid-1990s. BASE I ran over leased lines between member banks and Visa's centres. Terminals dialled: a countertop unit picked up a phone line, dialled a modem pool and took fifteen to thirty seconds, which is why floor limits and offline authorisation, covered in Volume II, were not an optimisation but a necessity. Behind the scenes ran X.25, and in bank environments IBM's SNA and SDLC.

Private packet networks, the 1990s and 2000s. Frame relay and then MPLS replaced point-to-point circuits with managed private clouds, and the scheme endpoint became a physical appliance installed in the customer's data centre — a Visa access point, or Mastercard's Interface Processor, the MIP. The scheme shipped you a box, and the box was the border.

IP with cryptographic protection, the 2010s. Terminals moved to ADSL, then Ethernet, then 4G, and the transport became the ordinary internet with TLS carrying the security a private circuit had previously carried by being private. PCI DSS Requirement 4 governs this directly: strong cryptography must safeguard the primary account number during transmission over open, public networks, with TLS 1.2 or higher the practical floor. Version 4 added Requirement 4.2.1, that certificates used to protect PAN in transit are confirmed valid and not expired or revoked, and 4.2.1.1, that an inventory of those

trusted keys and certificates is maintained. The second catches firms out, because certificates expire on a schedule nobody is monitoring until the morning they do.

Cloud and colocation endpoints, now. The physical box in the customer's data centre is being retired. Mastercard publishes three connectivity flavours side by side: Mastercard Edge for customers in their own on-premises data centres, Co-Lo Edge for those in third-party colocation facilities, and Cloud Edge for those in the public cloud, delivered in collaboration with cloud providers including AWS. A processor that runs no data centre of its own can now hold a scheme endpoint, which was not true a decade ago. Merchant-side, the same arc ends in softPOS: a commodity phone acting as the contactless reader, with no dedicated hardware between the card and the internet at all.

Concentration, and the critical third parties regime

Moving payment infrastructure to a small number of cloud providers improves the resilience of any single firm and concentrates the risk of all of them.

On 20 October 2025 that argument stopped being theoretical. Beginning at approximately 06:48 UTC, a latent race condition in DynamoDB's automated DNS management in Amazon Web Services' us-east-1 region produced an empty DNS record for the regional DynamoDB endpoint, which the automation could not repair. DNS resolution was restored by 09:40 UTC, but the cascade through dependent services — around 140 AWS services, including EC2 — meant full recovery took over fifteen hours. Firms whose control planes lived in that region discovered that their own multi-availability-zone architecture had not made them independent of it.

Regulators had already moved. The Critical Third Parties regime was created by the Financial Services and Markets Act 2023, and its rules were finalised jointly by the Bank of England, the PRA and the FCA on 12 November 2024, published as PS16/24 by the Bank and PRA and PS24/16 by the FCA. Designation is HM Treasury's decision, not the regulators'. On 10 July 2026 the Treasury made the first designations, effective 13 July 2026: Microsoft Ireland Operations Limited, Google Cloud EMEA Limited, Amazon Web Services EMEA SARL and Oracle Corporation UK Limited. Oversight applies only to the systemic services those firms provide to the financial sector, and the regime does not transfer responsibility: firms remain accountable for managing the risks in their own third-party arrangements. In the European Union the parallel instrument is the Digital Operational Resilience Act, which has applied since 17 January 2025.

Design rules that follow

Count trust boundaries, not companies. Every boundary has a timer, a key relationship and a different party to telephone at three in the morning, and your runbook needs all three written down before you need them.

Budget latency by segment and label each segment physics or software. You can optimise software; you cannot optimise distance, so decide where the issuer processor physically sits before you write any code. And never open a TLS session per transaction: pool, persist, keep warm and echo.

Treat a timeout as a financial event. Every one must produce a reversal or a reconciliation exception, and an exception nobody works is a customer complaint with a delay fuse.

Assume partial failure. Health checks must exercise the actual work rather than answer a ping, because a component that passes its heartbeat while corrupting its output will never be evicted by any automatic mechanism. Then rehearse the cutover, and rehearse the decision to cut over, separately. The technical failover is the easy half.

Keep the key inventory and the certificate inventory as first-class operational assets, and when you make a compliance claim about an HSM, name the validated module and its certificate rather than the appliance. On a due diligence questionnaire, precision reads as competence.

33.98 Common wrong ideas

Wrong: the five desks are five companies in five buildings. **Right:** they are five roles, and Adyen or Stripe may hold three of them in one stack while a single apparent hop conceals a gateway, a token vault, a risk engine, a switch, a scheme endpoint and a repository write.

Wrong: the card network is what makes a card machine slow. **Right:** the scheme accounts for about forty of the eight hundred milliseconds; the phone-to-reader exchange and the issuer's decision are where the time actually goes.

Wrong: the schemes publish a two-second rule for authorisation. **Right:** no such specification exists, only a nest of separate timers with separate owners — terminal, acquirer, scheme, issuer and the customer — expiring at different moments with different consequences.

Wrong: a timeout produces a slow approval. **Right:** it produces a different transaction: a reversal, a stand-in decision made by somebody who is not your bank, or a decline.

Wrong: an HSM is a safe that hides your keys. **Right:** almost every key lives as ciphertext in an ordinary database and only the local master key sits inside the tamper boundary; the security comes from a deliberately impoverished instruction set with no command capable of revealing a PIN.

Wrong: zeroisation protects your keys. **Right:** it makes every copy of them, everywhere, permanently unreadable, which is worse than losing the box.

Wrong: having two data centres means you have failover. **Right:** capacity is the easy half; detecting that a site is unwell, distinguishing unwell from briefly busy, and committing to a cutover that is itself disruptive is the part that fails.

Wrong: six-nines availability means outages are negligible. **Right:** 99.9999 per cent is 31.6 seconds a year, and the Visa Europe outage consumed roughly one thousand one hundred and fifty years of that budget in a single afternoon.

Wrong: authorisation debits the customer's account. **Right:** it writes an authorisation record — shadow balance, open-to-buy or memo post — that reduces available funds, and the ledger moves later when the clearing record arrives.

Wrong: "our HSM is FIPS 140-2 Level 3 certified" describes the appliance. **Right:** NIST certificate #3610 covers the Thales Advanced Security Platform, a module inside the payShield 10000 family, and an honest answer names the module and the certificate rather than the box.

33.99 Chapter summary in 20 lines

1. This book has described a payment as messages, but messages do not travel; optical and electrical signals travel, through glass and switches into racks in buildings with concrete walls and diesel tanks.
2. An authorisation passes through five jobs — the terminal, the merchant's payments company, the scheme network, the issuer's card system and the bank's ledger — and the answer returns down the same chain.

3. Timed end to end, a typical contactless approval takes about 790 milliseconds, comfortably inside the two seconds the industry talks about.
4. Almost two thirds of that is the phone and the reader talking before anything leaves the shop, the scheme is about five per cent, and all six network legs together are seventy milliseconds.
5. The remaining margin exists for busy issuers, retransmitted packets and geography, and geography is the only one money cannot fix.
6. Light in single-mode fibre travels about 204,200 kilometres per second, roughly 4.9 microseconds per kilometre, so London to Ashburn and back has a floor near sixty-four milliseconds however much you spend.
7. Because a full TLS handshake costs one or two round trips on top of that, payment links are long-lived, pooled and kept warm rather than opened per transaction.
8. The five desks are roles rather than companies, so the honest unit of analysis is the trust boundary, and every boundary carries a timer, a key relationship and a different party to telephone at three in the morning.
9. Only the acquirer and the issuer are scheme members, which is why the party a merchant shouts at is usually the processor while the party that owes the merchant money is always the acquirer.
10. The scheme switch itself is unglamorous — an account range table maps the leading digits of the card number to an issuer endpoint — and fraud scoring, tokenisation, currency conversion and stand-in are bolted around that lookup.
11. Visa's published 322 billion transactions a year against a peak of 83,000 messages a second implies a peak-to-mean ratio of about eight to one, which is why these networks are sized for the Friday before Christmas and why redundancy is comparatively cheap.
12. On the issuer side, authorisation verifies the chip's cryptogram, the PIN and the risk position and then writes a shadow balance rather than touching the ledger, which is what every pending transaction really is.
13. A PIN is four digits and an issuer master key can forge a whole portfolio, so neither may exist in the addressable memory of a general-purpose computer, and key operations happen inside tamper-responsive hardware that erases itself when attacked.
14. Only the local master key lives inside that hardware; everything else is a database column of ciphertext, which is what makes the architecture affordable and zeroisation catastrophic.
15. DUKPT gives each terminal a key that changes every transaction, so compromising a counter-top device yields nothing before the compromise and nothing about any other device.
16. Key blocks bind a key's permitted usage to the key itself, and PCI PIN Security Requirement 18-3 has extended that obligation to merchant hosts, point-of-sale devices and ATMs since 1 January 2025.
17. A PIN block is translated rather than revealed at each hop, existing in the clear for microseconds inside three organisations' tamper boundaries and never entering the memory of any switch.
18. Redundancy fails at the decision rather than the capacity: an 8xx echo evicts a dead link in seconds, but a link that answers every heartbeat correctly while mishandling the traffic behind it will never be evicted by any automatic mechanism.
19. That is exactly what happened to Visa Europe on 1 June 2018, when a partial switch failure poisoned the healthy site, took nearly five hours to isolate, and left 5.2 million European transactions unprocessed with UK failure rates oscillating between 7 and 35 per cent.
20. The rules that follow are to count trust boundaries rather than companies, to label every latency segment physics or software, to treat every timeout as a financial event that must produce a reversal, to health-check the real work rather than a ping, and to rehearse the decision to cut over separately from the cutover itself.

Sources: Visa corporate publications on VisaNet and the Ashburn data centre (October 2025) and Visa Europe fact sheet; Visa Europe's letter of 15 June 2018 to the Treasury Committee, with reporting by Finextra and The Guardian; Visa Developer response code documentation; Mastercard Edge Connectivity and Transaction Processing Rules; PCI SSC bulletin on PIN Security Requirement 18-3 and the PTS approved devices listing for approval 4-40266; NIST CMVP certificate #3610; Thales payShield 10K and Utimaco payment HSM specifications; EMVCo contactless specifications; FCA PS21/3 and the joint Bank of England, PRA and FCA policy statements PS16/24 and PS24/16; HM Treasury's designation announcement of 10 July 2026; AWS post-event summary and ThousandEyes analysis of the us-east-1 incident of 20 October 2025.

VOLUME IV

Moving Money Without Cards

Batch against real time. Bacs, Faster Payments, CHAPS, SEPA, SWIFT and UPI, and why each exists.

34

CHAPTER

Batch Against Real Time

34.0 What this chapter gives you

1. You will be able to explain why a £2,400 salary lands on Thursday while a £248,000 house completion lands before lunch, without reaching for "one system is old".
2. You will be able to place any rail on two axes — continuous or cyclical, gross or net — and say which of the four boxes CHAPS, Bacs and Faster Payments occupy.
3. You will be able to separate clearing from settlement, and say precisely what happens in seconds in Faster Payments and what happens three times each business day.
4. You will be able to explain how netting discharged £6.05 trillion of Bacs payments in 2025 with average daily interbank settlement of £5.6 billion.
5. You will be able to say what prefunding is, why it converts credit risk into opportunity cost, and what a Net Sender Cap caps.
6. You will be able to read the Bank of England's RTGS daily timetable and explain why Bacs settles at 9.30am and the card schemes at 11.15am and 12.15pm rather than the other way round.
7. You will be able to say why the £1.8 million average CHAPS payment matters less than the £4,586 median, and what that median reveals about why people use CHAPS.
8. You will be able to explain Herstatt risk from the events of 26 June 1974, and name the four tools that shrink, collateralise, cap or eliminate the gap it describes.
9. You will be able to argue that batch is not simply legacy, citing the recall window, file-level control and predictable funding that real-time rails cannot offer.
10. You will be able to explain why the Bank of England is extending settlement hours, and why that is an argument for making batch safer rather than for abolishing it.

Two payments leave the same British bank at nine o'clock on a Tuesday morning. One is a salary of £2,400 for a warehouse supervisor in Wolverhampton. The other is £248,000 for the completion of a house purchase in Hove. Both are in sterling, both electronic, both instructed by people at screens inside regulated institutions. The salary lands on Thursday. The house money lands before lunch.

The obvious explanation is that one system is old and one is new, and that given enough investment everything will eventually land before lunch. That explanation is wrong, and believing it will make the rest of this volume incomprehensible. The salary is slow on purpose; the house money is expensive on purpose. They are two answers to a question every payment system in every country must settle before it processes a single instruction: when a payment is made, do you act on it now, or write it down and act on a pile of them later?

That is the divide. It runs underneath Bacs and Faster Payments and CHAPS, underneath SEPA and the American ACH, underneath every card scheme in Volume III, and underneath every promise a

fintech has made about instant settlement. It is not a divide between good and bad, or modern and legacy. It is a trade between cost, risk, liquidity and operational calm, and Britain runs several payment systems side by side because different payments want different points on that trade.

The plain version

Imagine a street with two shops on it. Priya runs the bike shop; Tom runs the café opposite. They buy from each other all day long — Tom buying inner tubes and puncture kits for the bikes he lends to customers, Priya buying coffee and sandwiches for herself and her staff. There is no card machine and no bank in this story, just a cash box in each shop and a road between them.

There are two ways to run this street.

The first is that every time money is owed, somebody walks. Tom buys a £6.50 puncture kit, so he takes £6.50 out of his cash box, crosses the road and puts it in Priya's. Twenty minutes later Priya buys two coffees for £5.40, so she walks back with £5.40. Every purchase is a walk. Nobody ever owes anybody anything, because the moment a debt exists it is already paid.

The second is that they keep a slate — a chalkboard by the till. Tom writes "£6.50 to Priya". Priya writes "£5.40 to Tom". Nobody walks. At six o'clock they compare slates, work out who is ahead, and one person makes one walk with the difference.

Now put real numbers on a real Tuesday. Tom buys from Priya eleven times, totalling £54.30. Priya buys from Tom nine times, totalling £49.80. Under the first system there are twenty walks across the road, moving £104.10 of coins. Under the second there is one walk at six o'clock, moving £4.50.

Four pounds fifty instead of a hundred and four pounds ten; one crossing instead of twenty. The slate made nobody richer or poorer — Tom and Priya end the day exactly where they would have either way. What it did was cancel the payments against each other before any coins moved. That cancelling is the single most useful trick in money, and the whole first half of this book has been building towards it.

Why the slate gets better as the street gets longer

Add shops. Now there are twelve businesses on the street and by the end of the day they have made two hundred payments to each other. Under the walking system that is two hundred crossings. Under the slate system, all twelve slates go to the newsagent at six o'clock, who works out that seven of the twelve owe money overall and five are owed it, and that the totals come to £312. Seven people walk to the newsagent with money; five collect from her. Twelve walks, not two hundred.

That is roughly what Bacs does with British payroll, and the numbers are not small. In 2025, Bacs handled 6.86 billion payments worth £6.05 trillion. Because it settles once each working day, the amount that actually had to be moved between banks averaged £5.6 billion a day — somewhere near £1.4 trillion across a working year to discharge £6.05 trillion of payments. Roughly four pounds of payments for every pound that moves. The other three pounds cancelled themselves out on the slate.

What the slate costs you

The slate is not free, and a bright twelve-year-old spots the problem before an adult does: what happens if a shop shuts before six o'clock?

Suppose Tom sells the café at four and disappears. Priya has been serving him all day on the slate. Under the walking system she would already have his coins in her box; under the slate system she

has chalk. She spent the day handing over inner tubes in exchange for a number on a board, and the number is now worthless.

That gap — the hours between the goods changing hands and the money changing hands — is the price of the slate. Everybody on the street is trusting everybody else to still be solvent at six o'clock. It usually works. Occasionally, catastrophically, it does not, and that is why some payments do not use a slate at all.

The trolley and the runner

The other half of the divide is about timing rather than cancelling, and the school post room is the better picture for it.

A school office has a trolley. Letters posted into the internal tray are taken round the building once a day at four o'clock. This is efficient: one person, one circuit, hundreds of letters. It also means a letter dropped in the tray at 4:05pm sits there for twenty-four hours, not because anybody is being slow but because the trolley has gone. There is a cut-off, and being one minute the wrong side of it costs a whole day.

The alternative is a runner, who takes each letter to its recipient immediately. That is fast for you, expensive per letter, and it means the school must keep a runner sitting available at all times, doing nothing, in case somebody has a letter.

Both exist in British money. The trolley is Bacs, which takes three working days on a fixed timetable and carries most salaries and Direct Debits. The runner is CHAPS, which will move a house purchase across London this afternoon and charge for it. And there is a third thing, Faster Payments, which looks like a runner to you and behaves like a trolley behind the scenes: the message about your money sprints across in seconds, while the coins between the banks go on a trolley three times a day.

The numbers that make the point

Here is the whole argument in one comparison. In 2025, Bacs handled 6.86 billion payments; CHAPS handled 53.3 million, about one for every hundred and twenty-nine Bacs payments. But the money that had to be moved between banks to settle Bacs averaged £5.6 billion a day, while CHAPS moved £371.3 billion a day. The system with a hundred and twenty-nine times fewer payments moved sixty-six times more actual money.

That is not a paradox. It is the design. Bacs carries an enormous number of small, predictable, cancellable payments and squeezes them down before anything moves. CHAPS carries a small number of large, urgent, un-cancellable payments and moves each one on its own, immediately, in full.

You should now be able to say the following at dinner and be right. Some payment systems collect instructions and process them together, which is cheap and lets payments cancel each other out, but leaves a gap in which somebody could fail. Others act on each instruction the moment it arrives, closing the gap but requiring everybody to keep a great deal of money sitting ready at all times. Britain runs both, because British payments are not all the same kind of payment.

Where the plain version stops being true

The street with the slate is a good picture of netting and a poor picture of a payment system. Four corrections, and each one matters for what follows.

The first correction is that "batch" and "slow" are different things, and so are "real time" and "instant". The analogy makes it sound as though the trolley is slow because collecting things takes time.

It is not. A Bacs file of four hundred thousand salary payments is validated, sorted and distributed by the central infrastructure in a tiny fraction of the three days it takes to reach people's accounts. The three days are a rulebook, not a speed limit — a deliberate agreement about which day is input, which is processing and which is entry, so that every bank knows exactly when to expect what. Double the computing power tomorrow and the cycle would still be three days.

The reverse correction is sharper. Faster Payments credits the person you are paying in seconds, twenty-four hours a day. But the banks do not settle with each other in seconds. As of writing, the net obligations arising from Faster Payments settle in the Bank of England's system three times each business day. What happens in seconds is the clearing — the exchange of the instruction and the crediting of the customer. What happens three times a day is the settlement — the movement of central bank money between the banks. Your money is available immediately; the banks square up later. Confusing those two clocks is the commonest error in payments writing.

The second correction is that the slate hides who is carrying the risk, and modern systems do not leave it lying around. In the analogy Priya simply absorbs the loss if Tom disappears. That is not how a designed payment system works. The customer receiving a Faster Payment gets usable funds immediately while the sending bank's obligation is not discharged until the next settlement cycle, so the exposure is real — and it is managed rather than tolerated. Bacs, Faster Payments and the cheque-based Image Clearing System all use prefunding: each settling participant must hold cash in a segregated account at the Bank of England covering the largest net debit it could possibly reach before the next settlement, and if it fails, that ring-fenced cash completes the settlement. The Bank's own description is unambiguous — prefunding eliminates settlement risk in the systems that use it, at the cost of tying up liquidity.

So the honest version of the slate is that Priya may trade with Tom on chalk only up to a limit, and Tom deposited that limit in cash with a trusted third party before the day began. The risk did not vanish; it became a liquidity cost.

The third correction is that there is one cash box in the analogy and two tiers of money in reality. When Bacs nets four hundred thousand salary payments down to a single figure, what finally moves is not "money" in the sense a customer means. It is a balance in a settlement account at the Bank of England — central bank money, the only kind of sterling that can extinguish an obligation between banks with finality. The salary in the supervisor's account is commercial bank money: a liability of his bank to him. Netting compresses obligations denominated in commercial bank money; settlement discharges the residue in central bank money. A system that clears in one form of money and settles in another has a seam in it, and every argument in this volume about risk, finality and failure lives in that seam.

The fourth correction is that "batch versus real time" is not one axis but two, and there are four boxes, not two. The plain version blurs the question of *when* obligations are calculated with the question of *how*. A system can process each payment individually the moment it arrives, or in periodic cycles; independently, it can settle each payment at face value, or settle only the net difference. CHAPS is individual and gross. Bacs is cyclical and net. Faster Payments is individual for clearing and cyclical and net for settlement — a hybrid that did not exist in Britain before 2008 and now carries most of the country's non-card retail payments. Once you see the two axes, the industry's vocabulary stops sounding contradictory.

The technical version

Two axes, four quadrants

Clearing is the exchange, validation and reconciliation of payment instructions between participants, including calculation of the resulting obligations. **Settlement** is the discharge of those obligations by the transfer of funds — for systemically important sterling systems, in central bank money across accounts at the Bank of England. Chapter 7 established this distinction; here it becomes the coordinate system.

The first axis is timing: does the system act on each instruction as it arrives, or accumulate instructions and act on a defined cycle? The second is obligation calculation: does each payment settle at face value, or are payments offset so that only a net position settles?

	Gross (each payment at face value)	Net (offset positions only)
Continuous / real time	Real-time gross settlement. CHAPS.	Rare; continuous offsetting exists as a liquidity tool inside RTGS, not a settlement model.
Cyclical / deferred	Deferred gross settlement. Uncommon in sterling retail.	Deferred net settlement. Bacs, Faster Payments, LINK, Image Clearing System, Visa Europe, Mastercard Europe, PEXA.

The bottom-right box is where most payments in most countries live, because the arithmetic is cheapest there. The top-left box is where most *value* lives, because the risk is smallest.

What the Bank of England actually operates

As of writing (August 2026), the Bank of England supports four settlement models across its real-time gross settlement service: real-time gross settlement, delivery versus payment, and prefunded and unfunded variants of deferred net settlement.

Real-time gross settlement is used by exactly one system: CHAPS. Delivery versus payment is used by the payment leg embedded in CREST, the securities settlement system operated by Euroclear UK & International, where sterling settlement takes place in high-frequency cycles through the day, cash settling against securities in real time against segregated liquidity.

Deferred net settlement is used by seven sterling systems, all settling their net obligations across accounts in RTGS: Bacs, Faster Payments, the cheque-based Image Clearing System, LINK, Mastercard Europe, PEXA and Visa Europe. Each determines its own access criteria and its own number and duration of settlement cycles. Sterling settlement for CLS, the foreign exchange settlement system, runs through CLS's direct participation in CHAPS rather than as a separate model.

The relative sizes are worth stating precisely, because most people's intuition is wrong by two orders of magnitude. In the second quarter of 2026, average daily settlement across RTGS totalled £902,018 million: CREST delivery-versus-payment £477,841 million, CHAPS £411,562 million, and the seven retail net settlement systems between them £12,615 million — Bacs £6,205 million, Visa Europe £2,356 million, Faster Payments £2,298 million, Mastercard £1,480 million, LINK £238 million, cheque imaging £37 million, PEXA £1 million. Every retail payment made in Britain by every non-card rail and both major card schemes, netted down, comes to about three per cent of what CHAPS settles the same day.

The RTGS daily timetable

The clearest single artefact in British payments is the Bank of England's published timetable of settlement events, the skeleton on which everything else in this volume hangs. Accurate at time of writing, the sterling settlement day runs as follows, with the caveat the Bank itself attaches: timings can vary in contingency.

Time	Event
5.15am	Transfers between own accounts and user interface access enabled; Note Circulation Scheme settlement
6am	Start of CHAPS settlement; start of CREST DvP, free-of-payment and delivery-by-value settlement
7am to 11am	Hourly CLS pay-in and pay-out deadlines
7am	Faster Payments settlement (first cycle)
8.45am	PEXA settlement
9.30am	Bacs settlement
10.45am	PEXA settlement
11am	LINK settlement
11.15am	Mastercard settlement
12.15pm	Visa settlement
12.45pm	PEXA settlement
1pm	Faster Payments settlement (second cycle)
2.45pm	PEXA settlement
3.45pm	PEXA settlement; end of equity and gilt DvP settlement
4.30pm	Image Clearing System settlement
4.45pm	PEXA settlement
5pm	Faster Payments settlement (third cycle)
5.30pm	End of delivery-by-value settlement
5.40pm	End of CHAPS settlement for customer payments (pacs.008)
6pm	End of CHAPS settlement for financial institution payments (pacs.009); end of free-of-payment settlement
6.30pm	Note Circulation Scheme settlement
8pm	Latest end of contingency extension for RTGS

Two things deserve underlining. First, the deferred systems are not scattered randomly; their cycles are placed so that money arriving from one is available to fund obligations in the next — Bacs at 9.30am, before LINK at 11am and the card schemes at 11.15am and 12.15pm. Second, CHAPS has two closing times: 5.40pm for customer payments carried in the ISO 20022 `pacs.008` message, and 6pm for financial institution payments in `pacs.009`. The gap gives banks twenty minutes to move liquidity between themselves after the last customer payment.

Bacs: the shape of a three-working-day cycle

Bacs is the archetype of deferred net settlement in Britain, running since 1968; over 183.6 billion transactions have been credited or debited to British accounts through it. It carries two products: Direct Debit, where the beneficiary pulls funds under a mandate, and Bacs Direct Credit, where the payer

pushes them. In 2025 it processed 6.86 billion payments worth £6.05 trillion, including 5.0 billion Direct Debits, an annual high.

The cycle has three named days. **Input day** (**day one**) is the last day on which a service user or a Bacs approved bureau may submit a payment file for that cycle. Files must be transmitted to Bacs between 07.00 and 22.30, normally over Bacstel-IP, and may be submitted up to thirty days ahead of the intended payment date — which is why a payroll bureau can build December's run in November. **Processing day** (**day two**) is when the central infrastructure sorts the file and delivers the resulting items to each receiving payment service provider, which processes each payment against its own customer accounts. **Entry day** (**day three**) is when payments are credited to recipients' accounts and debited from the payer's, and it is the day the interbank obligation is settled: Bacs net settlement takes place once each business day in RTGS, at 9.30am.

"Day" here means working day. A file submitted on a Thursday enters accounts the following Monday; a file submitted before a bank holiday weekend waits for the calendar. That is the source of most of the frustration ordinary people feel with Bacs, and of most of its value: because the timetable is fixed and published, a payroll department knows in October exactly which day in December the money becomes irreversible, and can plan its funding to the pound.

There is also a control feature real-time systems cannot offer. Because the file sits in the cycle for two days before entry, payments can be recalled after submission, provided the payer's provider is notified before a specified cut-off. A submitted batch is not yet a payment made.

Faster Payments: real-time clearing, deferred net settlement

The Faster Payment System, operated by Pay.UK, is the hybrid, and understanding it correctly is the point at which most people's model of British payments either becomes accurate or stays wrong forever.

On the customer-facing side it is a real-time system: payments are cleared and the beneficiary credited within seconds, day and night, three hundred and sixty-five days a year. It supports payments of up to £1 million, and it is worth being exact about that limit. £1 million is the value limit applied by the central infrastructure. Individual banks and building societies apply their own, usually much lower, limits, which vary by channel — mobile app, online banking, telephone and branch may face four different ceilings at the same institution. When a customer says "my bank's Faster Payments limit is £25,000", they are describing their bank's policy, not the rail.

On the interbank side it is a deferred net settlement system, settling three times each business day in RTGS, at 7am, 1pm and 5pm as of writing. In 2025 Faster Payments processed 5.55 billion transactions worth £4.84 trillion, while the net obligations settling in RTGS averaged £2,070 million a day. On a 365-day denominator that is around £13 billion of clearing a day compressed to around £2 billion of settlement, better than six to one. The comparison is indicative rather than exact, because clearing runs every calendar day while settlement runs only on business days, so weekend traffic is carried into Monday's cycles.

The exposure between cycles is handled by prefunding. Each direct settling participant holds funds in a dedicated prefunding account at the Bank of England sufficient to cover the maximum net debit position it could reach before the next settlement. The cap on that position is the participant's **Net Sender Cap**. If a participant fails intra-cycle, the ring-fenced cash completes the settlement and no other participant takes a credit loss.

That is an elegant answer to the problem the plain version raised, and it is not free: prefunding converts credit risk into opportunity cost, because cash in a segregated account is cash not being used

for anything else. Pay.UK's framework for setting these caps has historically been prescriptive, built around a Minimum Net Sender Cap and a Peak Contingency Value that participants had to hold at all times, and Pay.UK announced in 2026 that it is moving to a more flexible model. Treat the detail as live; the principle — a capped, collateralised maximum debit — is stable.

The cost of getting prefunding wrong is not theoretical. The Bank's service records note that on 8 September 2021 the first Faster Payments settlement cycle of the day was delayed by forty-five minutes because a participant had insufficient funds in its settlement account. A system instant to customers can still be held up at the settlement layer by one institution's liquidity management.

CHAPS: gross, real time, and what that costs

CHAPS is the sterling high-value system, operated by the Bank of England since November 2017, with more than thirty direct participants and several thousand institutions accessing it indirectly through them. Payment obligations between settlement participants are settled individually, on a gross basis, in RTGS, throughout the business day, and settlement is final and irrevocable between direct participants at the moment it occurs. There is no minimum and no maximum payment value.

The system is usually open from 6am to 6pm, Monday to Friday, excluding bank and public holidays in England and Wales, with customer payments closing at 5.40pm. Until 17 June 2016 the settlement day was 6am to 4.20pm; it was extended by one hour and forty minutes from 20 June 2016, and in the following summer an average of 6.1 per cent of CHAPS daily values and 5.3 per cent of volumes settled in the additional hundred minutes.

This is where the phrase "early-afternoon cut-off" needs handling carefully, because it is repeated everywhere and it is not a system fact. The scheme cut-off for customer payments is 5.40pm. What is early-afternoon is the internal cut-off imposed by individual banks on their own customers, set earlier — often much earlier — to leave room for checks, funding and exception handling. The Bank states this explicitly: customer cut-off times set by providers may be earlier. A solicitor who says "we must instruct by two o'clock" is describing their bank, not CHAPS.

Direct participants may submit payments as **urgent**, settling immediately subject to available liquidity, or as **non-urgent**, accepting some delay in exchange for liquidity efficiency. Non-urgent settlement is usually available from 6am until 5.30pm, after which outstanding items are automatically promoted to urgent. Non-urgent payments run through a liquidity saving mechanism of matching cycles, offsetting payments so that a smaller amount of liquidity discharges a larger amount of value; on average they settle around eight minutes after submission. Even a real-time gross system, then, quietly borrows the netting idea — not to defer settlement but to reduce the liquidity needed to achieve it.

The scale figures for 2025 are these. CHAPS volumes grew 1.1 per cent to a record 53.3 million payments, an average of 210,482 per working day. Total value settled rose 7.3 per cent to £93.9 trillion, an average of £371.3 billion a day. The average payment value was £1.8 million; the median was £4,586. Ninety-four per cent of CHAPS payments were for £1 million or less. Financial institution payments in `pacs.009` were 24 per cent of payments but 73 per cent of value. CHAPS represents 0.4 per cent of UK payment volumes but 91 per cent of sterling payment values, excluding flows internalised within providers, and it settles the annual GDP of the United Kingdom every nine working days.

The median matters. If you believe CHAPS is only for enormous wholesale transfers, the £1.8 million average confirms it; the £4,586 median tells the truth. The typical CHAPS payment goes through CHAPS not because it is large but because it must be certain and it must be today.

The arithmetic of netting, and the arithmetic of liquidity

Under gross settlement, a participant must fund every outgoing payment at the moment it settles, from liquidity it holds. Its peak requirement over the day depends on the timing and size of its outflows relative to its inflows: pay out £3 billion before receiving anything and you need £3 billion available. CHAPS direct participants manage this with their own reserves, secured intraday liquidity provided by the Bank against eligible collateral, and the liquidity saving mechanism, which reduces the amount required by matching offsetting payments rather than deferring them.

Under deferred net settlement, a participant needs only enough liquidity to cover its net position at the settlement point, and if it is a net receiver, none at all. Against that, it must either accept credit exposure to other participants between cycles or prefund the maximum net debit it could reach, pinning down liquidity for the whole cycle regardless of its actual position. Set the two side by side using published figures.

	Bacs	Faster Payments	CHAPS
Payments, 2025	6.86 billion	5.55 billion	53.3 million
Value cleared, 2025	£6.05 trillion	£4.84 trillion	£93.9 trillion
Average value per payment	about £880	about £870	£1.8 million (median £4,586)
Average daily value settled in RTGS, 2025	£5,604 million	£2,070 million	£371,332 million
Settlement model	Deferred net, prefunded	Deferred net, prefunded	Real-time gross
Settlement frequency	Once per business day	Three times per business day	Continuous

Bacs and Faster Payments carry payments of almost identical average size — £880 against £870 — yet one takes three working days and the other takes seconds. The difference is not size but purpose: Bacs carries predictable, scheduled flows where certainty of date is worth more than speed, and Faster Payments carries unscheduled ones where speed is the point. And CHAPS, carrying 0.8 per cent of the payment count of Bacs, moves sixty-six times as much central bank money each day. Real-time gross settlement is not a tweak to netting; it is a different liquidity regime, affordable only because the payments that use it are few.

Settlement risk, and why anybody bothers

The reason the whole industry does not simply net everything and settle once a month is a Thursday in June 1974.

Bankhaus Herstatt was a German bank active in foreign exchange. On 26 June 1974 its banking licence was withdrawn and it was ordered into liquidation during the German banking day. Counterparties had already paid Deutsche Marks to Herstatt in Frankfurt that morning; the corresponding dollar payments, due in New York where the business day had not yet ended, were never made. They had irrevocably delivered one leg and never received the other. The exposure created by settling two legs of a transaction at different times, in different time zones, with a gap in between, has been called **Herstatt risk** ever since, and it is the reason CLS exists — a system settling both legs of a foreign exchange transaction simultaneously, whose sterling settlement runs through direct participation in CHAPS, with pay-in and pay-out deadlines every hour between 7am and 11am on the timetable above.

Herstatt is the general case of Priya and Tom. Any gap between the creation of an obligation and its discharge is a window in which a participant can fail and somebody else discovers they were holding

a promise rather than money. The toolkit for that window is small: shrink it by settling more often, collateralise it by prefunding, cap it with net sender caps, or eliminate it by settling gross, in real time, in central bank money.

Finality is the legal counterpart. A settlement that has occurred in RTGS cannot be unwound by a participant's subsequent insolvency. In the United Kingdom that protection comes from the Financial Markets and Insolvency (Settlement Finality) Regulations 1999, which shield transfer orders in designated systems from insolvency law that would otherwise reverse them, together with the recognition regime for payment system operators under the Banking Act 2009. Without the legal certainty the mechanical certainty would be worthless: a liquidator could reach back and undo it.

The Bank of England's current work on settlement hours is framed in exactly these terms. Its consultation paper of 18 May 2026 notes that longer RTGS hours would reduce settlement risk by enabling more frequent net settlement overnight and at weekends, observing that over an Easter weekend the gap between settlements runs from Thursday evening to Tuesday morning. Four and a half days of retail payments accumulating with no settlement in central bank money is the largest exposure window in the sterling system, and it exists because the calendar has holidays in it.

Cost per transaction, and what batch buys you

The cost argument is usually asserted and rarely evidenced. It can be evidenced, starting with how the settlement agent charges. The Bank of England operates RTGS and CHAPS on a full cost recovery basis. The CHAPS tariff is an annual participation fee for each direct participant plus a **per-item fee** on debits. For the retail net settlement systems, the Bank charges an annual fee per settlement participant and **no per-item fee at all**. That is the economics of the divide, written into a published price list: a gross system consumes settlement resource per payment and is priced per payment, while a net system consumes it per cycle and its price does not vary with how many payments the cycle contained.

The same shape recurs at every layer. In a batch rail, the marginal cost of the four-hundred-thousandth record in a payroll file is a few dozen bytes and a fraction of a second on infrastructure that was going to run anyway. In a real-time rail it is a full request-and-response round trip with validation, fraud scoring, sanctions screening, an account credit, a customer notification and a durable audit record, executed inside a latency budget of single-digit seconds, on infrastructure sized for peak rather than average and available continuously.

Operational cost differs in kind rather than degree. A batch system has a natural quiet period and an obvious point at which yesterday is closed and reconciliation can run against a fixed, complete set of records; it can be patched and tested between cycles; and its usual failure mode is a late file, recoverable within the cycle without any customer seeing anything. A twenty-four-hour real-time system has none of that: no natural end of day, so reconciliation is cut artificially against a moving target; no maintenance window, so upgrades happen live; and a failure mode that is immediately visible, because the customer is standing there watching for the money. Chapter 53 argues that reconciliation is where fintechs actually die; the reason is that real time removes the seam reconciliation was designed to use.

Batch has three further properties the rhetoric of the last fifteen years has treated as legacy defects and which are in fact advantages. It gives a recall window: a Bacs submission can be withdrawn before the cut-off, whereas a completed Faster Payment cannot be recalled at all, recovery being a request for goodwill rather than a mechanism — which is why authorised push payment fraud is a Faster Payments problem far more than a Bacs problem. It gives file-level control: a failed file is rejected, corrected and resubmitted as a unit, whereas a systematic error on a real-time rail produces thousands of individually completed wrong payments. And it gives predictability of funding: a treasurer running payroll on Bacs knows the exact date and amount days in advance.

Why real time is not simply better

Real time eliminates settlement risk in the gross model and shortens it in the hybrid. It gives customers what they want, and it enables use cases — refunds while the customer is still in the shop, gig-economy payouts on completion — that a three-day cycle cannot serve at all.

Against that, it costs more per payment at every layer from the settlement agent downwards. It demands far more liquidity, as intraday balances for gross settlement or prefunded collateral for the hybrid. It removes the recall window, converting fraud from a controllable operational problem into a customer-harm problem with statutory consequences. It removes the maintenance window and the reconciliation seam — the Bank's own consultation weighs a twenty-two-hour settlement day, preserving a defined window for maintenance, upgrades and reconciliation, against a 23.5-hour day that maximises availability but reduces the opportunity to maintain overnight. And it makes availability a systemic obligation rather than a service target: a batch system down for two hours in the afternoon has usually harmed nobody, whereas a real-time system down for two hours has visibly failed thousands of people.

The Bank's published availability record makes the last point concrete. On 18 July 2024 a 245-minute outage to CHAPS settlement was caused by an issue with Swift's Y-Copy service globally; settlement was extended by around ninety minutes and everything submitted settled successfully. On 31 July 2024 a 91-minute disruption caused by an expired certificate in the Bank's IT infrastructure delayed CHAPS and a number of retail settlements. On 14 August 2023 a technical issue kept RTGS closed from 6am until 12.20pm, everything nonetheless settling before the end of the day. These are the incident records of a well-run system, and they are instructive precisely because the deferred systems absorbed the disruption while the real-time one propagated it.

The spectrum, and where the rest of this volume sits

Every rail in the chapters that follow can be placed on the two axes.

Rail	Clearing	Settlement	Frequency	Availability	Value limit
Bacs	Batch, three working days	Deferred net, prefunded	Once per business day, 9.30am	Input 07.00 to 22.30 on working days	No scheme limit
Faster Payments	Real time, seconds	Deferred net, prefunded	Three times per business day: 7am, 1pm, 5pm	24/7/365	£1 million central limit; lower bank limits
CHAPS	Real time	Real-time gross, central bank money	Continuous	6am to 5.40pm customer, 6pm interbank, working days	None
Image Clearing System	Batch, two-day cycle	Deferred net, prefunded	Once per business day, 4.30pm	Working days	Not a scheme limit
Card schemes (Visa Europe, Mastercard Europe)	Real-time authorisation, batch clearing	Deferred net	24-hour cycles; weekend and holiday cycles settle next business day	24/7 authorisation	Scheme and issuer limits

Rail	Clearing	Settlement	Frequency	Availability	Value limit
SEPA Instant Credit Transfer	Real time, nine seconds under the 2025 rulebook	Varies by clearing and settlement mechanism	Varies	24/7/365 with permitted planned maintenance	No scheme maximum since the 2025 rulebook
Open Banking payment initiation	An API layer, not a rail; initiates over Faster Payments	Whatever the underlying rail does	Underlying rail	Underlying rail	Underlying rail

The last row is the one people get wrong most often. Open Banking is an interface standard that lets an authorised third party initiate a payment from your account with your consent. It does not clear, does not net, does not settle, and holds no accounts at the Bank of England. When a payment initiated through an Open Banking journey arrives in seconds, that is Faster Payments arriving in seconds. Chapter 44 develops this; the point to carry forward is that an API is not a settlement model, and no API design changes where a payment sits on the two axes above.

What is moving, as of writing

The batch–real-time frontier is where most of the change is happening. The following are live as of August 2026 and should each be checked against the primary source before being relied upon.

The Bank of England migrated RTGS to a new core ledger and settlement engine on 28 April 2025, replacing the Enquiry Link user interface with BERTI. In February 2026 it announced that CHAPS settlement will open at 01:30 rather than 06:00, Monday to Friday, from September 2027, subject to final confirmation of timelines with affected direct participants. Its consultation paper of 18 May 2026 sets out the next steps towards near-24x7 settlement: an additional settlement day at the weekend, most likely Sunday, with certain UK bank holidays, but not before 2029; then twenty-two hours a day across Monday to Friday and one weekend day, not before 2031; and in the longer term either twenty-two hours every day with a daily maintenance window, or near-continuous settlement with a short daily downtime. Today there are twelve hours in which CHAPS does not settle, reducing to seven and a half in September 2027.

Notice what that sequence is. It is not the abolition of batch. It is the systematic shortening of the windows during which deferred systems accumulate unsettled obligations, by making the gross system available at more times. The Bank's stated reason for prioritising weekend settlement is that it addresses the longest period during which settlement is unavailable and reduces the risk and liquidity costs borne by the net settlement systems. Real time is being extended to make batch safer.

In the euro area, Regulation (EU) 2024/886 — the Instant Payments Regulation — entered into force in April 2024, with obligations taking effect for euro-area credit institutions on 9 January 2025 and 9 October 2025, and later for electronic money institutions, payment institutions and providers in non-euro EEA countries. The European Payments Council's 2025 SEPA Instant Credit Transfer rulebook, in effect since 5 October 2025, gives the payer's provider, the payee's provider and the clearing and settlement mechanism nine seconds in total to make funds available, within the ten-second maximum set by the amended SEPA Regulation, and removes the scheme-level maximum transaction amount.

In the United States, the Nacha rules raised the Same Day ACH per-payment limit to $1 million from 18 March 2022, across three daily processing windows; entries above the limit submitted into a same-day window are not rejected but processed for next-day settlement in the next available window. Nacha

has announced a further increase to $10 million effective 17 September 2027. That is a batch system being pushed up the value curve rather than replaced by a real-time one — the clearest evidence available that the two designs are converging rather than one winning.

A note for implementers

Three design consequences follow, and they are why this chapter comes before the rail-specific ones.

Value date is not processing date, and conflating them produces wrong statements, wrong interest and wrong reconciliation. In a batch rail the debit, credit and settlement all carry the same value date by rulebook design, and that date is not the date you submitted; in a hybrid rail the customer credit and the interbank settlement fall on different clocks and sometimes different days.

Cut-offs are business logic, not configuration. A submission at 22:31 on input day is not late; it is a submission into the next cycle, with a different entry day, value date and funding requirement. Systems that treat a cut-off as a warning rather than a state transition generate an expensive class of incident.

And real-time rails require idempotency while batch rails require reconciliation, neither substituting for the other. A real-time payment that times out may or may not have happened, and the only safe response is a retry that cannot double-pay. A batch file that was accepted may still contain items rejected downstream, and the only safe response is to reconcile the returned reports against what you submitted, item by item.

Everything from here is a specific answer to the question this chapter posed. Bacs answers with three days and one settlement. Faster Payments answers with seconds for the customer and three settlements for the banks. CHAPS answers by refusing to accumulate anything at all. SEPA, SWIFT correspondent banking, UPI and the instant rails elsewhere each answer differently again, shaped by their own regulators, their own liquidity conditions and their own tolerance for the gap between a promise and a payment.

34.98 Common wrong ideas

Wrong: batch rails are slow because collecting and processing takes time. **Right:** a Bacs file of four hundred thousand salary payments is validated, sorted and distributed in a tiny fraction of the cycle; the three days are a rulebook, and doubling the computing power tomorrow would not change them.

Wrong: Faster Payments settles in seconds. **Right:** the beneficiary is credited in seconds, but the net interbank obligations settle in RTGS three times each business day; clearing and settlement run on different clocks.

Wrong: deferred net settlement leaves participants exposed to each other between cycles. **Right:** Bacs, Faster Payments and the Image Clearing System all prefund the maximum net debit in segregated accounts at the Bank of England, so the exposure becomes a liquidity cost rather than a credit loss.

Wrong: what settles between banks is the same money the customer sees. **Right:** netting compresses obligations denominated in commercial bank money, while settlement discharges the residue in central bank money — two kinds of money and two sets of books.

Wrong: CHAPS has an early-afternoon cut-off. **Right:** the scheme cut-off for customer payments is 5.40pm; the early-afternoon deadline is the individual bank's internal cut-off imposed on its own customers.

Wrong: CHAPS is only for enormous wholesale transfers. **Right:** the average is £1.8 million but the median is £4,586 and 94 per cent of payments are for £1 million or less, so the typical CHAPS payment goes that way because it must be certain and today.

Wrong: a bank quoting a £25,000 Faster Payments limit is describing the rail. **Right:** the central infrastructure limit is £1 million, and bank and channel limits are policy layered on top of it.

Wrong: Open Banking is a payment rail. **Right:** it is an interface standard that initiates over Faster Payments; it does not clear, net or settle, and holds no account at the Bank of England.

Wrong: a real-time gross system never nets anything. **Right:** CHAPS non-urgent payments run through a liquidity saving mechanism of matching cycles that offsets payments so less liquidity discharges more value, settling on average about eight minutes after submission.

Wrong: the direction of travel is the abolition of batch. **Right:** the Bank is extending gross settlement hours to shorten the windows in which deferred systems accumulate unsettled obligations, and Nacha is pushing Same Day ACH up the value curve — the two designs are converging, not competing.

34.99 Chapter summary in 20 lines

1. Every payment system must settle one question before it processes a single instruction: act on each payment now, or write it down and act on a pile of them later.
2. Acting later lets payments cancel against each other, which is the most useful trick in money: twelve businesses making two hundred payments can square up with twelve walks rather than two hundred.
3. Bacs is the industrial version of that trick, discharging £6.05 trillion of payments in 2025 with an average of £5.6 billion a day actually moving between banks.
4. The price of netting is the gap between goods changing hands and money changing hands, during which a participant can fail.
5. The other half of the divide is timing: a trolley that runs a fixed circuit with a cut-off, against a runner who leaves immediately and costs more per letter.
6. Bacs is the trolley, CHAPS is the runner, and Faster Payments looks like a runner to the customer while behaving like a trolley between the banks.
7. Batch is not the same as slow, because the three Bacs days are an agreement about input, processing and entry rather than a limit on computing speed.
8. Real time is not the same as instant, because Faster Payments clears in seconds and settles its net obligations in RTGS three times each business day.
9. Designed systems do not leave the settlement gap lying around: prefunding in segregated Bank of England accounts converts credit risk into an opportunity cost.
10. Clearing happens in commercial bank money and settlement in central bank money, and every argument in this volume about risk, finality and failure lives in that seam.
11. Timing and obligation calculation are two independent axes giving four boxes, and CHAPS is continuous and gross, Bacs is cyclical and net, and Faster Payments is real time for clearing and cyclical and net for settlement.
12. The Bank of England runs real-time gross settlement for CHAPS alone, delivery versus payment for CREST, and prefunded and unfunded deferred net settlement for seven sterling systems.
13. In the second quarter of 2026 those seven retail systems together settled £12,615 million a day against CHAPS's £411,562 million, so every retail rail netted down is about three per cent of what CHAPS settles.

14. The published RTGS timetable is the skeleton of the sterling day, and its cycles are ordered so that money arriving from one settlement is available to fund obligations in the next.
15. Bacs and Faster Payments carry payments of almost identical average size, about £880 against £870, so the difference between them is purpose rather than value: scheduled certainty against unscheduled speed.
16. CHAPS carries 0.4 per cent of UK payment volumes and 91 per cent of sterling payment values, and settles the annual GDP of the United Kingdom every nine working days.
17. Herstatt's failure on 26 June 1974 is the general case of the settlement gap, and the toolkit for it is small: settle more often, prefund it, cap it, or settle gross in central bank money.
18. Legal finality under the Settlement Finality Regulations 1999 is what stops a liquidator undoing what RTGS has done, without which the mechanical certainty would be worthless.
19. Gross settlement is priced per item and net settlement per cycle, and batch also buys a recall window, file-level control and predictable funding, which is why authorised push payment fraud is a Faster Payments problem far more than a Bacs one.
20. The frontier is moving through earlier CHAPS opening, weekend settlement and a $10 million Same Day ACH limit, but the direction is convergence: real time is being extended to make batch safer, not to replace it.

Sources: Bank of England (payment and settlement pages, summary of RTGS daily timetable, payment and settlement statistics, CHAPS pages, and the consultation paper "Extending RTGS and CHAPS settlement hours", 18 May 2026); Pay.UK and Bacs (scheme pages, Bacs processing cycle, Net Sender Cap publications); European Payments Council (SCT Inst scheme and 2025 rulebook); Nacha and Federal Reserve Financial Services (Same Day ACH rules). All timings, limits and figures accurate at time of writing, August 2026.

35

CHAPTER

Bacs and the Three-Day Cycle

35.0 What this chapter gives you

1. You will be able to say what happens on each of the three Bacs days, and name the 22:30 deadline that decides which cycle a file belongs to.
2. You will be able to work out the latest input day for any payroll date, including the case where a bank holiday makes the answer Thursday rather than Friday.
3. You will be able to explain why five calendar days can pass inside a three-day cycle without anything having gone wrong.
4. You will be able to say where the money is on day one and day two, and why nothing is in transit.
5. You will be able to separate the clearing that happens on day two from the 09:30 settlement on day three, and say which kind of money each one moves.
6. You will be able to read a Standard 18 payment record by counting characters, and say what fields 7, 9, 10 and 11 carry.
7. You will be able to explain why a Bacs file must balance and what the contra record does for reconciliation.
8. You will be able to distinguish Direct Debit from Bacs Direct Credit by direction, authority, return mechanism and transaction code, and say what sending code 19 by accident does to a mandate.
9. You will be able to say what a Service User Number is, why one organisation may hold several, and how a facilities management arrangement differs from a bureau.
10. You will be able to describe what a Bacs Approved Bureau inspection covers, how the volume tier sets its cadence, and what a single Fail does to the result.

There is a system that moved 6.86 billion payments worth £6.05 trillion in a single year, that pays most of the salaries in Britain and collects most of the household bills, and that almost nobody outside the payments industry can name. It has been running since 1968. It has never been fast, it has never tried to be, and its slowness is not a defect.

Bacs is the oldest of the sterling rails in this volume and the least glamorous. Faster Payments arrives in seconds and gets the headlines. CHAPS carries the property transactions and gets the respect. Bacs carries the *routine* — the salary that arrives on the first of the month and the gas bill that leaves on the eighth — on a schedule so predictable that entire industries have been built on the assumption that it will not change.

That schedule is three working days long. Every technical decision in this chapter follows from it, and so does every operational mistake people make with it. The most common of those mistakes is to believe that "three days" means three days.

This chapter explains what happens on each of those days, what physically travels between the parties, what the six digits of a Service User Number authorise, and why a Monday in late August can push a payroll into September. It ends with the approval regime governing anyone who submits Bacs files on somebody else's behalf.

The plain version

Imagine a very large sorting office in the middle of a town.

Every business in the town — the electricity company, the water company, the factory that pays four hundred wages, the council — needs to move money in and out of thousands of ordinary bank accounts. The electricity company wants to take £61.40 from Mrs Okafor and £48.10 from Mr Reid and similar small amounts from two hundred thousand other people. The factory wants to put £1,842.16 into one worker's account and £2,103.44 into another's.

None of them talks to the banks directly. They all send their lists to the sorting office.

Here is the rule of the sorting office. If your list arrives before half past ten on Monday night, then on Tuesday the office sorts it, and on Wednesday morning the money moves. Not before. Not after. Wednesday, for everybody, at once.

That is the three-day cycle. Day one you hand in the list. Day two the office sorts. Day three the money moves.

What the sorting office actually does on day two

You might think the sorting job is obvious: split the pile into "letters for Barclays", "letters for Lloyds", "letters for Nationwide", and send each bank its bundle. The office does do that. But it does something cleverer as well, and the cleverer thing is the reason the whole arrangement exists.

Suppose that on Tuesday the office finds it is holding instructions worth £4,200,000 that must move from Barclays customers to Lloyds customers, and instructions worth £3,900,000 that must move from Lloyds customers to Barclays customers. It could send two lorries of cash in opposite directions. Instead it does the subtraction, and on Wednesday morning one payment of £300,000 goes from Barclays to Lloyds and the matter is closed.

Eight million instructions have become a couple of dozen numbers. That is what the office is for. It is not a courier. It is an accountant.

A worked example, with real numbers

Barrow Lane Bakery employs fourteen people and pays them monthly. September's pay date is Tuesday 1 September 2026. The total wage bill is £31,420.00.

On Thursday 27 August, some time between seven in the morning and half past ten at night, the bakery's office manager sends one file to the sorting office. It contains fourteen lines. Each has a sort code, an account number, an amount in pence, and a short piece of text that will appear on the employee's bank statement — "BARROW LANE PAY", say. There is also a fifteenth line, which says that £31,420.00 in total is to come *out* of the bakery's own account. That fifteenth line is what makes the file add up to zero.

Nothing happens on Thursday except that the bakery gets a receipt saying the file arrived and looked sensible.

On Friday 28 August the sorting office reads the file, checks it, and works out which bank each of the fourteen employees uses. Fourteen tiny instructions join a national pile of tens of millions. The bakery's money has not moved. The employees have no idea anything is happening.

Then there is a weekend. Then Monday 31 August, the summer bank holiday, and the sorting office is shut.

On Tuesday 1 September, at half past nine in the morning at the very latest and usually a good deal earlier — most people see it before six — the fourteen employees are paid, and £31,420.00 leaves the bakery's account. Both things happen on the same morning. The bakery's money was in the bakery's account, fully usable, right up until then.

Now do the same story backwards. The bakery has a broadband contract at £34.99 a month, and the broadband company collects by Direct Debit. The broadband company sends *its* file on day one; the office sorts on day two; and on day three £34.99 leaves the bakery's account and arrives in the broadband company's. Same three days, same sorting office, same file format. The only difference is which direction the arrow points and who is allowed to draw it.

That is genuinely the whole idea. One rail, two directions. Money out of your account on somebody else's instruction is a Direct Debit. Money into somebody else's account on your instruction is a Direct Credit.

Why anyone puts up with three days

Because the alternative costs more and tells you less.

The broadband company has 220,000 customers on £34.99 a month. Collected by instant transfer, that is 220,000 separate real-time transactions to pay for and 220,000 confirmations to chase, and — crucially — it could not *make* the customer pay. Instant payments are pushed by the payer. Direct Debit is pulled by the biller, on an authority the customer signed once.

And a factory paying four hundred wages does not want them arriving at four hundred slightly different moments. It wants them all on the first, so that its statement shows one line of £742,000, its payroll report shows four hundred lines adding to £742,000, and reconciling the two takes a minute rather than a morning.

Predictability is the product. Three days is the price.

Where the plain version stops being true

The three days are not days. They are English bank working days, which excludes Saturdays, Sundays and public holidays, and the definition is *English* regardless of where in the United Kingdom the money is going. A Scottish bank holiday that is not an English one does not stop the cycle; an English one does, even for money moving between two Scottish accounts. The bakery's payroll above spent 27, 28, 29, 30 and 31 August somewhere between submitted and paid — five calendar days for a three-day cycle. That is not a delay. That is the cycle behaving exactly as specified.

Nothing is "in transit" on days one and two, because there is nothing to be in transit. The sorting-office image quietly suggests that your money is somewhere in the middle of the process, in a van, in a vault, in limbo. It is not. On day one and day two your money is in your account and it is entirely yours. What exists between day one and day three is not money, it is a *record of an intention*. The transfer of value happens once, on day three, and it happens in two different places at once: individual customer accounts are debited and credited at the banks, and separately a much smaller set of net amounts

moves between the banks themselves at the Bank of England. Those are two different kinds of money and two different sets of books, and the technical section separates them properly.

Day three is not the end. The plain version ends when the money lands, which is when most people stop thinking about a payment. Bacs does not stop there. A Direct Debit that lands on day three can be returned unpaid afterwards, and the service user finds out through a report that arrives one or two working days later. A credit sent to a closed account bounces back the same way. And a Direct Debit can be reversed under the Guarantee long after everyone has forgotten it, through an indemnity claim that hits the collecting organisation, not the bank. Any system built on the assumption that entry day is final will eventually produce a reconciliation break that nobody can explain.

"Simultaneously debited and credited" is a phrase to be careful with. Bacs' own published description of the cycle says, for Direct Credits, that on entry day "payments are simultaneously credited to the recipients' accounts and debited from your account", and for Direct Debits that they are "simultaneously debited from the recipients' accounts and credited to your account" — where in the second case the party being debited is the payer, not any sort of recipient. The scheme's plain-language copy blurs the direction, and so does most vendor documentation copied from it. Whenever you read a Bacs description, stop and ask which party is being debited, which is being credited, and which of them initiated the file. Those three answers, not the word "simultaneously", are what distinguish the two products.

The technical version

Who operates what

Bacs is one of the retail payment systems owned and operated by Pay.UK, the company established as the New Payment System Operator in 2018. Bacs Payment Schemes Limited became a wholly owned subsidiary of the NPSO on 1 May 2018, and the NPSO was renamed Pay.UK on 18 October 2018. The scheme itself dates from 1968; Pay.UK reports that over 183.6 billion transactions have been debited or credited to British bank accounts through it since then.

Pay.UK owns the rules. It does not run the machines. The central infrastructure that physically processes Direct Debit and Bacs Direct Credit and maintains the network is operated by Vocalink, now part of Mastercard, under contract to Pay.UK. When practitioners say "Bacs went down" they mean the infrastructure; when they say "Bacs says you must" they mean the rulebook.

Beneath Pay.UK sit the direct participants: the banks and payment institutions that clear and settle Bacs items in their own right. The published Bacs participant list ran to thirty-two institutions when last checked, Pay.UK's page having been updated on 7 April 2026, and it is a more interesting list than it once was. Alongside Barclays, HSBC, Lloyds, NatWest, Santander and Nationwide sit ClearBank, Modulr, Monzo, Starling, PayrNet, The Bank of London and the Bank of England itself. Everyone else reaches Bacs indirectly, through an agency arrangement with a direct participant.

And beneath *them* sit the service users: the organisations that actually send files. A service user is sponsored into the scheme by a payment service provider, which is on the hook for the service user's behaviour. This is the single most important structural fact about Bacs and it has no equivalent on the card rails. You do not sign up to Bacs. You are vouched for.

The three days, precisely

Bacs' own definition, as published, runs as follows.

Day	Name	What happens
1	Input day	The latest day a service user or bureau may submit a payment file for that cycle. Files must be transmitted between 07:00 and 22:30.
2	Processing day	Files are delivered to the recipient PSPs, which then process each payment. Interbank totals are calculated.
3	Entry day	Accounts are debited and credited. Settlement of the net interbank positions takes place.

Several precisions matter.

The input deadline is 22:30 on input day. A transmission completing at 22:31 belongs to the next cycle, whose entry day is a day later. Practitioners run payroll cut-offs hours before this, not minutes.

Input day is the *latest* day of submission, not the only one. Files may be sent up to 30 calendar days ahead of the processing day and held — the scheme term is *warehousing* — until the relevant input day arrives. Most large service users warehouse, because it converts a hard nightly deadline into a soft monthly one. Bacs' glossary consequently describes the processing cycle as having four stages rather than three: arrival, input, processing and entry. Arrival is when your file physically reaches Bacs; input day is when the cycle formally starts for it. For a warehoused file those are different dates.

On input day the service user receives an input report confirming that the file was received and describing what was found in it. Collecting it is not optional in any practical sense: a file that failed validation and was never chased is a payroll that does not run.

On entry day, paying PSPs apply the debits and credits to customer accounts no later than 09:30. For Bacs Direct Credit specifically, Pay.UK states that "it is not possible to guarantee the exact time the payment will reach the recipient's account, although at least 90% of Direct Credit payments are credited to beneficiary account by 06:00am". That 90 per cent figure is the closest thing the scheme offers to a service-level promise on timing, and it is a statistical statement about the population, not a guarantee about your file.

What settles, and in what money

Bacs is a deferred net settlement system. The distinction between clearing and settlement, laid out in Volume I, is load-bearing here and gets skipped constantly.

Clearing is the exchange and sorting of the payment instructions themselves, which happens on day two. Settlement is the movement of central bank money between the direct participants to discharge the obligations that the clearing created, and it happens on day three. Pay.UK's own assessment against the international principles for financial market infrastructures states it precisely: "There is one Bacs settlement cycle each working day at 09:30 and this settles the values calculated at the end of day two." Final settlement occurs at the Bank of England, over the Real-Time Gross Settlement infrastructure, in central bank money.

The netting is multilateral, not bilateral. Each direct participant ends day two with a single number: the net amount it owes to, or is owed by, the system as a whole. At 09:30 on day three those positions are

posted across settlement accounts at the Bank of England, and a system that handled tens of millions of instructions discharges them with a few dozen entries.

The credit risk that netting would otherwise create — the risk that a participant with a large net debit fails between day two and settlement — is removed by prefunding. Pay.UK records that all direct participants hold a cash deposit sufficient to cover their own net transactions in a segregated account at the Bank of England, and that this removes credit risk between direct participants from the settlement process. Bacs is deferred, but it does not have to wonder whether it will be paid.

There is a consequence for the reader who is building something. On entry day your customers' balances change in *commercial* bank money — an increase in what their bank owes them — while the banks square up in *central* bank money at the Bank of England. The two events are related by the rules, not by the same transaction. A payment can be applied to a customer account and appear on a statement before the interbank settlement that funds it has posted. This is normal, and it is one of the reasons prefunding exists.

Direct Debit and Bacs Direct Credit

Two products, one rail, one file format, opposite directions.

	Direct Debit	Bacs Direct Credit
Who submits the file	The collecting organisation (service user)	The paying organisation (service user)
Direction on entry day	Payer's account debited, service user's account credited	Service user's account debited, beneficiaries' accounts credited
Standing authority required	Yes: a Direct Debit Instruction from the payer	No
Typical use	Utilities, insurance, subscriptions, council tax, loan repayments	Payroll, pensions, benefits, supplier payments, expenses, refunds, dividends
Amount	May vary between collections	Set by the payer each time
Consumer protection	The Direct Debit Guarantee	None specific to the scheme
Return mechanism	ARUDD, returned unpaid	ARUCS, returned unapplied
Change advices	ADDACS	AWACS
Transaction codes	01, 17, 18, 19 (and 0N, 0S, 0C for instructions)	99 for credits, with a contra debit

Pay.UK describes a Direct Debit as "an occasional or regular payment instruction set up by the end user in advance, authorising an organisation to collect a pre-agreed amount of money from their account, on a prearranged date", and Bacs Direct Credit as "a regular payment, normally B2C, that pays the beneficiary on a regular date (i.e. first of the month) and is mainly used for paying wages and salaries".

The transaction codes are worth memorising because they carry meaning the amount field does not. Within Direct Debit, code **01** marks the first collection under an instruction; **17** marks a subsequent collection; **18** marks a collection represented after a failure; and **19** marks the final collection under the instruction. Code 19 is not decoration: it tells the paying PSP to delete the instruction from its records, and it may cause a "final collection" message to appear on the payer's statement. Sending 19 by accident cancels the mandate.

Separately, the AUDDIS service carries the instructions themselves rather than money: **0N** lodges a new Direct Debit Instruction electronically at the payer's PSP, **0S** converts an existing manually lodged

instruction to AUDDIS, and **0C** cancels one. AUDDIS is mandatory for all new service users that submit direct to Bacs. It travels on the same three-day cycle in the same file format, which is why a new customer's instruction and their first collection cannot sensibly be sent in the same submission.

The advance notice rule is the other thing to know. A Direct Debit service user must tell the payer, before collecting, the date and amount to be debited. The default notice period recorded against the service user is 10 working days plus postal time, and it can be varied downwards by agreement with the sponsoring PSP. Shorter notice periods are common for utilities and rare for anything unusual. The mandate, the Guarantee and the indemnity mechanism are the subject of the next chapter; here it is enough to say that they attach to Direct Debit and have no counterpart for Direct Credit.

Why payroll and utility billing run on this

Four reasons, in decreasing order of how often they get mentioned and increasing order of how much they matter.

The cheap reason is unit cost. A Bacs item is priced by the sponsoring PSP as a bulk item and a Faster Payment is priced as a transaction. Over 220,000 collections a month the difference is the whole business case.

The better reason is direction. Direct Debit is the only sterling mechanism in common use that lets the person owed money initiate the transfer under a standing authority. Everything else — Faster Payments, CHAPS, Open Banking payment initiation — is a push, which requires the payer to act, which means a proportion of payers will not act. For a utility with millions of accounts, the difference between pull and push is the difference between a collections department and a call centre.

The third reason is that the value date is knowable in advance and identical for everyone in the file. Payroll is not only a payments problem, it is a *fairness* problem: everybody must be paid on the same day, and a system where 3 per cent of staff are paid four hours late is worse than one where everybody is paid at 09:00. Over 150,000 organisations use Bacs Direct Credit, and the reason they do not migrate to instant rails is not inertia; it is that a batch with a fixed value date is the correct shape for the problem.

The fourth reason is reconciliation, and it is the one that gets fintechs killed. A Bacs file is self-balancing. The contra record forces the sum of the payment records to equal the single amount that hits the service user's own bank account, so the bank statement shows one figure that must reconcile to one file. A thousand individual Faster Payments show a thousand statement lines with a thousand chances of a mismatch. Batch is not an old-fashioned way of doing real time. It is a different, and for these purposes better, control structure.

Payroll carries one extra thread, because it explains a field in the file format. Under Real Time Information, HMRC for some years asked employers paying by Bacs to include a hash cross-reference on the Full Payment Submission, built partly from a four-character random string inserted into field 7 of the Bacs payment record, so that HMRC could match reported payroll data to payments actually made. HMRC withdrew the requirement for that field, data item 118, from the 2023/24 tax year onwards. The Service User Number remains reportable on the FPS. The field in the file is still there and payroll software still fills it, which is a small monument to the rule that payment formats accumulate requirements faster than they shed them.

What a Bacs file physically is

The format is called Standard 18, and Bacs' glossary defines it in one line as "the Bacs standard file/record format used by the Direct Debit and Bacs Direct Credit schemes". It is a flat, fixed-width, fixed-length text file. There is no XML, no JSON, no delimiter, no escaping. Every field is found by

counting characters from the start of the line, exactly as it was when these files travelled on magnetic tape.

Two record lengths coexist. The label records that wrap the file are 80 characters. The payment records inside it are 100 characters, and there is a longer 106-character variant that appends a processing date to each record.

A submission is a sandwich:

Record	Length	Purpose
VOL1	80	Volume header. Identifies the submitting party; carries the Service User Number.
HDR1	80	First file header. File identifier, creation date, expiration date.
HDR2	80	Second file header. Declares record format F, block length `02000`, record length `00100`.
UHL1	80	User header label. Carries the Bacs processing day and the work code.
Payment records	100 (or 106)	One per payment, plus one contra.
UTL1	80	User trailer label. Debit and credit totals and counts.
EOF1	80	End of file label, mirroring HDR1.
EOF2	80	End of file label, mirroring HDR2.

The label records are declarative rather than interesting, with the exception of UHL1, which carries the processing date as five Julian digits preceded by a space, in the form `YYDDD`. The work code sits in the same record and reads `1 DAILY`. Bacs' glossary defines the Julian convention exactly: "Julian dates run serially throughout the year. 1 January is always 001 and 31 December is 365 (366 if a leap year)." Thursday 27 August 2026 is Julian 239. Tuesday 1 September 2026 is Julian 244.

The payment record is where the work is. Counting characters from position 1:

Field	Positions	Length	Contents
1	1–6	6	Destination sorting code
2	7–14	8	Destination account number
3	15	1	Destination account type
4	16–17	2	Transaction code
5	18–23	6	Originating sorting code
6	24–31	8	Originating account number
7	32–35	4	Free format
8	36–46	11	Amount, in pence
9	47–64	18	Originating account name
10	65–82	18	Reference
11	83–100	18	Destination account name

Bacs' glossary names fields 9, 10 and 11 individually, which tells you which three the scheme considers most likely to be got wrong. Field 9 is the short name of the service user, up to 18 characters, and is

what may appear on the payer's statement. Field 10 carries the reference, and the glossary calls it "a critical field for both credits and debits allowing the recipient to identify what the payment relates to". Field 11 is the destination account name, and for Direct Debits it "must be the name of the person who is paying the Direct Debit and has authorised the Direct Debit Instruction". Field 7, the four-character free-format field, is the one HMRC borrowed for the RTI hash.

Three properties of this layout deserve emphasis.

The amount has no decimal point and no currency symbol. It is eleven digits of pence, right-justified and zero-filled, so £34.99 is written `00000003499` and £31,420.00 is written `00003142000`. Eleven digits will express any amount below one thousand million pounds; the limits that actually bite are the account limits set by the sponsoring PSP, not the width of the field.

The names and references are eighteen characters and no more. Not eighteen and a continuation, not eighteen displayed characters of a longer stored value. Eighteen. Every organisation that has wondered why its customers' statements say `THAMESIDE PROPERTY` rather than the full company name is looking at field 9 doing exactly what it was built to do in 1968.

And the file must balance. Alongside the payment records sits a contra record, the mirror entry against the service user's own account. In a Bacs Direct Credit file the individual credits carry transaction code **99** and the contra carries a debit code; in a Direct Debit collection file the individual debits carry 01, 17, 18 or 19 and the contra carries **99**. The UTL1 trailer then states total debits, total credits and the counts of each, right-justified and zero-filled. Bacs' glossary defines a file, tersely and correctly, as "a self balancing set of records from a service user". If it does not balance it is not a file, it is a rejection.

The definitive specification is the Service User's Guide and Rules to Bacstel-IP, which the glossary describes as "the technical specification for submission to Bacs". It is issued to sponsored service users by their PSP and is not public, so no publicly available scheme document sets out the internal offsets of the reserved and system fields inside the eighty-character labels. Two things fill the gap, and neither is the scheme speaking. The labels are not a Bacs invention at all: they are the American national standard for the file structure and labelling of magnetic tapes, ANSI X3.27, which is why VOL1 and HDR1 carry serial numbers, generation numbers, block counts and buffer offsets that no payment system has ever had a use for. And several sponsoring banks publish their own Standard 18 message implementation guides, HSBC's February 2022 edition among them, which do give the character position of every reserved field in VOL1, HDR1 and HDR2. Those guides agree with one another on the payment record and very largely on the labels, but they are the bank's document rather than the scheme's, and a sponsor is free to impose requirements of its own or to read a field differently. Take the offsets from the guide your own sponsor hands you, not from any book, including this one.

Pay.UK also publishes an ISO 20022 to Bacs translation guide and a translation specification with XSDs covering all Bacs Standard 18 messages, so that organisations speaking ISO 20022 internally can generate compliant submissions. As of writing, the format on the wire is still Standard 18; ISO 20022 is a translation layer over it, not a replacement.

Service User Numbers

A Service User Number is, in Bacs' own words, "the unique six digit number allocated to organisations authorised to use the Bacs service". Six digits. That is the entire identifier, and it is doing more work than its size suggests.

The SUN is not an account number and it is not a customer reference. It is the scheme's identity for a sponsored originator, and it is what a paying PSP matches against when a Direct Debit arrives claiming authority to take money from one of its customers. The Direct Debit Originator database

holds the record behind it: contact details, advance notice period, dormancy period, AUDDIS status. When a payer telephones their bank to dispute a collection, the SUN is how the bank knows who to point at.

One organisation may hold several SUNs. Large groups routinely run a separate SUN per operating company, per brand, or per application — one for payroll credits, one for subscription collections — because the SUN is the unit against which scheme configuration, reporting and, eventually, blame are recorded. In-house group bureaux, Bacs notes explicitly, may cover "a single organisation that has different Service User Numbers for different applications".

An organisation that cannot obtain its own SUN is not shut out. Under a facilities management arrangement, a registered FM provider acts as the service user, using its own SUN, and collects Direct Debits on behalf of an FM client that has not been sponsored into the scheme. The client's customers see the provider's short name in field 9. Pay.UK maintains a directory of sponsored FM providers, and the arrangement is fundamentally different from a bureau: a bureau transmits your file under *your* SUN, an FM provider collects under *its own*.

The SUN also appears inside the file, in the owner identification area of the VOL1 label. A file whose VOL1 SUN does not match the submitter's credentials does not get processed.

Getting the file to Bacs: Bacstel-IP, smartcards and the Payment Services Website

Bacstel-IP is the channel. The glossary describes it as "a service providing a highly secure access channel into Bacs" using internet technologies and public key infrastructure to allow access to payment services, and notes that it "carries out some online validation of submissions". That last clause is the one practitioners live by: some errors are caught at submission, and the rest are caught later and reported.

Access to Bacstel-IP requires software that Bacs has approved for the purpose, and it requires cryptographic credentials. Direct submitters are issued smartcards, which authenticate approved personnel to the Payment Services Website and sign and send the payment files themselves. Bacs' guidance is strict: cards cannot be transferred between people, a new card must be applied for through the sponsoring PSP when staff change, and the old card must be returned to the PSP to be destroyed. Other staff, and organisations that submit through a bureau, may instead be given a password and ID granting access to the Payment Services Website for reports only, without the ability to sign a submission.

The Payment Services Website is separate from the public bacs.co.uk site, a distinction Bacs itself makes in capitals because people confuse the two. It is where reports and advices are collected. Bacs' own submission checklist reduces to this: submit the file; collect the input report, available on the day of submission; monitor regularly for ADDACS and AWACS advices, ideally daily; and collect any ARUDD and ARUCS reports, which are available within one to two PSP working days after the payment or collection date.

Acting on those advices is a rule, not a courtesy. Service users must action them within three working days. An ADDACS advice saying a customer has cancelled their instruction, ignored for a month, produces a stream of collections that will fail and may produce indemnity claims. If a bureau or software supplier collects the reports for you, you are still the party required to act on them.

Why bank holidays extend the cycle

Bacs defines working days, for scheme purposes, as "English PSP working days excluding Saturdays, Sundays and Public Holidays". The three-day cycle counts in those days and only those days. It follows mechanically that a public holiday does not delay a Bacs payment by one day; it removes one

of the three slots, and whether that costs you one day or four depends entirely on where in the week the holiday falls.

Take the summer bank holiday in England and Wales in 2026, which falls on Monday 31 August.

Wanted entry date	Latest input day (by 22:30)	Processing day	Entry day
Friday 28 August	Wednesday 26 August	Thursday 27 August	Friday 28 August
Tuesday 1 September	Thursday 27 August	Friday 28 August	Tuesday 1 September
Wednesday 2 September	Friday 28 August	Tuesday 1 September	Wednesday 2 September

Read the middle row carefully, because it is the one that catches people. A payroll with a value date of Tuesday 1 September 2026 must be submitted by 22:30 on **Thursday 27 August**, not Friday. The Friday is the processing day. An organisation that has internalised "submit two days before" as "submit on the Friday for the Tuesday" will find its staff unpaid, and will find out on Tuesday morning, which is the worst possible moment to find out.

Note also that there is no entry day at all on Monday 31 August. Pay.UK's rule for Bacs Direct Credit is that payments can be sent Monday to Friday excluding bank holidays, and "if the prearranged date falls on a weekend or bank holiday, the payment is made on the next working day". A salary contractually payable on the last day of August 2026 is paid on 1 September.

Christmas compresses this further. In England and Wales in 2026 both Christmas Day, Friday 25 December, and the Boxing Day substitute, Monday 28 December, are holidays, and New Year's Day 2027 falls on the Friday. A payment dated Thursday 31 December 2026 requires input on Tuesday 29 December, and the next available entry day after that is Monday 4 January 2027.

For exactly this reason Pay.UK publishes an annual Bacs processing calendar, in landscape and portrait, marking non-processing days and giving the Julian date for every day of the year. The 2026 calendar's legend distinguishes non-input and non-processing days from days that are a banking holiday in Northern Ireland only, and carries a standing instruction to consult your payment service provider for entry date and processing day arrangements for accounts held in Northern Ireland. That footnote is the honest admission that "English bank working days" is a simplification of a United Kingdom with four sets of holidays, and that where it matters you ask your sponsor rather than a calendar.

If you build software that schedules Bacs submissions, hard-code nothing. Load the processing calendar as data, refresh it annually, and treat "is this a Bacs processing day" as a lookup rather than a computation. Every engineer who has tried to derive English bank holidays algorithmically has eventually met the substitute day rule, the royal proclamation, or a one-off national holiday, and has lost.

Bacs approved bureaux and Bacs approved software

Most organisations that use these schemes never submit anything themselves. Pay.UK states that over half of those making Direct Debit, Bacs Direct Credit and Faster Payment submissions do so through approved bureaux rather than directly, and that there are approximately 700 approved bureaux worldwide.

A bureau is defined by the scheme as an organisation that sends payments to Bacs on behalf of another organisation. Pay.UK distinguishes commercial bureaux, which submit on behalf of independent third parties; PSP bureaux, offering participant services direct to service users; in-house commercial group

bureaux, working solely within their own group; and a residual "other" category. The distinction that matters for approval is between in-house and commercial: an in-house bureau submitting only for entities within the same legal organisation need not register, while a commercial bureau submitting for third-party service users that are separate legal entities must achieve Bacs Approved Bureau status. Pay.UK must approve any organisation that submits Direct Debits, Bacs Direct Credits or Faster Payments transactions on behalf of third parties.

The route in runs through a sponsor. An applicant must first be authorised by its sponsoring participant; Pay.UK's Bureau Inspection team then formalises the process; a one-off registration fee is payable; a questionnaire, supporting documents and a signed agreement follow; an inspection must be passed; and only then does Pay.UK issue a certificate and the sponsor enable the live service. An application not completed within 12 months expires.

Approved bureaux are tiered by annual transaction volume, and the tier sets the inspection cadence.

Tier	Annual transaction volume	Cadence over four years
A	Over 5 million	Full (on-site), interim (on-site), full (on-site), interim (on-site)
B	250,000 to 5 million	Full (on-site) in years 1 and 3; Annual Compliance Statement in years 2 and 4
C	25,000 to 250,000	Full (remote) in year 1; ACS in years 2 and 3; full (remote) in year 4
D	Under 25,000	As tier C

An inspection assesses end-to-end payment processing, from the receipt of Bacs-related information through to the point of transmission to the Bacs clearing system. The questionnaire is structured around ISO/IEC 27001 and covers eight areas: organisation and policy; professional services and commercial arrangements; physical security; network environment; systems management; logical access control; business continuity and disaster recovery; and Bacs processing and data controls. Two annexes attach where relevant: Annex A for third-party risk management where outsourcing is material, and Annex B for hardware security module and cryptographic key management.

An on-site visit typically runs five to seven hours: a business overview, a tour of processing and server areas, a walk through the questionnaire responses with the staff who own them, observation of operations, and review of supporting documents — security policies, process guides, customer agreements, business continuity plans.

Each category is rated Pass, Pass with requirements, Fail, or Not Applicable. A Fail in any single category fails the whole inspection, and a failed bureau must be re-inspected within a maximum of six months. Requirements are mandatory and time-bound; recommendations are best practice, and one ignored may be restated as a requirement next time. The report follows within five working days. Cancelling an inspection with less than ten working days' notice attracts a fee. Annual fees are levied in April at bureau number level, based on the transactions submitted during the previous calendar year, with new bureaux exempt for their joining year; invoices are payable within 30 calendar days. Terminating BAB status requires three months' notice.

Approved bureaux receive the BAB logo. Pay.UK is careful about what it means: bureaux awarded it "have been inspected and have met appropriate standards of business, security, technical and operational competence at the time of the inspection", status "is based upon point in time inspections", and

Pay.UK "does not make any representation in respect of the suitability, or otherwise, of organisations approved under the scheme for any purpose". It is an attestation about a date, not a warranty.

Running in parallel is the Bacs Approved Software Service. A Bacs approved software supplier is a developer and supplier providing software, hardware and consultancy services to Bacs Payment System customers to enable them to use the service. Testing under BASS examines authentication of parties communicating with Bacstel-IP or Secure-IP; secure communication, meaning confidentiality and integrity, between those channels and the approved software; correct formatting of submissions and reports; handling of errors and warnings; and trademark usage. Approval requires an application, mandatory testing, training for named staff on the Bacs Payment System and the relevant interfaces, an approval fee and an ongoing annual maintenance fee. Full business and technical specifications, including a sorting code directory, are released to applicants.

The two schemes answer different questions. BASS asks whether a piece of software constructs and transmits submissions correctly. The Bacs Approved Bureau Scheme asks whether an organisation can be trusted to operate the process around that software — who holds the smartcards, what happens when the building is unavailable, how third-party clients are onboarded and offboarded, whether someone is reading the ADDACS advices. A vendor can hold approved software status and not be a bureau; a bureau can be approved while running somebody else's approved software. An organisation intending to do both applies to both, and should expect the bureau inspection to look hardest at the questionnaire areas that have nothing to do with code.

The scale of it, as of writing

Measure	Figure	Period
Bacs payments processed	6.86 billion	2025
Value processed	£6.05 trillion	2025
Direct Debits	5.0 billion, an annual high	2025
Transactions since inception	183.6 billion	1968 to date
Direct Debits over five decades	Over 113 billion	to date
Organisations using Bacs Direct Credit	Over 150,000	as published
Direct participants	32	list updated 7 April 2026
Approved bureaux	Approximately 700	as published

Set that against the rails on either side. Faster Payments settles in seconds, runs 24 hours a day, and carries a maximum of £1 million per transaction under the central scheme limit, with individual institutions imposing lower limits of their own. CHAPS settles same-day, irrevocably, gross, one payment at a time across the Bank of England's RTGS system, with an early-afternoon cut-off for the same-day guarantee, and carries the large majority of sterling value on a tiny fraction of the volume. Bacs sits between them and does something neither can: it moves enormous numbers of small, scheduled, two-directional payments at a known value date, cheaply, in a format that reconciles itself.

Three days is what that costs. It has been three days since 1968, and the reason it is still three days is not that nobody has thought about it. It is that the three days buy the netting, the netting buys the cost, and the cost is why your salary and your gas bill both arrive without anybody thinking about them at all.

35.98 Common wrong ideas

Wrong: three days means three days. **Right:** it means three English bank working days, so a file submitted on Thursday 27 August 2026 enters accounts on Tuesday 1 September — five calendar days later, and exactly as specified.

Wrong: a Scottish or Northern Irish bank holiday moves the cycle. **Right:** the definition is English working days regardless of where in the United Kingdom the money is going, and Pay.UK's calendar tells you to ask your sponsor about accounts held in Northern Ireland.

Wrong: the money is somewhere in the middle of the process on days one and two. **Right:** nothing is in transit, because what exists between input and entry is a record of an intention, and value moves once, on day three.

Wrong: "submit two days before" is a safe rule of thumb. **Right:** for entry on Tuesday 1 September 2026 the file must be in by 22:30 on Thursday 27 August, because the Friday is the processing day and the Monday is a holiday.

Wrong: entry day is the end of the payment. **Right:** Direct Debits come back through ARUDD and credits through ARUCS one or two working days later, and a collection can be reversed by indemnity claim long after everyone has forgotten it.

Wrong: input day is the day you send the file. **Right:** input day is the latest day you may send it, files may be warehoused up to thirty calendar days ahead, and the scheme's own glossary therefore counts four stages rather than three.

Wrong: Pay.UK runs the Bacs machines. **Right:** Pay.UK owns the rulebook and Vocalink, now part of Mastercard, operates the central infrastructure under contract, so "Bacs went down" and "Bacs says you must" are statements about different organisations.

Wrong: you sign up to Bacs. **Right:** you are vouched for, because a service user is sponsored into the scheme by a payment service provider that is on the hook for its behaviour.

Wrong: the 90 per cent by 06:00 figure is a promise about your file. **Right:** it is a statistical statement about the population of Direct Credit payments, and the only commitment is that paying PSPs apply entries no later than 09:30.

Wrong: the Standard 18 offsets in a vendor or bank guide are authoritative. **Right:** the definitive specification is the Service User's Guide and Rules to Bacstel-IP, issued by your sponsor and not public, and a sponsor may impose requirements of its own.

35.99 Chapter summary in 20 lines

1. Bacs has run since 1968, moved 6.86 billion payments worth £6.05 trillion in 2025, and pays most of the salaries and collects most of the household bills in Britain.
2. It behaves like a sorting office: lists arrive by 22:30 on day one, are sorted on day two, and money moves on day three for everybody at once.
3. The clever part of day two is not sorting but subtraction, because millions of instructions are netted down to a few dozen interbank figures.
4. One rail carries two products in opposite directions: Direct Debit pulls under a standing authority, and Bacs Direct Credit pushes.
5. The three days are English bank working days, so a payroll dated Tuesday 1 September 2026 must be submitted by 22:30 on Thursday 27 August.

6. Nothing is in transit during the cycle, because the payer's money stays in the payer's account and fully usable until entry day.
7. On entry day two different things happen in two kinds of money: customer accounts move in commercial bank money, and the netted interbank positions settle at 09:30 in central bank money at the Bank of England.
8. Prefunding removes the credit risk that deferral would otherwise create, because every direct participant holds a segregated cash deposit covering its own net position.
9. Entry day is not the end, since ARUDD, ARUCS, ADDACS and AWACS all arrive afterwards and an indemnity claim can reverse a collection much later still.
10. Pay.UK owns the rules, Vocalink runs the infrastructure, thirty-two direct participants clear and settle, and everyone else reaches Bacs through an agency arrangement or a sponsor.
11. Input day is the latest day of submission rather than the only one, and warehousing up to thirty calendar days ahead is why the glossary describes arrival, input, processing and entry.
12. Standard 18 is a flat, fixed-width text file inherited from magnetic tape, with 80-character labels wrapped around 100-character payment records and no delimiters of any kind.
13. Every field is found by counting characters: amounts are eleven digits of pence with no decimal point, and names and references are eighteen characters and no more.
14. The file must balance, because a contra record mirrors the payment records against the service user's own account, which is why one bank statement line reconciles to one file.
15. Transaction codes carry meaning the amount field does not — 01, 17, 18 and 19 for collections, 99 for credits, and 0N, 0S and 0C for AUDDIS instructions.
16. A Service User Number is six digits, is the scheme's identity for a sponsored originator, and is what a paying bank matches against and eventually attributes blame to.
17. Files travel over Bacstel-IP using approved software and non-transferable smartcards, while reports are collected from the Payment Services Website and must be actioned within three working days.
18. Bank holidays do not delay the cycle by a day; they remove a slot, and where the holiday falls decides whether that costs one day or four.
19. Anyone submitting for third parties must hold Bacs Approved Bureau status, inspected on a cadence set by volume tier, and a Fail in any single category fails the whole inspection.
20. Three days is what the netting costs, the netting is what makes the payments cheap, and the cheapness is why a salary and a gas bill both arrive without anybody thinking about them.

Sources: Pay.UK and Bacs published scheme material (bacs.co.uk processing cycle, glossary, processing calendar 2026 and The Little Bacs Book; wearepay.uk Bacs Payment System pages, participant list, Bacs Approved Bureau Scheme supporting guidelines and Bacs Approved Software Service), Pay.UK's PFMI self-assessment report, the Bank of England's payment and settlement pages, GOV.UK bank holidays, and CIPP guidance on the withdrawal of the RTI Bacs hash. Standard 18 record layouts cross-checked against independent implementation documentation; the authoritative specification is the Service User's Guide and Rules to Bacstel-IP, issued by sponsoring PSPs. All timings, limits and figures accurate at time of writing.

36

CHAPTER

The Mandate

36.0 What this chapter gives you

1. You will be able to say what a Direct Debit Instruction legally is, and why it is an authority given to the service user rather than a contract with the payer's bank.
2. You will be able to state what the Direct Debit Guarantee actually promises, who makes the promise, and who ends up paying for it.
3. You will be able to trace the money in a disputed collection from the payer's immediate refund, through the paying bank, to the indemnity claim that debits the service user's own account.
4. You will be able to count an advance notice period correctly and say what the notice must contain if it is to win a challenge.
5. You will be able to choose between paper, AUDDIS and Paperless Direct Debit for setting up a mandate, and say what obligations each route carries.
6. You will be able to read a DDICA reason code and say whether the claim is about notice, cancellation, authority or the name on the statement.
7. You will be able to work out the challenge deadline from the day a claim is reported, and say what happens at the end of working day nine and working day eleven.
8. You will be able to explain why the Guarantee is faster and broader than the Payment Services Regulations 2017, and where regulation 79(3) does and does not apply.
9. You will be able to explain why cancelling a Direct Debit cancels nothing else, and why a refunded payer may still owe the money.
10. You will be able to name the design choices that keep a collections product alive: storing the notice document, actioning ADDACS within three working days, treating day nine as code rather than a runbook, and putting the trading name in field 9.

Every other payment in this book is something you *do*. You tap a card, you press send, you hand over a note. A Direct Debit is something you *stop doing*. You do it once, and thereafter the money leaves without you touching anything, on dates you may not remember, in amounts you did not choose, for as long as neither side gets round to stopping it.

That is an extraordinary thing to hand to a stranger, and roughly five billion times a year in the United Kingdom people hand it over without a second thought. In 2025, Bacs processed 5.03 billion Direct Debits — a record for the system — as part of 6.86 billion Bacs payments worth £6.05 trillion in total. Nine in ten UK adults with a bank account have at least one Direct Debit running.

They do it because of a single short document that almost nobody reads: the Direct Debit Guarantee, five bullet points long, printed on the tear-off half of every mandate form and reproduced on every biller's website. It is the most generous consumer protection attached to any sterling payment rail.

It is more generous than the statutory rights that sit underneath it. And it is generous in a way that costs the person collecting the money, not the person paying it, which is why it works and why it is expensive to abuse.

The previous chapter covered the three-day cycle that carries the collection. This chapter covers the authority that permits the collection in the first place: what a Direct Debit Instruction legally is, how one is created on paper, by file, or over the telephone, what notice must be given before money moves, what happens when the payer objects, and — the question that decides whether a collections business survives — who ends up out of pocket.

The plain version

Imagine you have an account at the corner shop.

The old way of settling it was that you walked in every month and paid what you owed. That works, but it means the shopkeeper has to chase you when you forget, and you have to remember. So you try something else. You write a short note and give it to the shopkeeper. The note says: *take what I owe you, out of my money-box, on the first of every month*. You sign it. The shopkeeper keeps it.

Two things about that note matter more than anything else in this chapter.

The first is that the note does not say an amount. You have not said "take £30". You have said "take what I owe you". The shopkeeper decides how much, and when, within whatever you agreed about the goods. That is what makes the arrangement useful — your bill genuinely does change month to month — and it is also what makes it frightening.

The second is who the note is addressed to. It looks as though you are giving permission to the shopkeeper. What you are actually doing is telling your *bank* that the shopkeeper is allowed to reach in. The bank is the one holding the money-box. It will honour requests from anybody you have named on a note like this, and it will not ring you up first.

The promise that makes it safe

If that were the whole arrangement, nobody sensible would sign the note. So there is a second document, and it is the reason Direct Debit is trusted in Britain in a way that no equivalent arrangement is trusted in most other countries.

Your bank makes you a promise. It is short enough to say in full, and it is worth saying in full, because the exact words are the whole point:

If there are any changes to the amount, date or frequency of your Direct Debit, the organisation will notify you — normally ten working days — in advance of your account being debited, or as otherwise agreed. If an error is made in the payment of your Direct Debit, by the organisation or by your bank or building society, you are entitled to a full and immediate refund of the amount paid from your bank or building society. If you receive a refund you are not entitled to, you must pay it back when the organisation asks you to. You can cancel a Direct Debit at any time by simply contacting your bank or building society.

Read the second sentence again, slowly. *You are entitled to a full and immediate refund from your bank.* Not from the shopkeeper. Not after an investigation. Not if the bank agrees with you. You go to your own bank, you say an error has been made, and the money comes back.

There is no minimum amount. There is no stated deadline in the promise itself. There is no requirement to argue with the shopkeeper first. The bank refunds you and then goes and sorts it out with the shopkeeper afterwards, at its own inconvenience.

Where the money actually comes back from

Here is the part people miss. The bank refunds you out of its own till, but it does not stay out of pocket. It turns round and takes the money back off the shopkeeper — directly out of the shopkeeper's own bank account, without asking the shopkeeper first.

So the sequence is: shopkeeper takes £39.99 from you; you object; your bank immediately gives you £39.99; your bank then debits £39.99 from the shopkeeper. The shopkeeper can dispute that, with evidence, within a short window. If the shopkeeper does nothing, the money simply goes.

That is the entire economics of the Direct Debit Guarantee in one paragraph. The person collecting the money carries the risk of being wrong. Not the payer, and not, in the end, the bank.

A worked example, with real numbers

Northgate Leisure runs a gym. It collects £34.99 a month by Direct Debit from 4,100 members on the first working day of each month.

In September 2026 it decides to raise the price to £39.99 with effect from the October collection, which falls on Thursday 1 October 2026.

The Guarantee's default notice period is ten working days. Counting back ten working days from Thursday 1 October gives Thursday 17 September, and the notice has to *arrive* by then, so in practice Northgate posts and emails its members in the second week of September. The notice says the new amount, £39.99, and the date, 1 October, and it says who is collecting it in the name members will actually recognise on their statements.

On 1 October, 4,100 collections of £39.99 land. Total: £163,959.

One member, Mr Hale, does not remember any letter. On Monday 5 October he telephones his bank and says the gym has taken the wrong amount. His bank does not ring the gym. It does not ask him for the letter. Under the Guarantee it credits £39.99 back to his account that day.

His bank is now £39.99 down, so it raises an indemnity claim against Northgate Leisure. The claim reaches Northgate through a report on Tuesday 6 October. That day counts as working day one. Northgate has until the end of working day nine — Friday 16 October 2026 — to challenge it, and the only thing that will win the challenge is a copy of the advance notice showing the amount, the date and the address it was sent to.

If Northgate can produce it, the claim is cancelled and Northgate keeps the £39.99. If Northgate cannot produce it, or simply does not act in time, £39.99 comes out of Northgate's bank account and Mr Hale keeps his refund.

And here is the sting in the tail: Mr Hale still owes the gym for October. The refund undid the *payment*. It did not undo the *membership*.

Why anybody collects money this way

Given all that, why would Northgate use Direct Debit at all?

Because the alternative is worse. Cards expire, get reissued and get declined, and every failure needs a human being to chase it. With Direct Debit, Northgate knows on a fixed date that £163,959 will arrive

in one lump, it reconciles only the exceptions, and the bill collects itself for years without either party doing anything. The Guarantee is the price of that, and for a business genuinely giving correct notice and taking correct amounts the price is low, because most claims never happen: most collections are right.

The Guarantee is not a tax on collecting money. It is a tax on collecting money carelessly.

Where the plain version stops being true

The note is not addressed to your bank, and the Guarantee is not law. The plain version says you are telling your bank that the shopkeeper may reach in. Legally, a Direct Debit Instruction is an authority given by the payer *to the service user* — Bacs' own definition is "an authority (paper, AUDDIS or PDD) from the customer allowing the service user to collect Direct Debit payments from their account" — and the Bacs guidance for anyone producing a mandate states flatly that "the DDI is not evidence of any contract between you and your customer's payment service provider". A copy is lodged with the paying bank so that it knows the collection is expected, but the mandate is not an agreement with that bank. Nor is the Guarantee a statute. It is a scheme rule: a promise that Pay.UK's Bacs rulebook obliges every participating bank and building society to offer, enforced through contracts between scheme participants rather than through an Act of Parliament. Underneath it sit genuine statutory rights, in the Payment Services Regulations 2017, and those rights are in several respects *narrower* than the Guarantee. The technical section sets them side by side.

"Immediate" describes one side of the transaction only. The payer's refund is immediate. Everything on the other side runs on working-day clocks: the claim reaches the collecting organisation through a batch report, the challenge window is counted in English bank working days, and the paying bank's response has its own cut-off. A payer refunded on a Monday afternoon may be looking at a business whose money does not actually move for another fortnight. Cash-flow models that treat indemnity claims as instantaneous are wrong in both directions.

Almost no modern mandate is signed. The corner-shop image has a signed note, kept by the shopkeeper. In 2026 most new Direct Debits are set up over a website or a telephone with no signature anywhere in the process — Paperless Direct Debit — and the mandate travels to the payer's bank as an electronic record through AUDDIS. That shifts a large burden. When there was a signature, the paying bank checked it. When there is not, the collecting organisation is responsible for establishing that the person giving the account details is entitled to give them, and it carries the loss if they were not. Modulus checking, which most systems do and which the scheme recommends, confirms only that a sort code and account number *could* exist together. It says nothing whatever about who owns the account.

Cancelling the Direct Debit does not cancel anything else. Telling your bank to stop paying the gym stops the payments and nothing more. The membership contract survives, the debt keeps accruing, and the gym is entitled to pursue it. The Guarantee is explicit that it does not affect the contract between payer and biller: you can claim the money back and still owe it. In the other direction, a mandate that nobody uses does not last forever either — paying banks remove dormant instructions after a set period, so a service user that stops collecting for a year or more may find the authority has quietly disappeared.

The Guarantee covers the mechanics, not the merits. It protects you if the amount, date or set-up was wrong. It does not protect you if the gym is terrible, if the goods never arrived, or if the company goes into administration owing you nine months of prepaid membership. Those are contract disputes, and the route for them is the biller, then the biller's complaints process, then the courts — not the bank.

Section 75 of the Consumer Credit Act and card chargebacks, which do reach into the merits, do not apply to Direct Debit at all.

The technical version

The parties, and what each of them signs

Five roles matter, and confusing any two of them produces most of the bad documentation on this subject.

The **payer** is the account holder whose account is debited. The **service user** is the organisation collecting, identified by a six-digit **Service User Number** (**SUN**) allocated by its bank. The **paying payment service provider** (**paying PSP**) holds the payer's account, lodges the instruction on its records, applies the collection and, critically, is the party that owes the Guarantee. The **sponsoring PSP** is the bank that authorises the service user to submit to Bacs at all, vets it, sets its limits and stands behind it. And **Pay.UK** is the scheme operator and rule-maker, with Vocalink as the infrastructure provider that physically processes the files.

The service user's own signature appears on a document the payer never sees: the **indemnity**. Bacs defines it as "a document under the Direct Debit Scheme that contains a legally binding undertaking to make payment to any paying PSP in response to an indemnity claim", and adds that "an indemnity, in standard form, is an essential requirement of the Direct Debit Scheme". That sentence is the legal engine of everything in this chapter. The Guarantee is a promise the paying bank makes to the payer; the indemnity is a promise the service user makes to every paying bank in the country, in advance, that it will make good whatever the Guarantee costs them.

Against the SUN, the scheme maintains a record on the **Direct Debit Originator database** (**DDO**): contact details, AUDDIS status, dormancy period, and the service user's agreed advance notice period. That last field is not decorative. It is the number a paying bank looks at when deciding whether a claim about insufficient notice is well founded.

The mandate document

The Direct Debit Instruction is a controlled document. Bacs publishes ten standard templates in A4, A5 and small formats, with and without an "official use" box, and the rule is that all DDIs must follow all of the wording and the order of the information in those templates. The design must be approved in writing by the sponsoring PSP before use, and again after any change. The Direct Debit logo is mandatory on every DDI.

The instruction half of the standard form carries these fields: the service user's name and address, the six-digit Service User Number, the name or names of the account holder or holders, the branch sort code, the bank or building society account number, the name and full postal address of the payer's bank or building society, a reference, a signature and a date. The operative sentence is fixed:

Please pay [*service user*] *Direct Debits from the account detailed in this Instruction subject to the safeguards assured by the Direct Debit Guarantee. I understand that this Instruction may remain with* [*service user*] *and, if so, details will be passed electronically to my Bank/Building Society.*

The Guarantee sits on a detachable portion, headed with the instruction that it "should be detached and retained by the payer".

Several operational rules follow from the form and are routinely broken by first-time service users. The customer must not add amount or date restrictions to the mandate; if they have, the mandate is not valid as submitted. The mandate may only be lodged with the paying PSP by the service user,

never posted in by the customer. Tear-off slips carrying extra marketing or account information must be detached before the form is sent onward. And when a mandate is submitted electronically the amount field must be zero — every Direct Debit Instruction is for an unlimited amount, and a non-zero value produces an AUDDIS rejection with reason code 7, "DDI amount not zero".

Three ways to create a mandate

Non-AUDDIS (paper). The historic method. The payer signs a paper DDI, the service user sends the physical form to the payer's bank branch or to that bank's central lodgement address, and the bank keys it in. Slow, error-prone, and shrinking. Pay.UK has announced that from 1 July 2027 no new non-AUDDIS Service User Numbers will be created by PSPs. Existing non-AUDDIS SUNs may continue to lodge new paper instructions and to collect against existing ones, but any SUN created after that date will be electronic only. Note also that a service user cannot mix the two on one number: separate SUNs are required for AUDDIS and non-AUDDIS instructions.

AUDDIS (Automated Direct Debit Instruction Service). The service user keeps the signed original and transmits the mandate details electronically through Bacs to the paying PSP, which validates them and, if it accepts, sets up the instruction on its database. AUDDIS is mandatory for all new service users that submit directly to Bacs. Three transaction codes carry the traffic:

Code	Meaning
0N	New Direct Debit Instruction lodged
0S	Conversion of an existing manually lodged DDI to AUDDIS
0C	Cancellation of an existing DDI by the service user

Rejections come back through the Bacs messaging service and appear on the AUDDIS Bank Rejected Direct Debit Instruction Report. Bacs states that notification of any rejected instruction will reach the service user within five working days of the instruction being sent to the paying PSP. Some failures — a failed modulus check, for instance — are caught by Bacs before the file ever reaches the paying bank and are returned immediately.

Paperless Direct Debit (PDD). Sign-up by telephone, website, telephone keypad, face to face or interactive television, with no signature at any point. PDD is available only to service users who already have live AUDDIS status and who satisfy additional criteria, and it carries a distinct set of obligations:

- The service user, not the bank, is responsible for verifying the customer's identity, address and account details. Bacs publishes a *Direct Debit verification measures* document listing acceptable methods.
- Modulus checking at the point of sale is strongly recommended. It does not verify that the account belongs to the payer.
- Every telephone script, internet screen, confirmation letter and item of supporting material must be approved by the sponsoring PSP before use, and again after any change.
- Two members of staff must successfully complete the mandatory Paperless Direct Debit training course.
- One type of application per SUN is recommended, so that different sign-up channels can be monitored and controlled separately.
- Voice recording of telephone sign-ups is recommended.
- Service users may be subject to annual review of their PDD processes, and failure to comply with the scheme rules may result in removal of sponsorship and exclusion from the Direct Debit Scheme altogether.

PDD carries its own confirmation rule, and it is one of the most frequently misquoted requirements in the scheme. The service user must send confirmation to the customer within three working days of the verbal or internet instruction. Alternatively — and only if the first collection is made within a month of the Direct Debit being set up — the confirmation may incorporate the advance notice, in which case it must be received by the customer no later than the service user's advance notice period before the first collection.

Two further timing rules apply to all mandates regardless of channel. All Direct Debit Instructions should be submitted to the paying PSP within ten working days of the customer signing, even if the first collection is months away. And if a paying PSP asks to see a copy of a mandate, the service user must provide it within seven working days of the request; failure to do so exposes the service user to liability for every Direct Debit collected under that instruction. For a PDD service user there is no signed copy to produce, which is precisely why a disputed authority nearly always ends in a refund.

Mandates also expire from disuse. Bacs' published material is not internally consistent here, and practitioners should check the current rulebook rather than rely on either figure quoted second-hand: the Bacs glossary describes a default dormancy period of thirteen months, noting that this was temporarily increased to twenty-four months to limit the impact of pandemic payment holidays, while the Bacs FAQ states that paying PSPs hold DDI details for a minimum of twenty-four months from receipt or last payment. An extended dormancy period may be applied for through the sponsoring PSP at the point of joining AUDDIS. Whichever number applies, the operational consequence is the same: a service user collecting annually, or seasonally, must know its dormancy setting or its mandates will evaporate between collections.

Advance notice

Advance notice is the single most important operational obligation in the scheme, because it is the obligation that most indemnity claims allege was breached.

The default period, in the absence of any other agreement, is a minimum of ten working days plus postal time. Working days for scheme purposes means English bank working days, excluding Saturdays, Sundays and public holidays, regardless of where in the United Kingdom either party sits. The period is recorded against the SUN on the Direct Debit Originator database, and it can be shortened by agreement with the payer — many subscription businesses run on three or five working days, stated in their terms — but the shortened period must be agreed, documented and reflected on the DDO, not simply adopted.

Notice is required in two circumstances. Before the first collection under a new mandate. And before any change to the amount, date or frequency of an existing one. Note what is not on that list: notice is not required before an unchanged, already-notified collection, which is why an annual notice covering a fixed schedule is sufficient for a fixed-amount arrangement.

The notice must reach the payer — the account holder, not a spouse, not an account administrator, not a purchasing department that happens to have signed the contract. It must state the amount and the date. It must identify the service user in the name the payer will actually see on the bank statement, which for organisations trading under a different name from their registered name means using the trading name that appears in field 9 of the Bacs record. And the notice document itself, like every other piece of Direct Debit literature, requires written approval from the sponsoring PSP before use.

Where the Guarantee says "or as otherwise agreed", it is also worth noting the sentence that follows in the published wording: if the payer *requests* the organisation to collect a payment, confirmation of the amount and date is given at the time of the request rather than in advance. That is the clause that

makes one-off, payer-initiated collections workable, and it is the clause that subscription businesses most often try to stretch beyond its meaning.

Failures of advance notice surface in three different message streams, which is a good indication of how seriously the scheme takes it: a collection returned unpaid with ARUDD reason code 4, "advance notice disputed"; a mandate amended or cancelled with ADDACS reason code D, "advance notice disputed"; and indemnity claims under DDICA reason codes 1 and 2. Bacs publishes best-practice material specifically aimed at reducing indemnity claim volumes, and clear, timely, correctly addressed advance notice is the first item on it.

The Guarantee, clause by clause

The authoritative text, as published by Bacs and reproduced on every compliant mandate, is five bullets. Taken one at a time:

"This Guarantee is offered by all banks and building societies that accept instructions to pay Direct Debits." The promisor is the payer's own bank, not the biller and not Pay.UK. A payer never has to deal with the collecting organisation to obtain a refund, and a bank cannot decline on the basis that it disagrees with the payer's account of events.

"If there are any changes to the amount, date or frequency of your Direct Debit [*the organisation*] *will notify you* (*normally 10 working days*) *in advance of your account being debited or as otherwise agreed. If you request* [*the organisation*] *to collect a payment, confirmation of the amount and date will be given to you at the time of the request."* The notice obligation, restated to the payer as a right. The parenthesis is why "ten working days" is universally quoted as the rule despite being a default rather than a fixed requirement.

"If an error is made in the payment of your Direct Debit, by [*the organisation*] *or your bank or building society, you are entitled to a full and immediate refund of the amount paid from your bank or building society."* The operative promise. Note the breadth: it covers errors by either party, it is full rather than partial, and it is immediate. There is no stated time limit, no minimum value, and no condition that the payer exhaust the biller's complaints process first.

"If you receive a refund you are not entitled to, you must pay it back when [*the organisation*] *asks you to."* The clawback. Practically, this is the clause that makes the whole design tolerable to service users: a refund obtained wrongly is recoverable as a debt, and a service user with good records can pursue it.

"You can cancel a Direct Debit at any time by simply contacting your bank or building society. Written confirmation may be required. Please also notify [*the organisation*]*."* Cancellation is unilateral, immediate and requires no justification.

How that compares with the statutory position

It is worth being precise about the relationship between the Guarantee and the Payment Services Regulations 2017 (SI 2017/752), because a good deal of published commentary treats them as the same thing.

Under regulation 67, a payment transaction is authorised if the payer has consented to its execution; a Direct Debit Instruction is that consent, given in advance and for an unspecified amount.

Regulation 79 then gives the payer a right to a refund from its own PSP for an authorised transaction initiated by or through the payee, but only where two conditions are both met: the authorisation did not specify the exact amount, and the amount exceeded what the payer could reasonably have expected given previous spending, the framework contract and the circumstances. Regulation 80 requires the

payer to make that request within eight weeks of the debit, and requires the PSP either to refund or to justify its refusal within ten business days.

Regulation 79(3) does provide an *unconditional* refund right, but it applies to euro direct debits within the scope of Regulation (EU) 260/2012 — the SEPA regulation — and not to sterling Bacs Direct Debits. This is the point most often got wrong: for a sterling Direct Debit, the unconditional right comes from the scheme Guarantee, not from statute.

Regulation 74 sets the long-stop: a payment service user loses the right to redress for an unauthorised or incorrectly executed transaction unless it notifies its PSP without undue delay and in any event no later than thirteen months after the debit date. Where a transaction genuinely was unauthorised — a mandate the payer never gave — regulation 76 requires the PSP to refund as soon as practicable and in any event no later than the end of the business day following the day it becomes aware, subject to a suspected-fraud exception.

Set side by side:

	Direct Debit Guarantee	PSRs 2017
Source	Bacs scheme rules (Pay.UK)	SI 2017/752
Trigger	Any error in the payment of a Direct Debit	Unauthorised transaction, or authorised but unexpectedly large
Conditions	None stated	Reg 79(2) reasonableness test for authorised transactions
Speed	Full and immediate refund	Reg 76: end of next business day (unauthorised); reg 80: 10 business days to refund or justify
Deadline to claim	None stated	Reg 80: 8 weeks; reg 74: 13 months long-stop
Who refunds	Payer's own bank or building society	Payer's own PSP

The practical consequence is that a UK Direct Debit payer almost never needs to invoke the statute. The Guarantee is faster, broader and unconditional, and banks apply it as a matter of course. Where a bank refuses, the payer's route is the bank's own complaints procedure and then the Financial Ombudsman Service.

Cancellation, and what it generates

A payer cancels by contacting the bank or building society — increasingly by switching a toggle in a banking app — and the scheme's own consumer guidance adds two pieces of practical advice: written confirmation may be required, and the cancellation should be made at least a full working day before a collection is due, because a collection already inside the three-day cycle may be too far along to stop. If money is nonetheless taken after cancellation, that is squarely within the Guarantee.

Cancellations and amendments reach the service user through **ADDACS**, the Automated Direct Debit Amendment and Cancellation Service. The reason codes a service user will actually see include:

Code	Meaning
0	Instruction cancelled — refer to payer
1	Instruction cancelled by payer

Code	Meaning
2	Payer deceased
3	Account transferred to a new bank or building society (new details supplied; new DDI required)
B	Account closed
C	Account transferred within the same bank or building society (no new DDI required)
D	Advance notice disputed
E	Instruction amended
R	Instruction reinstated

ADDACS advices must be actioned immediately, or within three working days of receipt. That deadline is a scheme rule, not a courtesy, and repeated failure to observe it is a compliance matter with the sponsoring PSP. Reports are available on the Payment Services Website for a limited retention period — most Bacs reports for 37 calendar days, ADDACS and AWACS for 60 calendar days — after which they are gone from the service user's view.

Cancellation from the service user's side uses transaction code 0C, and the scheme also provides a cleaner mechanism for a planned ending: the collection transaction codes distinguish the first collection under a mandate (01), a subsequent collection (17), a re-presentation of a previously unpaid item (18), and the final collection (19). Submitting a collection as code 19 tells the paying PSP to delete the instruction from its records, and may produce a "final collection" annotation on the payer's statement. Using it correctly avoids the common failure mode of a mandate left live for years after the last payment.

One structural point about cancellation deserves emphasis, because it explains a recurring category of indemnity claim. A payer who cancels with their bank and never tells the service user leaves that service user submitting collections against an instruction that no longer exists. Those collections will be returned unpaid, and if the payer is annoyed enough they may also generate a claim. The mirror-image case — the payer who cancels with the service user but not the bank, and the service user who fails to submit a 0C — leaves a live authority nobody intends to use.

Indemnity claims

An indemnity claim is defined by Bacs as "a claim made by the paying PSP in respect of an incorrect Direct Debit being applied to an account". It is the mechanism by which the Guarantee's cost is passed from the paying bank to the service user, under the standing indemnity the service user signed when it was sponsored.

Claims arrive as **DDICA** advices in the service user's Bacs reports, carrying one of eight reason codes:

Code	Reason
1	Amount and/or date of Direct Debit differs from advance notice
2	No advance notice received by payer, or the amount is disputed
3	DDI cancelled by paying bank
4	Payer cancelled DDI direct with the service user
5	No instruction held; payer disputes having given authority

Code	Reason
6	Signature is fraudulent or not in accordance with authorised signatory
7	Claim raised at the service user's request after the Direct Debit was applied to the payer's account
8	Service user name disputed; payer does not recognise the service user collecting the Direct Debit

Codes 1 and 2 are advance notice failures. Codes 3 and 4 are cancellation failures. Codes 5 and 6 are authority failures, and code 6 is the reason a Paperless Direct Debit service user must have documented verification: with no signature to produce, the claim is close to unanswerable. Code 8 is the quiet killer of subscription businesses trading under names their customers have never heard of.

Code 7 is different in kind. It is the route by which a service user raises a claim *on behalf of* a payer who has come to it directly asking for money back — a refund the service user has decided to make. Rather than sending the customer to their bank, the service user initiates the reversal itself through the Payment Services Website, and may enter details for up to twelve Direct Debits for a single payer in one request.

The challenge process. Since 4 March 2024 challenges are raised centrally through the Payment Services Website rather than by email to each individual paying bank; the old bank-by-bank process was closed on 31 May 2024. The timetable, as published in the Bacs DDIC Challenge Process service user guide and accurate at time of writing, works like this. The day the claim is raised and the report made available counts as working day one. The service user must raise any challenge before the end of working day nine; a claim neither accepted nor challenged by that point is automatically accepted, and the money leaves. Up to five pieces of supporting evidence may be attached — typically a copy of the Direct Debit Instruction, a copy of the advance notice, or a call recording. The paying bank then reviews and either accepts or rejects; a rejected challenge may be relodged with better evidence, still within the day-nine window. If the paying bank has neither accepted nor rejected a challenge by the end of working day eleven, the claim is automatically cancelled in the service user's favour.

Older vendor documentation frequently quotes a fourteen-working-day window. That figure belongs to the pre-2024 arrangements; the operative deadline today is the end of working day nine. The authoritative text is the *Service User's Guide and Rules to the Direct Debit Scheme* — version 5.9 at the time of writing — available to registered users on the Bacs site, and it is what a sponsoring bank will hold a service user to.

Four supporting reports are available through the Payment Services Website: a monthly challenge and response report summarising claims raised, challenged, accepted and rejected; an ad hoc challenge and response audit report produced within fifteen minutes of request; an in-progress challenge and response report; and an in-progress report for reason code 7 requests.

Refund requests are not indemnity claims. The scheme distinguishes the two. A refund request is a claim by a paying PSP for money back where the error was the paying PSP's own — for example, it failed to action a cancellation and paid the collection anyway — and therefore falls outside the valid indemnity claim criteria. Settlement of a refund request is at the discretion of the service user. A service user that treats every incoming request as an obligation is paying for other people's mistakes.

Who bears the loss, in order

Putting the sequence in one place, because it is the answer to the question everyone actually has:

1. The payer bears no loss. The Guarantee makes them whole immediately, from their own bank, without conditions.
2. The paying PSP bears the loss temporarily. It funds the refund out of its own money for the days it takes an indemnity claim to run.
3. The service user bears the loss finally, in the ordinary case. The claim debits its bank account under the standing indemnity, whether or not it agrees, unless it challenges successfully inside the window.
4. The sponsoring PSP bears the loss if the service user cannot. That is the whole reason sponsorship exists, and the whole reason it involves financial assessment, account limits, document approval, mandatory training and the power to withdraw. A sponsor that lets an insolvent or careless service user run up claims is buying those claims.
5. The payer may still owe the underlying debt. A successful claim reverses the payment and leaves the contract intact. If the service is delivered and the contract is valid, the service user retains its ordinary legal remedies for the debt, including debt collection and the county court. The bank's decision to refund is an administrative act under scheme rules, not a judgment on the contract.

That ordering is why the Guarantee can be as generous as it is. The cost of generosity is loaded onto the party best placed to prevent the error, and that party has posted a legally binding undertaking, is vetted annually, and can be removed from the scheme. Fraudulent set-up is a good illustration: Bacs' own guidance states that where a payment is made against a fraudulent signature, the service user is liable, but the paying PSP must refund the customer immediately and then reclaim through an indemnity claim. The customer's position is never in question. Only the allocation between the two institutions is.

The scale of it, as of writing

Direct Debit is not a niche instrument, and the Guarantee therefore stands behind a very large fraction of British household finance. Pay.UK's published Bacs statistics for 2025 record 5,029,499 thousand Direct Debits — 5.03 billion — from a Bacs total of 6.86 billion payments worth £6.05 trillion. Over 113 billion Direct Debits have been collected since the mechanism began, out of more than 183.6 billion Bacs transactions since 1968.

The 2025 breakdown gives a sense of what the mandate actually carries:

Sector	Direct Debits, 2025 (thousands)
Utilities and household bills	1,840,984
Insurance premiums	930,489
Subscriptions	562,919
Business to business collections	136,344

Within those, the individual lines are instructive: 648 million mobile telephone collections, 398 million general insurance, 304 million domestic fuel, 232 million council tax, 188 million water, 157 million television licences and 149 million road tax. These are not discretionary purchases. They are the fixed costs of running a household, and they are collected on an authority the payer can revoke in ten seconds and reverse without giving a reason.

A note for implementers

If you are building a collections product, the following is the short list that separates systems that survive their first year from systems that do not.

Store the advance notice, not just the fact that one was sent. A challenge under DDICA code 1 or 2 is won or lost on producing the document, its amount, its date and its delivery address. A boolean flag is worthless.

Reconcile ADDACS within three working days, automatically. Every code 1, 3, B and C changes what you may submit next month, and the three-working-day rule is a scheme obligation with a compliance consequence.

Treat day nine as a hard deadline in code, not in a runbook. Claims arrive in batch reports, weekends and English bank holidays do not count, and an unattended inbox in the first fortnight of January will cost real money.

Use transaction code 19 for final collections and submit 0C when a customer leaves. Live mandates against departed customers are how code 4 claims are born.

If you are running Paperless Direct Debit, treat verification as the product, not the paperwork. You have no signature. Under code 5 or code 6 you will be asked to prove authority within seven working days, and if you cannot, you pay. Voice recordings, timestamped screens, address matching and audit trails are your entire defence.

And put your trading name — the one the customer will recognise — in field 9. Code 8 claims are almost always self-inflicted.

The mandate is a small document. It authorises an indefinite series of payments of unspecified amounts, and it is backed by a promise that the payer can call in at any time, for any reason, and get their money back the same day. That combination sounds unworkable, and it would be, were it not for the fact that the entire cost of being wrong sits with the one party who can choose not to be. That is the design. It has held for over fifty years, across five billion collections a year, and it holds precisely because nobody collecting money by Direct Debit can afford to be casual about it.

36.98 Common wrong ideas

Wrong: the mandate is an instruction you give your bank. **Right:** it is an authority given by the payer to the service user, and Bacs states plainly that the DDI is not evidence of any contract between the service user and the payer's payment service provider.

Wrong: the Direct Debit Guarantee is law. **Right:** it is a scheme rule in Pay.UK's Bacs rulebook, enforced through contracts between participants, and the statutory rights underneath it in the PSRs 2017 are in several respects narrower.

Wrong: the unconditional refund right for a sterling Direct Debit comes from the Payment Services Regulations. **Right:** regulation 79(3)'s unconditional right applies to euro direct debits under Regulation (EU) 260/2012, and for sterling Bacs it is the scheme Guarantee that is unconditional.

Wrong: "immediate" describes the whole transaction. **Right:** only the payer's refund is immediate, while the claim, the challenge window and the paying bank's response all run on English bank working days, so the service user's money may not move for another fortnight.

Wrong: modulus checking confirms the customer owns the account. **Right:** it confirms only that a sort code and account number could exist together and says nothing whatever about ownership, which is why a Paperless service user must verify identity itself.

Wrong: the challenge window is fourteen working days. **Right:** that figure belongs to the arrangements closed in 2024, and the operative deadline is the end of working day nine, after which an unchallenged claim is automatically accepted and the money leaves.

Wrong: a refund under the Guarantee cancels what you owe. **Right:** the refund undoes the payment and not the contract, and the service user keeps its ordinary legal remedies for the debt.

Wrong: the Guarantee covers you if the goods are bad or the company collapses. **Right:** it covers the mechanics — amount, date and set-up — and not the merits, and Section 75 and card chargebacks, which do reach the merits, do not apply to Direct Debit at all.

Wrong: every claim a paying bank raises must be paid. **Right:** a refund request, where the error was the paying PSP's own, falls outside the valid indemnity claim criteria and is settled at the service user's discretion.

Wrong: an unused mandate stays live indefinitely. **Right:** paying PSPs remove dormant instructions after a set period, so a service user collecting annually or seasonally must know its dormancy setting or find the authority has quietly evaporated.

36.99 Chapter summary in 20 lines

1. A Direct Debit is the one payment you set up once and then stop doing, and Britain ran 5.03 billion of them in 2025 out of 6.86 billion Bacs payments worth £6.05 trillion.
2. People hand over that authority because of the Direct Debit Guarantee, five bullet points promising a full and immediate refund from the payer's own bank if an error is made.
3. The refund is unconditional in practice: no minimum amount, no stated deadline, and no requirement to argue with the biller first.
4. The bank does not stay out of pocket, because it raises an indemnity claim and debits the collecting organisation's own account.
5. That is the entire economics of the scheme — the party collecting the money carries the risk of being wrong.
6. Legally the mandate is an authority given by the payer to the service user rather than a contract with the payer's bank, and the Guarantee is a scheme rule rather than a statute.
7. Five roles matter: the payer, the service user with its six-digit Service User Number, the paying PSP that owes the Guarantee, the sponsoring PSP that vouches for the service user, and Pay.UK as rule-maker.
8. The service user's own signature appears on the indemnity, a standing, legally binding undertaking to reimburse any paying bank in the country.
9. The mandate itself is a controlled document with fixed wording and fixed field order, requiring written approval from the sponsor before use and again after any change.
10. Every Direct Debit Instruction is for an unlimited amount, which is why an electronic mandate carrying a non-zero amount is rejected under AUDDIS reason code 7.
11. Mandates are created on paper, through AUDDIS, or with no signature at all through Paperless Direct Debit, and no new non-AUDDIS Service User Numbers will be created after 1 July 2027.
12. Paperless sign-up moves verification from the bank to the service user, and brings script approval, mandatory training, recommended voice recording and possible annual review with it.

13. Advance notice, by default ten working days plus postal time and recorded against the SUN, is the obligation that most indemnity claims allege was breached.
14. Notice is required before a first collection and before any change of amount, date or frequency, must reach the payer personally, and must name the service user as the statement will show it.
15. Cancellations and amendments arrive through ADDACS and must be actioned within three working days, and a payer who cancels with the bank but not the biller leaves collections that will fail.
16. Indemnity claims arrive as DDICA advices under eight reason codes covering notice failures, cancellation failures, authority failures, service-user-requested refunds and disputed trading names.
17. The day a claim is reported counts as working day one, a challenge must be raised by the end of working day nine, and a challenge the paying bank has neither accepted nor rejected by the end of day eleven is cancelled in the service user's favour.
18. Set against the Payment Services Regulations 2017, the Guarantee is faster, broader and unconditional, so a UK payer almost never needs to invoke the statute.
19. The loss runs in a fixed order: never the payer, temporarily the paying bank, finally the service user, and the sponsoring PSP if the service user cannot pay — while the payer may still owe the underlying debt.
20. The design holds because the whole cost of being wrong sits with the one party who can choose not to be, which is why nobody collecting by Direct Debit can afford to be casual about it.

Sources: Pay.UK and Bacs published scheme material (bacs.co.uk pages on Direct Debit, AUDDIS, Paperless Direct Debit, indemnity claims, Direct Debit Instruction templates and logo, the glossary, the FAQs, the guide and rules index and the notice on non-AUDDIS service user numbers after 1 July 2027; the DDI 1 template; The Little Bacs Book), directdebit.co.uk (the Direct Debit Guarantee, how to claim, cancelling a Direct Debit, your rights and safeguards), Pay.UK's Bacs Payment System pages and Bacs annual processing statistics 2025, the Payment Services Regulations 2017 (SI 2017/752, regulations 67, 74, 76, 79 and 80) as published on legislation.gov.uk, and the published description of the Direct Debit Indemnity Claim challenge process via the Payment Services Website. Bacs message and indemnity reason codes cross-checked against independent implementation documentation; the authoritative text is The Service User's Guide and Rules to the Direct Debit Scheme, v5.9 at the time of writing, available to registered users. All timings, limits and figures accurate at time of writing.

37

CHAPTER

Faster Payments

37.0 What this chapter gives you

1. You will be able to explain why a payment that reaches the plumber's account in one second does not move a penny between the two banks until Monday.
2. You will be able to say what deferred net settlement means, and why the Faster Payment System sits in exactly the same category as Bacs.
3. You will be able to work out why a payment made on Good Friday is spendable that afternoon and is not settled between banks until the following Tuesday.
4. You will be able to explain why the £1 million scheme cap is almost never the limit a customer actually meets, and name what binds instead.
5. You will be able to describe the three participation models and say which one requires a settlement account at the Bank of England.
6. You will be able to explain what prefunding and the Net Sender Cap are for, and why the July 2026 flexible model lowers the cost of entry for non-banks.
7. You will be able to distinguish the four transaction types and say why standing orders break the assumption that everything runs 24/7.
8. You will be able to state precisely what "irrevocable" means here, and which remedies still exist after a payment has gone.
9. You will be able to explain why open banking is a way of instructing a rail rather than a rail in its own right.
10. You will be able to size the netting efficiency — roughly six to nine pounds of customer payments discharged for every pound of central bank money that moves.

At 11:47 on a Sunday night in July, a man in Leeds opens his banking app, taps in a sort code, an account number and £340, and pays a plumber who fixed his boiler that afternoon. The plumber's phone buzzes before the man has locked his screen. The money is there. She can spend it.

Nothing about that sentence would have been possible in Britain in 2007. Before May 2008, a payment instructed by telephone or internet banking went into the Bacs cycle and arrived three working days later. A Sunday-night instruction would have reached the plumber on Wednesday. There was no Sunday, because Bacs does not run at the weekend, so in practice it would have been Thursday. The plumber would have gone to bed on Sunday with an invoice and nothing else, and would have chased the man on Tuesday to ask whether he had paid.

The Faster Payment System changed that, and it changed it more thoroughly than most people realise. It did not merely speed up an existing thing. It created a new expectation — that a payment between two ordinary British bank accounts is a real-time event, available at any hour, on any day,

with no batch, no cut-off and no working-day arithmetic — and that expectation has since spread underneath a great deal of British commercial life. Payroll advances, gig-economy earnings, marketplace payouts, insurance settlements, refunds, the entire account-to-account payments industry and every open banking payment initiation in the country all sit on top of it.

And almost none of it is true in the way it appears. The plumber's £340 did not travel from one bank to another on Sunday night. Nothing travelled. What travelled was a message, and the plumber's bank paid her out of its own money on the strength of it. The actual money between the two institutions moved during Monday, in one of three settlement cycles, netted against every other payment those institutions had exchanged. The seconds you can see and the settlement you cannot are two different systems bolted together, and the gap between them is where the risk, the liquidity cost and most of the interesting engineering live.

This chapter explains both halves: the fast one you experience, and the slow one underneath it that makes the fast one affordable.

The plain version

Imagine two villages either side of a wide river. There is one bridge, and it is a long way round.

Each village has a post office that looks after everybody's money. If Mrs Ali in the north village wants to send £180 to Mr Grant in the south village, the old way was simple and slow: the north postmaster counts £180 out of his drawer, puts it in a locked box, and waits for the van. The van crosses the bridge once a day. Mr Grant gets his money the following afternoon, or the one after that if the van has already gone.

Now somebody installs a telephone line between the two post offices.

The north postmaster picks up the phone and says: "Mrs Ali is sending £180 to Mr Grant. Pay him out of your drawer now. I'll square it with you when the van comes."

The south postmaster puts down the phone, opens his own drawer, and hands Mr Grant £180 of his own cash. Mr Grant has his money in under a minute. Nothing has crossed the bridge. There is no van in this story at all yet.

That is the whole trick, and it is worth sitting with for a moment because everything else follows from it. The speed does not come from moving money quickly. It comes from *not moving money at all* and having somebody at the far end pay out on a promise. The message is fast because messages are cheap and light. The money is slow because money is heavy, and the clever part of the design is realising you do not have to move the money in order to make the payment happen.

What the van still does

The van has not been abolished. It runs three times during each working day, and it carries the difference.

Suppose that during Monday morning the north post office phones the south fourteen times, for payments totalling £8,420, and the south phones the north nine times, for payments totalling £7,955. At lunchtime the two postmasters compare notes. North owes south £8,420; south owes north £7,955. Subtract one from the other and north owes south £465.

The van crosses the bridge carrying £465. Twenty-three payments worth £16,375 have been settled by moving four hundred and sixty-five pounds.

This is why the arrangement is cheap enough to give away free. If the van had to carry every payment separately it would need to cross twenty-three times with sixteen thousand pounds aboard, and somebody would have to pay for that. Instead it crosses three times a day with the leftovers.

What the south postmaster is worried about

Here is the part a bright twelve-year-old spots straight away, usually before an adult does.

The south postmaster handed Mr Grant £180 of his own cash on the strength of a phone call. What if, between the phone call and the van, the north post office burns down? Or the north postmaster turns out to have been lying, or has run out of money, or has simply disappeared? The south postmaster has given away real cash in exchange for a sentence spoken down a telephone line.

That worry is the single most important fact about this system, and it is invisible to everybody using it.

The way it is dealt with is not trust. It is a locked box. Before the north post office is allowed to use the telephone at all, it must take a quantity of its own cash to the county courthouse — a building neither village controls, which will not lose the money and will not go bust — and lock it in a box there. The box is not the north post office's spending money. It sits untouched. Its only job is to be there if the north post office fails, so the south postmaster can be paid from it and nobody who took money out of a drawer that morning is left short.

The size of the box is set by a rule: the north post office may never, between one van and the next, run up a debt to the other village larger than what is in the box. Once it has phoned through enough payments to reach that figure, the telephone stops working for outgoing calls until the van has been and the slate has been cleared.

So the picture is: a telephone that is always on, a van that runs three times a working day, and a locked box at the courthouse that makes the telephone safe.

The rules on the telephone

There are three more rules, and each of them has a reason.

The first is that there is a ceiling. The south postmaster will accept a phone call for anything up to one million pounds. Above that, the money goes by armoured van, immediately, with two guards and a signature — that is a different service, and it costs money, and it is what people buying houses use. The telephone is for ordinary amounts, and one million pounds is where "ordinary" is judged to stop.

The second is that each post office may set a *lower* ceiling of its own for its own customers, and they all do. The village rule says a million. Your own postmaster may say that you personally may phone through twenty-five thousand pounds and no more, because he has known you for two years and that is as far as his nerve goes. Both rules apply, and the lower one is the one that stops you.

The third is that the telephone is on all the time, and the van is not. The telephone works at three in the morning on Christmas Day. The van runs on working days only. That is fine, because of the locked box — but it means that over a long bank holiday weekend the north post office can be phoning payments across for four days before a single van crosses the bridge.

The worked example

Take the man in Leeds and the plumber.

He instructs £340 at 11:47pm on Sunday 12 July 2026. His bank checks he has the money, checks the payment against its fraud rules, and sends a message to the central system. The central system

routes it to the plumber's bank, which decides in under a second whether it can accept the payment, and answers. The whole round trip — sending bank to central system to receiving bank and back — is designed to complete inside fifteen seconds, and normally takes a small fraction of that. The plumber's bank credits her account immediately, out of its own funds. She can spend the £340 at 11:48pm on Sunday night.

The two banks square up on Monday. Three times during Monday's working hours, every bank in the system compares what it owes with what it is owed and one number per bank moves at the Bank of England. The man's £340 is a rounding error inside that number. Neither he nor the plumber will ever know which of Monday's three cycles carried it, and neither of them needs to.

Now scale it up. In 2025 the Faster Payment System carried 5.55 billion payments worth £4.84 trillion. That is roughly fifteen million payments a day, averaging about £870 each. Yet the money that actually had to move between the banks to settle all of it averaged around £2.3 billion a day in the first half of 2026. Somewhere between six and nine pounds of customer payments are being discharged for every pound of real money that moves. The rest cancels out — north owing south, south owing north, and the subtraction doing the work.

You should now be able to say the following at dinner and be right. When you send money by bank transfer in Britain, nothing is transferred. A message goes across in about a second, the receiving bank pays its customer out of its own pocket, and the banks square up with each other three times on each working day by moving only the difference. It feels instant because it is instant for you. Underneath, it is a promise with a locked box behind it.

Where the plain version stops being true

The village is a good picture of the mechanism and a misleading picture of the risk, the availability and the guarantees. Four corrections.

The first correction is that "instant" describes the customer experience, and only the customer experience. No money moves between institutions in seconds. Not some of it, not most of it — none. The Faster Payment System is a deferred net settlement system, which is precisely the same category as Bacs, and the only reason it does not feel like Bacs is that the receiving institution is contractually obliged to credit its customer immediately rather than waiting for settlement. The interbank leg happens three times each business day in the Bank of England's real-time gross settlement system, and the Bank's own description of the lag is worth quoting because it is more candid than anything the industry says in public: the gap between a customer payment and its settlement "could be several hours for Faster Payments during the week", but over an Easter weekend "the gap between settlements is currently from between Thursday evening and Tuesday morning". Read that again. A payment made on Good Friday is spendable on Good Friday and is not settled between banks until the following Tuesday. For four days, one institution has paid out real money and holds nothing but an obligation. That is not a defect and it is not a secret; it is the design, and the prefunded cash at the Bank of England exists precisely because of it. But anybody who tells you Faster Payments "settles in real time" has confused two layers of the system, and if they are designing a treasury function on that belief they will get their liquidity model wrong.

The second correction is that "24/7" is true of the service and false of the settlement infrastructure. The Faster Payment System is available twenty-four hours a day, every day of the year. The Bank of England's RTGS system is not. As of the time of writing, RTGS operates from 06:00 to 18:00, and the Bank has announced an early-morning extension to 01:30 from September 2027; a move towards near round-the-clock settlement, including a possible additional weekend settlement day, is at the

consultation stage, with the consultation paper published on 18 May 2026. So the always-on property that everybody markets belongs to the messaging layer. Beneath it sits a settlement layer that keeps banking hours, does not work weekends, and is only now being asked whether it might. The mismatch between the two is not a temporary lag in modernisation — it is the reason the prefunding regime exists, and any change to settlement hours changes the economics of prefunding directly.

The third correction is that the £1 million cap is almost never the limit you actually hit. One million pounds is the central scheme ceiling per transaction, and it has been since 10 February 2022, when it was raised from £250,000. Every participating institution sets its own lower limits, and those limits vary enormously by institution, by channel and by whether the account is personal or business. On Pay.UK's own published comparison, as last updated in July 2026, HSBC's online personal limit was £25,000 against a business limit of £1,000,000; Lloyds was £25,000 personal and £99,999 business; Barclays £50,000 personal; NatWest £100,000 personal; Metro Bank £21,000 personal. Starling, Monzo and several others publish the full £1,000,000 for both, subject to their own additional checks. So "Faster Payments can carry a million pounds" is a statement about the scheme, not about your account, and in most retail cases the binding constraint is two orders of magnitude below the headline. The cap is also per transaction rather than per day, and institutions layer daily aggregate limits, new-payee cooling-off periods and value-triggered verification on top of it. None of that is in the rulebook figure.

The fourth correction is that "irrevocable" and "guaranteed" are being used to mean four different things, and none of them is quite what the customer thinks. It is true that once a Faster Payment has been sent, the payer cannot cancel it — Pay.UK states this plainly, and it follows from the real-time design. But that is not the same as saying nothing can be done. Where a payment has gone to the wrong account by mistake, the industry operates a Credit Payment Recovery process. Where it has gone to the wrong account because the payer was defrauded, a mandatory reimbursement regime has applied since 7 October 2024. And "instant" itself carries an exception: the scheme's own published expectation is that funds are usually available almost immediately but can take up to two hours, and a very small proportion of UK accounts — Pay.UK puts it at under 0.1%, including some savings accounts — do not accept Faster Payments at all. A receiving institution that cannot post immediately is expected to tell the customer when the funds will arrive, not to fail the payment. So the honest formulation is: unstoppable by the sender, usually instant, occasionally two hours, sometimes recoverable afterwards, and increasingly reimbursable if it was a scam.

The technical version

What the system is, and who runs what

The Faster Payment System is one of the retail interbank payment systems operated by Pay.UK, the recognised operator and standards body for UK interbank retail payments. Pay.UK owns the scheme: the rules, the participation criteria, the standards and the liability arrangements. It does not build or run the machines. The central infrastructure is supplied and operated under contract by Vocalink, a Mastercard company, which also provides the central infrastructure for Bacs and for the Image Clearing System. On 17 December 2025 Pay.UK announced multi-year extensions to all three of those infrastructure contracts, providing continuity into the early 2030s.

Settlement is provided by the Bank of England. The Bank is not the operator of Faster Payments and has no role in the scheme rules; it is the settlement agent, holding the accounts in which the interbank obligations are finally discharged in central bank money.

The system launched on 27 May 2008, following a testing period in which very small-value transactions were processed. Its origins lie in the Office of Fair Trading's Payment Systems Task Force,

established in March 2004, which by May 2005 had secured an industry agreement to reduce clearing times for telephone, internet and standing order payments. Vocalink was awarded the infrastructure contract in October 2005 and has operated it ever since. Ten institutions went live at launch: Abbey (now Santander UK), Barclays, HSBC, Lloyds TSB, The Royal Bank of Scotland, Citi, Bank of Scotland, Clydesdale Bank, The Co-operative Bank and Nationwide Building Society.

The programme intended to replace the underlying infrastructure — the New Payments Architecture — has been through a substantial reset. Following the cancellation of the NPA procurement and the government's National Payments Vision, the work was renamed Interbank Infrastructure Renewal. The Payment Systems Regulator revoked Specific Direction 3 in May 2025 and Specific Direction 2 in August 2025, and closed its NPA project page on 14 October 2025. The stated long-term destination remains a single central infrastructure clearing and settling Bacs, cheques and Faster Payments, rather than three separate ones. The Vocalink contract extensions announced in December 2025 should be read against that: the existing infrastructure is now expected to carry the traffic into the early 2030s.

Scale, as reported by Pay.UK: 5.55 billion transactions worth £4.84 trillion during 2025. For comparison, the system handled 967.6 million transactions worth £771.4 billion in 2013.

The four transaction types

Pay.UK defines four categories of Faster Payment. They differ in initiation channel, availability and timing, not in the underlying clearing mechanism.

Type	Availability	Central limit	Timing characteristics
Single Immediate Payment	24 hours, 7 days	£1,000,000	Usually near-instant; may take up to two hours; central infrastructure returns accept or reject to the sending PSP within 15 seconds
Forward-dated payment	Can be set up 24 hours, 7 days	£1,000,000	One-off, executed on a pre-arranged future date
Standing order	Monday to Friday, excluding bank holidays	£1,000,000	Fixed amount, recurring date; rolls to next working day if the date falls on a weekend or bank holiday; at least 90% sent by 6am
Direct Corporate Access	24 hours	£1,000,000	Bulk file submission by business customers, same-day transmission; not offered by all participants

Several points in that table repay attention.

The fifteen-second figure is a property of the central infrastructure, not of the end-to-end customer experience. It is the window within which the sending PSP receives confirmation that the receiving PSP has accepted or rejected the payment, and where the answer is a rejection, the receiving PSP supplies a reason. This is the mechanism that makes the customer-facing "sent" and "received" states

meaningful. The two-hour outer bound covers the cases where the receiving institution cannot post immediately — an operational problem at its end, or an account type that does not permit immediate posting — and in those cases the scheme expects the customer to be given an indication of when the funds will arrive rather than a bare failure.

Standing orders are the outlier and are frequently misunderstood by engineers who assume the 24/7 property extends to everything. It does not. Standing orders execute on working days only, roll forward when the nominal date is not a working day, and cluster heavily in the early morning: Pay.UK's published expectation is that at least 90% are sent by 6am. If you are building a system whose behaviour depends on when a standing order will land, that early-morning clustering and the working-day roll are the two facts that matter.

Direct Corporate Access is the least discussed of the four and the most relevant to anyone with a bulk payment file. It allows a business customer to submit files of payment instructions directly into the Faster Payment System, using a secure-IP solution comparable to Bacstel-IP and, notably, the same standard file format as Bacs transactions. That last detail is the practical bridge for organisations that already produce Standard 18 files for Bacs: the file format does not have to change to move the same payment run onto a same-day rail. Not every participant offers Direct Corporate Access, and those that do not typically offer an alternative bulk mechanism of their own. A number of software providers are accredited to supply Direct Corporate Access services.

The value cap and its history

The central scheme limit is £1,000,000 per transaction. It has stood at that level since 10 February 2022, when Pay.UK raised it from £250,000; at the time of the increase there were 38 directly connected payment service providers and around 400 PSPs connected indirectly through sponsors. Earlier in the system's life the ceiling was substantially lower, having been raised to £100,000 for all payment types on 6 September 2010.

Institution-set limits are what bind in practice. Pay.UK maintains a published comparison of the limits set by individual banks and building societies; the figures below are drawn from that page as last updated on 24 July 2026, and are accurate at time of writing. They will change.

Institution	Online personal	Online business
HSBC	£25,000	£1,000,000
Barclays	£50,000	£1,000,000 (high value)
Lloyds Bank	£25,000	£99,999
NatWest	£100,000	£250,000 to £1,000,000
Metro Bank	£21,000	£250,000
Starling Bank	£1,000,000	£1,000,000
Monzo	£1,000,000	£1,000,000

Monzo's entry carries a rider worth generalising: payments above £10,000 are subject to further checks. Value-triggered friction of that kind is now routine across the industry and is not visible in any published limit table. A £1,000,000 headline limit and a £10,000 verification threshold coexist comfortably, and the second is what a customer will actually encounter.

Access: three participation models

Pay.UK defines three ways to participate, and the distinction between them is the most commercially consequential thing in this chapter for anybody building a payments business.

A **Directly Connected Settling Participant** holds its own technical connection to the central infrastructure and its own settlement account at the Bank of England, and performs its own settlement. It sets its own transaction limits. This is the full-fat model, appropriate to institutions with high volumes and eligibility for a Bank of England settlement account.

A **Directly Connected Non-Settling Participant** holds its own technical connection to the central infrastructure but does not settle at the Bank of England. A sponsoring participant performs the Bank of England settlement on its behalf. This model exists precisely for organisations that want the operational control and latency of a direct connection but are not eligible for, or do not want, a settlement account.

An **Indirect PSP** has neither its own connection nor its own settlement. It sends and receives Faster Payments through a sponsoring participant, which also performs the Bank of England settlement. The sponsor allocates the indirect participant a sort code from its own range and registers it with the payment systems and with the industry reference data. The indirect participant still processes its own payment messages, handles its own exception flows, and manages its own settlement and safeguarding accounts with the sponsor — those latter accounts being commercial bank accounts, and not to be confused with a Bank of England settlement account.

The eligibility criteria for the settlement account are set by the Bank of England, not by Pay.UK. To hold a settlement account, a firm must either hold a reserves account or be a financial market infrastructure, or an e-money or payment institution authorised in the UK by the Financial Conduct Authority, and must meet the settlement participation criteria of the relevant payment system.

That last clause is the one that changed the shape of the industry. In July 2017 the Bank of England extended direct access to RTGS settlement accounts to non-bank payment service providers. The consequences arrived quickly: Ebury became the first directly connected non-settling participant in February 2018, and TransferWise — now Wise — became the first non-bank settling participant in April 2018, followed by Equals Money in February 2019. Pay.UK's own access programme, launched in 2014, has seen the number of direct Faster Payment participants more than triple.

The current shape of that is visible in the published participant lists. As last updated in April 2026, the Faster Payment System participant list named 47 organisations, against 32 for Bacs. Among the 47 are Adyen, Banking Circle, ClearBank, Corpay, Modulr, PPS, Revolut, Soldo, Square, Stripe UK, SumUp, The Bank of London and Wise — a set that would have been unimaginable as direct participants in a UK clearing system fifteen years ago. Faster Payments is now, by participant count, materially more open than Bacs.

Pay.UK also operates an accreditation regime for technical aggregators, solution providers and consultants, with a published fee structure for Faster Payment System technical accreditation. For most new entrants the practical route to a direct connection runs through one of those accredited providers rather than through building the gateway in-house.

Settlement: deferred net, prefunded

Settlement of Faster Payments takes place three times each business day in the Bank of England's RTGS system. The Bank's classification is explicit: this is one of four settlement models it supports, and Faster Payments sits in the prefunded deferred net settlement category, alongside Bacs and the Image Clearing System.

The mechanics are multilateral netting. Between settlement cycles, participants exchange customer payments through the central infrastructure, entirely outside RTGS — the Bank notes that RTGS does not need to be open for this exchange to happen, which is what makes the 24/7 service possible at all. Obligations accrue. At each settlement cycle, the operator calculates each settling participant's multilateral net position against all others, and a single net amount per participant is settled across accounts in RTGS.

Because obligations accrue between cycles, settlement risk arises between cycles. The mitigation is prefunding, introduced for both Bacs and Faster Payments in September 2015. Each settling participant holds cash in a dedicated prefunding account in RTGS. That balance caps the participant's maximum net obligation to the system. If a settlement participant defaults, the segregated cash is used to complete settlement. The effect, in the Bank's own words, is that prefunding "eliminates settlement risk in systems that use it by capping the maximum net obligations of settlement participants" and "eliminates credit risk between settlement participants". Before 2015, the exposure was managed by a mutualised loss-sharing arrangement among participants; prefunding removed that mutualisation entirely.

The cap itself is the Net Sender Cap. It limits a participant's indebtedness to other participants within each settlement cycle, and it is backed by the prefunded balance. Historically Pay.UK set Net Sender Caps prescriptively, using a Minimum Net Sender Cap together with a Peak Contingency Value that participants had to hold at all times. The observed problem was that participants held more liquidity than they needed, with the high-usage cases occurring only infrequently — dead capital sitting in a segregated account at the Bank of England against a peak that rarely came.

On 1 July 2026, Pay.UK went live with a flexible model. Participants may now determine their own Net Sender Cap values according to their own business requirements, submitting plans quarterly for Pay.UK review. Adoption is optional; participants may remain on the existing prescriptive framework. Real-time alerts and the existing settlement protections are retained, and Pay.UK states that no technical change is required to adopt the new model. The intended effects are lower liquidity cost for existing participants and a lower barrier to entry for new ones, particularly non-banks, for whom the prefunding requirement is a direct and highly visible cost of participation.

The numbers give a sense of the netting efficiency. The Bank of England's published average daily RTGS settlement values for Faster Payments were £2,008 million in Q3 2025, £2,104 million in Q4 2025, £2,287 million in Q1 2026 and £2,298 million in Q2 2026. Set that against Pay.UK's reported £4.84 trillion of Faster Payments in 2025 — roughly £13.3 billion for every day of the calendar year, and closer to £19 billion for every business day, since payments accrue over weekends and settle on working days. The ratio of customer payments to central bank money moved therefore sits somewhere between six and nine to one depending on how the days are counted. Either way, the great majority of the value cancels before anything settles. It is also worth noting that Bacs, with far fewer participants and a slower cycle, settles a larger daily net value than Faster Payments: £6,205 million average daily in Q2 2026. Faster Payments dominates by volume and by public visibility, not by settled value.

RTGS operating hours are the constraint on all of this. At the time of writing they run 06:00 to 18:00. The Bank has announced that CHAPS will open at 01:30 from September 2027, giving a 16.5-hour

settlement window, and on 18 May 2026 published a consultation paper on further extension towards near 24x7 settlement, including the possibility of an additional weekend settlement day, most likely Sunday. The Bank identifies liquidity and risk management as one of the three principal drivers, noting explicitly that longer operating hours "could enable more frequent net settlement cycles, reduce settlement risk, and allow liquidity to be used more efficiently, lowering prefunding and liquidity cost".

The Bank's own worked illustration of the current exposure is the sharpest statement of the seam in this chapter. Between settlement cycles on a normal weekday, the lag "could be several hours". Over an Easter weekend, the gap between settlements runs "from between Thursday evening and Tuesday morning". In 2026 that means the last settlement before Good Friday on 3 April, and the next on Tuesday 7 April — four full days during which Faster Payments continue to flow at full volume, receiving institutions continue to credit customers, and the obligations pile up behind prefunded balances. The Bank makes a related observation in the same paper that deserves to be read by anyone modelling stress: outflows from banks can already take place at the weekend through Faster Payments, because it operates 24x7, at a time when most other sterling flows are closed.

Messages and standards

The Faster Payment System's native message set is derived from ISO 8583 — the same lineage as the card messages in Volume III, which is why an engineer who has worked on card switching finds the FPS message structure oddly familiar. Pay.UK maintains an FPS ISO 20022 Standards Library, described by Pay.UK as "a reference source providing information on the recommended conversion of ISO 8583 messages to the ISO 20022 messaging standard". The library sits behind registration on Pay.UK's standards portal rather than on the open website.

Beyond the payment messages themselves, participants exchange status reporting — in Faster Payments terminology, the fate of a payment — and a set of exception messages covering recalls, returns and related error flows. A direct or indirect participant must be able to generate, receive and process all of them; supporting only the happy path is not participation.

Confirmation of Payee

Confirmation of Payee is an overlay service, not part of the Faster Payment System itself, but no honest account of a UK account-to-account payment in 2026 omits it. Launched by Pay.UK in 2020, it checks the payee's account name against the account details entered before the payment is initiated, and returns a result to the payer. Its purpose is to reduce misdirected payments and to attack the specific fraud pattern in which a customer is persuaded to send money to an account in a different name from the one they believe they are paying.

Adoption is now broad: Pay.UK reports over 300 organisations participating and more than two million checks completed every day. Participation follows the same shape as the payment systems themselves — direct participants that are account-holding PSPs, and indirect participants reaching the service through accredited PSP aggregators. The Payment Systems Regulator mandated an expansion of coverage in 2024.

Liability: who pays when it goes wrong

Two regimes matter.

For payments sent to the wrong account in error, the industry operates the Credit Payment Recovery process. It is a best-efforts recovery mechanism, not a right of reversal; the receiving customer's consent, or a legal process, is generally required to get the money back.

For authorised push payment fraud — where the customer was deceived into authorising the payment themselves — a mandatory reimbursement requirement has applied to Faster Payments since 7 October 2024. The maximum level of mandatory reimbursement is £85,000 per claim, confirmed by the Payment Systems Regulator in policy statement PS24/7; the Bank of England, as operator of CHAPS, set the same £85,000 maximum for CHAPS to keep the two consistent. The PSR's assessment was that £85,000 covers 99.8% of Faster Payments APP scams by volume and 90% by value. Firms may apply an optional excess of up to £100, but may not apply it to vulnerable consumers. Claims must be reported within 13 months of the payment. Reimbursement is due within five business days of the claim, with the clock stoppable where the firm needs more information, and a decision required within 35 business days. The cost is split 50:50 between the sending and the receiving payment service provider — a design choice that deliberately puts the receiving institution, which onboarded the mule account, on the hook for half.

Underneath both regimes sits the statutory backstop in the UK's payment services regulations, derived from PSD2: a payment instructed by mobile, internet or telephone banking must arrive by the end of the following business day at the latest. Faster Payments beats that by roughly a day and a half. The regulation is the floor, not the standard.

What Faster Payments is not

It is not the rail for large value. Above £1 million, and for anything requiring same-day irrevocable settlement in central bank money on a per-payment basis, the answer is CHAPS, which settles gross across RTGS with an early-afternoon cut-off for the same-day guarantee and carries the large majority of sterling value on a tiny fraction of the volume.

It is not real-time settlement, and describing it as such in a prospectus, an architecture document or a liquidity model is a material error rather than a simplification.

It is not the same thing as open banking. Open banking is an API layer that allows a third party to initiate a payment from a customer's account with the customer's consent; the payment it initiates is, overwhelmingly, a Faster Payment. Open banking has no clearing, no settlement, no participant list and no prefunding. It is a way of instructing a rail, not a rail.

And it is not, any longer, a banks-only club. Nearly half the named direct participants are non-banks, they hold settlement accounts at the Bank of England on the same terms as the clearers, and the July 2026 change to the Net Sender Cap regime was explicitly designed to make that easier still.

What this means if you are building on it

Three practical consequences follow from the technical picture, and they are the ones most often got wrong.

The first is that prefunding is a real and continuous cost of direct settling participation, sized to your peak net sending position between cycles rather than to your average. It is cash, it is segregated, and until the flexible Net Sender Cap model it was sized conservatively by rule rather than by your own traffic. If your business is heavily one-directional — a payout business that sends far more than it receives — your net sender position, and therefore your prefunded balance, is close to your gross volume, and the economics of direct settling participation look very different from those of an institution with balanced two-way flow.

The second is that there is no cut-off to design around, and that is an unfamiliar discipline for teams coming from Bacs. There is no submission window, no processing day, no entry day. There is a queue, a fifteen-second response expectation, a two-hour outer bound, and a settlement cycle you do not

control and cannot see. Reconciliation must therefore be built against your own message-level fate reporting rather than against a settlement file, because the settlement file tells you about a net figure that bears no reconstructable relationship to individual payments.

The third is that the weekend is the risk. Volume does not stop on Friday evening; settlement does. Whatever exposure your model tolerates on a Tuesday afternoon, multiply it by the length of the longest bank holiday weekend in the calendar and check that the prefunded balance and the Net Sender Cap still work. That is the calculation the Bank of England is effectively inviting the industry to revisit in its May 2026 consultation, and it is the single most instructive number in the whole system: the measure of how far a payment that arrives in one second can outrun the money behind it.

37.98 Common wrong ideas

Wrong: Faster Payments settles in real time. **Right:** It is a deferred net settlement system; the interbank leg settles three times each business day at the Bank of England, and only the credit to the receiving customer is immediate.

Wrong: The whole system runs twenty-four hours a day. **Right:** The messaging service does; RTGS runs 06:00 to 18:00 on working days, with an extension to 01:30 announced for September 2027 and near round-the-clock settlement still at consultation.

Wrong: You can send £1 million by Faster Payment from any account. **Right:** £1 million is the central scheme ceiling per transaction; institution-set limits bind first, with HSBC's online personal limit at £25,000 and Metro Bank's at £21,000 as published in July 2026.

Wrong: A Faster Payment is always instant. **Right:** Funds are usually available almost immediately, but the scheme's own published expectation allows up to two hours, and under 0.1% of UK accounts do not accept Faster Payments at all.

Wrong: Once sent, nothing whatever can be done. **Right:** The payer cannot cancel, but Credit Payment Recovery exists for payments sent to the wrong account in error, and a mandatory reimbursement regime capped at £85,000 per claim has applied to APP fraud since 7 October 2024.

Wrong: Faster Payments settles more money than Bacs. **Right:** Bacs settles a larger daily net value — £6,205 million average daily in Q2 2026 against about £2,298 million for Faster Payments — because Faster Payments dominates by volume and visibility, not by settled value.

Wrong: Open banking is a rail competing with Faster Payments. **Right:** Open banking has no clearing, no settlement, no participant list and no prefunding; it is an API layer, and the payment it initiates is overwhelmingly a Faster Payment.

Wrong: Standing orders inherit the 24/7 property of the scheme. **Right:** They execute on working days only, roll forward when the nominal date is a weekend or bank holiday, and cluster early, with at least 90% sent by 6am.

Wrong: Faster Payments is a banks-only club. **Right:** The participant list named 47 organisations as last updated in April 2026, including Adyen, ClearBank, Modulr, Revolut, Square, Stripe UK, SumUp and Wise, making it materially more open than Bacs.

Wrong: You can reconcile individual payments against the settlement file. **Right:** The settlement file reports one net figure per participant per cycle, with no reconstructable relationship to individual payments; reconciliation must be built against message-level fate reporting.

37.99 Chapter summary in 20 lines

1. Before May 2008 a British payment instructed by telephone or internet banking went into the Bacs cycle and arrived three working days later.
2. The Faster Payment System launched on 27 May 2008 and created the expectation that an account-to-account payment is a real-time event at any hour on any day.
3. Nothing actually travels between the banks when you send one: a message goes across, and the receiving bank credits its customer out of its own funds.
4. The speed comes from not moving money at all and having somebody at the far end pay out on a promise.
5. The banks square up three times each business day, netting everything they have exchanged and moving one figure per participant at the Bank of England.
6. That netting is what makes the service cheap enough to give away, with somewhere between six and nine pounds of customer payments discharged for every pound of central bank money moved.
7. The exposure created between the message and the settlement is capped by prefunded cash held in a segregated account in RTGS.
8. Prefunding, introduced in September 2015, replaced a mutualised loss-sharing arrangement and caps each participant's maximum net obligation through the Net Sender Cap.
9. On 1 July 2026 Pay.UK went live with a flexible model letting participants set their own Net Sender Cap values, lowering liquidity cost and the barrier to entry.
10. "Instant" describes the customer experience only, because Faster Payments is a deferred net settlement system in precisely the same category as Bacs.
11. The gap between a customer payment and its settlement can be several hours on a weekday and, over an Easter weekend, runs from Thursday evening to Tuesday morning.
12. "24/7" belongs to the messaging layer, while the settlement layer beneath it keeps banking hours and does not work weekends.
13. The £1 million per-transaction ceiling, in force since 10 February 2022, is rarely the binding limit, because institution-set limits are often two orders of magnitude lower.
14. Four transaction types share one clearing mechanism: Single Immediate Payments, forward-dated payments, standing orders and Direct Corporate Access.
15. Standing orders are the exception to everything always-on, running on working days only, rolling forward, and clustering before 6am.
16. Participation comes in three shapes — directly connected settling, directly connected non-settling, and indirect through a sponsor — and only the first holds its own Bank of England settlement account.
17. Since the Bank opened settlement accounts to non-bank payment service providers in July 2017, the direct participant list has grown to 47 organisations, nearly half of them non-banks.
18. The sender cannot cancel a payment, but Credit Payment Recovery covers misdirection and mandatory reimbursement, split 50:50 between sending and receiving firms, covers APP fraud.
19. Faster Payments is not the rail for large value, is not real-time settlement, and is not the same thing as open banking.
20. If you build on it, prefunding is a continuous cost sized to your peak net sending position, there is no cut-off to design around, and the weekend is where a payment that arrives in one second most outruns the money behind it.

Sources: Pay.UK published scheme material (wearepay.uk Faster Payment System pages covering how Faster Payments work, transaction types, transaction limits, participation, participant lists and the FPS ISO 20022 Standards Library; Confirmation of Payee pages; the 1 July 2026 flexible Net Sender Cap announcement and

explainer; the 17 December 2025 Vocalink contract extension announcement; and the February 2022 £1 million limit press release), *the Bank of England* (*payment and settlement pages, payment and settlement statistics, and the consultation paper "Extending RTGS and CHAPS settlement hours — next steps towards near 24x7 settlement", published 18 May 2026, together with the 2016 RTGS PFMI self-assessment for the September 2015 introduction of prefunding*), *and the Payment Systems Regulator* (*PS24/7 on the maximum level of reimbursement, APP fraud reimbursement protections, and the Interbank Infrastructure Renewal project page*). *Launch history and early participant milestones cross-checked against public record. All limits, timings, figures and participant counts accurate at time of writing.*

38

CHAPS and RTGS

38.0 What this chapter gives you

1. You will be able to explain why a conveyancer waits for CHAPS rather than sending a Faster Payment, even though Faster Payments is quicker and free.
2. You will be able to state the trade at the heart of gross settlement: it removes the risk that a counterparty fails while owing you, and charges for it in cash kept parked and idle.
3. You will be able to describe how a direct participant manufactures intraday liquidity by pledging collateral, and why the real constraint is encumbrance rather than owned cash.
4. You will be able to explain what the Liquidity Saving Mechanism does, and why marking a payment urgent or non-urgent is a real decision with consequences in minutes and in liquidity.
5. You will be able to name the three points of finality in the CHAPS rulebook and say which one makes a payment unrecallable.
6. You will be able to explain why designation under the Settlement Finality Regulations 1999 is what lets the seller's solicitor pick up the phone and release the keys.
7. You will be able to work backwards from the 2pm conveyancing convention through your own bank's cut-off to the scheme's 17:40 deadline, and say why the earliest one governs.
8. You will be able to explain tiering, and why "settled in central bank money" describes one leg of the journey only.
9. You will be able to recount what happened on 20 October 2014 and say why the contingency service was available and was not invoked.
10. You will be able to decide, for a given payment, whether CHAPS is worth £25 to £30 or whether you are buying a certainty you do not need.

At about eleven on a Tuesday morning, in an office above a dry cleaner's in a market town, a licensed conveyancer refreshes a screen. She has been refreshing it for twenty minutes. On the other end of a telephone line is a client sitting in a hired van outside a house she does not yet own, with a cat in a carrier on the passenger seat and a removal crew being paid by the hour.

Nothing about this scene is unusual, and nothing about it is optional. Until a specific number appears on the conveyancer's client account screen, the house belongs to somebody else, the keys stay in the estate agent's safe, and the van stays parked. When the number appears, everything happens at once: a phone call, a release of keys, a chain of four other households doing the same thing in sequence within the next ninety minutes.

That number appears because of CHAPS, and CHAPS works because of a machine at the Bank of England called RTGS. Together they are the most important piece of financial infrastructure in the United Kingdom that almost nobody has heard of. In 2024 CHAPS carried 52.7 million payments

worth £87.5 trillion, which is roughly 0.5% of all UK payment volumes and 91% of all sterling payment value. The average CHAPS payment was £1.7 million. Most people use it once or twice in a lifetime, pay between £25 and £30 for the privilege, and assume it is simply an expensive bank transfer that goes a bit faster.

It is not faster. Faster Payments is faster, and free, and works at midnight on a Sunday. What CHAPS gives you is something else entirely, and this chapter is about what that something is: finality, in central bank money, that cannot be undone, and what the country pays to have it.

All timings, limits, fees and figures in this chapter are accurate at the time of writing, August 2026. Payment infrastructure is a moving target, and the Bank of England has published plans to change several of these numbers before 2030.

The plain version

Imagine a school where every class has a tub of counters in the head teacher's office. Class 4 has a tub, Class 5 has a tub, Class 6 has a tub. Nobody carries counters around the school. If Class 5 owes Class 6 forty counters, somebody walks to the office, the head teacher takes forty counters out of Class 5's tub and puts them in Class 6's tub, and that is that. The counters never leave the office. What moves is which tub they are sitting in.

Now, the classes owe each other things all day long. Hundreds of times a day. There are two ways to run this.

The first way is the whiteboard. Every time one class owes another, you write it on a whiteboard in the corridor. Class 5 owes Class 6 forty. Class 6 owes Class 4 twenty. Class 4 owes Class 5 fifteen. At four o'clock the head teacher looks at the whiteboard, works out who is up and who is down overall, and moves counters between the tubs once, at the end of the day. This is enormously efficient. Hundreds of debts get settled by moving a handful of counters. Nobody has to keep many counters in their tub, because most of what they owe cancels against most of what they are owed.

There is one problem with the whiteboard, and it is a serious one. At half past three, Class 5 might collapse. The teacher walks out, the class dissolves, there is nobody to pay. All those entries on the whiteboard saying "Class 5 owes" are now just chalk. Class 6 spent the afternoon handing out biscuits on the strength of a promise that Class 5 would pay at four, and at four Class 5 does not exist. Worse, Class 6 was counting on that money to pay Class 4, so now Class 4 is short too, and the whole whiteboard has to be rubbed out and recalculated as though Class 5 had never traded at all. One collapse at 3.30 rewrites everybody's day.

The second way is to walk to the office every single time. Class 5 owes Class 6 forty counters? Somebody walks to the office, forty counters move from one tub to the other, immediately, and the matter is closed forever. There is no whiteboard. There is nothing to rub out. If Class 5 collapses at 3.31, every payment it made before 3.30 is simply done, and nobody else is affected at all.

That second way is real-time gross settlement. "Real-time" means it happens when you ask, not at four o'clock. "Gross" means each payment moves on its own, at its full size, rather than being cancelled off against other payments first. RTGS is the head teacher's office. CHAPS is the queue of people walking to it.

The second way is safer. It is also more expensive, and the expense is a specific and slightly surprising one: you need a lot more counters.

Here is why. Suppose Class 6 is going to send 900 counters' worth of payments today and receive 850 back. Under the whiteboard system it only ever needs to find the 50 counter difference at four o'clock. Under the walk-to-the-office system, it depends entirely on the order. If all 900 of Class 6's payments have to go out in the morning and the 850 coming in do not arrive until the afternoon, then Class 6 needs 900 counters sitting in its tub at nine in the morning, doing nothing, just waiting. Those 900 counters could have been doing something useful. Having them sit in a tub all morning is the price of never having a whiteboard.

That is the whole trade, and it is worth saying in one sentence, because it is the sentence this chapter exists to explain. Gross settlement removes the risk that somebody fails while owing you, and it charges you for that in cash you have to keep parked and idle.

The house

Now the conveyancer above the dry cleaner's.

Her client is buying a house for £340,000. At exchange of contracts, three weeks ago, the client paid a 10% deposit of £34,000. On completion day the remaining £306,000 has to reach the seller's solicitor. Of that, £246,000 has come from a mortgage lender, and £60,000 is the client's own savings plus the deposit already held.

At 10.52, the conveyancer instructs her bank to send £306,000 by CHAPS to the seller's solicitor's client account. Her firm is charged about £25 by its bank for that single instruction. Somewhere inside the machinery, the Bank of England charges her firm's bank 48.7 pence for it, which is the 2026 CHAPS tariff per payment sent. The rest is the bank's margin, its own liquidity cost, and its compliance department.

Within seconds, a message reaches the Bank of England. The buyer's bank has an account there. The seller's solicitor's bank has an account there. £306,000 comes off one and goes onto the other. Not a promise, not an entry on a whiteboard: the actual balance changes, at the Bank of England, immediately and permanently.

And here is the part that matters to the woman in the van with the cat. From the instant that balance changes, the money cannot come back. There is no cooling-off period. The buyer cannot ring up and cancel. The bank cannot reverse it because it made a mistake. A court can order someone to pay it back afterwards, but that is a new payment going the other way, not an undoing of the first one. The seller's solicitor, seeing the credit on her screen, can lift the phone to the estate agent and say "release the keys", and she can do that with total confidence, because the money on her screen is not a claim on the buyer's bank. It is settled.

That confidence is the product. Everything else, including the fee, is the cost of manufacturing it.

Now multiply it. That house is in a chain. The seller of the £340,000 house is buying a £455,000 house two streets away, and cannot send that money until the £306,000 arrives. The seller of the £455,000 house is buying a flat for £238,000. Five households, five vans, five sets of keys, all keyed off a sequence of payments that must each be final before the next one can start. If any link in that chain were reversible, the whole chain would have to wait until everyone was certain, which in practice means days. The reason a chain can complete in a single morning is that each payment in it is dead certain the moment it lands.

	The whiteboard way	The walk-to-the-office way
What moves	One net figure at the end of the day	Every payment, one at a time, in full
Cash you must keep idle	Very little	A great deal
If a member collapses mid-afternoon	Everyone's day is rewritten	Nothing already paid is affected
Cost per payment	Fractions of a penny	Tens of pence to the bank, tens of pounds to you
UK examples	Bacs, Faster Payments	CHAPS

Where the plain version stops being true

The counters are not scarce in the way the analogy implies

The school story says a bank needs a big pile of money sitting idle to make payments early in the day. That is the right shape but the wrong mechanism, and the difference is the single most important thing a practitioner knows that a layperson does not.

A CHAPS direct participant does not fund its day only out of its own accumulated cash. It can, within limits, manufacture settlement money during the day by pledging assets. Eligible participants in the Bank's Sterling Monetary Framework can generate intraday liquidity against collateral: they pledge gilts and other very high quality assets into a pool at the Bank, the Bank values that collateral, applies a haircut, and credits the participant with settlement funds it can spend across the day. At the end of the day that liquidity is extinguished and the collateral is released. Provided the collateral is there and the day balances by the close, the participant has borrowed several billion pounds from the central bank and given it back before dinner.

So the real constraint is not "how much cash does the bank own" but "how much eligible collateral can it encumber, and what is it giving up by encumbering it". That is a subtler and more interesting cost, and it changed the industry's behaviour after the post-crisis liquidity rules made intraday positions expensive to carry. The tub of counters is really a tub plus an overdraft facility secured on the school's silverware.

It is not one payment at a time, economically speaking

The plain version says each payment walks to the office on its own. Legally that is exactly right, and it stays right. Economically it stopped being the whole story on 15 April 2013, when the Bank introduced a Liquidity Saving Mechanism into CHAPS.

Participants can mark a payment as urgent or non-urgent. Urgent payments settle one at a time, immediately, exactly as described. Non-urgent payments go into a central scheduler and wait a little. Every couple of minutes the system pauses the urgent stream, runs a matching algorithm for around twenty seconds, and looks for groups of payments across different banks that broadly offset each other. Those it finds, it settles all at the same instant. Each of them is still a separate gross settlement in law, with its own finality; but because they land simultaneously, the outflows are funded by the inflows and nobody has to pre-fund the full amount.

The measured effect was large. Before the mechanism, the combined intraday liquidity requirement across CHAPS averaged around £21.2 billion. In the months after, it averaged around £16.9 billion, and the Bank's own analysis attributed roughly £4 billion of that, about a 20% saving, to the mechanism itself. So the honest summary is that CHAPS is a gross system that has quietly borrowed one of netting's best tricks, while keeping gross settlement's legal guarantees. The plain version's clean

opposition between "the whiteboard" and "the walk" is a teaching device, not a description of the modern machine.

Almost nobody is actually in the room

The school has three classes and three tubs. CHAPS has, at the time of writing, over 35 direct participants, and several thousand financial institutions that reach it through one of those participants. More than 70 organisations hold accounts allowing them to settle directly in one or more payment systems at the Bank, but the CHAPS membership itself is a small club.

This is called tiering, and it means that the sentence "CHAPS is settled in central bank money and is irrevocable" describes one leg of the journey only. If your bank is not a direct participant, your payment reaches CHAPS through a correspondent, and the leg between your bank and its correspondent is an ordinary commercial arrangement with ordinary commercial credit risk. The Bank of England has worried publicly about this for years: a 2013 Bank Quarterly Bulletin article on tiering counted around 4,500 indirect participants and identified the credit, liquidity and operational risks that concentration creates, and the Bank subsequently pressed the largest indirect participants to join directly. Those particular figures are more than a decade old and the population has changed, but the structural point has not. Central bank money is the settlement asset between direct participants. Between everybody else it is commercial bank money wearing the same coat.

And your own leg, as a customer, is not central bank money at all. What lands in the seller's solicitor's client account is a claim on her bank. What settled at the Bank of England is a transfer between two banks. They happen close together in time, they are causally linked, and they are not the same event.

"Same day" and "irrevocable" both need footnotes

The plain version implies you can send a CHAPS payment whenever you like during the working day and it will land. Two things complicate that.

First, cut-offs come in layers. The CHAPS system opens at 6am each working day and closes at 6pm for bank-to-bank payments, with customer payments having to be submitted by 5.40pm. That is the scheme deadline. It is not your deadline. Every bank sets its own, earlier, customer cut-off, and those are frequently in the early afternoon, sometimes considerably earlier for instructions given in a branch or by a business channel that needs a callback. On top of that sits a market convention: under the Law Society's Standard Conditions of Sale, if the completion money is received after 2pm, completion is treated as taking place the next working day for the purposes of the compensation and apportionment provisions. So the practical deadline in a house purchase is not 5.40pm and not the bank's cut-off but 2pm, and every conveyancer in the country works backwards from it. The chapter brief for a house sale is early afternoon; the scheme's own limit is nearly four hours later.

Second, "irrevocable" is a statement about the settlement, not a statement about your money being unreachable. A settled CHAPS payment cannot be pulled back; the receiving bank can be asked to return funds and may agree, but it is not obliged, and there is nothing resembling a card chargeback. That is the point of the rail. But since 7 October 2024 the Bank of England has applied reimbursement rules to in-scope retail CHAPS payments broadly aligned with those for Faster Payments, with a maximum reimbursement of £85,000 per claim for authorised push payment fraud. So a victim of a conveyancing fraud is not necessarily without remedy. The payment is still final. The remedy is a different payment, running the other way, from a different party, under a different rulebook.

CHAPS is also, second by second, not a system without limits in the way people assume. Faster Payments caps a single transaction at £1 million. CHAPS scheme rules set no maximum value at all,

which is why a £4 billion money-market payment and a £306,000 house purchase travel on the same rail. What limits your payment is your own bank's policy, not the scheme.

The technical version

RTGS is a ledger, not a payment system

The first correction a practitioner makes is terminological. RTGS is not a payment scheme. It is the Bank of England's accounting infrastructure: the ledger on which sterling settlement accounts are held and on which balances move. The Bank describes it as an accounting system that underpins the settlement of sterling payments. CHAPS is a payment system that settles on it. So do others.

That distinction matters because the ledger is shared. On an average working day, over £800 billion settles across RTGS. That total is not CHAPS alone. It includes CREST, the securities settlement system, whose deliveries against payment settle in central bank money on the same ledger; it includes the net settlement obligations arising from Bacs, Faster Payments, LINK, cheque clearing and card schemes, each of which computes its own multilateral net positions and then discharges them by a small number of movements across RTGS accounts. The retail rails you use every day are, from the Bank's point of view, a handful of large entries per day. CHAPS is the system that pushes each payment individually onto the ledger.

The core ledger itself was replaced under the Bank's RTGS Renewal Programme. The renewed core settlement engine, RT2, completed its final transition stage and went live on 28 April 2025, and the Bank's revised tariff framework took effect on the same date.

Settlement account balances at the Bank are central bank money: a liability of the central bank rather than of a commercial bank, with no credit risk in any conventional sense. The Bank characterises it as the ultimate secure and liquid sterling asset. When a CHAPS payment settles, the sending participant's balance falls and the receiving participant's balance rises, in the Bank's books, in real time.

What "real-time gross settlement" removes, precisely

The risk RTGS removes is settlement risk in its principal form: the risk that you have given value and your counterparty fails before giving value back. In a deferred net settlement system, an obligation exists from the moment a payment is exchanged until the moment the net position settles. Between those two moments there is an exposure. In the retail systems that exposure is managed by prefunding, caps and loss-sharing. In a high-value system moving hundreds of billions a day, an intraday exposure of that size between a small number of institutions would be a financial stability problem in its own right, and there is no plausible amount of collateral that would make it comfortable.

Real-time gross settlement removes it by construction. Because settlement occurs in real time, in central bank money, participant by participant, no intraday exposures accumulate between account holders at all. There is nothing to unwind, so there is no unwind risk: the scenario in which one participant's failure forces recalculation of everybody else's net positions simply cannot arise.

What it does not remove is liquidity risk, and this is where the design cost sits. Because payments settle one by one, they settle only if funds are available at that instant. A participant that has sent more than it has received needs a balance to cover the difference. Aggregate across the system and you get a queueing problem with a free-rider incentive: everybody would prefer to receive early and send late, and if everybody acts on that preference the system gridlocks. The architecture responds to that incentive in three ways: liquidity supply, liquidity saving, and behavioural rules.

Liquidity supply

Direct participants fund their CHAPS activity from their settlement account balances. Sterling Monetary Framework participants, and certain other account holders case by case, can generate intraday liquidity secured against the very highest quality collateral with prudent haircuts, using a collateral pooling arrangement at the Bank. In the Bank's published service description, intraday liquidity provision to support CHAPS averaged around £18 billion a day; that figure dates from the mid-2010s edition of that document and should be read as indicative of scale rather than current. CREST separately generated large volumes of liquidity through auto-collateralising repo.

The point of design principle is that the central bank is willing to supply intraday liquidity generously, against good collateral, because the alternative is a gridlocked high-value system. The constraint on a participant is therefore its collateral, its opportunity cost, and the regulatory treatment of intraday positions, rather than a hard cash balance.

Liquidity saving

The Liquidity Saving Mechanism, live in CHAPS since 15 April 2013 and retained in the renewed service, splits the flow into urgent and non-urgent streams via a central scheduler. Urgent payments settle individually and immediately. Non-urgent payments queue for a matching cycle, which as documented runs for around twenty seconds with roughly two-minute gaps between cycles, during which the algorithm identifies groups of broadly offsetting payments from different participants and settles them simultaneously.

The legal characterisation is unchanged: each payment settles gross, individually, with its own finality. The economic characterisation is that the participants economise, because the simultaneity means outflows are funded by matched inflows. The Bank measured combined intraday liquidity requirements falling from around £21.2 billion before implementation to around £16.9 billion afterwards, and attributed around £4 billion, roughly 20%, to the mechanism.

A practitioner reading this should note the operational consequence. Marking a payment non-urgent is cheaper in liquidity terms and slower in wall-clock terms, by a matter of minutes. For a completion payment at 1.50pm, urgent is the correct choice and the extra liquidity cost is the bank's problem. For an internal treasury sweep at 9.30am, it is not.

Behavioural rules

The Bank sets throughput targets for CHAPS direct participants, varying by category, and monitors adherence to them. The CHAPS Reference Manual distinguishes Category A direct participants as those whose average daily CHAPS volumes, sent and received, are greater than or equal to 40,000 payments, or whose CHAPS values are greater than or equal to 3.5% of total payment values, measured over a rolling twelve-month period. The specific percentage-by-time obligations sit in participant-facing operational documentation rather than the published reference manual, so no figures are quoted here.

Alongside throughput, the scheme sets availability obligations across the day. Participants must be open to receive by 8am and must send by 10am, which prevents the pathology of a participant that logs on at noon and drains everybody else's liquidity for two hours.

The CHAPS operating day

The published timetable, accurate at the time of writing, is as follows. All times are UK local time on a working day, meaning Monday to Friday excluding bank and public holidays in England and Wales.

Time	Event
06:00	CHAPS opens for settlement
08:00	Direct participants must be open to receive
10:00	Direct participants must have sent
17:40	Deadline for submission of customer payments
18:00	CHAPS closes for bank-to-bank payments
20:00	Latest contingency extension of the settlement day

RTGS itself does not close when CHAPS does. The Bank's 2026 consultation describes CHAPS as operating twelve hours a day, 06:00 to 18:00, Monday to Friday, with RTGS closing between 19:00 and 00:15 after end-of-day processes.

The customer and bank-to-bank distinction is a message-type distinction. Since the June 2023 migration, CHAPS uses ISO 20022. A customer credit transfer is a pacs.008; a financial institution transfer is a pacs.009. The 17:40 deadline is the pacs.008 deadline, and the twenty minutes to 18:00 exist so that participants can square their own positions between themselves after the customer flow has stopped.

Finality, in three named points

The most precise thing in the CHAPS rulebook, and the thing most often garbled in secondary sources, is that finality is not one event. The scheme defines three.

The Point of Entry is when the payment message has entered the Swift network and is acknowledged within that network by Swift. The Point of Irrevocability is after the point at which the settlement status in RTGS changes to "settling". The Point of Final Settlement is the point at which the settlement status in RTGS changes to "settled". Under contingency operation in MIRS Active Mode these points are modified, with finality occurring when the settlement status in MIRS changes to "settled".

Underpinning this is statute. CHAPS is a designated system under the Financial Markets and Insolvency (Settlement Finality) Regulations 1999. Designation is what makes finality survive an insolvency: a transfer order that has entered a designated system cannot be unwound by the ordinary retrospective effect of insolvency law, and the system's default rules take precedence. Without designation, the whole architecture would rest on contract and would be vulnerable to a liquidator arguing that payments made on the morning of a failure should be clawed back. With designation, the settled payment is settled, and the conveyancer can release the keys.

This is also why "recall" is a misnomer in CHAPS. A participant may transmit a request to the receiving participant asking for funds to be returned, and the receiving participant may co-operate, subject to its own obligations to its customer. That is a request for a new payment in the opposite direction. Nothing reverses.

Who participates, and what it costs

There are over 35 direct participants and several thousand financial institutions that make CHAPS payments through one of them. The Bank has progressively widened access: non-bank payment service providers became eligible for settlement accounts from 2017, and an omnibus account model was introduced in 2021 for certain arrangements. The Bank took over direct delivery of the CHAPS service itself in November 2017, having previously overseen a separate scheme company.

The economics of participation, at the time of writing, are published. Direct participants pay a per-item tariff of 42.7 pence per CHAPS payment sent for 2025 and 48.7 pence for 2026, plus an annual participation fee calculated from each participant's percentage share of total CHAPS values over a

defined historical window, applied to half of total CHAPS values. Joining is a capital project in its own right: the published onboarding fees include £30,000 to open a settlement account and £90,000 to join CHAPS, before a participant has spent anything on its own systems, testing, liquidity or staffing. Certain manual contingency operations carry fixed charges, such as £550 for a manual defund, and party-to-party transfers attract a £550 fixed administration fee if used more than twice a year.

Those numbers explain the shape of the market. The tariff is trivial per payment. The fixed costs of participation, technical conformance and round-the-clock operational cover are not, which is why the direct membership is a few dozen institutions and the indirect population is several thousand.

Against that, the end customer typically pays between £25 and £30 per CHAPS payment, and indirect participants have historically paid their correspondents in the region of £2 to £3 per payment with a maximum of about £30, according to figures published by the Bank. The gap between 48.7 pence and £25 is not primarily infrastructure cost. It is the sending bank's liquidity, its manual checks on high-value payments, its fraud controls and its acceptance of operational risk on an irreversible instruction.

What CHAPS is actually used for

Three families of use dominate.

Wholesale financial activity comes first by value. Financial institutions and the largest businesses use CHAPS to settle money market transactions and the sterling legs of foreign exchange trades. These are the payments that make the average transaction value £1.7 million and produce the extraordinary ratio of 91% of sterling value on 0.5% of volume.

Property is the use the public sees. CHAPS is the standard instrument for completion in England, Wales, Scotland and Northern Ireland conveyancing, because completion requires cleared, final, same-day funds and there is no other sterling rail that provides all three above £1 million, and none at all that provides finality of this legal quality. The Standard Conditions of Sale contemplate payment on completion by direct credit to the seller's solicitor's bank, which in practice means CHAPS.

Corporate treasury is the third. Time-critical supplier payments, tax payments, margin calls, intra-group funding and anything where a missed cut-off has a contractual or regulatory consequence.

The ISO 20022 enhanced data mandate has made the property use case explicit in the data. From 1 May 2025 the Bank mandated purpose codes and Legal Entity Identifiers for defined categories of CHAPS payment: payments between financial institutions, meaning all pacs.009 messages and pacs.008 messages where both ultimate parties fall within defined categories such as PRA-authorised deposit-takers, broker-dealers, Bank-supervised financial market infrastructures or RTGS account holders; and property payments, which must carry one of a defined set of codes.

Code	Meaning
PCOM	Property completion payment
PDEP	Property deposit
PLDS	Property loan disbursement
HLRP	Property loan repayment
HLST	Property loan settlement
PLRF	Property loan refinancing

The mandate applies to payments originated through channels controlled by the direct participant. The wider ISO 20022 programme continues: structured addresses were encouraged from November

2025 and unstructured addresses are to be rejected from November 2026, with a mandate for structured remittance data expected from November 2027.

The size of the thing

Measure	Figure	Period
CHAPS payments	52.7 million, a record, up 3.1%	2024
CHAPS value	£87.5 trillion	2024
Average daily value	£344.4 billion	2024
Average daily volume	207,609 payments	2024
Average payment value	£1.7 million	2024
Share of sterling payment value	91%	2024
Share of UK payment volume	around 0.5%	2024
Average daily value	£365.7 billion	2025, to Q3
Average daily volume	209,401 payments	2025, to Q3
Total daily settlement across RTGS	over £800 billion	2024/25

For comparison against the rails covered elsewhere in this volume: Bacs runs a three working-day cycle of submission, processing and entry, and settles on a deferred net basis; Faster Payments settles in seconds, runs 24/7, and caps a single transaction at £1 million; CHAPS is same-day, gross, final, and uncapped by the scheme.

When it stops: 20 October 2014

The best way to understand what RTGS is for is to read the report of the day it did not work, and the Bank commissioned and published one.

RTGS opened at 06:00 on Monday 20 October 2014. At 06:02 a system error was detected. The proximate cause was a set of configuration changes made over the weekend of 18 and 19 October to transfer a CHAPS membership from one bank to another. Those changes exposed three latent defects that had been sitting in the system unnoticed: one design defect introduced in April 2014, and two functional defects dating from the implementation of the contingency service. Nothing was hacked. Nothing was overloaded. A routine membership change met three sleeping bugs at the same moment.

The cause was identified between 07:00 and 07:32. A remediation plan was finalised at 10:30. A controlled restart began between 14:15 and 14:30. Settlement resumed at 15:15. The Bank extended the settlement day from its then-scheduled 16:20 close to 20:00. By 18:00 around 99% of payments had settled by both value and volume, and by 20:00 all of them had. The day's total was 142,759 CHAPS payments worth £289.3 billion, against a forecast of about 145,000 payments and £276 billion. In other words, the day completed. Nothing was lost. Nine hours of the country's high-value payment capability were simply unavailable, and then they were available again, and the backlog cleared in five hours.

The human cost landed on the property market, because property completions are the CHAPS use case with a hard deadline and vans outside. A Law Society survey with 157 responses recorded that 18% of transactions completed within the usual times, 27% were delayed by up to three hours, 24% by more than three hours, and 30% did not complete until the next day or later. Against a typical Monday of around 2,340 residential sales, the review estimated approximately 700 transactions may not have completed until the next day.

The most instructive part of the report is not the failure but the contingency decision. The Bank had, and has, a contingency service: MIRS, the Market Infrastructure Resiliency Service, operated by Swift

on alternative technology at geographically remote sites, designed to let CHAPS keep settling without falling back to a deferred net model. It was not invoked. The review sets out why. Switching to MIRS was believed to be a one-way move in practice, because failing back to RTGS intraweek was a manual process carrying an increased risk of error if attempted overnight. MIRS lacked full functionality, notably the Liquidity Saving Mechanism, and offered reduced resilience relative to RTGS. And at the 14:20 decision meeting the operational and technical team advised that the fix-forward plan was well advanced and RTGS should be running by 15:00 to 15:15, which it was.

That is a rational decision and it is also the anatomy of a near miss. The contingency existed and was not used because using it was expensive and irreversible, which is exactly the condition under which contingency arrangements decay. The independent review's recommendations, all of which the Bank accepted, addressed precisely that: test intraweek failback from MIRS so that invoking it is not a one-way door; educate participants on MIRS scenarios so the decision is not made cold; build a loss-of-integrity contingency manual with reconciliation scripts and explicit MIRS decision criteria. Alongside those came governance and engineering changes, including reconstituting the RTGS board under the Deputy Governor for Markets and Banking, which first met in that form on 12 March 2015, deferring further functional changes pending business assurance, and building a proper testing regime with independent testing, regression testing and separated test and pre-production environments.

What "RTGS goes down" actually means

There is no second CHAPS. This is the uncomfortable structural fact. The £344 billion a day that moves through CHAPS has no alternative rail: Faster Payments cannot take a £4 billion money market payment, both because of the £1 million cap and because deferred net settlement of that value would create precisely the exposure CHAPS exists to eliminate. So resilience in CHAPS cannot be delivered by diversity of rails. It has to be delivered inside the one rail.

It is delivered in four layers. First, the primary RTGS infrastructure, now RT2, with its own redundancy. Second, MIRS, which is a genuinely different technology stack run by a different organisation in different locations, so that a defect in RTGS does not propagate. Third, the ability to extend the settlement day, to 20:00 as a contingency, which converts an availability failure into a lateness problem as long as the fault is fixed within the day. Fourth, manual and bilateral fallbacks such as party-to-party transfers and contingency transfers, which are slow, expensive, individually authorised and intended for a handful of critical payments rather than the day's flow.

The Bank now frames its operational performance in terms of impact tolerances for important business services rather than a single uptime number. In its 2024/25 annual report it documented five incidents between July 2024 and March 2025 and stated that impact tolerances were met in every case, with all payments settling by end of day. That framing is deliberate and worth understanding: for a system like this, the question is not "was there an outage" but "did the day complete". A ninety-minute interruption at 07:00 is an inconvenience. The same interruption at 16:30 is a housing chain that does not complete and a money market position that has to be rolled.

What changes next

Two changes are confirmed or consulted on at the time of writing, and both are worth tracking by anyone building on these rails.

The first is hours. On 24 February 2026 the Bank published a policy statement confirming an early morning extension: CHAPS settlement will begin at 01:30 rather than 06:00, targeted for September 2027 subject to confirmation with participants. Sending during the extended window will be optional for direct participants, but all participants will be able to receive from 01:30, and each institution de-

cides for itself when to credit received funds to customer accounts. The Bank will run an alert-and-respond support model in the early hours, with 24/7 cover retained for critical incidents.

On 18 May 2026 the Bank went further, consulting until 10 August 2026 on the path beyond that: weekend, specifically Sunday, and bank holiday settlement from 01:30 to 18:00 not before 2029; 22-hours-a-day operation across six days not before 2031; and a long-run target of either 22x7 or 23.5x7, with short daily closures for housekeeping. The stated drivers are innovation, including synchronisation and settlement of tokenised assets and stablecoin arrangements, better alignment with overseas markets for cross-border payments, and liquidity efficiency from more frequent settlement.

The second is data. The ISO 20022 enhanced data programme is not finished. Purpose codes and LEIs became mandatory for defined payment types from 1 May 2025; structured addresses become compulsory in effect from November 2026 when unstructured ones start being rejected; a structured remittance mandate is expected from November 2027. For anyone integrating with CHAPS through a direct participant, these are hard dates with rejection consequences, not aspirations.

A practitioner's summary

If you are choosing a rail, the decision tree is short. If the value exceeds £1 million, or the payment must be irrevocable on receipt, or a third party will act irreversibly on the credit the moment it lands, use CHAPS. If none of those is true, you are paying £25 to £30 for a certainty you do not need.

If you are operating CHAPS payments, five things matter more than the rest. Know your own bank's customer cut-off, which is earlier than the scheme's 17:40, and know that the conveyancing convention of 2pm is earlier still. Get the beneficiary details right the first time, because the rail has no reverse gear and your remedy is capped at £85,000 per claim for in-scope retail payments under the reimbursement rules that applied from 7 October 2024. Populate the purpose code correctly for property and financial institution payments, because since 1 May 2025 it is mandatory, not decorative. Understand that urgent and non-urgent are real settings with real consequences in minutes and in your bank's liquidity cost. And build your operational process on the assumption that one day in some year, RTGS will be unavailable at 11am, the day will be extended, and everything will settle by evening, because that is what the record actually shows happens.

The woman in the van gets her keys because a ledger at the Bank of England changed by £306,000 and cannot change back. Everything in this chapter, all the collateral, the matching cycles, the throughput targets, the contingency stack, the three named points of finality, exists to make that one sentence true 209,401 times a day.

38.98 Common wrong ideas

Wrong: CHAPS is the fastest UK payment rail, which is what the fee buys. **Right:** Faster Payments is faster, free and works at midnight on a Sunday; what CHAPS sells is finality in central bank money that cannot be undone.

Wrong: A bank needs a large pile of its own cash sitting idle to make CHAPS payments in the morning. **Right:** Sterling Monetary Framework participants generate intraday liquidity against pledged high-quality collateral with haircuts, so the constraint is eligible collateral and its opportunity cost, not owned cash.

Wrong: Every CHAPS payment settles strictly one at a time. **Right:** Legally each one does, but since 15 April 2013 the Liquidity Saving Mechanism settles groups of broadly offsetting non-urgent payments simultaneously, taking combined intraday requirements from around £21.2 billion to around £16.9 billion.

Wrong: The money in the seller's solicitor's client account is central bank money. **Right:** It is a claim on her bank; what settled at the Bank of England was a transfer between two banks, a causally linked but different event.

Wrong: The CHAPS deadline is 17:40. **Right:** That is the scheme deadline for customer payments; every bank sets an earlier customer cut-off, and the conveyancing convention drawn from the Standard Conditions of Sale puts the practical deadline at 2pm.

Wrong: Because a settled CHAPS payment is irrevocable, a fraud victim has no remedy. **Right:** The payment stays final, but since 7 October 2024 the Bank has applied reimbursement rules to in-scope retail CHAPS payments with a maximum of £85,000 per claim; the remedy is a different payment under a different rulebook.

Wrong: CHAPS has a maximum transaction value in the way Faster Payments does. **Right:** The scheme rules set no maximum at all, which is why a £4 billion money-market payment and a £306,000 house purchase travel on the same rail; what limits you is your own bank's policy.

Wrong: RTGS is a payment system. **Right:** RTGS is the Bank of England's accounting ledger, over £800 billion a day, carrying CREST and the net obligations of Bacs, Faster Payments, LINK, cheques and cards as well as CHAPS.

Wrong: A CHAPS payment can be recalled if you spot the error quickly. **Right:** A participant may request that the receiving participant return the funds, and it may refuse; a return is a new payment in the opposite direction, and nothing reverses.

Wrong: Resilience comes from being able to route round CHAPS on another rail. **Right:** There is no second CHAPS, so resilience is delivered inside the one rail through RT2, MIRS, extension of the settlement day to 20:00, and slow manual fallbacks for a handful of critical payments.

38.99 Chapter summary in 20 lines

1. CHAPS carried 52.7 million payments worth £87.5 trillion in 2024, roughly 0.5% of UK payment volume and 91% of sterling payment value, averaging £1.7 million a payment.
2. It is not fast, since Faster Payments is faster and free, and what it sells instead is finality in central bank money that cannot be undone.
3. Deferred net settlement is efficient because obligations cancel, but a failure before the net settles forces everybody's positions to be recalculated.
4. Real-time gross settlement removes that by settling each payment individually, in full, in central bank money, at the moment it is made.
5. The cost of gross settlement is liquidity, because a participant that sends before it receives needs a balance sitting ready to cover the difference.
6. Direct participants do not fund that from their own cash alone; they generate intraday liquidity against pledged collateral, extinguished by the end of the day.
7. Since 15 April 2013 the Liquidity Saving Mechanism has settled groups of offsetting non-urgent payments simultaneously, saving around £4 billion of intraday liquidity, about 20%.
8. Each of those payments still settles gross and individually in law, so CHAPS has borrowed netting's best trick without giving up gross settlement's guarantees.
9. RTGS is not a payment scheme but the Bank of England's ledger, on which over £800 billion settles daily, including CREST and the net positions of every retail rail.
10. The renewed core settlement engine, RT2, went live on 28 April 2025 alongside a revised tariff framework.

11. Finality is not one event: the rulebook names the Point of Entry, the Point of Irrevocability and the Point of Final Settlement.
12. Designation under the Financial Markets and Insolvency (Settlement Finality) Regulations 1999 is what makes that finality survive an insolvency, and therefore what lets the keys be released.
13. Tiering means the guarantee describes one leg only, with over 35 direct participants sitting above several thousand institutions reaching CHAPS through a correspondent on ordinary commercial credit terms.
14. Cut-offs come in layers — 17:40 at the scheme, earlier at each bank, and 2pm by conveyancing convention — and the earliest one governs your day.
15. Direct participants pay 48.7 pence per payment sent in 2026 plus a participation fee, £30,000 to open a settlement account and £90,000 to join, while the end customer pays £25 to £30.
16. That gap is not infrastructure cost but liquidity, manual high-value checks, fraud controls and the acceptance of operational risk on an irreversible instruction.
17. Wholesale financial activity dominates by value, property completion is the use the public sees, and corporate treasury is the third family.
18. Since 1 May 2025 purpose codes and Legal Entity Identifiers have been mandatory for defined categories, with PCOM, PDEP, PLDS, HLRP, HLST and PLRF naming the property cases.
19. The outage of 20 October 2014 shows the shape of failure: a routine membership change met three latent defects, nine hours were lost, MIRS was not invoked because using it was effectively a one-way door, and the day still completed by 20:00.
20. Everything in the system — collateral, matching cycles, throughput targets, the contingency stack, the three named points of finality — exists so that a ledger entry at the Bank of England changes and cannot change back.

Sources: Bank of England published material, principally the CHAPS service pages (operating hours, direct participant numbers, uses, and reimbursement position), "A brief introduction to the Real-Time Gross Settlement system and CHAPS", the payment and settlement statistics pages, the CHAPS Reference Manual version 28 April 2025 (points of entry, irrevocability and final settlement, Category A definition, contingency modes), the RTGS and CHAPS Annual Report 2024/25 (RT2 go-live, ISO 20022 milestones, incidents and impact tolerances, daily RTGS settlement value), the 2023 self-assessment against the Principles for Financial Market Infrastructures (Settlement Finality Regulations designation, intraday liquidity and collateral, throughput targets, MIRS), the RTGS service description (daily timetable and liquidity provision, mid-2010s edition), the RTGS and CHAPS fees pages (2025 and 2026 tariffs, participation and onboarding fees), the Quarterly Bulletin articles "How has the Liquidity Saving Mechanism reduced banks' intraday liquidity costs in CHAPS?" (2014) and "Tiering in CHAPS" (2013), the guidance on mandating ISO 20022 enhanced data in CHAPS (purpose codes and LEIs), the policy statement "Extending RTGS and CHAPS settlement hours – early morning extension" (24 February 2026) and the consultation "Extending RTGS and CHAPS settlement hours – next steps towards near 24x7 settlement" (18 May 2026), together with the independent review of the RTGS outage on 20 October 2014 and the Bank's published response (March 2015). Conveyancing completion practice from the Law Society Standard Conditions of Sale, fifth edition. All figures, fees and timings accurate at time of writing, August 2026.

39

CHAPTER

Addressing: Sort Codes, IBANs and Confirmation of Payee

39.0 What this chapter gives you

1. You will be able to explain why a payment to a valid account in somebody else's name arrives without a single system in the chain failing.
2. You will be able to take a sort code and account number apart and say exactly what each half addresses, and what neither of them proves.
3. You will be able to run a MOD11 and a DBLAL modulus check by hand, and say why summing the products instead of the digits of the products gives the wrong answer.
4. You will be able to state what a passing modulus check is worth, given that a MOD10 rule admits about one in ten arbitrary numbers and a MOD11 rule about one in eleven.
5. You will be able to generate and validate the two check digits of a British IBAN, and explain why a correctly formed IBAN always leaves a remainder of exactly 1 modulo 97.
6. You will be able to explain why a BIC can never address an account, and why its presence in a directory does not prove network reachability.
7. You will be able to read the Confirmation of Payee reason codes and say which ones signal the shape of invoice redirection fraud.
8. You will be able to explain why allocation of a sorting code is not reachability, and why the EISCD needs refreshing weekly rather than inferring institutions from digit ranges.
9. You will be able to say why a collection account moves the real address into an eighteen-character reference field with no check digits and no validation.
10. You will be able to explain why over 99 per cent of Faster Payments initiators now check names and APP losses still rose 19 per cent in 2025.

A payroll clerk at a scaffolding firm in Wolverhampton opens an email from a supplier she has paid twenty-two times. The invoice is for £61,400. The reference is the one she expects, the letterhead is right, the VAT number is right, and the message arrives inside the same thread as the last four invoices. Exactly one thing has changed. At the bottom of the page the bank details are different, with a short apologetic note: new banking provider, please update your records before you run Friday's payment.

She updates the records. The payment goes out on Friday. It arrives, in seconds, at an account that exists, at a bank that is regulated, in the name of somebody who is not the supplier. Nothing failed. Every system in the chain did exactly what it was built to do, which was to take an address and deliver to it.

This chapter is about that address: what it is made of, what the numbers inside it prove, what they cannot prove, and what was bolted on top of the whole arrangement in 2020 to try to catch precisely the

Friday afternoon described above. All limits, timings, figures and version numbers here are accurate at the time of writing, August 2026, and payments infrastructure changes.

The plain version

Think about how a letter gets to a house.

You write two different kinds of thing on an envelope. You write a name, and you write an address. Only one of them is used. The postal system reads the postcode and the house number, sorts the letter through a series of increasingly local buildings, and pushes it through a letterbox. Nobody at any stage compares the name on the envelope with the name of the person who lives there. If you put the right name and the wrong house number, the letter goes to the wrong house. If you put the wrong name and the right house number, the letter goes to the right house.

Bank payments were built the same way, and for the same reason: it is far faster to sort by number than to check by name.

The two numbers

In Britain the address has two parts.

The first is the **sort code**: six digits, usually printed in three pairs, like 20-00-00. Think of it as the postcode. It says which organisation, and historically which branch of that organisation, the money is going to.

The second is the **account number**: eight digits, like 55779911. Think of it as the house number on that street. It says which specific customer, inside that organisation, gets the money.

Together they are fourteen digits, and those fourteen digits are the entire address. There is no third element. When you pay a plumber £2,400 from your phone, you type a name into a box, but the name is a label for your own records. The money is moved by the fourteen digits and nothing else.

The trick hidden inside the numbers

Now, fourteen digits is a lot of digits to type, and people mistype. If one digit in a postcode is wrong, the letter usually bounces, because most wrong postcodes do not exist. But almost every combination of fourteen digits looks like a plausible bank address. So the banks did something clever: they built a hidden test into the account number itself.

Here is the idea in miniature. Suppose account numbers were only four digits long, and suppose the rule was this: multiply the first digit by 4, the second by 3, the third by 2 and the fourth by 1, add up the answers, and the total must divide exactly by 11.

Take the number 1325.

Digit	1	3	2	5
Multiplier	4	3	2	1
Answer	4	9	4	5

Four plus nine plus four plus five is 22. Twenty-two divided by eleven is exactly two, with nothing left over. So 1325 passes.

Now mistype it. Swap two digits, so you type 1352: 4 plus 9 plus 10 plus 2 is 25, which does not divide by 11. Rejected. Change one digit, so you type 1825: 4 plus 24 plus 4 plus 5 is 37, which does not divide by 11. Rejected.

That is the whole idea, and the real system is the same idea made bigger. The real test uses all fourteen digits, sort code and account number together, and each bank has its own set of multipliers. It is called a **modulus check**, and it runs before the payment screen will even let you press Next. It catches the overwhelming majority of typing mistakes, which is exactly what it was designed for and exactly the limit of what it does.

Writing the address for abroad

Every country invented its own address format, and they do not fit together. A German account is eighteen digits arranged differently; a French one has letters in it. So Europe agreed on a standard envelope you can put any country's domestic address inside, called an **IBAN**, which stands for International Bank Account Number.

A British IBAN looks like this:

```
GB29 NWBK 6016 1331 9268 19
```

Take it apart and there are no surprises in it:

Piece	Meaning
GB	The country: Great Britain
29	Two check digits, explained in a moment
NWBK	A short code for the bank, in this case NatWest
601613	The sort code you already know
31926819	The account number you already know

So a British IBAN is a British bank address with a country label and a checksum stapled to the front. There is nothing new inside it.

The two check digits are the interesting part, because they protect the whole string rather than just the account number. The recipe is odd but it works. First, move the four characters at the front to the back. Then turn every letter into a number, where A is 10, B is 11, and so on up to Z as 35. You now have one very long number. Divide it by 97. If the IBAN was typed correctly, the remainder is exactly 1. Every time. For any country.

Working it the other way produces the check digits in the first place. Start with the account details and two zeros where the check digits will go, convert to a number the same way, divide by 97, and you get a remainder of 69. Subtract that from 98 and you get 29. That is where the 29 in GB29 comes from. It is not a reference number, it is not chosen by the bank, and it is not a secret. It is arithmetic.

There is one more code you will see on international paperwork, the **BIC** or SWIFT code, which looks like NWBKGB2L. It names the bank, not the account: four letters for the institution, two for the country, two for the town. It is the equivalent of writing "Barclays, Leeds" on an envelope. Useful, but on its own it will not get money to a person.

The name on the envelope, finally

Return to the scaffolding firm in Wolverhampton. Every system did its job. The digits were valid, the sort code existed, the account existed, and the money arrived. What nobody did was look at the name.

In 2020, British banks started doing exactly that. The service is called **Confirmation of Payee**. When you type in a new payee, your bank sends the sort code, the account number and the name you typed to the bank at the other end, and asks a single question: is this the name on that account? The other

bank answers in about a second, and you get one of four answers on your screen before you have sent anything.

Yes. The name matches. Carry on.

Close. It is nearly right, and here is the actual name on the account: you typed "Dan Okafor" and the account says "Daniel Okafor". Check it and decide.

No. The name is wrong. You are shown no name at all, only a warning, because telling you the real name would let anyone with a keyboard harvest account holders' names.

Cannot check. The other bank did not answer in time, or the account holder has opted out, or that kind of account is not covered.

The scale of the thing it is fighting is not small. In 2025, people and businesses in the United Kingdom lost £576.4 million to payments they authorised themselves after being deceived, which was 19 per cent more than the year before, and about a third of all payment fraud losses. Banks reimbursed £354.3 million of it, roughly 61 per cent. That is the number Confirmation of Payee exists to bring down: not money stolen from accounts, but money pushed out of them by the account holder, into an address that was perfectly valid and completely wrong.

Where the plain version stops being true

The envelope is a good picture of routing and a poor picture of nearly everything else. Four corrections.

The first correction is that the address is not a place, is not permanent, and is very often not unique to one person. A house stands still. A sort code and account number pair is a row in a directory, and directories are edited constantly. Sort codes move between institutions when a bank is bought or a brand retired. Crucially, a sort code does not reliably tell you which bank holds the money: under indirect access, a sponsoring bank allocates sort codes from its own range to smaller firms that settle through it, so the institution named against the sort code and the institution where your payee actually banks can be two different companies. Meanwhile, the Current Account Switch Service moves customers between banks and redirects payments arriving at the old address to the new one, silently, for years. And a very large number of real-world accounts are **collection accounts**: one sort code and account number belonging to a platform, a broker or a savings marketplace, into which tens of thousands of customers pay, with the individual identified only by the payment reference. In that arrangement the fourteen digits are the building and the reference field is the flat number, and the reference field has no check digits, no validation and no protection whatsoever.

The second correction is that every check described above tests the number against itself, and none of them tests the number against reality. This is the single most expensive misunderstanding in the field. A modulus check answers one question: could this pattern of digits be a legitimate combination for this sort code? It does not say the account exists, is open, has ever been issued, or belongs to anyone. Worse, passing is weak evidence by construction. A modulus 10 check accepts roughly one in ten random numbers by chance; a modulus 11 check roughly one in eleven. Failing is strong evidence; passing is barely evidence at all. The IBAN check digits are weaker still in this respect: they verify that a string was transcribed correctly, and say nothing about whether the account exists or whether the bank code and sort code inside it even agree. And for some sort codes there is no arithmetic at all. The published rules are explicit that when a sort code falls outside every range in the weight table, no modulus check can be performed and the details should be presumed valid unless other evidence suggests otherwise. Presumed valid. That is the honest position of the whole domestic validation layer, and it sits a long way from the confidence implied by a green tick.

The third correction is that Confirmation of Payee is advisory, partial and deliberately fuzzy, and is none of the three things people assume. It is not a block: the payer sees the result and can carry on regardless, and in a competently run scam the payer has already been told to expect a mismatch and given a reason for it. It is not comprehensive: the responding bank can decline to answer, the account holder can opt out, certain account types are not supported, and payments where either party is outside the United Kingdom are outside the requirement entirely. And it is not exact. Name matching is intentionally tolerant, because real account names contain initials, married names, trading names, punctuation and the ordinary noise of human typing, and a literal comparison would reject correct payments all day. The "close match" answer exists because rejecting every payment to "J Smith" when the account says "John Smith" would have made the service unusable. Somebody at each responding institution tuned that tolerance, and the tuning is not published. Most fundamentally, the service checks whether a name matches an account. It cannot check whether the person behind that account is who they claim to be, or whether the goods will ever arrive. Against purchase scams, where the payee's real name really is on the account, it does almost nothing.

The fourth correction is that there is no single worldwide address format, the IBAN is not it, and Britain does not use IBANs domestically anyway. IBAN covers Europe and a long tail of adopters elsewhere; the United States, Canada, India, China, Japan and Australia do not use it. A domestic Faster Payment, a domestic CHAPS payment and a Bacs Direct Debit are every one of them addressed with a bare sort code and account number, and a British consumer can bank for forty years without seeing an IBAN. Nor is the BIC an address in the same sense: it identifies an institution, or a branch or department within one, and never an account. An IBAN without a BIC is usually a complete destination. A BIC without an account identifier is a building with no room number.

The technical version

The domestic identifier pair

The addressing primitive for all three principal sterling interbank rails — Bacs, Faster Payments and CHAPS — is a six-digit sorting code paired with an account number. Nothing in the domestic schemes routes on name, and nothing routes on IBAN.

Sorting codes are allocated by Pay.UK on behalf of the industry. Direct participants receive codes through the onboarding process for the payment system concerned; indirect participants receive codes sub-allocated by their sponsoring direct participant, and settle through that sponsor's agency arrangement. Pay.UK also allocates sorting codes to firms that issue GB IBANs for international payments even where those firms do not participate in a UK payment system at all. The operational rules are overseen by the Clearing Codes Operational Group, whose membership includes direct participants, Pay.UK and Vocalink.

The governing rule for practitioners is that allocation is not reachability. A sorting code must be separately registered with each payment system before it can be used in that system. A code registered for Bacs Direct Credit is not thereby usable for Direct Debit; a code registered for Faster Payments is not thereby usable for CHAPS. Systems that validate a sorting code's existence but not its scheme registration produce a characteristic class of rejection that looks like a data error and is in fact a reachability error.

The Extended Industry Sort Code Directory

The public extract of the central sorting code database is the **Extended Industry Sort Code Directory** (**EISCD**), successor to the Industry Sort Code Directory, published under Pay.UK's governance and distributed by Vocalink. It is the authority, and no heuristic about digit ranges should ever be used in its place.

For each sorting code the EISCD carries, among other fields, the branch title, the postal address and telephone number, the associated BIC, and clearing participation records showing which schemes that code is registered for and which settlement member it clears through. Subscription is on a quarterly fee basis with unlimited downloads; the guidance is that monthly refresh is the minimum and weekly is advisable, and many originators take it through their Bacs software rather than directly. For a Bacs originator or approved supplier this file is not optional infrastructure. It is the difference between rejecting a payment at capture and discovering the problem in an ARUCS or ARUDD report three working days later.

Historically the leading digits carry institutional meaning: blocks were assigned to the clearing banks when the six-digit format was introduced, and the legacy is visible in the ranges the large groups still occupy. It is a legacy and not a rule. New entrants have been allocated ranges with no historical logic, groups have merged and codes have transferred. Any code that infers the institution from the first two digits will eventually be wrong, silently, in production.

Account number normalisation

The account number is nominally eight digits, and institutions have never been uniform about it. Before any validation is attempted, the account number must be normalised to eight digits according to the published rules. Shorter numbers are zero-prefixed. Several institutions issue nine- or ten-digit numbers and specify which eight digits are significant, and in one case specify that a digit of the account number displaces a digit of the sorting code before checking. Building societies add a further layer: many hold balances against a **roll number** rather than an account number, with payments routed to a single society account and allocated internally from the reference field. Roll numbers have institution-specific formats, cannot be modulus checked, and are the reason the Confirmation of Payee schema carries a secondary identification field at all.

Modulus checking

Modulus checking is the domestic self-consistency test on the fourteen-digit pair. The specification is published by Vocalink as *Validating Account Numbers: UK Modulus Checking*; at the time of writing the current release is version 8.70, dated 10 February 2026. Two reference files accompany it: `valacdos.txt`, the weighting table, and `scsubtab.txt`, the sorting code substitution table used by certain exceptions.

Positions are named by letter throughout the specification. The six sorting code digits are `u v w x y z`; the eight account number digits are `a b c d e f g h`. Every rule is expressed against those fourteen positions.

Rows in `valacdos.txt` are fixed-length, space-separated, and consist of a sorting code range start, a range end, a modulus check type, fourteen weights for `u` through `h`, and an optional exception number. A row therefore looks like this in shape:

```
010004 016715 MOD11 0 0 0 0 0 0 8 7 6 5 4 3 2 1
```

A sorting code may appear in two rows. That means two checks are specified, and both are ordinarily required to pass, except where a numbered exception states that either check passing is sufficient.

Three algorithms are defined.

Type	Method
MOD10	Multiply each of the fourteen digits by its weight, sum the products, divide by 10. Remainder zero passes.
MOD11	Multiply each of the fourteen digits by its weight, sum the products, divide by 11. Remainder zero passes.
DBLAL	Multiply each digit by its weight, then sum the *individual digits* of every product rather than the products themselves, and divide the total by 10. Remainder zero passes.

The digit-summing step in DBLAL is where most reimplementations go wrong, and it is worth doing once by hand.

Worked MOD11. Sorting code 000000, account number 58177632, weights `0 0 0 0 0 0 7 5 8 3 4 6 2 1`.

Position	u	v	w	x	y	z	a	b	c	d	e	f	g	h
Digit	0	0	0	0	0	0	5	8	1	7	7	6	3	2
Weight	0	0	0	0	0	0	7	5	8	3	4	6	2	1
Product	0	0	0	0	0	0	35	40	8	21	28	36	6	2

The sum is 176. 176 divided by 11 is 16 with no remainder. The check passes.

Worked DBLAL. Sorting code 499273, account number 12345678, weights `2 1 2 1 2 1 2 1 2 1 2 1 2 1`.

Position	u	v	w	x	y	z	a	b	c	d	e	f	g	h
Digit	4	9	9	2	7	3	1	2	3	4	5	6	7	8
Weight	2	1	2	1	2	1	2	1	2	1	2	1	2	1
Product	8	9	18	2	14	3	2	2	6	4	10	6	14	8
Digits summed	8	9	9	2	5	3	2	2	6	4	1	6	5	8

The sum of the bottom row is 70. 70 divided by 10 is 7 with no remainder. The check passes. Note that summing the *products* instead would give 106, and 106 fails. The two routes disagree, and only one of them is the specification.

Fourteen numbered exceptions modify the standard procedure. They exist because individual institutions carried their own pre-existing check schemes into the industry standard and were not going to renumber their customers. Summarised, and always to be implemented from the current specification rather than from any summary:

Exception	Effect
1	Add 27 to the total before dividing.

Exception	Effect
2 and 9	Alternative weightings selected by the values of `a` and `g`, for certain euro accounts.
3	Skip the double alternate check where `c` is 6 or 9.
4	The remainder must equal the two-digit check value formed by `g` and `h`.
5	Substitute a sorting code from `scsubtab.txt` and apply specific check-digit logic.
6	Foreign currency accounts: treat as valid where `a` is 4 to 8 and `g` equals `h`.
7	Zeroise the weights for positions `u` to `b` where `g` is 9.
8	Substitute sorting code 090126 for checking purposes only.
10 and 11	Either check passing deems the account valid.
12 and 13	Either check passing deems the account valid.
14	If the first MOD11 check fails and `h` is 0, 1 or 9, drop `h`, prefix a zero, and recheck.

Exception 14 is worth working through because it shows how much of the specification is history rather than mathematics. Sorting code 180002, weights `0 0 0 0 0 0 8 7 6 5 4 3 2 1`.

Account 98093517 gives products 72, 56, 0, 45, 12, 15, 2 and 7 across positions `a` to `h`, summing to 209. 209 divided by 11 is 19 exactly. It passes on the first check and exception 14 is never invoked.

Account 00000190 sums to 21. 21 divided by 11 leaves a remainder of 10, so the first check fails. Because `h` is 0, exception 14 applies: drop the trailing digit and prefix a zero to make 00000019. That sums to 11, which divides by 11 exactly. The account passes.

Two properties of modulus checking must be stated plainly to anyone building on it. First, the specification itself directs that where a sorting code is not found in the weighting table, no check can be performed and the details should be presumed valid. Second, a passing check carries very little information. A MOD10 rule admits about one in ten arbitrary numbers and a MOD11 rule about one in eleven. Modulus checking is a typo filter of very high value at the point of capture and a fraud control of no value whatsoever.

The IBAN

The IBAN is specified by **ISO 13616**, with Swift acting as the registration authority and publishing the IBAN Registry that lists each participating country's format.

The United Kingdom's entry is:

Attribute	Value
IBAN structure	`GB2!n4!a6!n8!n`
IBAN length	22
BBAN structure	`4!a6!n8!n`
BBAN length	18
Bank identifier	Positions 1 to 4 of the BBAN, `4!a`
Branch identifier	Positions 5 to 10 of the BBAN, `6!n`
Registry example	`GB29NWBK60161331926819`

Read in the registry's notation, `2!n` means exactly two numeric characters, `4!a` exactly four alphabetic, and `!` denotes a fixed rather than maximum length. So a GB IBAN is: the literal country code GB, two check digits, a four-letter bank identifier which is conventionally the first four characters of the institution's BIC, the six-digit sorting code as the branch identifier, and the eight-digit account number. The sorting code and account number are carried unaltered. A GB IBAN contains no information a domestic payment instruction does not already have.

Other formats vary in length and content, which is why length-based validation alone is insufficient:

Country	Structure	Length	Registry example
United Kingdom	`GB2!n4!a6!n8!n`	22	`GB29NWBK60161331926819`
Germany	`DE2!n8!n10!n`	22	`DE89370400440532013000`
France	`FR2!n5!n5!n11!c2!n`	27	`FR1420041010050500013M02606`
Netherlands	`NL2!n4!a10!n`	18	`NL91ABNA0417164300`

The check digits are computed under **ISO/IEC 7064, MOD 97-10**. The generation procedure is:

1. Assemble the BBAN, then append the country code and the two literal characters `00`.
2. Replace each letter with two digits: A is 10, B is 11, through to Z as 35.
3. Interpret the result as a single integer and take it modulo 97.
4. Subtract the remainder from 98. If the answer is a single digit, pad it to two digits with a leading zero.

Worked through for the registry example. The BBAN is `NWBK60161331926819`. Appending `GB00` gives `NWBK60161331926819GB00`. Converting the letters — N is 23, W is 32, B is 11, K is 20, G is 16, B is 11 — produces:

```
233211206016133192681916 1100
```

That integer modulo 97 is 69. 98 minus 69 is 29, giving the check digits `29` and the complete IBAN `GB29NWBK60161331926819`.

Validation is the same arithmetic run the other way. Move the first four characters to the end, convert letters to digits, and take the result modulo 97. A correctly formed IBAN always yields exactly 1. For the example, rearranging gives `NWBK60161331926819GB29`, which converts to `233211206016133192681916 1129`, and that modulo 97 is 1.

That integer is twenty-eight digits long and will overflow a 64-bit type, so implementations that do not use arbitrary-precision arithmetic must compute the modulus piecewise. The standard technique takes the first nine digits, reduces modulo 97, then repeatedly prefixes the running remainder to the next seven digits and reduces again. For the string above:

Step	Value reduced	Remainder
1	233211206	23
2	230161331	22
3	229268191	58
4	5861100	69

Which is the 69 the generation step needs. Nine and seven are chosen because the largest intermediate value stays inside the range of a 32-bit signed integer.

MOD 97-10 detects all single-character substitutions and all transpositions of adjacent characters, which is the entire design goal. It detects nothing about the account behind the string.

The BIC

The BIC is specified by **ISO 9362**, with Swift as registration authority. Since the 2022 revision the expansion is Business Identifier Code rather than Bank Identifier Code, reflecting that non-bank entities hold them.

The structure is four characters of business party prefix, two characters of ISO 3166-1 country code, and two characters of business party suffix, giving the eight-character BIC8. An optional three-character branch identifier extends this to the eleven-character BIC11, used to identify a specific location, department, service or unit of the same business party. By convention `XXX` in the branch position denotes the head office, and a BIC8 and the corresponding BIC11 ending `XXX` address the same party.

Swift distinguishes **connected** BICs, which have access to the Swift network, from **non-connected** BICs, registered for reference purposes only and with no right or authorisation to exchange messages over it. A BIC present in a directory is therefore not proof of reachability, any more than an EISCD sorting code is proof of scheme registration. The second character of the business party suffix carries a convention of its own, and Swift has published enough about it to settle the point. Swift's information paper on the ISO 9362 implementation states that addresses ending in a zero in position eight are used only for the Test and Training service, that they are not published in the directory of BICs, and that this convention will not change. A one in the same position historically meant the BIC was not connected, and Swift's published BIC Policy still illustrates a non-connected BIC as an eight-character BIC with a one in the eighth position. The connectivity reading, however, is no longer safe to compute. Since November 2018 an organisation connecting or disconnecting no longer needs a new BIC, and Swift's own guidance is that users should rely on the network connectivity status attribute in the SWIFTRef directories rather than on the intelligence once carried in position eight.

The essential point is categorical: the BIC identifies an institution or a unit within one, never an account. In cross-border messaging it names the parties and the route; the account is named by the IBAN or by a domestic identifier carried alongside.

Confirmation of Payee

Confirmation of Payee is an account name-checking overlay operated under Pay.UK's standards, launched in 2020. It is not a payment system and it moves no money. Architecturally it is a peer-to-peer API service: there is no central switch, only a directory identifying the organisations in the ecosystem, and requests pass directly between the requesting PSP and the responding PSP. Account-holding PSPs may build directly or use a technical service provider; smaller PSPs may reach the service through a CoP aggregator, of which Pay.UK lists nine approved.

The service is payment-system agnostic: Pay.UK describes it as applicable to domestic UK payments across CHAPS, Faster Payments and Bacs, covering both credit and debit flows, and a companion capability, Payer Name Verification, extends name checking to the setting up of Bacs Direct Debits.

A request carries a small, fixed set of fields:

Field	Content
SchemeName	`SortCodeAccountNumber`
Identification	Fourteen numerics: six-digit sorting code followed by eight-digit account number

Field	Content
Name	The payee name as entered, up to 140 characters
AccountType	`Personal` or `Business`
SecondaryIdentification	Optional secondary reference data, up to 140 characters

The matching is performed by the responding PSP against its own records; the requester never receives the account name on a negative result. The response carries a boolean match indicator and, where the match is not a straight yes, a reason code. The codes in use are:

Code	Meaning
ANNM	The responder performed the match and confirms the name does not match
MBAM	Close match; the responder returns the actual account name
BANM	A personal account was expected; a business account was found, with a name match
PANM	A business account was expected; a personal account was found, with a name match
BAMM	A personal account was expected; a business account was found, with a close name match
PAMM	A business account was expected; a personal account was found, with a close name match
AC01	The account does not exist in the responder's books
IVCR	The secondary reference data did not locate the account
ACNS	The account is not supported for CoP
OPTO	The payee has opted out of CoP
CASS	The account has been switched through the Current Account Switch Service
SCNS	The sorting code does not belong to the responding participant
SCNF	Sorting code not found at the routing layer

The account type distinction and the four codes covering it are frequently overlooked in implementation and are among the most operationally useful signals the service produces. A payment addressed to a business but landing on a personal account is a common shape for invoice redirection fraud.

Phase 1, Phase 2 and the regulatory timetable

Phase 1 went live in 2020 and covered the six largest banking groups, which between them accounted for around 90 per cent of Faster Payments and CHAPS transactions. It required full Open Banking membership to participate and, structurally, only supported PSPs holding their own unique sorting codes and identifying customers by sorting code and account number alone. That left two gaps: firms without their own sorting code, and accounts identified by something else, such as building society roll numbers or credit card numbers.

Phase 2 addressed both. A dedicated CoP-only role profile was created in the Open Banking Directory so that a PSP could enrol for name checking without taking on the full Open Banking estate, and

the standard was extended to carry **secondary reference data** (**SRD**) so that roll numbers and card numbers could identify the target account. Pay.UK's guidance gives the two canonical SRD cases as a sixteen-digit card number when paying a credit card bill and a roll number for some building society accounts.

The regulatory backing sits with the Payment Systems Regulator. Specific Direction 10 mandated Phase 1 for the six largest groups. **Specific Direction 17**, issued in October 2022, expanded the mandate in two tranches: Group 1 PSPs, named in the direction, by 31 October 2023, and Group 2 PSPs by 31 October 2024. Group 2 is defined by having a unique sorting code, or being a building society, and sending funds for customers over CHAPS or the Faster Payment Service into accounts consumers and businesses can pay into and be paid from. Directed firms must implement both the sending and the responding sides unless they hold only account types where responding alone is appropriate.

The exclusions matter as much as the scope. The direction does not apply where either party is outside the United Kingdom, to loan repayments including mortgages, credit cards and buy-now-pay-later, to a PSP moving its own money, or to bulk and batch payments submitted by indirect PSPs.

On coverage, the PSR's compliance report of 18 November 2025 recorded that over 99 per cent of organisations initiating Faster Payments transactions in the United Kingdom offer CoP checks, that over 320 organisations offer the service, and that over two million checks are completed each day.

Where the address sits on each rail

Addressing is the same fourteen digits everywhere, but the container differs.

Bacs carries payment instructions in the Standard 18 format: fixed-position records of 100 characters, with an optional extension for multi-processing files. The destination sorting code occupies positions 1 to 6 and the destination account number positions 7 to 14, so the fourteen-digit address is literally the first fourteen characters of the record. Position 15 is the destination account type, positions 16 to 17 the transaction code, positions 18 to 23 and 24 to 31 the originating sorting code and account number, positions 32 to 35 a four-character free format field, positions 36 to 46 the amount in pence, positions 47 to 64 the originator's name in eighteen characters, and positions 65 to 82 the eighteen-character payment reference. Multi-processing files carry a processing date at positions 101 to 106 in the format bYYDDD.

The eighteen-character reference is the pinch point. In a collection-account architecture the reference is the real address, and eighteen characters with no validation is all the protection it has. Pay.UK also publishes an ISO 20022 to Standard 18 translation guide for originators moving between the two.

CHAPS has carried ISO 20022 messages since 19 June 2023, with customer credit transfers as `pacs.008`. The Bank of England has been progressively mandating enhanced data within those messages. From 1 May 2025, Legal Entity Identifiers became mandatory for payments between financial institutions, and Purpose Codes became mandatory both for payments between financial institutions and for property transactions, using six specified codes: HLRP, HLST, PLDS, PDEP, PCOM and PLRF. Hybrid addresses — town name, country and up to two unstructured lines — were enabled from November 2025, and fully unstructured addresses are to be rejected by the CHAPS validation library from November 2026. The Bank has consulted on extending mandatory enhanced data to all initiation channels and Purpose Codes to all CHAPS payments from November 2027. The direction of travel is unambiguous: the address is being made structured and machine-readable, party by party and field by field.

Faster Payments carries the same sorting code and account number in scheme messages, with Pay.UK publishing an ISO 20022 standards library as part of the migration away from the legacy message

format. The relevant scheme constraint for addressing purposes is the £1 million maximum per transaction; individual PSPs set their own lower limits by channel, account type and payment type, and those limits, not the scheme limit, are what a customer meets in practice.

Open Banking payment initiation is an API layer over these rails, chiefly Faster Payments. It changes who constructs the instruction, not how the destination is addressed.

APP fraud and the reimbursement regime

Confirmation of Payee was introduced because addressing by number alone left one specific hole, and the hole has a name: authorised push payment fraud, in which the customer is deceived into instructing a payment to an account the fraudster controls. Every technical control described in this chapter is satisfied in an APP scam. The modulus check passes. The IBAN, if there is one, validates. The sorting code is registered. The account exists.

The regulatory response went beyond name checking. Since **7 October 2024**, the PSR's reimbursement requirement has applied to in-scope APP scam payments made over Faster Payments and CHAPS. The consolidated policy sets a maximum of £85,000 per claim, permits a claim excess of up to £100 which a PSP may set lower or not apply at all, and requires reimbursement within five business days, with defined circumstances in which the clock may be stopped and an absolute backstop of the thirty-fifth business day following the report. The cost is split evenly: receiving PSPs must pay sending PSPs 50 per cent of the reimbursement made to the consumer. A consumer standard of caution exception allows a claim to be refused for gross negligence against four defined requirements, including responding to interventions, reporting within thirteen months, sharing information and reporting to the police. Neither the standard of caution exception nor the excess may be applied to vulnerable consumers.

The scale, from UK Finance's 2026 Annual Fraud Report covering 2025: total payment fraud losses of £1.28 billion across more than four million cases, of which APP fraud accounted for £576.4 million, up 19 per cent year on year and 32 per cent of all losses. Investment fraud within that rose 40 per cent to £221.5 million. Around two-thirds of APP fraud originated online and 17 per cent through telecommunications. Firms reimbursed £354.3 million, some 61 per cent of APP losses.

Those two facts sit together and explain the shape of the whole control stack. Coverage of name checking is above 99 per cent of Faster Payments initiators, and APP losses still rose 19 per cent. Confirmation of Payee closed the misdirection problem. It did not close the deception problem, because it was never capable of doing so.

The summary a practitioner should carry

A UK domestic payment is addressed by six digits of sorting code and eight digits of account number, and by nothing else. The sorting code is allocated by Pay.UK, sub-allocated by sponsors for indirect participants, and must be separately registered with each payment system it is to be used in; the EISCD is the authority for both ownership and scheme registration, and it should be refreshed weekly. Account numbers must be normalised to eight digits before validation.

Modulus checking, specified by Vocalink and driven by `valacdos.txt` and `scsubtab.txt`, runs one or two of MOD10, MOD11 or DBLAL against the fourteen positions `u` to `h`, subject to fourteen numbered exceptions. It is a high-value typo filter, it admits roughly one in ten or one in eleven arbitrary numbers by chance, and where a sorting code is absent from the table the specification directs that the details be presumed valid.

The IBAN under ISO 13616 wraps the same domestic pair in a country code and two MOD 97-10 check digits computed under ISO/IEC 7064; validation is the rearranged string modulo 97 equalling 1, and the check protects transcription and nothing else. The BIC under ISO 9362 names an institution or a unit within one, never an account, and its presence in a directory does not imply network connectivity.

Confirmation of Payee is a peer-to-peer API overlay, mandated in tranches under PSR Specific Direction 17 to 31 October 2023 and 31 October 2024, covering Faster Payments and CHAPS and excluding international payments, loan repayments, a PSP's own funds and indirect bulk submissions. It returns a match indicator and a reason code, performs matching at the responding PSP, and never reveals a name on a mismatch. It catches the wrong address. It cannot catch the wrong intention, which is why the reimbursement regime of 7 October 2024 exists alongside it and why, in a system where over 99 per cent of Faster Payments initiators check names, £576.4 million still went to the wrong people in 2025.

The scaffolding firm's payroll clerk had every number right. That was always the problem.

39.98 Common wrong ideas

Wrong: The payee name is part of the address and is checked before delivery. **Right:** A UK domestic payment is routed by six digits of sorting code and eight of account number and by nothing else; the name is a label for the payer's own records.

Wrong: A passing modulus check means the account exists. **Right:** It tests the fourteen digits against themselves only, and a MOD10 rule admits about one in ten arbitrary numbers while a MOD11 rule admits about one in eleven; failing is strong evidence, passing is barely evidence at all.

Wrong: Every sorting code can be modulus checked. **Right:** Where a sorting code falls outside every range in the weighting table no check can be performed, and the published specification directs that the details be presumed valid.

Wrong: A sorting code identifies the institution holding the money, and once allocated it can be paid on any rail. **Right:** Sponsors sub-allocate codes from their own range to indirect participants, so the named institution and the payee's actual bank can differ, and a code must be separately registered with each payment system before it is usable there.

Wrong: The IBAN check digits prove the account is real. **Right:** MOD 97-10 detects all single-character substitutions and all adjacent transpositions and nothing else, saying nothing about whether the account exists or whether the bank code and sorting code inside the string even agree.

Wrong: A BIC identifies an account and proves you can send to it. **Right:** The BIC names an institution or a unit within one, never an account, and non-connected BICs are registered for reference only with no right to exchange messages over the network.

Wrong: British payments are addressed by IBAN. **Right:** Domestic Faster Payments, CHAPS and Bacs Direct Debits all carry a bare sorting code and account number, and a British consumer can bank for forty years without seeing an IBAN.

Wrong: Confirmation of Payee blocks a payment when the name does not match. **Right:** It is advisory; the payer sees the result and may carry on, and in a competently run scam the payer has already been told to expect a mismatch and given a reason for it.

Wrong: Confirmation of Payee covers every payment. **Right:** The direction excludes payments where either party is outside the United Kingdom, loan repayments, a PSP moving its own money and indirect bulk submissions, and the payee can opt out.

Wrong: Name matching is an exact comparison. **Right:** It is deliberately tolerant, which is why the close-match answer exists, because rejecting every payment to "J Smith" against an account reading "John Smith" would make the service unusable; each responder tunes that tolerance and the tuning is not published.

39.99 Chapter summary in 20 lines

1. A UK domestic payment is addressed by a six-digit sorting code and an eight-digit account number, and by nothing else.
2. The name typed into the payee box is a label for the payer's records, because no domestic sterling rail routes on name.
3. Modulus checking is a self-consistency test built into those fourteen digits, specified by Vocalink and driven by the weighting table `valacdos.txt` and the substitution table `scsubtab.txt`.
4. Three algorithms are defined, MOD10, MOD11 and DBLAL, and DBLAL's step of summing the digits of each product rather than the products themselves is where reimplementations go wrong.
5. Fourteen numbered exceptions modify the standard procedure, because institutions carried their own pre-existing check schemes into the industry standard rather than renumber their customers.
6. A passing check carries very little information, and where the sorting code is absent from the weighting table the specification directs that the details be presumed valid.
7. Modulus checking is therefore a typo filter of very high value at the point of capture and a fraud control of no value whatsoever.
8. Sorting codes are allocated by Pay.UK, sub-allocated by sponsors to indirect participants, and must be separately registered with each payment system, so allocation is not reachability.
9. The Extended Industry Sort Code Directory is the authority for both ownership and scheme registration, and should be refreshed weekly rather than guessed at from leading digits.
10. Account numbers must be normalised to eight digits before validation, and building society roll numbers cannot be modulus checked at all.
11. Very many real accounts are collection accounts, where the fourteen digits are the building and an unvalidated eighteen-character reference is the flat number.
12. The IBAN under ISO 13616 wraps the same domestic pair in a country code and two check digits, adding no information a domestic instruction does not already carry.
13. Those check digits are computed under ISO/IEC 7064 MOD 97-10, and validation rearranges the string, converts letters to digits and takes the remainder modulo 97, which is always exactly 1.
14. The resulting integer overflows a 64-bit type, so implementations without arbitrary-precision arithmetic must reduce it piecewise, nine digits and then seven at a time.
15. MOD 97-10 protects transcription and detects nothing about the account behind the string.
16. The BIC under ISO 9362 names an institution or a unit within one, never an account, and directory presence does not imply network connectivity.
17. Confirmation of Payee, launched in 2020, is a peer-to-peer API overlay with no central switch, matching at the responding PSP and never revealing a name on a mismatch.
18. Its reason codes carry more than a yes or no, and the four account-type codes flag a business payment landing on a personal account, a common shape for invoice redirection fraud.
19. Specific Direction 17 mandated the service in tranches to 31 October 2023 and 31 October 2024, excluding international payments, loan repayments, a PSP's own funds and indirect bulk submissions.
20. Over 99 per cent of Faster Payments initiators now check names and APP losses still rose 19 per cent to £576.4 million in 2025, because the address was never the thing that was wrong.

Sources: Pay.UK, including the Confirmation of Payee service pages and FAQs, the Extended Industry Sort Code Directory pages, sort code information and allocation guidance, Faster Payment System transaction limits, and the ISO 20022 to Standard 18 translation guide; Vocalink, Validating Account Numbers: UK Modulus Checking, version 8.70 dated 10 February 2026; Swift, the IBAN Registry as ISO 13616 registration authority, and the BIC pages for ISO 9362; Payment Systems Regulator, Specific Direction 17 and the associated requirements pages, CP21/6 on Confirmation of Payee Phase 2, the Confirmation of Payee compliance report of 18 November 2025, and PS25/5, the consolidated policy statement on the APP scams reimbursement requirement; Bank of England, policy statement on mandating ISO 20022 enhanced data in CHAPS; UK Finance, Annual Fraud Report 2026 covering 2025; Bacs Standard 18 message implementation guidance.

40

CHAPTER

SEPA

40.0 What this chapter gives you

1. You will be able to explain why there is no SEPA computer and no SEPA clearing house, and say what Europe built instead.
2. You will be able to distinguish scheme scope from legal scope, and explain why a euro transfer from Dublin to Zurich uses the same rulebook without the EU charge-equality rule.
3. You will be able to state the SCT Inst timing budget as five, seven and nine seconds, and say what the tenth second belongs to.
4. You will be able to explain why sanctions screening had to be taken out of the transaction path before instant payments could be made compulsory.
5. You will be able to say who holds a SEPA mandate, why the debtor's bank never sees the signature, and what the Unique Mandate Reference and Creditor Identifier are for.
6. You will be able to compare SEPA Direct Debit Core with UK Bacs Direct Debit and explain why the British model can promise an immediate refund with a straight face.
7. You will be able to say why the business-to-business scheme abolishes the refund right and what obligation it puts on the paying institution instead.
8. You will be able to name the five things that must all be true at once for a Cork to Lisbon payment to be domestic in character.
9. You will be able to explain why SEPA never had to migrate to ISO 20022, and what the 15 November 2026 address deadline requires of every message you send.
10. You will be able to list what SEPA did not unify: cards, currencies, the front end and national variation.

Europe did something in payments that almost nobody else has managed. It took thirty-odd countries with thirty-odd national payment systems, thirty-odd account numbering conventions, thirty-odd sets of file formats and thirty-odd sets of consumer protection rules, and it made a euro payment between any two of them behave, from the customer's side of the counter, like a payment inside one country.

It did not do this by building a single machine. There is no SEPA computer. There is no SEPA clearing house. What Europe built instead was a set of rulebooks, a legal obligation to follow them, and a shared message format, and then it let the existing infrastructure compete underneath. That distinction is the whole chapter. Understand it and the rest of SEPA falls into place. Miss it and you will spend years looking for a network that does not exist.

This chapter covers what SEPA unified and what it conspicuously did not, the four schemes that carry the traffic, the mandate model that makes European direct debits work differently from British ones,

the reachability rules that make a payment from Cork to Lisbon domestic in character, and the fact that SEPA was born speaking ISO 20022 while everyone else was still translating.

The plain version

Imagine that every country in Europe has its own postal service, and that each one uses a different shape of envelope.

In Germany the envelopes are tall and narrow. In France they are square. In Ireland the address goes on the back. Each country has its own rules about how long a letter takes, its own price for a stamp, and its own way of writing a house number. Posting a letter inside your own country is easy, because you learned the shape of your own envelope as a child. Posting a letter to the next country along is a small nightmare. You need a special form, you pay a special price, and it takes an unpredictable number of days.

Now imagine that all of those countries sat down together and agreed three things.

First, one envelope. Same size, same shape, same place for the address, everywhere.

Second, one way of writing an address, which always starts with the country. So instead of a house number that means something only to the local postman, everyone writes an address that any post office on the continent can read at a glance.

Third, and this is the one that changed people's lives, the same price. Posting a letter from Ireland to Portugal costs what posting a letter from Cork to Dublin costs. Not a bit more. The same.

That is SEPA. It stands for the Single Euro Payments Area, and it is an agreement about the shape of the envelope.

A payment, with real numbers

Aoife lives in Cork. She has found a second-hand guitar advertised by Nuno, who lives in Lisbon, for 480 euros.

In 1999 this would have been an adventure. She would have gone into a branch, filled in an international payment form, given the name and address of Nuno's bank, paid a fee somewhere between fifteen and thirty euros, and waited three to five days while wondering whether Nuno would receive 480 euros or 462 euros after somebody along the way took a slice.

Today she opens her banking app and types one thing: Nuno's account address. It looks like this.

PT50 0002 0123 1234 5678 9015 4

The first two letters are PT, for Portugal. That is the whole trick. Aoife's bank does not need to know anything about Portuguese banking to know where this payment is going, because the address tells it. Her own account address starts IE. Her German friend's starts DE. Everyone in the area writes their account address the same way, and it always begins with the country.

She types the amount, 480.00, and Nuno's name. Her app checks the name against the account before she confirms, and tells her it matches. She presses send.

The money arrives in Nuno's account in about four seconds. He gets a notification on his phone while Aoife is still looking at hers. It is a Sunday evening. No bank in either country is open.

The fee Aoife pays is whatever her bank charges her to send 480 euros to someone in Cork. If that is nothing, this is nothing. Nuno receives 480.00, not 479.20. Nobody in the middle takes a slice, because the rules say nobody may.

A regular payment, with real numbers

The second thing SEPA agreed a shape for is the opposite arrow: the payment you let somebody else pull.

Nuno subscribes to a language app run by a company in Dublin. It costs 12.99 euros a month. Instead of him remembering to pay every month, he signs a permission slip that says the Dublin company may take 12.99 euros from his Portuguese account on the fifteenth of each month.

That slip is the important object. In Europe, the company keeps it. Nuno's bank in Lisbon never sees his signature. What his bank sees, when the first collection turns up, is a claim from the Dublin company that says, in effect, we have a signed permission from this person, here is its reference number, here is our registered collector number, please pay.

Three rules protect Nuno.

The company must tell him in advance what it is going to take and when. The default is fourteen days' notice, though he and the company can agree something different.

The collection has to arrive at his bank at least one working day before the day it is due, so that his bank has time to look at it rather than having it dumped on the doorstep at the last minute.

And if he changes his mind, he can ring his bank within eight weeks of the money leaving and ask for it back, and his bank must give it back without asking him why. If he says he never signed anything at all, he has thirteen months.

That is a SEPA Direct Debit. It works identically whether the collector is in Dublin and the payer in Lisbon, or both are in the same street in Munich.

What SEPA is not

Two things are worth saying plainly at the dinner table, because they are the source of most confusion.

SEPA is euros only. If Aoife wanted to send Swedish kronor to Stockholm, none of this applies. The euro is the point. The area is defined by the currency, not by the European Union.

And SEPA is not a company. There is no SEPA. If you go looking for the building, there is no building. It is an agreement about the envelope, written down as a set of rulebooks, made compulsory by European law. The actual sorting offices, the machines that move the money, are separate businesses that compete with each other, and every one of them has agreed to use the same envelope.

Where the plain version stops being true

SEPA is a rulebook, not a rail

The envelope analogy hides the entire infrastructure layer, and the infrastructure layer is where a practitioner spends their life.

A SEPA payment does not travel over "SEPA". It travels over a clearing and settlement mechanism, of which there are many. In the euro area alone the European Central Bank counted thirty-five retail payment systems processing around 55.7 billion transactions in the first half of 2025. The pan-European ones are STEP2-T and RT1, both operated by EBA CLEARING, and TIPS, operated by the Eurosystem.

There are also large national systems, of which CORE in France is the biggest. Two of the three largest retail systems in the euro area by volume, STEP2-T and CORE, are SEPA systems. The largest, MCMS, is Mastercard's, which tells you something about what SEPA did not cover.

Underneath all of them sits central bank money. Deferred net obligations from STEP2-T settle in T2, the Eurosystem's real-time gross settlement service. Instant payments settle across dedicated accounts in TIPS. When you say "SEPA payment" you have named the rulebook and said nothing about which of these carried it, and the choice matters enormously for cut-offs, liquidity and failure modes.

"Domestic in character" is a legal claim, and it has edges

Aoife's payment to Nuno is domestic in character because European law makes it so, not because the physics changed. Three specific edges are hidden.

The charging rule is a law with a geographical limit. It applies within the European Union. But the geographical scope of the EPC schemes covers forty-one countries and territories, eleven of which are outside the European Economic Area, including the United Kingdom, Switzerland, Monaco, Andorra, San Marino and Serbia. A euro credit transfer from Dublin to Zurich uses the same rulebook and the same message, but the EU charge-equality rule and the instant payments law do not reach Switzerland. Same envelope, different legal weather.

Non-EEA legs also change the data you must send. Under the EU funds transfer rules, when either the sending or the receiving payment service provider sits in a SEPA country outside the European Economic Area, the payer's full address must be carried in the message. Within the EEA it is optional. So a payment to a British account is not, in fact, format-identical to a payment to a Belgian one.

And the promise that nobody may refuse your account because of the country it is in is a law precisely because it kept happening, and it still does. Payroll systems, insurers, utilities and subscription platforms that only accept a national IBAN are a live enforcement problem across the area, not a solved one.

"Instant" is a defined term with a budget and a set of trapdoors

Four seconds is what the customer sees. The scheme sees a timing budget with named deadlines, and the deadlines have consequences.

The rulebook does not say "fast". It says that the confirmation must have reached the originating institution by a specific second, and that if it has not, the payer's institution must put the money back. Money that "arrived instantly" may in fact have been reserved rather than debited, and may be unreserved a few seconds later.

There are also caps that the plain version never mentions. At launch in November 2017 the scheme had a hard ceiling of 15,000 euros per transaction. That rose to 100,000 euros in 2020, and the scheme-level ceiling has since been removed, but individual institutions still set their own value limits according to their own risk appetite, and European law explicitly lets each customer set a personal per-transaction or per-day limit. "No limit" at scheme level does not mean no limit in your app.

Finally, the reason instant payments could be made compulsory at all is a legal manoeuvre most people never notice. Sanctions screening was taken out of the transaction path. Institutions now screen their own customer base against targeted financial restrictive measures at least once every calendar day, and are forbidden from screening the payer and payee again during the execution of an instant transfer. Without that change, ten-second settlement across thirty countries would not have been achievable.

Direct debit is the least unified of the unified things

The plain version implies that a European direct debit is one thing. It is closer to two things wearing a shared uniform, sitting on top of thirty national habits.

There are two schemes, not one, and they behave very differently. The consumer scheme gives an eight-week no-questions-asked refund. The business-to-business scheme abolishes the refund entirely and instead requires the payer's bank to check the mandate before it debits, which is a completely different risk model wearing a similar-looking message.

Adoption is wildly uneven. In the first half of 2025, direct debits were around 14 per cent of non-cash payments across the euro area, but in Germany they were around 31 per cent of all non-cash payments. A German product manager and an Italian one are describing different worlds when they say "our customers pay by direct debit".

And the electronic mandate, the thing that ought to make all this frictionless, is still a minority. Of euro area direct debits in the first half of 2025, those with an electronic mandate accounted for 12 per cent by number. Eighty-eight per cent still rest on consent given in some other form, which in practice means paper, or a national scheme's own e-mandate service, or a recorded telephone call.

The technical version

What SEPA is, institutionally

SEPA is a European Union policy objective, delivered through two instruments that are frequently confused.

The first is scheme rules. These are written and maintained by the European Payments Council, an international non-profit association under Belgian law which is explicitly not part of the EU institutional framework. The EPC publishes rulebooks, each institution adheres to them individually and appears on a published register of scheme participants, and adherence is a contractual matter between the participant and the EPC.

The second is law. Regulation (EU) No 260/2012, universally called the SEPA Regulation, establishes technical and business requirements for credit transfers and direct debits in euro. It was amended in 2024 by Regulation (EU) 2024/886, the Instant Payments Regulation, which inserted new Articles 5a to 5d dealing with instant credit transfers, charges, verification of payee, and sanctions screening.

The geographical scope of the EPC schemes is set out in EPC409-09, which reached version 8.0 on 24 December 2025 and lists forty-one countries and territories. These comprise the twenty-seven EU Member States, three EEA countries (Iceland, Liechtenstein and Norway), eleven non-EEA countries (Albania, Andorra, Moldova, Monaco, Montenegro, North Macedonia, San Marino, Serbia, Switzerland, the United Kingdom and Vatican City State) and four non-EEA territories (Guernsey, Jersey, the Isle of Man and Saint-Pierre-et-Miquelon). All figures and dates in this chapter are accurate at time of writing.

The critical structural point is that scheme scope and legal scope are not the same set. Scheme rules bind a Serbian or British participant. EU regulations do not.

SEPA Credit Transfer

The SEPA Credit Transfer scheme, launched in January 2008, is the ordinary push payment. The governing document at time of writing is the 2025 SCT rulebook version 1.1 (EPC125-05), effective 5 October 2025.

Its core parameters:

Parameter	SEPA Credit Transfer
Currency	Euro at all process stages, including exception handling
Account identifier	IBAN, mandatory
Maximum execution time	One banking business day following the point in time of receipt of the instruction
Scheme-level amount ceiling	None; settlement and value limits may exist between participants
Remittance information	Up to 140 characters, structured or unstructured
Reject	Same day, at the latest the next banking business day
Return	Within three banking business days after settlement date
Recall	Ten business days for duplicate or technical error; thirteen months for fraud
Recall response	Within fifteen banking business days of receipt of the recall

Only one recall is permitted per transaction. The recall regime is worth dwelling on, because it is the structural answer to a question every treasury team asks: what happens when we send the wrong payment. The answer in SCT is that you may ask, the beneficiary institution must answer within a defined window, and it may say no. There is no unilateral clawback.

Volumes are large. The EPC puts SEPA credit transfer traffic at more than twenty-nine billion transactions a year across the area. The ECB's own euro area figures for the first half of 2025 record 16.8 billion credit transfers worth 107.3 trillion euros, which was 22 per cent of non-cash payments by number and 92 per cent by value.

SEPA Instant Credit Transfer

SCT Inst went live in November 2017, described by the EPC at the time as a world first for a region of this size. The governing document at time of writing is the 2025 SCT Inst rulebook version 1.1 (EPC004-16), published 5 October 2025 and stated to remain in force until 21 November 2027 at 03:30 CET.

The scheme is euro-only, operates twenty-four hours a day on every calendar day of the year subject to short and foreseeable planned maintenance, and has no scheme-level maximum amount. Value limits are applied by the originator's institution according to its own risk appetite and risk management controls.

The timing model is the part practitioners get wrong most often. It is not one number. The 2025 rulebook replaced the original 10 / 20 / 25 second structure with a 5 / 7 / 9 second structure, measured from the time stamp applied by the originator's institution at the time of receipt. Within seven seconds of that time stamp, the clearing mechanism serving the beneficiary's institution must have received a positive or negative confirmation message: this is the Timeout Deadline. That confirmation must then reach the originator's institution within a further two seconds, that is to say by the ninth second at the latest.

The tenth second belongs to the law rather than the scheme. Under Article 5a of Regulation (EU) No 260/2012 as inserted by the Instant Payments Regulation, the payee's institution must make the

amount available on the payee's account and confirm completion within ten seconds of the time of receipt of the payment order by the payer's institution; and where the payer's institution has received no such confirmation within ten seconds, it must immediately restore the payer's account to the state it would have been in had the transaction not taken place. The 2025 rulebook also moved the time stamp to millisecond precision.

Two further legal points matter. Time of receipt for an instant credit transfer is the moment the order is received by the payer's institution, regardless of hour or calendar day, notwithstanding the ordinary cut-off provisions of PSD2. And credit value date on the payee's account must be the same date as the date of crediting.

The Instant Payments Regulation timetable

The regulation converts SCT Inst from a commercial option into an obligation, on a staggered timetable.

Obligation	Euro area PSPs	Non-euro area PSPs
Receive instant credit transfers	9 January 2025	9 January 2027
Send instant credit transfers	9 October 2025	9 July 2027
Charge parity with non-instant transfers (Article 5b)	9 January 2025	9 January 2027
Verification of payee (Article 5c)	9 October 2025	9 July 2027
Daily sanctions screening (Article 5d)	9 January 2025	9 January 2025

Payment and electronic money institutions have their own dates, 9 April 2027 in the euro area and 9 July 2027 outside it, reflecting their later admission to settlement systems. Non-euro area institutions have a further deadline of 9 June 2028 relating to sending outside business hours in national currency.

The substantive obligations are worth stating precisely, because they are what a compliance officer will be measured against.

Article 5a(1) requires that any institution offering its customers a service of sending and receiving credit transfers must offer sending and receiving of instant credit transfers to all of its customers, and must ensure that all payment accounts reachable for credit transfers are also reachable for instant credit transfers twenty-four hours a day on any calendar day.

Article 5a(6) requires institutions, on request, to let a customer set a maximum amount per day or per transaction for instant credit transfers, modifiable at any time before an order is placed.

Article 5b(1) provides that charges for sending and receiving instant credit transfers shall not be higher than charges for other credit transfers of corresponding type, and Article 5b(2) requires that the verification service be provided free of charge.

Article 5d(1) requires institutions offering instant credit transfers to verify whether any of their customers are persons or entities subject to targeted financial restrictive measures, immediately after any new or amended measures enter into force and at least once every calendar day. Article 5d(2) then forbids them from screening payer and payee again during execution of an instant credit transfer.

There is one derogation of note. An institution in a Member State whose currency is not the euro may, with prior permission from its competent authority based on an assessment of its access to euro liquidity, decline to send instant euro transfers above a per-transaction limit from accounts denominated in

the national currency, during hours when it neither sends nor receives non-instant euro credit transfers. That limit is set by the competent authority and may not be lower than 25,000 euros. Permissions run for one year and may be extended.

Uptake is now material rather than marginal. In the first half of 2025, instant credit transfers accounted for 23 per cent of the number and 7 per cent of the value of credit transfer transactions processed by euro area retail payment systems.

Verification of Payee

The EPC's Verification Of Payee scheme (EPC218-23) is the scheme-level answer to Article 5c. Version 1.0 was published in October 2024; version 1.1 was issued on 16 March 2026 with an effective date of 20 September 2026.

VOP is not an ISO 20022 message flow in the way the payment schemes are. It uses application programming interface technology making use of ISO 20022 resource elements, exchanged through Routing and Verification Mechanisms of each participant's choosing, with participants registered in the EPC Directory Service.

There are four possible outcomes: match; no match; close match, in which case the name associated with the account is shown to the payer; and verification check not possible. The rulebook sets a maximum execution time of five seconds for the requesting institution to obtain the VOP response, and states a preference for one second or less.

Two things about scope catch people out. VOP applies to all credit transfers under Article 5c, not only instant ones. And it must be performed immediately after the payer provides the payee's details and before the payer is offered the possibility of authorising the transfer, regardless of the initiation channel.

SEPA Direct Debit Core

SDD Core launched in November 2009. The governing document at time of writing is the 2025 SDD Core rulebook version 1.1 (EPC016-06), effective 5 October 2025.

The defining architectural choice is the creditor-driven mandate flow. The debtor signs a mandate, on paper or electronically, in favour of the creditor. The creditor stores it. The debtor's institution does not receive the signed instrument. Each mandate is uniquely identified by the combination of the Unique Mandate Reference (attribute AT-M001) chosen by the creditor and the Creditor Identifier (attribute AT-E005) issued to the creditor under national arrangements. That pair, and not any document held at the paying institution, is what makes a collection traceable.

The operational parameters:

Parameter	SDD Core
Currency	Euro at inter-PSP level
Pre-notification to debtor	At least 14 calendar days before due date, unless another timeline is agreed between debtor and creditor
Presentation	Debtor's PSP must receive the collection at the latest one inter-PSP business day before due date (D-1)
Returns	Within five inter-PSP business days following settlement date

Parameter	SDD Core
Refusal	Debtor may instruct its PSP before settlement; handled as a rejection
Refund, authorised collection	Eight weeks from the debit date, on a no-questions-asked basis
Refund, unauthorised collection	Claim to be presented within thirteen months of the debit date
Mandate dormancy	If no collection is presented for 36 months, the creditor must cancel the mandate

A point of law that is easy to skip: offering SDD Core is not optional. Under the reachability provisions, an institution reachable for national direct debits in euro must be reachable for consumer direct debits under a Union-wide scheme, which in practice means SDD Core.

SEPA Direct Debit B2B

The business-to-business scheme launched alongside Core in November 2009 and is optional for institutions. The governing document at time of writing is the 2025 SDD B2B rulebook version 1.0 (EPC222-07), issued 28 November 2024 and effective 5 October 2025.

Three differences define it.

The debtor may not be a consumer. Participation is limited to debtors authorised under national law to opt out of the refund right.

There is no refund right for an authorised transaction. The debtor cannot ask its institution for the money back.

Because the refund right is gone, the risk control moves upstream. Before debiting the account, the debtor's institution must check that the mandate-related data received in the collection matches mandate-related data received from, or confirmed by, the debtor. This obligation is the reason B2B onboarding is slow: somebody at the paying bank has to be told about the mandate before the first collection, not after it.

Presentation is the same D-1 as Core. Returns are permitted up to three inter-PSP business days after settlement date, a shorter window than Core's five, and the mandate dormancy rule at 36 months is the same.

The mandate model compared with UK Direct Debit

For a British reader, and for anyone building a product that has to do both, this is the comparison that matters. The two systems reach similar outcomes by opposite routes.

The clearest way to see the difference is to ask a single question: after the mandate is signed, who has a copy of it?

In the United Kingdom, under AUDDIS, the signed Direct Debit Instruction is retained by the service user, but the instruction itself is lodged electronically with the paying institution through Bacs. Bacs guidance is explicit that the one major change AUDDIS made was that the signed DDI is retained by the service user and not by the paying institution, while the details still travel to the paying institution. The paying bank therefore holds a record of the instruction, can validate collections against it, and can bounce a collection for which no instruction exists.

In SEPA Core, no equivalent lodgement takes place. The mandate data reaches the debtor's institution with the first collection, and the debtor's institution has nothing to check it against unless it has built its own mandate store. The B2B scheme exists precisely to plug that gap for corporate payers.

	SEPA Direct Debit Core	UK Bacs Direct Debit
Who holds the signed mandate	Creditor	Service user
Does the payer's bank hold the instruction	No, receives mandate data with the first collection	Yes, lodged via AUDDIS
Mandate identifiers	Unique Mandate Reference plus Creditor Identifier	Service User Number plus payer reference
Advance notice, default	14 calendar days before due date, unless otherwise agreed	Minimum 10 working days plus postal time, in the absence of any other agreement
Collection cycle	Collection received by debtor's PSP at latest D-1	Three working day cycle: submission, processing, entry
Settlement	Deferred net at the clearing mechanism	Deferred net through Bacs
Refund, authorised	8 weeks, no questions asked	Direct Debit Guarantee: immediate refund by the paying institution, then indemnity claim against the service user
Refund, unauthorised	Claim within 13 months	Direct Debit Guarantee, with the service user liable on indemnity
Producing the mandate on request	Creditor must be able to produce it	Copy required within seven working days of request
Dormancy	Creditor must cancel after 36 months without a collection	Paying institutions hold DDI details for a minimum of 24 months from receipt or last payment
Currency and geography	Euro, 41 SEPA countries and territories	Sterling, United Kingdom

Three practical consequences follow.

The British model puts the arbiter at the payer's bank, which is why the Direct Debit Guarantee can promise an immediate refund with a straight face: the paying institution refunds first and recovers from the service user afterwards through the indemnity process. The SEPA model puts the arbiter at the creditor, and compensates by giving the payer a broad, time-boxed, no-questions-asked right that the paying institution honours from the settlement chain.

The British model is therefore harder to join and easier to police. Service users are sponsored by a payment service provider, hold a Service User Number, and are vetted. Note also that from 1 July 2027 it will no longer be possible to apply for a non-AUDDIS Service User Number, which completes the migration to electronic lodgement.

And the notice periods are not equivalent even though they look similar. Ten working days plus postal time in the United Kingdom is a longer real period than fourteen calendar days in SEPA, and the UK figure is a default that can be varied only with the sponsor's agreement, while the SEPA figure can be varied by agreement between creditor and debtor directly.

Reachability obligations

Reachability is the provision that turns a scheme into an area. Without it, SEPA would be a format standard that some banks supported for some corridors.

Article 3 of Regulation (EU) No 260/2012 states the principle in two directions. A payee's payment service provider that is reachable for a national credit transfer under a payment scheme must be reachable for credit transfers initiated through a scheme at Union level. A payer's payment service provider that is reachable for a national direct debit must be reachable for direct debits initiated through a scheme at Union level.

The effect is that reachability cannot be sold as a premium corridor. If an institution accepts euro credit transfers from across the street, it must accept them from across the continent.

The Instant Payments Regulation extends this to time as well as geography. Article 5a(1) requires that all payment accounts reachable for credit transfers must also be reachable for instant credit transfers twenty-four hours a day on any calendar day.

Underneath the legal obligation sits an interoperability problem, because reachability is only real if the clearing mechanisms interconnect. This is why RT1 is designed to reach all institutions adhering to the SCT Inst scheme, whether directly, through other clearing mechanisms connected to RT1, or through Eurosystem reachability measures, and why participants can use a single interface for transactions settling in RT1, in TIPS, or across the two. Reachability in law required reachability in plumbing, and the plumbing had to be built.

Why a euro payment from Ireland to Portugal is domestic in character

Five separate things have to be true at once for the Cork to Lisbon payment to feel like a Cork to Dublin payment, and SEPA supplies all five.

The identifier is the same. IBAN is mandatory in both cases. There is no separate international addressing scheme to learn, and the SEPA Regulation removed the requirement for customers to supply a BIC.

The message is the same. The same ISO 20022 messages, in the same versions, with the same mandatory fields, carry a domestic and a cross-border euro transfer.

The rulebook is the same. Execution time, return windows, recall procedure, remittance capacity and exception handling do not change at the border.

The reachability obligation removes the question of whether the far end can be paid at all, and Article 9 of the SEPA Regulation removes the question of whether the far end will accept the account: a payer must not be required to specify the Member State in which a payee's account is located, and a payee must not be required to specify the Member State of a payer's account, provided the account is reachable. This is the provision that makes refusing a foreign IBAN unlawful.

And the price is the same. Regulation (EC) No 924/2009, as amended by Regulation (EU) 2019/518, requires that charges for a cross-border payment in euro be the same as those for a corresponding national payment of the same value in the same currency. The Instant Payments Regulation then adds the second parity rule inside the instant product: instant may not cost more than non-instant of corresponding type.

Domestic in character is therefore a precise claim about identifiers, formats, rules, access and price. It is not a claim about routing, and it does not survive being carried outside the European Union, where the pricing and instant-payment obligations no longer apply even though the scheme rules still do.

ISO 20022 as SEPA's native format

Most payment systems in the world are migrating to ISO 20022. SEPA never had to, because it started there.

The first SEPA Credit Transfer rulebook in 2008 specified ISO 20022 XML, at a point when the standard was barely four years old and almost nothing else used it. Two years before the euro area migration deadline, the SEPA Regulation made it law: the Annex to Regulation (EU) No 260/2012 requires that the message format be the ISO 20022 XML standard, that the account identifier be the IBAN, that at least 140 characters of remittance information be carried, and that processing be fully automated and electronic without manual intervention.

This has three consequences that a practitioner feels every day.

There is no legacy format to translate from. Elsewhere, ISO 20022 adoption is a truncation problem: fields that existed in a fixed-width or tag-based legacy format have to be squeezed into or expanded out of the new one, and data is lost at the seams. SEPA has no such seam. The customer-to-bank message, the interbank message and the reporting message were designed as one family.

The versions are pinned. Scheme messages are not "ISO 20022" in the abstract, they are named versions specified in the implementation guidelines. At time of writing the SEPA schemes use the 2019 message versions.

Scheme	Message	Purpose
SCT and SCT Inst	pacs.008.001.08	FI to FI customer credit transfer, the payment itself
SCT and SCT Inst	pacs.002.001.10	FI to FI payment status report, positive and negative confirmation
SCT and SCT Inst	pacs.004.001.09	Payment return, and positive response to a recall
SCT and SCT Inst	camt.056.001.08	FI to FI payment cancellation request, used for recall
SCT and SCT Inst	camt.029.001.09	Resolution of investigation, negative response to a recall
SCT Inst	pacs.028.001.03	FI to FI payment status request
SDD Core and B2B	pacs.003.001.08	FI to FI customer direct debit, the collection
SDD Core and B2B	pacs.002.001.10	FI to FI payment status report, used for rejects
SDD Core and B2B	pacs.004.001.09	Payment return, covering returns and refunds
SDD Core and B2B	pacs.007.001.09	Payment reversal, creditor-initiated

And the format still moves, which is the answer to anyone who thinks a native standard means a frozen one. The current migration is addresses. Until 5 October 2025 only unstructured and structured address formats were permitted. From 5 October 2025 to 15 November 2026 all three formats are allowed, including a hybrid combining structured elements with address lines. From 15 November 2026 only structured and hybrid addresses will be allowed. The 2025 SCT Inst rulebook version 1.1 exists in large part to move that date from 22 November 2026 to 15 November 2026.

Address rules interact with the geography in a way already flagged. Where either institution is located in a SEPA country outside the European Economic Area, the payer's full address must be provided in the message under the EU funds transfer rules. Within the EEA it is optional. Screening requirements in practice only need the structured town and country elements, which is part of the argument for the structured migration.

The infrastructure underneath

Three systems carry most of the pan-European traffic.

STEP2-T is EBA CLEARING's pan-European automated clearing house for retail payments in euro, and one of the three largest retail payment systems in the euro area. It settles by continuous gross settlement: payments settle at any time the settlement windows are open, transaction by transaction, in immediately available central bank funds held on a TARGET Technical Account, with queuing and optimisation where liquidity is short.

RT1 is EBA CLEARING's instant service, live since 2017, providing real-time gross settlement of instant payments in central bank funds and reaching all institutions adhering to the SCT Inst scheme, directly or through connected mechanisms.

TIPS is the Eurosystem's TARGET Instant Payment Settlement service. It settles in central bank money on a 24/7/365 basis, processes transactions in under 100 milliseconds, and provides final and irrevocable settlement. It is compliant with the SCT Inst scheme. It currently handles euro, Swedish krona and Danish krone. It operates on cost recovery with no entry fee and a fixed charge of 0.2 euro cent per transaction, split equally between the sending and receiving participants.

Above them sits T2, the Eurosystem's real-time gross settlement service for euro payments in central bank money, which since March 2023 has replaced the former TARGET2 within the consolidated TARGET Services platform, and alongside it EURO1, EBA CLEARING's multilateral net large-value system whose final balances settle end of day in T2. In the first half of 2025, euro area large-value systems settled 74.0 million payments worth 235.1 trillion euros.

What SEPA did not unify

The honest summary of SEPA's scope is that it unified two instruments completely, one instrument partially, and left everything else alone.

It did not unify cards. The SEPA Cards Framework did not produce a pan-European card scheme, and the largest retail payment system in the euro area by volume is Mastercard's clearing system. Card payments were 57 per cent of euro area non-cash payments by number in the first half of 2025, and none of that traffic is governed by a SEPA scheme.

It did not unify currencies. Everything in this chapter is euro-denominated at the interbank level. Non-euro EU members participate as scheme countries and their institutions can hold euro accounts, but a domestic payment in Polish złoty or Swedish krona is outside SEPA entirely.

It did not unify the front end. Request-to-pay, e-invoicing, mobile initiation, wallets and account-to-account overlays sit above the schemes, not inside them. The EPC publishes a SEPA Request-To-Pay rulebook, published in November 2020, which is a messaging layer for asking, not a payment scheme. Commercial overlay services build on SCT Inst rather than replacing it.

It did not eliminate national variation. Additional Optional Services remain, national mandate services differ, and adoption patterns diverge so sharply that the same scheme supports 31 per cent of non-cash payments in one country and a rounding error in another.

And it did not, by itself, make anything instant. That took twelve more years, a second scheme, and finally a regulation with dates in it.

Timeline

Date	Milestone
January 2008	SEPA Credit Transfer scheme launched
November 2009	SEPA Direct Debit Core and B2B schemes launched
February 2012	Regulation (EU) No 260/2012, the SEPA Regulation, adopted
1 August 2014	End of migration for euro area countries
31 October 2016	End of migration for non-euro area countries
November 2017	SCT Inst launched, initial ceiling 15,000 euros
2020	SCT Inst ceiling raised to 100,000 euros
November 2020	SEPA Request-To-Pay rulebook published
November 2018	TIPS launched by the Eurosystem
March 2023	One-Leg Out Instant Credit Transfer rulebook published; T2 replaces TARGET2
13 March 2024	Regulation (EU) 2024/886, the Instant Payments Regulation, adopted
October 2024	First Verification Of Payee scheme rulebook published
9 January 2025	Euro area PSPs must receive instant credit transfers; charge parity; daily sanctions screening
5 October 2025	2025 rulebooks enter force for SCT, SCT Inst, SDD Core and SDD B2B
9 October 2025	Euro area PSPs must send instant credit transfers and offer verification of payee
15 November 2026	Only structured and hybrid address formats permitted
9 January 2027 onwards	Staggered obligations for non-euro area PSPs and for payment and e-money institutions

What to check before you build

If you are integrating, five questions decide most of your design.

Which scheme, and therefore which timing model: D+1 for SCT, seconds for SCT Inst, D-1 presentation for both direct debit schemes.

Which clearing mechanism your institution uses, because that determines your real cut-offs, your file formats at the edges, and your liquidity behaviour, none of which the rulebook tells you.

Whether any leg touches a SEPA country outside the European Economic Area, which changes your address data obligations and removes your customer's EU pricing and instant-payment protections.

Whether your direct debit population contains consumers, which decides Core versus B2B and therefore whether you are exposed to an eight-week no-questions-asked refund or to a mandate-checking obligation at the paying institution.

And whether your address data is structured. That deadline is 15 November 2026 and it applies to every message you send.

40.98 Common wrong ideas

Wrong: SEPA is a payment system that carries the money. **Right:** SEPA is a set of rulebooks made compulsory by law, and the traffic runs over competing clearing and settlement mechanisms such as STEP2-T, RT1, TIPS and national systems like CORE.

Wrong: SEPA covers the European Union. **Right:** The area is defined by the currency rather than the Union, and the EPC schemes cover forty-one countries and territories, eleven of them outside the EEA, including the United Kingdom and Switzerland.

Wrong: A euro payment anywhere in SEPA costs what a domestic one costs. **Right:** The charge-equality rule is EU law with a geographical limit, and it does not reach Switzerland or the United Kingdom even though the scheme rules do.

Wrong: A SEPA message is format-identical wherever it is sent. **Right:** Where either institution sits in a SEPA country outside the European Economic Area, the EU funds transfer rules require the payer's full address, which is optional within the EEA.

Wrong: SCT Inst has no value limit now that the scheme ceiling has gone. **Right:** Institutions still apply their own limits according to their risk appetite, and Article 5a(6) requires them to let a customer set a personal per-transaction or per-day maximum.

Wrong: Money that arrived instantly has definitely arrived. **Right:** It may have been reserved rather than debited, and where no confirmation reaches the payer's institution within ten seconds it must restore the payer's account to the state it would have been in.

Wrong: A European direct debit is one thing. **Right:** There are two schemes with opposite risk models: Core with an eight-week no-questions-asked refund, and B2B with no refund right for an authorised collection and a mandate check at the paying institution before it debits.

Wrong: The debtor's bank holds the mandate and can validate collections against it, as in the United Kingdom. **Right:** In SEPA the creditor holds it, the mandate data reaches the debtor's institution with the first collection, and that institution has nothing to check against unless it has built its own mandate store.

Wrong: European direct debits now run on electronic mandates. **Right:** In the first half of 2025 only 12 per cent of euro area direct debits by number carried an electronic mandate, with the rest resting on paper, national e-mandate services or recorded calls.

Wrong: Verification of Payee applies to instant credit transfers. **Right:** Article 5c applies it to all credit transfers, and it must be performed immediately after the payee's details are entered and before the payer is offered the possibility of authorising.

40.99 Chapter summary in 20 lines

1. SEPA made a euro payment between any two of thirty-odd countries behave, from the customer's side of the counter, like a payment inside one country.
2. It did this without building a machine, because what Europe built was a set of rulebooks, a legal obligation to follow them and a shared message format.
3. The rules are written by the European Payments Council, a Belgian non-profit explicitly outside the EU institutional framework, and adherence is contractual.
4. The law is Regulation (EU) No 260/2012, amended in 2024 by the Instant Payments Regulation, which inserted Articles 5a to 5d.

5. Scheme scope and legal scope are different sets, since the schemes cover forty-one countries and territories, eleven outside the EEA, where EU regulations do not bind.
6. SEPA Credit Transfer, launched in January 2008, executes by the following banking business day, uses IBAN, carries 140 characters of remittance information, and permits a request to recall but no unilateral clawback.
7. SEPA Instant Credit Transfer went live in November 2017 and now runs a five, seven and nine second scheme budget inside a ten-second legal obligation.
8. If no confirmation reaches the payer's institution within ten seconds, that institution must immediately restore the payer's account as though the transaction had never happened.
9. The scheme-level ceiling, once 15,000 euros and then 100,000, has been removed, but institutional limits and customer-set limits remain.
10. Instant payments could only be made compulsory because sanctions screening was moved out of the transaction path into a daily check by each institution against its own customer base.
11. Verification of Payee under Article 5c applies to all credit transfers, returns four possible outcomes, and must run before the payer is offered the chance to authorise.
12. SEPA Direct Debit Core, launched in November 2009, is creditor-driven, with the creditor holding the signed mandate and identifying it by Unique Mandate Reference plus Creditor Identifier.
13. Core gives the debtor fourteen days' default pre-notification, D-1 presentation, an eight-week no-questions-asked refund, and thirteen months for an unauthorised collection.
14. The B2B scheme abolishes the refund right for authorised collections and moves the control upstream, requiring the debtor's institution to check mandate data before it debits.
15. The British scheme reaches similar outcomes by the opposite route, because AUDDIS lodges the instruction with the paying bank, which is why the Direct Debit Guarantee can refund immediately and recover afterwards by indemnity.
16. Reachability under Article 3 is what turns a scheme into an area, since an institution reachable for a national euro credit transfer or direct debit must be reachable for the Union-wide scheme, now twenty-four hours a day.
17. Five things make Cork to Lisbon domestic in character: the same identifier, the same message, the same rulebook, the reachability and non-discrimination obligations, and the same price.
18. SEPA was born speaking ISO 20022, so there is no legacy seam to translate across, but the versions are pinned and the format still moves, with only structured and hybrid addresses permitted from 15 November 2026.
19. SEPA did not unify cards, currencies, the front end or national habit, and card payments were 57 per cent of euro area non-cash payments by number in the first half of 2025 with none of it governed by a SEPA scheme.
20. Before you build, decide which scheme and timing model, which clearing mechanism, whether any leg leaves the European Economic Area, whether your debtors are consumers, and whether your address data is structured.

Sources: European Payments Council rulebooks and guidance, namely EPC125-05 2025 SCT rulebook v1.1, EPC004-16 2025 SCT Inst rulebook v1.1, EPC016-06 2025 SDD Core rulebook v1.1, EPC222-07 2025 SDD B2B rulebook v1.0, EPC218-23 Verification Of Payee rulebook v1.1, EPC114-06 and EPC122-16 implementation guidelines, EPC409-09 List of SEPA Scheme Countries v8.0, EPC153-22 v2.1 guidance on the provision of addresses, and the EPC SEPA timeline; Regulation (EU) No 260/2012 and Regulation (EU) 2024/886 as published in the Official Journal; Regulation (EC) No 924/2009 as amended by Regulation (EU) 2019/518; European Central Bank, TARGET Instant Payment Settlement service pages and "Payments statistics: first half of 2025" (29 January 2026); EBA CLEARING STEP2-T settlement and RT1 SCT Inst service pages; Bacs, AUDDIS and Direct Debit scheme FAQs and the Service User's Guide and Rules to the Direct Debit Scheme.

41

CHAPTER

SWIFT and Correspondent Banking

41.0 What this chapter gives you

1. You will be able to explain why SWIFT has never moved a penny, and say precisely what it does move instead.
2. You will be able to draw the correspondent chain for a dollar payment from Sheffield to Izmir and identify the point at which a dollar left the United States, which is never.
3. You will be able to account for eighty-five missing dollars on a USD 214,000 payment by naming the charge basis that entitled each bank to take its cut.
4. You will be able to separate the four causes of a three-day international payment, cut-off times, business calendars, value dating and compliance screening, and say which of them a faster network would fix.
5. You will be able to say which side of which balance sheet a nostro sits on and which side a vostro sits on, and why one account carries two names.
6. You will be able to map an MT 103 onto a pacs.008 and explain why translating back the other way loses data that cannot be recovered.
7. You will be able to say what changed on 22 November 2025, and what it now costs an institution that has not migrated.
8. You will be able to explain why SWIFT's "nearly 60 per cent within 30 minutes" and the Financial Stability Board's "50.6 per cent within one hour" are both honest figures.
9. You will be able to tell a customer what a UETR is, where it lives in the message, and what the gpi tracker will and will not tell them.
10. You will be able to argue that a correctly structured beneficiary address is a speed feature rather than a compliance chore, and show the mechanism by which it is one.

On a Tuesday morning a finance manager at an engineering firm in Sheffield instructs a payment of USD 214,000 to a steel mill outside Izmir. She has the mill's account number, the mill's bank, and a BIC. She types it into her bank's business portal at 09:40, approves it on a second device, and gets a reference number. The screen says the payment has been sent.

On Friday afternoon the mill emails to confirm receipt. USD 213,915 has arrived. Eighty-five dollars are missing, nobody in Sheffield authorised anyone to take them, and three working days have gone by. Her bank's helpline tells her the payment "left us on Tuesday", which is true, and that "it's with the correspondent", which is also true, and neither statement helps.

Nothing went wrong. That is the disorienting part. Every institution in the chain did precisely what it was designed to do, the money arrived, and the arithmetic is defensible line by line. An international payment is not a thing that travels. It is a sequence of separate domestic bookkeeping entries in differ-

ent countries, stitched together by messages, and this chapter is about the messages, the bookkeeping, and the gap between them.

All figures, dates and message versions here are accurate at the time of writing, August 2026. The standards in particular have just been through the largest change in their history.

The plain version

Start with the single most common misunderstanding in all of payments. Ask most people what SWIFT is and they will say it is the system that sends money abroad. It is not. SWIFT sends no money at all. It has never sent a penny, a cent or a lira, and it never will.

SWIFT sends messages.

Two shops and two tins

Imagine a corner shop in Sheffield and a corner shop in Istanbul. The two owners have known each other for years. Because their customers keep asking to send money to relatives in the other city, they set up an arrangement.

The Sheffield owner keeps a tin of Turkish lira at the Istanbul shop. The Istanbul owner keeps a tin of pounds at the Sheffield shop. Both tins are labelled, both are counted, and both live permanently in the shop where they sit. They do not travel.

Now a customer walks into the Sheffield shop and wants to send money to her brother in Istanbul. She hands over cash. The Sheffield owner picks up the phone and says: pay my customer's brother out of my tin. The Istanbul owner opens the Sheffield tin, counts out the lira, hands them to the brother, and writes down that the tin is now that much lighter.

Money has moved from one person to another across two thousand miles. Nothing physical crossed a border. The cash the customer handed over is still in Sheffield. The lira the brother received were already in Istanbul. All that crossed the border was a phone call.

That is correspondent banking. The tins are called **accounts**, the shops are called **banks**, and the phone call is called **SWIFT**.

Why SWIFT had to be invented

The phone call is the weak point. It can be misheard, it can be faked by someone who sounds convincing, and it cannot be proved afterwards. When there are not two shops but eleven thousand, in two hundred and twenty countries, all using different words for the same thing, "phone the other bank" collapses.

So in 1973 a group of banks built a shared, secure, standardised messaging network. They called it the Society for Worldwide Interbank Financial Telecommunication, which nobody says out loud, and it went live carrying real messages in 1977. It is a cooperative, owned by the banks that use it, based in Belgium.

It gives its members three things. A private network, so messages do not travel across the open internet. Cryptographic proof of who sent each message, so a fake instruction is very hard to inject. And, least glamorous but most important, a rigid vocabulary: a payment instruction has numbered fields, field 32A is always the value date, currency and amount, field 59 is always the beneficiary, and a bank in Japan and a bank in Chile fill in the same numbered boxes.

What SWIFT does not give its members is money. It has no view of anyone's balance, it holds no accounts, and it does not know whether the payment it carried actually happened. It delivers the instruction. The banks do the rest.

Why there is usually a chain

The two-shop story works because those two shopkeepers know each other and keep tins with each other. Real banks cannot do that with every other bank on earth. There are tens of thousands of banks. Maintaining an account with each one would mean tens of thousands of pots of idle money, tens of thousands of reconciliations, and tens of thousands of relationships to check for fraud and criminality.

So banks keep accounts with a small number of large banks, and those large banks keep accounts with each other. If you want to send dollars, the chain has to pass through a bank that has actual dollars in America, because dollars only really exist in America. If you want to send pounds, the chain has to pass through a bank that has actual pounds in Britain.

Here is the Sheffield firm's payment, drawn as a chain.

Step	Who	What they actually do
1	The firm's UK bank	Takes USD 214,000 of value from the firm. It has no dollars in America itself, but it keeps a dollar tin at a big bank in New York.
2	New York bank A	Opens the UK bank's dollar tin, takes out USD 214,000, and passes it to New York bank B. Both tins are in New York. Nothing crosses an ocean.
3	New York bank B	Puts the dollars into the Turkish bank's dollar tin, which it holds. Still in New York.
4	The Turkish bank	Sees that its New York tin is heavier, and credits the steel mill's account in Izmir with the matching amount.

Four institutions, three messages, and at no point did a dollar leave the United States. What travelled was a claim: at the end, the Turkish bank has more dollars in New York than it had before, and the UK bank has fewer.

Where the eighty-five dollars went, and the three days

Each of those middle banks did work. Each one has a price list. Because the payment was sent on shared-charge terms, which is the default for most business payments, the sending firm paid its own bank's fee up front and every bank downstream was entitled to take its cut out of the money itself.

Who	Deducted	Running total arriving
Sent		214,000
New York bank A	30	213,970
New York bank B	25	213,945
Turkish bank, receiving fee	30	213,915

Eighty-five dollars, all of it disclosed in somebody's tariff, none of it visible on the Sheffield screen at the moment of sending.

The days are a different story and a more interesting one. The messages themselves were delivered in seconds. What consumed Tuesday, Wednesday and Thursday was everything around the messages: a bank's daily cut-off time for dollar payments, after which your instruction is tomorrow's work; the fact that New York is five hours behind Sheffield and its dollar clearing system has its own opening hours; and, in this case, a compliance check. The steel mill's name resembled a name on a sanctions list closely enough for an automated filter to stop the payment and put it in a queue for a human being. The human being asked for the mill's full street address, the question travelled back up the chain to Sheffield, the answer travelled back down, and by the time the payment was released it had missed the next cut-off.

The parcel tracking number

For decades this was invisible. Once a payment left, the sending bank genuinely did not know where it was. It could ask, by sending another message, and wait.

Then in 2017 the banks agreed to attach a tracking number to every payment: a long, random, unique string generated when the payment is created and carried unchanged all the way down the chain. Every bank that touches the payment reports back to a central tracker: I received it at this time, I passed it on at this time, I took this much in fees, I used this exchange rate. The sending bank can now watch the payment move, the way you watch a parcel. That programme is called SWIFT gpi. It did not make payments faster by itself. It made them visible, which turns out to be most of what people wanted.

Where the plain version stops being true

The tin is not a tin

The most important simplification above is the pot of money sitting in the other shop. There is no pot. When a UK bank "holds dollars in New York", what it actually holds is a claim on the New York bank: a promise, recorded as a number on the New York bank's ledger, that the New York bank owes it that many dollars. It is a deposit, which means it is an unsecured loan to that institution. If the New York bank fails, the UK bank does not walk in and collect its dollars. It joins the queue of creditors.

This changes the shape of the whole business. It is why banks care intensely about which correspondents they use, why a correspondent relationship takes months of due diligence, and why a bank will keep balances with two or three correspondents in the same currency rather than one. It is also why the balance is not a passive store. It is a working balance, run deliberately thin, topped up during the day, drawn down by outgoing payments, and frequently supported by an intraday credit line from the correspondent. "Trapped liquidity" in nostro accounts means exactly this: money the bank cannot use for anything else, parked in other people's institutions in other people's jurisdictions, purely to be able to pay.

And because those balances sit inside another country's legal system, they are subject to that country's law. Dollar balances in New York are within reach of United States authorities, whoever owns them. That is the mechanism by which one country's sanctions policy becomes binding on banks with no other connection to it, and it is the single largest reason correspondent banking has become as cautious as it has.

The messages do not always follow the money

The chain drawn above implies one flow: the instruction walks down the chain alongside the value. That is one of the two ways it is done, and often not the one used.

In the **serial method** the payment instruction really does pass from bank to bank down the chain, each one debiting and crediting the accounts it holds and then forwarding a fresh instruction to the next.

In the **cover method** there are two separate flows. The originating bank sends the customer payment instruction directly to the beneficiary's bank, so the beneficiary's bank knows immediately that money is coming and for whom. Separately, it sends a bank-to-bank instruction into the correspondent chain to move the actual funds, the "cover". The customer message and the cover message travel different routes and can arrive at different times.

This matters more than it sounds. Under the cover method the intermediary banks moving the money historically saw only a bank-to-bank instruction with no detail of the underlying customer. That gap was exploited: institutions stripped out the names of sanctioned parties, and intermediaries screened a message that no longer contained anything to find. The standards were changed in 2009 to force the underlying customer details into cover messages, and the enforcement actions that followed are among the largest financial penalties ever imposed. The simple chain diagram hides all of this.

A day per hop is a symptom, not a mechanism

It is tempting to read the three-day journey as three banks each taking a day. Nothing in the system works like that. Delivery of the message is measured in seconds. The delay is made of four separate things, worth separating because they respond to different fixes.

The first is **cut-off times**. Every bank publishes a deadline for each currency, after which an instruction becomes next business day's work. Miss it by a minute and you have lost a day. These are bank-specific and published in each bank's own tariff and service schedules; read the ones that apply to you rather than any general claim about them.

The second is **business calendars**. A payment in dollars needs the American clearing system open, a payment in euros the European one. A chain crossing three currencies must satisfy three calendars, and every weekend and public holiday in any of them stops the clock.

The third is **value dating**, which is bookkeeping rather than movement. A payment can be credited today with a value date of tomorrow, so the recipient sees it but cannot use it.

The fourth is **compliance screening**, in practice the biggest single source of unexplained delay in the whole system. It gets its own treatment below.

None of these is fixed by making the network faster. The network was never the bottleneck.

SWIFT is not the only way, and not every hop takes a fee

Two further corrections. First, plenty of cross-border value moves without any SWIFT message. A bank moving money between its own branches in two countries does it on its own internal ledger. Card networks run their own messaging and settlement. The euro area's retail schemes run over their own clearing infrastructure. China and Russia have built their own interbank messaging arrangements. Money transfer operators run closed-loop networks where both ends are theirs and the "cross-border" element is an internal transfer plus local funding on each side. SWIFT is dominant in interbank correspondent messaging. It is not a synonym for international payment.

Second, the fee-per-hop is a consequence of a choice recorded in the message, not an automatic toll. Payments can be sent on terms where the sender bears every charge in the chain and the beneficiary

receives the full amount, on terms where charges are shared, or on terms where the beneficiary bears everything. Within the United Kingdom and the European Economic Area, the shared-charge basis is effectively mandated by payment services regulation for payments in the relevant currencies, which is why intra-European transfers rarely arrive short and transfers to Türkiye or Nigeria frequently do.

The technical version

What SWIFT actually is

The legal entity is a cooperative society under Belgian law, headquartered in La Hulpe, owned and controlled by its member institutions, governed by a board of twenty-five directors elected by shareholders. It was founded in 1973 and carried its first live traffic in 1977.

The scale, on SWIFT's own published figures for 2024: more than 11,500 connected institutions, more than 220 countries and territories, an average of more than 53 million FIN messages per day, and FIN availability of 99.99 per cent. Note carefully what that message count is and is not. FIN traffic includes securities, treasury, foreign exchange, trade finance and reporting messages as well as payments. It is not a count of payments, and each payment in a chain generates several messages.

The services are worth naming precisely, because practitioners use the names as shorthand.

Service	What it carries
SwiftNet	The underlying secure IP network and its interfaces
FIN	Store-and-forward messaging for the legacy MT message set
FINplus	Store-and-forward messaging for ISO 20022 messages, including CBPR+
InterAct	Real-time and store-and-forward exchange of structured XML messages
FileAct	Bulk file transfer, used for large payment files and reporting

Security of the endpoints, as opposed to the network, has been SWIFT's largest operational preoccupation since 2016, when attackers who had compromised a member's own systems used entirely valid credentials to issue fraudulent instructions from the Bangladesh Bank. The network worked perfectly; the customer's premises did not. The response was the Customer Security Programme and its Customer Security Controls Framework, against which members must attest annually. This is the correct frame for SWIFT security generally: the network is extremely hard to break, and every significant incident has been an endpoint compromise.

Governance and oversight

SWIFT is systemically important without being a financial institution, which makes it awkward to regulate. The arrangement the central banks settled on is cooperative oversight. The National Bank of Belgium is lead overseer, because SWIFT is incorporated in Belgium, supported by the G-10 central banks: Banca d'Italia, Bank of Canada, Bank of England, Bank of Japan, Banque de France, De Nederlandsche Bank, Deutsche Bundesbank, the European Central Bank, Sveriges Riksbank, the Swiss National Bank, and the Federal Reserve System, represented by the Federal Reserve Bank of New York and the Board of Governors. A wider SWIFT Oversight Forum brings in central banks from other major economies.

The oversight is conducted against a set of High Level Expectations covering risk identification and management, information security, reliability and resilience, technology planning, and communication with users. What it is not is prudential supervision, and the reason is stated plainly by SWIFT itself: it is "neither a payment nor a settlement system", and in its own words, "we don't actually move money." There is no settlement risk to supervise because there is no settlement, only operational and security risk, and that is exactly what the oversight covers.

Nostro and vostro

The vocabulary is Italian and, once you see the trick, trivial. The same account has two names depending on whose books you are reading it in.

Nostro means "ours". A nostro account is our account, held with you, denominated in your currency. On our balance sheet it is an asset: money owed to us.

Vostro means "yours". A vostro account is your account, held with us, denominated in our currency. On our balance sheet it is a liability: money we owe you. Older documentation sometimes uses **loro**, "theirs", for an account a bank services on behalf of a third party.

A UK bank's dollar account at a New York bank is that UK bank's nostro and the New York bank's vostro. One account, one balance, two labels, two sides of two balance sheets. The confusion in practice comes from people using "nostro" as a generic term for correspondent accounts regardless of viewpoint, which is harmless until it appears in a reconciliation specification.

Reconciling nostro accounts is a substantial operational function in its own right. The account servicer reports movements and balances, and the account holder matches them against its own expected entries. In ISO 20022 terms that is a camt.053 end-of-day statement, a camt.052 intraday report, and camt.054 debit and credit notifications, which replaced the MT 940 and MT 950 statements, the MT 942 interim report, and the MT 900 and MT 910 confirmations respectively. Aged unmatched items are a regulatory and audit concern, because an unexplained entry in a correspondent account is exactly what a fraud or a misdirected payment looks like.

The shrinking network

The correspondent network has been contracting for well over a decade, and the data is unusually good because it is derived from the messages themselves. The Committee on Payments and Market Infrastructures, working with SWIFT, published an annual quantitative review built on MT 103 and MT 202 traffic, excluding MT 202 COV, covering more than two hundred jurisdictions.

The headline is stark. The number of active correspondents worldwide fell by about 22 per cent between 2011 and 2019, with a further roughly 3 per cent fall in 2019 alone, while the number of active corridors fell by about 12 per cent over the same period. The decline is universal but wildly uneven: over 2011 to 2022 the fall was about 19.4 per cent in Northern America and about 47.1 per cent in the Americas excluding Northern America, with Oceania down about 37.8 per cent. Small island developing states were hit hardest of all. Meanwhile the volume and value of payments flowing through the shrinking network kept rising. Fewer banks, carrying more. Concentration in this business means a single institution's decision to exit a country can leave that country materially harder to pay.

The currency picture from the same data, as at December 2022, explains why the chain so often runs through New York. Of the value transferred, the US dollar accounted for 50.9 per cent, the euro for 27.3 per cent and sterling for 4.3 per cent.

The drivers of the decline are mostly regulatory in origin. A correspondent must satisfy itself about its respondent's anti-money-laundering controls, and to some degree about the respondent's own cus-

tomers. That due diligence is expensive, the penalties for getting it wrong are enormous, and the revenue from a small respondent in a high-risk jurisdiction is not. The rational response for a large bank is to exit, and it is called de-risking. Attempts to reduce the cost side of that calculation include the Wolfsberg Group's Correspondent Banking Due Diligence Questionnaire, which standardises what is asked, and SWIFT's KYC Registry, which standardises where the answers are kept.

The last mile is always domestic

An international payment always ends as a domestic settlement in the currency's home system, and this is the single most useful thing to hold in your head when diagnosing one. Dollars settle across the Federal Reserve's Fedwire Funds Service or across CHIPS. Euros settle across the Eurosystem's T2 or across EURO1. Sterling settles across CHAPS, inside the Bank of England's RTGS system, and for lower-value onward legs across Faster Payments. When a payment "arrives in the UK", what has happened is that a UK institution's sterling position at the Bank of England has increased, or a UK bank has credited a foreign bank's sterling vostro account.

For a UK practitioner the consequence is that inbound international payments land on domestic rails and behave like domestic payments. An inbound sterling payment reaching a UK clearing bank for onward delivery will go out over CHAPS if it is large or time-critical, and CHAPS is same-day, irrevocable, real-time gross settlement with an early-afternoon cut-off for the same-day guarantee. Miss that cut-off and it is tomorrow's payment. Alternatively the onward leg may go over Faster Payments, which settles in seconds and runs 24/7 with a maximum of GBP 1 million per transaction, subject to lower limits set by individual institutions. Neither rail knows or cares that the money came from abroad. They see a domestic instruction.

The domestic rails have themselves been converging on the same standard. The Fedwire Funds Service completed its migration to ISO 20022 on 14 July 2025. The Bank of England moved CHAPS to ISO 20022 and has been ratcheting up the data requirements, mandating Legal Entity Identifiers and purpose codes on certain CHAPS payments from 1 May 2025, with further enhanced data requirements consulted on for 2027. The point is that the same structured data now survives the whole journey instead of being flattened at each border.

MT and MX

This is where the vocabulary trips people, so take it slowly.

MT stands for Message Type. It is SWIFT's proprietary format, in service since 1977, carried over FIN. An MT message is a flat text structure of five blocks: block 1 the basic header, block 2 the application header, block 3 the user header, block 4 the text with the actual content, and block 5 the trailer. Inside block 4 the content sits in tagged fields, so field 20 is the sender's reference, field 32A is the value date, currency and amount, field 50 identifies the ordering customer, field 57 the account-with institution, field 59 the beneficiary, field 70 the remittance information, and field 71A the details of charges.

Message types are numbered by category, and the first digit tells you the business.

Category	Business
0	System messages
1	Customer payments and cheques
2	Financial institution transfers
3	Treasury markets: foreign exchange, money markets, derivatives
4	Collections and cash letters
5	Securities markets

Category	Business
6	Commodities and syndications
7	Documentary credits and guarantees
8	Travellers cheques
9	Cash management and customer status

So an MT 103 is a single customer credit transfer, an MT 202 is a general financial institution transfer, and an MT 940 is a customer statement. The messages ending in 92, 95, 96 and 99 in each category are the common group messages: cancellation requests, queries, answers and free-format.

The format's constraints were reasonable in 1977 and became crippling. Names and addresses are unstructured free text in lines of thirty-five characters, typically four of them. Remittance information in field 70 is four lines of thirty-five characters: a hundred and forty characters to describe why the payment is being made. The permitted character set is restricted, which mangles accented names. And anything too long is truncated silently. A beneficiary address running to five lines loses its fifth line, and the sanctions filter downstream screens what remains.

MX is the informal name for the ISO 20022 XML messages that replace MT. ISO 20022 is not a message format so much as a modelling methodology with a shared dictionary of business concepts, from which messages are generated. Message identifiers follow a fixed pattern: a four-letter business area, a three-digit message number, a three-digit variant and a two-digit version. So pacs.008.001.08 is business area pacs, message 008, variant 001, version 08.

The business areas you will meet in payments are pacs for payments clearing and settlement, pain for payment initiation, camt for cash management, acmt for account management, and head for the Business Application Header that wraps the message.

The broad correspondence is as follows. Treat this as a map of intent, not a field-level mapping, because the two formats do not correspond one to one and that is the entire point of the change.

MT	ISO 20022	Purpose
MT 103	pacs.008	Customer credit transfer
MT 202	pacs.009 core	Financial institution transfer
MT 202 COV	pacs.009 COV	Cover payment carrying underlying customer detail
MT 205	pacs.009	Domestic financial institution transfer
MT 103 return	pacs.004	Payment return
MT 210	camt.057	Notification to receive
MT 900 / MT 910	camt.054	Debit and credit confirmation
MT 940 / MT 950	camt.053	End-of-day statement
MT 942	camt.052	Intraday report
MT n92	camt.056	Cancellation request
MT n96	camt.029	Resolution of investigation
MT 101	pain.001	Request for transfer / credit transfer initiation

What ISO 20022 buys is data. Party names and addresses in discrete elements rather than free text, so a town is tagged as a town and a country as a country. Remittance information both structured

and unstructured, and far longer. Purpose codes drawn from a controlled list. Room for Legal Entity Identifiers alongside BICs. An extended character set. And end-to-end identifiers that survive the journey.

The migration, and the date that mattered

The cross-border implementation of ISO 20022 is governed by **CBPR+**, Cross-Border Payments and Reporting Plus: a set of usage guidelines defining exactly how the ISO 20022 messages are to be populated for cross-border traffic, developed with the Payments Market Practice Group and published through SWIFT's MyStandards. CBPR+ messages travel over FINplus, not FIN. The guidelines were released in tranches, with the initial set covering the core payment and reporting messages, later additions covering cheques, direct debits, cancellations, and charges notifications and requests.

The coexistence period, during which institutions could send either MT or ISO 20022 for cross-border payments and reporting, ended on **22 November 2025**. That is the date the industry had been working towards for years, and it is the reason a great deal of bank technology spending happened in 2024 and 2025.

What happens now, for an institution that has not migrated, is worth stating precisely because it is widely misreported. MT categories 1, 2 and 9 are no longer supported natively for cross-border payments and reporting. Institutions still sending MT are put through automatic conversion, with no opt-in, and institutions still needing to receive MT can use an in-flow translation service. From 1 January 2026, both the contingency processing for MT senders and the in-flow translation service are chargeable. So the MT world did not vanish on 22 November 2025; it became a paid fallback that degrades the data. Translation from ISO 20022 to MT is lossy by definition, because the structured fields have nowhere to go, and a bank running on translated MT is receiving less than it is being sent.

There is a second-order effect that shows up in the performance statistics. The Financial Stability Board's key performance indicators recorded a fall in the share of wholesale cross-border payments credited within one hour between 2023 and 2024, and attributed part of it to the migration itself: message translation takes additional processing time. Standardisation is worth it, and it is not free while it is happening.

The UETR and SWIFT gpi

The **UETR**, the Unique End-to-end Transaction Reference, is the identifier that makes tracking possible. It is a string of 36 characters, generated by the originator, and it must not be altered by anyone in the chain. In MT messages it lives in field 121 in block 3, the user header. In ISO 20022 it is carried in the Business Application Header and in the payment identification. It is required in the relevant payment instruction messages, including MT 103 and its variants and MT 202, MT 205 and their cover variants, for all SWIFT users originating payments, whether or not they are gpi members.

SWIFT gpi, the global payments innovation initiative, launched in 2017, is not a rail and not a message format. It is two things: a rulebook that participating banks sign up to, and a central tracker database. The rulebook commits members to same-day use of funds within the receiving bank's business day, transparency of the fees deducted, unaltered transmission of remittance information, and end-to-end tracking. The tracker is a database, hosted by SWIFT, that every bank in the chain updates as the payment passes through, recording status, timestamps, charges taken and exchange rate applied. Because the UETR is unaltered and unique, all those updates from different banks in different countries attach to the same payment.

SWIFT's own published performance figures are that nearly 60 per cent of gpi payments are credited to end beneficiaries within 30 minutes and almost 100 per cent within 24 hours, and that hundreds of the largest cash management banks send over USD 300 billion via gpi every day.

Around the tracker sits a set of services that turn visibility into action: case resolution, so investigations run as structured exchanges rather than free-format messages; stop and recall, so a payment can be chased down the chain if it was sent in error or under fraud; and pre-validation, so a beneficiary account can be checked before the payment is sent rather than bouncing days later.

Two sets of numbers that do not agree, and why

Here is a genuine trap. SWIFT reports that almost 60 per cent of gpi payments are credited within 30 minutes. The Financial Stability Board reports, for 2024, that 50.6 per cent of cross-border wholesale payments were credited within one hour and 92 per cent within one business day, with retail at 33.5 per cent and 69 per cent and remittances at 53 per cent and 76.6 per cent. The 2025 progress report describes only a slight improvement at global level since the indicators were first calculated in 2023, with wholesale and remittance speed improving and cost remaining stubborn.

Both sets of numbers are honest. They measure different things. The gpi figure covers gpi payments, sent by banks that joined the scheme along corridors they serve, and the clock is the bank-to-bank leg. The FSB indicators are drawn from a wider population and are closer to end-to-end. If you are asked why your bank's tracker says 30 minutes and the customer says three days, that gap is the answer, and it is mostly made of the domestic legs at each end, the cut-offs between them, and the compliance queue.

The G20 targets are the yardstick everyone is now measured against: by the end of 2027, 75 per cent of cross-border wholesale, retail and remittance payments credited or available within one hour and the remainder within one business day; a global average retail cross-border cost of no more than 1 per cent with no corridor above 3 per cent; and, restating the UN Sustainable Development Goal, a global average cost of sending a USD 200 remittance of no more than 3 per cent by 2030 with no corridor above 5 per cent. Against that last one, the World Bank's Remittance Prices Worldwide put the global average cost of sending USD 200 at 6.49 per cent in the first quarter of 2025, with digital services at 4.85 per cent and non-digital at 7.16 per cent, and the global weighted average at 4.92 per cent. The distance between 6.49 and 3 is the whole of the reform agenda.

Sanctions screening, and where the days actually go

Every institution in the chain screens every payment, and it screens the message, not the payment. This distinction is the source of most of the friction.

The lists screened against include, for a UK institution, the UK Sanctions List and the consolidated list of asset freeze targets maintained by the Office of Financial Sanctions Implementation; for anything touching dollars, the Specially Designated Nationals and Blocked Persons list and other lists maintained by the United States Office of Foreign Assets Control; the European Union's consolidated list; and the United Nations Security Council consolidated list. Institutions add their own lists of high-risk entities, jurisdictions, and typologies on top.

The screening engine reads names, addresses, countries, BICs, and free-text fields, and applies fuzzy matching, because criminals do not helpfully spell their names the way the list does. Fuzzy matching generates false positives at a rate that would astonish anyone outside the industry, and every one of them is a stop. The payment is held, an analyst reviews it, and if the message does not contain enough information to clear the alert, the bank sends a request for information back up the chain. In ISO 20022 that is a camt.026 unable to apply or a camt.028 additional payment information, answered by

camt.029; in the old world it was an MT 195 query and MT 196 answer. The round trip crosses time zones, and the answer often has to come from the originating customer, who is asleep.

Three consequences follow, and they are the real technical content of this section.

First, **data quality is a speed feature**. The single most effective way to make an international payment fast is to populate the beneficiary name and full structured address correctly. A truncated or free-text address is what turns a fifteen-second payment into a two-day investigation. This is the strongest practical argument for ISO 20022, and it is why the enhanced data mandates on CHAPS and elsewhere are framed around structured addresses, purpose codes and LEIs. Structured fields let the screening engine compare a town against a town rather than a hundred and forty characters of prose against everything.

Second, **the cover method has a history**. Plain MT 202 cover messages that omitted the underlying parties were used to defeat screening, deliberately, by stripping identifying data. MT 202 COV was introduced in the 2009 standards release specifically to force the underlying customer information into the cover message so that intermediaries screened the real parties. The enforcement that followed produced some of the largest financial penalties on record, including BNP Paribas's guilty plea and approximately USD 8.9 billion penalty in 2014 for sanctions violations. The lesson institutionalised across the industry is that an intermediary is responsible for what passes through it, and cannot rely on the sender.

Third, **the data requirements are about to get heavier**. At its June 2025 plenary the Financial Action Task Force agreed revisions to Recommendation 16, the wire transfer rule, published on 18 June 2025 and intended to take effect by the end of 2030. They require enhanced and verified originator information and enhanced beneficiary information on cross-border payments. Above a threshold of USD or EUR 1,000, the message must carry the name and account number of both originator and beneficiary, or a unique transaction reference; for individuals, the originator's address and the beneficiary's country and town; and for legal entities, a BIC, a Legal Entity Identifier, or another unique official identifier. Card transactions for the purchase of goods and services remain outside the full requirements. Below the threshold the data set is smaller and need not be verified absent suspicion.

Read that list of required fields against the ISO 20022 message structure and the design intent becomes obvious. The standards migration and the transparency rules are the same project seen from two directions.

Sanctions as an instrument, and the network as a chokepoint

Because SWIFT is a Belgian entity, it is bound by European Union law. When the EU has required it, SWIFT has disconnected named institutions from its network, most prominently a set of Russian banks from 2022. Be precise about what disconnection does. It does not freeze anyone's money; balances in correspondent accounts are frozen, if at all, by asset-freeze measures applied by the banks holding them, not by a messaging network. What disconnection removes is the standard, automated way to instruct. A disconnected bank can still be paid by any institution willing to use telex, email, a bilateral link or another network, subject to whether doing so is lawful.

That is why alternative arrangements exist and are growing: China's Cross-Border Interbank Payment System for renminbi, Russia's domestic financial messaging system, regional systems such as Buna in the Arab region, and a lengthening set of direct links between national instant payment systems. None approaches SWIFT's reach. All reduce the degree to which any single network is a chokepoint, which is precisely why some states are building them.

Where this is going

The standards are converging. ISO 20022 is now the format for cross-border messaging, for Fedwire, for CHAPS, for T2 and for a growing list of domestic high-value systems. The same structured payment can, in principle, travel from a corporate ledger through a domestic rail, across a correspondent chain and out through another domestic rail without being flattened.

The service levels are hardening. SWIFT gpi set rules for wholesale payments; SWIFT Go extended a service level to low-value payments; and on 25 September 2025 SWIFT announced new rules for retail cross-border payments on its network, developed with over thirty banks from seventeen countries, committing to upfront transparency on costs, full value delivery, end-to-end visibility, and instant settlement where domestic infrastructure and regulation allow.

And the underlying model is being questioned. The Bank for International Settlements has framed the correspondent banking problem explicitly in terms of the cost and failure rate of the current chain, and Project Agorá, its work with central banks and private banks on tokenised deposits and central bank money on a shared programmable platform, is an attempt to collapse the sequence of separate ledgers into one. Whether that works is not yet knowable. What is knowable is that the friction it targets is real, measured, and has not improved much in a decade.

The Sheffield payment, explained

The USD 214,000 never left the United States. Her bank's dollar position sat as a claim on a New York correspondent, was debited there, moved across a domestic dollar clearing system to a second New York bank, and was credited to the Turkish bank's dollar claim, also in New York. The steel mill was paid in Izmir out of the Turkish bank's own balance sheet, against the increase in what New York owed it. SWIFT carried the instruction, in seconds, and knew nothing about any of the balances involved. It could not have told her where the money was, because at every instant the money was in New York and only the entitlement to it was moving.

The eighty-five dollars were three tariffed charges taken under a shared-charge instruction, each disclosed in a document she had not read, none of them visible at the point of sending. Had the payment been sent on terms where the sender bears all charges, the mill would have received the full amount and her own bank would have billed her more. Had the corridor been intra-European, the shared basis would have been mandatory and the deductions would not have happened.

The three days were not three banks taking a day each. They were one cut-off missed by hours, one time zone, and one sanctions filter that stopped on a fuzzy name match and needed a full structured address the original message had never carried, because thirty-five characters was not enough.

Every part of that is fixable. Most of it is being fixed, and the fix is not a faster network. It is better data in the message, one identifier that survives the whole journey, and an agreement among the banks about what they owe the customer. That is what ISO 20022, the UETR and the gpi rulebook respectively are, and it is why a chapter about a messaging network turns out to be a chapter about bookkeeping, deadlines and paperwork.

41.98 Common wrong ideas

Wrong: SWIFT is the system that sends money abroad. **Right:** SWIFT sends messages, has no view of anyone's balance, holds no accounts, and does not know whether the payment it carried actually happened.

Wrong: a bank's dollars in New York sit in a pot of money that belongs to it. **Right:** they are an unsecured claim on the New York bank, recorded on that bank's ledger, and if the New York bank fails its correspondent joins the queue of creditors.

Wrong: a payment that takes three days means three banks each took a day. **Right:** message delivery is measured in seconds, and the days are made of cut-off times, business calendars, value dating and a compliance queue.

Wrong: every hop in the chain automatically takes a fee out of the money. **Right:** the charge basis is a choice recorded in the message, and within the United Kingdom and the European Economic Area the shared basis is effectively mandated for the relevant currencies.

Wrong: the payment instruction always travels alongside the funds. **Right:** under the cover method the customer message goes straight to the beneficiary's bank while a separate cover message moves the funds down the correspondent chain, and the two can arrive at different times.

Wrong: SWIFT is a synonym for international payment. **Right:** intragroup transfers, card networks, euro-area retail schemes, China's and Russia's own arrangements and money transfer operators' closed loops all move cross-border value without a SWIFT message.

Wrong: disconnecting a bank from SWIFT freezes its money. **Right:** disconnection removes the standard automated way to instruct; balances are frozen, if at all, by asset-freeze measures applied by the banks that hold them.

Wrong: MT messages ceased to exist on 22 November 2025. **Right:** the coexistence period ended, and MT became a chargeable fallback from 1 January 2026, delivered through automatic conversion and lossy in-flow translation.

Wrong: the tracker says thirty minutes, so a customer complaining of three days must be mistaken. **Right:** the gpi figure covers the bank-to-bank leg of gpi payments, while the wider indicators are closer to end to end, and the gap is the domestic legs, the cut-offs and the compliance queue.

Wrong: SWIFT's security record means the network has been broken into. **Right:** every significant incident, including the Bangladesh Bank fraud, has been an endpoint compromise using entirely valid credentials, which is why the Customer Security Programme is aimed at members' own premises.

41.99 Chapter summary in 20 lines

1. SWIFT is a Belgian messaging cooperative owned by its members, not a payment system: it has never moved money and holds no accounts.
2. An international payment is a sequence of separate domestic bookkeeping entries in different countries, stitched together by messages.
3. Correspondent banking means two banks holding accounts with each other, so that value is paid locally at each end while only the instruction crosses the border.
4. No bank can hold an account with every other bank, so payments run down a chain through a small number of large correspondents.

5. Dollars only really exist in America, so a dollar chain must pass through New York, and the dollars themselves never leave it.
6. What a bank "holds" at a correspondent is an unsecured claim on that institution, sitting inside another country's legal system and reachable by that country's authorities.
7. That is the mechanism by which one country's sanctions policy binds banks with no other connection to it, and the largest single reason correspondent banking has become so cautious.
8. The eighty-five dollars missing from the Sheffield payment were three tariffed charges, each disclosed somewhere, taken lawfully under a shared-charge instruction.
9. Within the United Kingdom and the European Economic Area the shared basis is effectively mandated for the relevant currencies, which is why intra-European transfers rarely arrive short.
10. The three days were not three banks taking a day each; the messages were delivered in seconds.
11. Delay is made of cut-off times, business calendars, value dating and compliance screening, and none of them is cured by a faster network.
12. Screening reads the message rather than the payment, so truncated or unstructured data is what turns a fifteen-second payment into a two-day investigation.
13. The cover method splits the customer message from the funds message, a gap that was exploited by stripping names until MT 202 COV forced the underlying parties back into the cover.
14. MT is the 1977 flat-text format with thirty-five-character lines and silent truncation; ISO 20022 replaces it with structured, tagged and far longer data.
15. CBPR+ defines how ISO 20022 is populated for cross-border traffic over FINplus, and the coexistence period ended on 22 November 2025, leaving MT as a chargeable, lossy fallback.
16. The UETR is a thirty-six-character identifier that nobody in the chain may alter, which is what lets updates from banks in different countries attach to one payment.
17. SWIFT gpi is a rulebook plus a central tracker, not a rail: it made payments visible rather than fast, which turned out to be most of what customers wanted.
18. SWIFT's thirty-minute figure and the Financial Stability Board's one-hour figure are both honest, because they measure different populations over different legs.
19. The correspondent network has been shrinking for over a decade through de-risking while carrying ever more traffic, which concentrates the system and can leave a whole country harder to pay.
20. Every international payment ends as a domestic settlement in the currency's home system, and the fix for its friction is better data in the message, one identifier that survives the journey, and an agreement about what the banks owe the customer.

Sources: Swift, on FIN traffic figures, organisation, governance and oversight, ISO 20022 and CBPR+ implementation, the UETR, gpi, and the retail cross-border rules announced 25 September 2025; National Bank of Belgium and Deutsche Bundesbank on the cooperative oversight of Swift; BIS Committee on Payments and Market Infrastructures correspondent banking data commentaries and chartpacks, and BIS Bulletin No 87; Financial Stability Board G20 targets and the 2024 and 2025 progress reports; World Bank Remittance Prices Worldwide, Issue 53, March 2025; Federal Reserve Financial Services on the Fedwire Funds Service ISO 20022 migration of 14 July 2025; Bank of England on ISO 20022 and CHAPS enhanced data; FATF, update to Recommendation 16, June 2025; ISO 20022 Registration Authority message catalogue.

42

CHAPTER

UPI

42.0 What this chapter gives you

1. You will be able to explain why NPCI never holds a rupee of Meera's forty and yet determines where every one of them goes.
2. You will be able to say what happens between Ravi's speaker announcing forty rupees and his bank actually being made whole, and name the risk his bank carries in the meantime.
3. You will be able to explain why UPI carries 85.5 per cent of India's digital payment volume and only 9.5 per cent of its value, and why that is the same shape as CHAPS and Faster Payments in Britain.
4. You will be able to describe single-click two-factor authentication and say why removing the OTP round trip is what made a forty-rupee payment worth making.
5. You will be able to tell a customer which twelve-digit number to write down when a payment goes wrong, and what the Reserve Bank of India's turnaround-time circular entitles them to without asking.
6. You will be able to distinguish a failed payment from a payment that succeeded to the wrong person, and give the different remedy for each.
7. You will be able to explain deemed status, and why a payment sitting pending for hours is neither a success nor a failure.
8. You will be able to read a UPI QR code as a `upi://pay?` deep link and say which parameters make it static and which make it dynamic.
9. You will be able to say what zero merchant discount rate is, what the amendment of 6 August 2026 changed, and why nobody should model UPI economics as though free acceptance were a law of nature.
10. You will be able to design a UPI integration that survives the API rate limits, the abolition of person-to-person collect, and the fact that every number in the specification moves.

There is a country where roughly two thirds of a billion payments happen every day, where the tea seller with a plastic stool and a kettle takes forty rupees by phone without paying a fee for the privilege, and where the whole thing runs on a piece of national infrastructure owned by a not-for-profit company that no single bank controls. That country is India, and the thing is the Unified Payments Interface. It is the highest-volume retail payment system on earth, and it is worth understanding properly, because almost everything written about it is either breathless or wrong.

The plain version

Imagine a very old-fashioned telephone exchange. Every house in town has a wire running to one building in the middle. Inside that building sits an operator with a board full of sockets. When you pick up your phone you do not dial a number, you say a name: "the baker on the high street, please." The operator knows which socket belongs to the baker, plugs your wire into his, and the two of you are connected. You never had to know which socket he was on. You never had to know that his wire runs down Mill Lane and under the canal. You said a name and the operator did the rest.

UPI is that operator, for money.

In India, every bank account has a long number and a branch code, exactly as a British account has a sort code and an account number. Nobody enjoys typing those. So UPI lets you register a short nickname that points at your account. The nickname looks like an email address: `meera@okhdfcbank`, `ravi@ybl`, `stalls9971@paytm`. It is called a virtual payment address. When somebody wants to pay you, they type the nickname, or they scan a square barcode that contains the nickname, and the operator in the middle looks it up and connects the two banks.

The operator is a company called NPCI, the National Payments Corporation of India. It sits between every bank in the country. It does not hold your money. It holds the address book and it passes the messages.

Now the worked example, with real numbers.

Meera stops at Ravi's tea stall on her way to work. Two glasses of chai, forty rupees, which is about thirty-five pence. Ravi has no card machine. He has a laminated square of paper taped to the counter with a black-and-white pattern printed on it. Meera opens her payments app, points the camera at the square, and the app fills in Ravi's nickname automatically. She types 40. The app shows her a name: **RAVI KUMAR**. That name did not come from the square of paper. It came back from Ravi's bank, a quarter of a second ago, as a check that she is paying who she thinks she is paying. She taps to confirm and types a four-digit PIN.

What happens next takes about a second and a half. Her app sends a message to her bank. Her bank sends it to NPCI. NPCI asks Ravi's bank to confirm the address is real. Meera's bank checks her PIN, checks she has forty rupees, and takes forty rupees off her balance. NPCI then tells Ravi's bank to put forty rupees on his balance. Ravi's bank does it and says so. A small speaker sitting next to Ravi's kettle, the kind every stallholder in India now owns, says out loud in Hindi: "Forty rupees received." Meera is already walking away.

Here is the part that surprises people. No money has actually moved between the two banks. Meera's bank is forty rupees richer. Ravi's bank is forty rupees poorer, because it credited Ravi out of its own pocket. Both banks made a note of it. Later that day, at one of ten scheduled moments, NPCI adds up every single thing that happened between every pair of banks, works out who owes whom on balance, and tells the Reserve Bank of India to shift the net amounts between the banks' accounts. Meera's forty rupees never travels on its own. It is swept up into one enormous sum with millions of other payments and settled in a lump.

Think of it as a group of friends who share a taxi every week and keep a tally on the fridge rather than passing coins about each time. At the end of the week one person hands over one note and the tally goes back to zero.

The scale of that tally is what makes UPI worth a chapter. In July 2026, the most recent full month at the time of writing, UPI processed **23,658.35 million transactions**, which is 23.66 billion, worth **INR**

29,87,880.49 crore, which is about INR 29.9 lakh crore or roughly 29.9 trillion rupees. **741 banks** were live on the system that month. Divide the volume by thirty-one days and you get about **763 million payments a day**. Divide that by 86,400 seconds and you get roughly **8,800 payments every second**, averaged flat across the whole day and night, which means the peaks are a great deal higher than that.

Divide the value by the volume and you get the number that explains everything: the average UPI payment in July 2026 was worth **INR 1,263**, about eleven pounds. This is not a system for buying flats. It is a system for buying chai, and vegetables, and a bus ticket, and topping up a phone, hundreds of millions of times an hour.

Every one of those payments gets a receipt number. It is usually twelve digits long, and it is the single most useful thing a customer can know, because when a payment goes wrong the receipt number is the only thing that lets a bank find it. More on that shortly.

Where the plain version stops being true

The operator does not move money, and "instant" describes only what the customer sees. The telephone-exchange picture makes it sound as though NPCI holds a great pot of rupees and shifts them about. It does not. NPCI is a switch: it routes messages, applies rules, keeps score, and produces settlement files. The actual money moves between banks' current accounts held at the Reserve Bank of India, and it moves in batches, not per payment. UPI is a **deferred net settlement** system dressed in real-time clothing. Since 3 November 2025, NPCI has run **ten settlement cycles a day for authorised transactions, between 09:00 and 21:00 at two-hour intervals**, plus two separate cycles a day for disputes, so that a chargeback can no longer delay a payments cycle. Between the moment Ravi's phone speaks and the moment his bank is actually made whole, his bank is carrying credit risk on Meera's bank. That risk is small per payment and enormous in aggregate, which is why NPCI caps how far into the red any member may go before it must top up, and why the number of daily cycles has crept upwards over the years. The customer experience is instant. The interbank plumbing is not, and any practitioner who confuses the two will size their liquidity wrong.

The app is not a bank, and the nickname hides less than it appears to. Meera's app may be PhonePe or Google Pay or Paytm. None of those is a bank. They are third-party application providers, and they ride on the licence of a sponsor bank called a payment service provider bank. Money never sits with the app; the app is a user interface and a fraud-scoring engine bolted onto somebody else's banking licence. That is a deliberate design choice and it is the single biggest reason UPI got good so fast, but it means that when an app has an outage, the correct question is which sponsor bank is down. As for the nickname: it hides the account number **from the payer**, which is the point, because a stolen account number plus branch code is a useful thing for a fraudster and a stolen `ravi@ybl` is not. It does not hide anything from NPCI, which resolves it, or from Ravi's bank, which owns it. And the resolution deliberately leaks something back the other way: the payee's registered name is returned to the payer before confirmation. That is an anti-fraud feature, and it is also a small, permanent, unavoidable disclosure of who owns a given address.

Volume is not value, and UPI is not where India's money is. This is the correction that most commentary gets wrong. According to the Reserve Bank of India's half-yearly Payment Systems Report covering the second half of 2025, UPI accounted for **85.5 per cent of India's digital payment transactions by volume** but only **9.5 per cent by value**. RTGS, the high-value real-time gross settlement system, accounted for **0.1 per cent of volume and 68.6 per cent of value**. NEFT took 3.6 per cent of volume and 14.9 per cent of value. The shape is identical to the United Kingdom's, where CHAPS carries a tiny fraction of the payments and the large majority of the sterling value. UPI is the everyday

rail. It is not the wholesale rail, it was never designed to be, and a person who says "India moves its money on UPI" is describing counts, not sums.

Free to the customer is not the same as free, and the law changed in August 2026. Since 1 January 2020, merchant discount rate on bank-account UPI and on RuPay debit has been set at zero by statute, through Section 10A of the Payment and Settlement Systems Act 2007 and Section 269SU of the Income-tax Act 1961, both inserted by the Finance (No. 2) Act 2019. Zero MDR is why Ravi accepts UPI: a card machine would have cost him a rental and a percentage, and a QR sticker costs him the price of the lamination. But zero MDR is a subsidised outcome, not a costless one. NPCI charges member banks a switching fee, banks carry the infrastructure, and the government has run an incentive scheme to compensate the ecosystem for the foregone revenue. On **6 August 2026 the Lok Sabha passed the Taxation and Other Laws (Amendment) Bill, 2026**, which amends Section 10A of the Payment and Settlement Systems Act to give the central government the power to permit charges on UPI and other notified electronic modes. At the time of writing no charge has been notified and small-value payments are widely expected to stay free, but the statutory bar that made zero MDR permanent has been removed. Anyone modelling UPI economics past 2026 should treat the current position as a policy choice rather than a law of nature. Note also that three flows already sit outside the zero-MDR perimeter today: RuPay credit card on UPI above INR 2,000, prepaid-instrument-on-UPI, and credit line on UPI.

The technical version

What NPCI is, and what it is not

The National Payments Corporation of India was incorporated in 2008 as a not-for-profit company, promoted by the Reserve Bank of India and the Indian Banks' Association, and it operates India's retail payment systems under authorisation from the RBI under the Payment and Settlement Systems Act, 2007. Its registered office is at 1001A, The Capital, B Wing, Bandra Kurla Complex, Mumbai 400 051, and its corporate identity number, U74990MH2008NPL189067, encodes the "not-for-profit public limited" status that shapes everything about how it behaves. It runs IMPS, NACH, RuPay, NETC FASTag, the National Financial Switch, AePS, BHIM and UPI, and it has two subsidiaries of note: NPCI International Payments Limited, which exports the stack, and NPCI Bharat BillPay.

The relevant comparison for a British reader is Pay.UK rather than Visa. NPCI is the scheme operator and the switch. It is not a bank, it is not a settlement agent, and it does not take a percentage of the transaction. It levies a switching fee on member banks, and NPCI's own circulars refer to rebate arrangements on that fee through the URCS back-office system, which is where reconciliation, adjustments and chargebacks are handled.

UPI went live as a pilot on **11 April 2016**, launched by Dr Raghuram G. Rajan, then Governor of the Reserve Bank of India, in Mumbai, with **21 banks**. As at March 2026 there were **703 banks live**, and as at July 2026, **741**.

The participant stack

A single UPI payment involves more legal entities than most people expect. The roles, in NPCI's own vocabulary, are these.

The **UPI app** is the interface the customer touches. The **payer PSP** is the payment service provider bank whose licence that app rides on; if the app is a bank's own app, the two are the same entity, and if the app is a third-party application provider such as PhonePe, Google Pay or Paytm, the PSP is a sponsor bank. Axis Bank and Yes Bank are the two largest sponsors by volume. The **remitter**

bank is the bank that actually holds the payer's account and performs the debit. The **payee PSP** is the equivalent on the receiving side, and the **beneficiary bank** holds the account that gets credited. For merchant payments there is also an **acquiring bank** and, very often, a payment aggregator sitting between the merchant and the acquirer.

The separation of the app layer from the account layer is the architectural decision that made UPI what it is. A company with no banking licence, no branch network and no balance sheet can build a payments app, compete purely on user experience, and reach every bank account in India on day one. That is why the market is so concentrated in the app layer and so fragmented in the bank layer: as at recent months PhonePe and Google Pay between them have run roughly four fifths of UPI volume, against an NPCI rule capping any single third-party application provider at 30 per cent. NPCI first announced that cap in November 2020 and has extended the compliance deadline three times; the current deadline is **31 December 2026**, accurate at the time of writing. Whether it is enforced is one of the genuinely open questions in Indian payments.

Addressing: the VPA, the UPI number and the mapper

The virtual payment address takes the form `handle@psp`. The left side is chosen by the customer or generated by the app; the right side identifies the PSP, so `@okhdfcbank`, `@oksbi`, `@ybl`, `@paytm`, `@axl`. The address is not a bank account. It is a pointer, resolved by NPCI's central mapper to a specific account at a specific bank. One customer may have several addresses pointing at the same account, or several accounts behind one address with one nominated as default.

Three other addressing forms exist. A **UPI Number** is a numeric alias, typically the customer's mobile number, mapped to an account through NPCI's mapper; NPCI's circular series on the Numeric UPI ID continues into FY 2026-27. A payer may also address a payment to a raw **account number plus IFSC**, which is how UPI absorbs the older IMPS use case. And a payment may be addressed by **QR code**, which is simply an encoded address plus optional parameters, covered below.

Before any money moves, the payer's app performs an address validation. The resolved **registered beneficiary name** is returned and displayed. This is mandatory, it is the reason UPI's mis-payment rate is as low as it is, and it is the reason a UPI address cannot be treated as private.

Authentication: device binding, the Common Library, and single-click two-factor

Indian regulation requires two-factor authentication on electronic payments. Card payments satisfy this with a card plus an OTP, which costs a round trip through a mobile network and is where a large share of card failures come from. UPI satisfies it without any round trip at all, and this is the second architectural decision that made the system work.

At registration, the app sends an outbound SMS from the handset to bind the device to the customer's mobile number as registered with the bank. That establishes the **device** as the first factor, something the customer has. The **UPI PIN**, four or six digits, is the second factor, something the customer knows. Because the device is already bound, no OTP is needed at payment time. NPCI calls the result **single-click two-factor authentication**, and it is worth pausing on how much latency and failure that removes.

The PIN is never seen by the app or by the PSP. It is captured inside NPCI's **Common Library**, a component embedded in every UPI app, which encrypts the keystrokes into a credential block under NPCI's public key. The block travels through the app, the PSP and the switch as opaque ciphertext and is decrypted and verified only at the remitter bank, inside a hardware security module. A compromised app cannot harvest PINs. A compromised PSP cannot replay them.

The PIN itself is set using a second-channel credential: historically the last six digits and expiry date of the customer's debit card, and more recently Aadhaar-based verification, which removed the requirement to hold a debit card at all and opened UPI to the very large population of Jan Dhan account holders who never had one. NPCI's circulars on user onboarding cover this route.

The two legs: debit and credit

A UPI push payment is two legs and a switch. Both legs are synchronous request-response messages over the UPI API, an XML message set carried over mutually authenticated TLS with digital signatures on the payload. The messages a practitioner meets are named in matched pairs: `ReqPay` and `RespPay` for the payment itself, `ReqValAdd` and `RespValAdd` for address validation, `ReqAuthDetails` and `RespAuthDetails` for the credential and authorisation exchange with the remitter bank, `ReqChkTxn` and `RespChkTxn` for status enquiry, `ReqBalEnq` for balance, `ReqListAccount` for account discovery, `ReqMandate` for e-mandates and `ReqComplaint` for disputes.

The sequence for Meera paying Ravi is as follows.

The app resolves `ravi@ybl` with `ReqValAdd`, which the switch routes to the payee PSP; the response carries the registered name, which the app displays. The customer enters the amount and the PIN; the Common Library produces the credential block. The app issues `ReqPay` to the payer PSP carrying a **transaction ID of up to 35 characters** generated by the initiating PSP, the amount, both addresses, a purpose code, and, for merchant payments, a merchant category code. The payer PSP signs and forwards it to the UPI switch.

The switch performs the **debit leg**: it presents the credential block and the debit instruction to the remitter bank, which validates the PIN in its HSM, applies its own velocity and value limits, checks the balance, debits the account and responds. Only on a successful debit does the switch perform the **credit leg**, presenting the credit instruction to the beneficiary bank, which credits the payee's account and responds. The switch then returns `RespPay` to both PSPs, and both customer apps show the outcome.

Three failure shapes matter, and they are the source of most customer distress.

If the debit fails, nothing has happened and the customer sees a decline. Common response codes are `00` for success, `Z9` for insufficient balance, `ZM` for an incorrect UPI PIN and `ZA` for a transaction declined by the customer; the authoritative list lives in NPCI's API and technical specification and runs to several hundred entries across issuer, acquirer and switch categories.

If the debit succeeds and the credit fails cleanly, the switch initiates a reversal and the money returns, usually within minutes.

If the debit succeeds and the credit **times out** without a definitive answer, the transaction enters an indeterminate state. NPCI's rules define **deemed approval** and **deemed debit** handling for exactly this case, with dedicated circulars on the subject in FY 2025-26 and FY 2026-27, and the position is resolved in the next reconciliation cycle through the URCS back office. This is the state in which the customer's money has genuinely left their account and has genuinely not arrived, and it is the state the next section is about.

A **collect** or pull payment inverts the first half of the flow. The payee's PSP issues `ReqPay` in collect mode; the switch delivers a pending request to the payer's PSP, which notifies the payer; the payer approves with a UPI PIN inside an expiry window set by the requester; the debit and credit legs then run exactly as above. Collect is how a merchant's website asks a customer's app for money without a redirect, and it is how UPI handles subscription first-payments and IPO applications.

Collect is also, historically, UPI's worst fraud surface, because a request for money looks superficially like an offer of money to an inexperienced user. Fraudsters exploited that relentlessly. NPCI's response was progressive restriction and then abolition: the person-to-person collect limit was reduced to INR 2,000, and **from 1 October 2025 NPCI discontinued person-to-person collect requests on UPI entirely**. Merchant collect remains available to verified merchants. If you are designing a UPI integration today, plan for a world in which pull only exists in the merchant direction.

Settlement

Settlement is deferred and net. NPCI accumulates every authorised transaction, computes each member's multilateral net position, and produces a settlement file. Members hold current accounts with the Reserve Bank of India, and the net positions are posted against those accounts. NPCI operates net debit caps and collateral requirements so that no member can accumulate an unfunded net debit position beyond a prescribed limit.

The cycle structure changed on **3 November 2025**. Before that date, authorised transactions and disputes shared the same cycles. From that date, authorised settlement runs in **ten cycles a day between 09:00 and 21:00 at two-hourly intervals**, and disputes run in **two separate cycles**, one covering midnight to 16:00 and one covering 16:00 to midnight. RTGS posting timelines were unchanged. The purpose is to stop a large chargeback batch from delaying a payments cycle, and to make each cycle's arithmetic cleaner for members' treasury teams. This is accurate at the time of writing; settlement cycle structures change, and a practitioner should read NPCI's current circular rather than a book.

For a British reader the mental map is: UPI's customer experience resembles Faster Payments, its settlement model resembles Faster Payments' deferred net settlement rather than CHAPS' real-time gross settlement, and its addressing and app layer occupy the space that Open Banking occupies in the United Kingdom. The difference is that in India those layers are one system with one rule book, whereas in the United Kingdom the rail, the addressing and the API layer are three separate things governed by three separate regimes.

	UPI (India)	Faster Payments (UK)	Bacs (UK)
Operator	NPCI	Pay.UK	Pay.UK
Customer experience	Seconds, 24/7	Seconds, 24/7	Three working day cycle
Settlement model	Deferred net	Deferred net	Deferred net
Addressing	VPA, UPI number, QR, account plus IFSC	Sort code and account number, Paym	Sort code and account number
Per-transaction ceiling	INR 1 lakh default, higher for listed categories	GBP 1 million	Set by scheme and sponsor
Recurring	UPI AutoPay e-mandate	Standing order	Direct Debit

Identifiers, and what to do when money vanishes

Every UPI transaction carries at least two identifiers, and confusing them wastes hours.

The **UPI transaction ID** is generated by the initiating PSP and may be up to **35 characters**. It is the app's handle on the transaction. It is excellent for talking to the app's support team and close to useless for talking to a bank.

The **Retrieval Reference Number**, twelve digits, is generated by the switch and carried through both legs. Consumer apps display it under labels including "UPI Ref No", "UPI transaction reference" and,

most commonly, **UTR**, for unique transaction reference. Strictly, UTR is the term the Indian banking system uses for the reference on a NEFT or RTGS credit and RRN is the term for a switched retail transaction, but in ordinary Indian usage the twelve-digit number on a UPI receipt is called the UTR by customers, banks and NPCI's own support material alike. It is the number that both the remitter bank and the beneficiary bank can search on. It is the number to write down.

Now the procedure, because this is the part a customer actually needs.

Money leaves the account and does not arrive. First, distinguish two cases, because they have completely different remedies. If the payment **failed** — the app showed an error, a pending state or a timeout — this is a system failure and the rules are on the customer's side. If the payment **succeeded but went to the wrong person** — the customer typed the wrong address, or was defrauded — this is not a system failure, there is no automatic reversal, and the remedy is a fraud report to the bank and to the National Cyber Crime Reporting Portal, plus, in practice, the beneficiary's consent.

For the failure case, the governing rule is the Reserve Bank of India's circular **RBI/2019-20/67 of 20 September 2019, "Harmonisation of Turn Around Time (TAT) and customer compensation for failed transactions using authorised Payment Systems"**. Its two UPI rows are precise. Where the account is debited but the beneficiary account is not credited on a funds transfer, the beneficiary bank must auto-reverse **latest on T+1 day**, and beyond that the customer is entitled to **INR 100 per day of delay**. Where the account is debited but transaction confirmation is not received at a merchant location, auto-reversal is **within T+5 days**, with the same INR 100 per day beyond it. The compensation is not discretionary and does not require the customer to ask.

The escalation ladder, in order, is: raise the complaint **inside the app**, because UPI carries a native dispute mechanism and apps are required to expose it; if unresolved, go to the **PSP bank**; if still unresolved after thirty days, escalate to NPCI through the UPI Help portal at `upihelp.npci.org.in`; and beyond that to the **Reserve Bank - Integrated Ombudsman Scheme** through the RBI's complaint management system. At every rung, the first thing anybody will ask for is the twelve-digit reference.

There is one more state worth knowing. A transaction may sit in **deemed** status, where the switch never received a definitive answer from one leg. To a customer it looks identical to a failure. Internally it is not a failure at all; it is an unresolved fact, and it is settled by reconciliation rather than by reversal. NPCI's response-code extensions for deemed debit on mandate execution, issued as an addendum to circular OC-128 in FY 2026-27, exist because this state kept producing wrong customer outcomes on AutoPay. If a customer's payment sits pending for hours rather than seconds, deemed status is usually why.

QR codes

A UPI QR code is not a special data format. It is a text string, encoded as a standard QR symbol, containing a **UPI deep link** of the form `upi://pay?` followed by URL query parameters. The parameters a practitioner meets are `pa` for the payee address, `pn` for the payee name, `am` for amount, `cu` for currency, always `INR`, `tr` for a transaction reference, `tn` for a note, `mc` for the merchant category code, `mode` and `sign` for signed intent. A static merchant QR omits `am`, so the customer types the amount; a dynamic QR, generated per bill by a till or a printer, includes both `am` and `tr`, which is what allows automatic reconciliation against an invoice.

The same string, invoked on a handset rather than printed, is an **intent**: tapping a `upi://pay?` link on an Android device offers the user their installed UPI apps and hands the parameters straight across. That is the mechanism behind every "Pay by UPI" button on an Indian checkout page, and it is why a UPI merchant integration requires no card fields, no 3-D Secure and no redirect.

NPCI publishes brand guidelines for merchant QR, with a circular series on mandatory BHIM UPI merchant QR branding running to an addendum in FY 2026-27, and separately supports EMVCo-compliant QR for interoperability with card schemes and for international acceptance. Circular OC No. 236 of FY 2026-27 covers enhancement of international mobile application payment acceptance through UPI QR in India, which is the mechanism that lets a visitor's foreign wallet scan an Indian merchant's sticker.

The economics of the QR are the whole story of UPI's merchant reach. A card terminal costs a merchant a device, a rental, a telephone or data line and a percentage of every sale. A UPI QR costs the price of printing. The Reserve Bank of India's Annual Report for 2024-25 recorded UPI QR codes rising 91.5 per cent to **65.8 crore**, that is 658 million, against a point-of-sale terminal estate roughly two orders of magnitude smaller. There is no version of the card model that reaches a tea stall. There is an obvious version of the sticker model that does.

Collect requests, mandates and UPI AutoPay

UPI AutoPay, launched in 2020, is the recurring-payments layer. It is a genuine e-mandate rather than a stored credential: the customer authorises a mandate once, with additional factor of authentication, and the mandate is registered against their account at the remitter bank with a **Unique Mandate Number**. Subsequent executions do not require the PIN.

The parameters a practitioner needs are these. The customer receives a **pre-debit notification at least 24 hours before any execution**, so that funds can be arranged; NPCI waives the pre-debit notification only for auto-replenishment of NETC FASTag and RuPay NCMC balances. The **additional factor of authentication limit** — the ceiling below which no per-execution authentication is needed — was raised from **INR 15,000 to INR 1,00,000** for credit card bill payment, mutual fund and insurance use cases. The customer can modify, revoke, pause and unpause a mandate from within any UPI app, and that control must be exposed in the app rather than buried with the merchant, which is a materially stronger consumer position than a card-on-file subscription.

Frequency options run from one-time through daily, weekly, fortnightly, monthly, bi-monthly, quarterly, half-yearly, yearly and as-presented, which is why AutoPay swallowed use cases as varied as OTT subscriptions, insurance premiums, SIP mutual fund instalments, loan EMIs and utility bills.

One operational constraint matters for anyone building on AutoPay. From **1 August 2025**, NPCI began rate-limiting UPI APIs to protect the switch at peak. Balance enquiries are capped at **50 per app per day**, non-customer-initiated API calls are restricted during peak hours, and **peak hours are defined as 10:00 to 13:00 and 17:00 to 21:30**. Mandate executions are pushed into off-peak windows. If your batch of mandate executions is scheduled for eleven in the morning, it will not run when you expect it to.

Limits

NPCI sets ceilings; member banks set their own limits within them, and often lower. The default per-transaction and per-day ceiling for ordinary person-to-person and person-to-merchant use is **INR 1,00,000**, with banks free to impose tighter caps, and many impose a first-24-hour cap on a newly linked account and a cap on the number of transactions per day.

Higher ceilings apply to specific merchant categories, and the governing document is NPCI circular **NPCI/UPI/2025-26/OC/185B dated 28 August 2025**, an addendum to OC 185A, with a compliance date of **15 September 2025**. The enhanced limits apply only to merchants classified as **Verified Merchant** and compliant with NPCI's merchant onboarding circulars. Person-to-person limits were explicitly left unchanged.

Category	Merchant category code	Per transaction	Cumulative, 24 hours
Capital markets	6211	INR 5 lakh	INR 10 lakh
Insurance	5960	INR 5 lakh	INR 10 lakh
Government e-Marketplace, EMD payments	9311	INR 5 lakh	INR 10 lakh
Travel	4722	INR 5 lakh	INR 10 lakh
Credit card bill payments	5413	INR 5 lakh	INR 6 lakh
Collections	7322	INR 5 lakh	INR 10 lakh
Business or merchant, including pre-approved payments	Payer MCC not 0000, purpose code 20 or 15	INR 5 lakh	not applicable
Jewellery	5944	INR 2 lakh	INR 6 lakh
FX-Retail via BBPS	6012	INR 5 lakh	INR 5 lakh
Digital account opening, term deposits	7410	INR 5 lakh	INR 5 lakh
Digital account opening, initial funding	7409	INR 2 lakh	INR 2 lakh

Two adjacent products carry their own ceilings. **UPI Lite** is an on-device wallet designed for payments **below INR 1,000**, processed without a UPI PIN and without a round trip to the bank's core banking system; it exists to take load off issuers' cores and to keep passbooks uncluttered, and the balance behaves like cash in that losing the device without disabling the wallet means going to the bank for a refund. **UPI Circle**, which lets a primary user delegate spending authority to a secondary user or to a trusted device, carries limits of **INR 5,000 per transaction per device, INR 15,000 per month per device, INR 2,000 in the first 24 hours after linking, and a maximum of five linked secondary devices or software agents**, per NPCI's published product page at the time of writing.

All of these figures are accurate at the time of writing and all of them have moved before. Read the current circular.

Why it reached these volumes

It is tempting to attribute UPI's scale to India's population, but population explains the ceiling, not the trajectory. Several deliberate design choices did the work, and they are transferable.

Interoperability was mandatory from the first day. Any app could pay any bank account. There was never a period in which a merchant had to accept four different wallets, and there was never a network effect that a single incumbent could capture. Compare the closed-wallet era that immediately preceded it, in which a customer holding one wallet simply could not pay a merchant who accepted another.

The app layer was separated from the licensed layer. Firms with no balance sheet could compete on user experience alone. That created ferocious product competition without creating prudential risk, because customer money never rested outside a bank.

Authentication was made cheap. Device binding plus PIN removed the OTP round trip, which removes both latency and a large class of failures. A payment that completes in under two seconds gets used for a forty-rupee purchase. A payment that takes twenty seconds and fails one time in twenty does not.

Addressing was abstracted. The VPA removed both the friction of typing an account number and the risk of publishing one. The mandatory name-return closed most of the mis-payment gap that abstraction would otherwise have opened.

Merchant acceptance was made nearly free, at both ends. Zero MDR removed the merchant's financial reason to refuse, and the printed QR removed the merchant's capital cost to accept. Those two together are why acceptance reached the informal economy, which no card scheme has ever managed anywhere.

The prerequisites were already in place. Bank accounts from the Jan Dhan financial inclusion programme, identity from Aadhaar, and mobile connectivity at some of the lowest data prices in the world gave UPI an addressable base it did not have to create. Demonetisation in November 2016, seven months after launch, provided an accelerant that no product plan could have arranged.

The specification kept moving. UPI 2.0 in August 2018 added linked overdraft accounts, one-time mandates, invoice-in-the-inbox and signed intent and QR. Later layers added AutoPay, UPI Lite, UPI 123Pay for feature phones, RuPay credit card on UPI, credit line on UPI, UPI Circle and IPO applications on UPI. Each layer reused the same switch, the same addressing and the same authentication, which is why each shipped in months rather than years.

UPI 123Pay deserves a specific mention because it is the clearest evidence that the design was about reach rather than about smartphones. It offers feature-phone users four routes: calling an interactive voice response number, an app running on the feature phone itself, a missed-call-based flow, and proximity sound-based payments. None of these requires a data connection in the ordinary sense.

The results, in NPCI and government figures published on **30 April 2026** to mark ten years of the system: annual volume grew from **2 crore transactions in FY 2016-17 to 24,161.69 crore in FY 2025-26**, an increase of roughly twelve thousand times; annual value in FY 2025-26 exceeded **INR 314 lakh crore**; the daily average through 2025 was about **60 crore transactions**, rising past 66 crore; person-to-merchant payments made up **63 per cent of volume** while person-to-person made up **71 per cent of value**; and **86 per cent of merchant payments were below INR 500**. The International Monetary Fund, in a June 2025 report, recognised UPI as the world's largest real-time payment system by transaction volume, at roughly **49 per cent of global real-time payment volume**.

The honest caveats

Concentration is real. Two apps run about four fifths of the volume, the rule says 30 per cent, and the deadline has moved three times. An outage at one of those apps, or at one of the two dominant sponsor banks, degrades payments for a very large fraction of the country at once. NPCI publishes uptime and downtime statistics per member precisely because this is the live risk.

Economics are unresolved. Zero MDR made acceptance universal and left the ecosystem dependent on a government incentive scheme that has consistently funded only a fraction of industry cost. The August 2026 amendment to Section 10A is the first legislative acknowledgement that this cannot continue indefinitely. Whatever emerges will be a compromise between the merchant reach that free acceptance bought and the revenue that maintaining the rail requires.

Fraud has migrated rather than disappeared. The collect request was closed off; social engineering, mule accounts and fake customer-support numbers were not. NPCI's FY 2026-27 circular OC No. 234 on safeguarding user information in UPI, and the revision of turnaround times for fraud and wrong-credit chargebacks in URCS, are the current front line.

And the system is not finished. Circulars in the current financial year cover a limit management member portal, on-device biometric authentication with an enhanced transaction ceiling, numeric UPI IDs, and international QR acceptance. Anything in this chapter with a number attached should be checked against NPCI's circular list before it is relied upon in production. That is not a weakness of the writing. It is the nature of a rail that is still being built at the same time as it carries eight thousand payments a second.

42.98 Common wrong ideas

Wrong: UPI is an instant payment system end to end. **Right:** the customer experience is instant and the interbank plumbing is deferred net settlement, in ten cycles a day between 09:00 and 21:00 since 3 November 2025, and a practitioner who confuses the two will size their liquidity wrong.

Wrong: NPCI holds the money and shifts it between banks. **Right:** NPCI is a switch that routes messages, applies rules, keeps score and produces settlement files; the money moves across banks' current accounts at the Reserve Bank of India.

Wrong: India moves its money on UPI. **Right:** UPI is 85.5 per cent of digital payment volume and 9.5 per cent of value, while RTGS is 0.1 per cent of volume and 68.6 per cent of value; that claim describes counts, not sums.

Wrong: PhonePe, Google Pay and Paytm are banks. **Right:** they are third-party application providers riding on a sponsor payment service provider bank's licence, which is why the right question during an outage is which sponsor bank is down.

Wrong: a virtual payment address keeps the account holder anonymous. **Right:** it hides the account number from the payer, and nothing at all from NPCI or the payee's bank, and the registered name is deliberately returned to the payer before confirmation.

Wrong: UPI is free. **Right:** zero merchant discount rate is a statutory policy subsidised by member banks, NPCI switching fees and a government incentive scheme, three flows already sit outside the perimeter, and the statutory bar was removed on 6 August 2026.

Wrong: the transaction ID shown in the app is the reference to quote when chasing a payment. **Right:** the up-to-35-character transaction ID is the app's handle; the twelve-digit retrieval reference number, commonly called the UTR, is the number both banks can search on.

Wrong: if money leaves the account and does not arrive, the customer must chase it and hope. **Right:** auto-reversal is due latest on T+1 for a funds transfer and within T+5 at a merchant location, with INR 100 per day of delay beyond that, and the compensation is not discretionary.

Wrong: a collect request is a normal way to ask an individual for money. **Right:** person-to-person collect was progressively restricted and then discontinued entirely from 1 October 2025, because a request for money looks like an offer of money to an inexperienced user.

Wrong: mandate executions and API calls can be scheduled whenever it suits the merchant. **Right:** since 1 August 2025 NPCI rate-limits UPI APIs, caps balance enquiries at fifty per app per day, and pushes non-customer-initiated calls out of the peak windows of 10:00 to 13:00 and 17:00 to 21:30.

42.99 Chapter summary in 20 lines

1. UPI is India's retail payment switch, operated by NPCI, a not-for-profit company promoted by the Reserve Bank of India and the Indian Banks' Association, live since 11 April 2016.
2. NPCI holds no money: it keeps the address book, routes the messages, applies the rules and produces the settlement files.
3. A payment is addressed by virtual payment address, UPI number, QR code or raw account number plus IFSC, and the address is resolved to an account by NPCI's central mapper.
4. Before any debit, the payee's registered name is returned and displayed, which is why mis-payment is rare and why a UPI address cannot be treated as private.
5. Two-factor authentication is satisfied by a bound device plus a PIN, with no OTP round trip, which NPCI calls single-click two-factor authentication.
6. The PIN is encrypted inside NPCI's Common Library and decrypted only in the remitter bank's hardware security module, so neither the app nor the PSP ever sees it.
7. A push payment is two synchronous legs: the switch performs the debit at the remitter bank and only on success performs the credit at the beneficiary bank.
8. Failures come in three shapes: a clean decline, a credit failure that reverses within minutes, and a timeout that leaves the transaction in deemed status until reconciliation resolves it.
9. Settlement is deferred and net across members' current accounts at the Reserve Bank of India, in ten cycles a day between 09:00 and 21:00 since 3 November 2025, with two separate cycles for disputes.
10. Between the customer's instant result and settlement the beneficiary bank carries credit risk on the remitter bank, which is why NPCI runs net debit caps and collateral requirements.
11. The app layer was separated from the licensed layer, so firms with no balance sheet compete on user experience while customer money never rests outside a bank.
12. That separation also produced extreme concentration: two apps run about four fifths of volume against a 30 per cent cap whose deadline has moved three times and now stands at 31 December 2026.
13. In July 2026 UPI carried 23.66 billion transactions worth about INR 29.9 lakh crore across 741 banks, roughly 763 million payments a day.
14. The average payment was INR 1,263, about eleven pounds, which is the number that explains everything: this is a rail for chai and vegetables, not for flats.
15. Zero merchant discount rate, statutory since 1 January 2020, is why a tea stall accepts a laminated sticker that cost the price of printing rather than a terminal with a rental and a percentage.
16. On 6 August 2026 the Lok Sabha passed an amendment removing the statutory bar on charging, so the free-to-use position is now a policy choice rather than a law of nature.
17. When money vanishes, the twelve-digit reference is the only thing that lets a bank find it, and the escalation ladder runs app, PSP bank, NPCI's UPI Help portal, then the Reserve Bank Integrated Ombudsman Scheme.
18. UPI AutoPay is a genuine e-mandate with pre-debit notification, customer-side control and frequency options from daily to as-presented, subject to peak-hour rate limits.
19. The scale came from mandatory interoperability, cheap authentication, abstracted addressing, near-free acceptance at both ends, an existing base of accounts, identity and cheap data, and a specification that never stopped moving.
20. The unresolved problems are concentration, the economics of zero MDR, and fraud that migrated to social engineering rather than disappearing, which is why every figure here should be checked against NPCI's current circular before production use.

Sources: NPCI (UPI product statistics for July 2026, About UPI, UPI AutoPay, UPI Lite, UPI Circle, UPI

123PAY, UPI circular list, and circular NPCI/UPI/2025-26/OC/185B of 28 August 2025); Reserve Bank of India (circular RBI/2019-20/67 of 20 September 2019 on turn around time and customer compensation, Annual Report 2024-25, and the half-yearly Payment Systems Report covering H2 2025); Press Information Bureau releases of 30 April 2026 and on zero MDR; the Payment and Settlement Systems Act 2007 as amended, and reporting on the Taxation and Other Laws (Amendment) Bill, 2026 passed by the Lok Sabha on 6 August 2026.

43

CHAPTER

Instant Rails Elsewhere

43.0 What this chapter gives you

1. You will be able to explain why Brazil, the United States and the euro area reached three different outcomes from the same forty-year-old technology.
2. You will be able to name the design choices that predict adoption, central operation or compulsion, mandated reach, a rule-based price cap and memorable addressing, and test any new system against them.
3. You will be able to separate Pix the scheme from SPI the settlement system and DICT the directory, which everyday usage runs together.
4. You will be able to explain why a Pix payment and a Faster Payment look identical to a customer and put risk and liquidity in completely different places for a treasurer.
5. You will be able to argue that the United States has more instant payment infrastructure than most countries and less instant payment, and say exactly what it is missing.
6. You will be able to quote the three Same Day ACH deadlines and say what happens to a file that misses all of them.
7. You will be able to state what the Instant Payments Regulation actually obliges, and the date each obligation bites for euro-area banks, non-euro-area banks and payment institutions.
8. You will be able to read the TIPS settlement dip of October 2025 as the expected cost of a well-executed mandate rather than as a failure.
9. You will be able to say who is paying for a "free" instant payment in each of the three jurisdictions, and whether they will still be paying in five years.
10. You will be able to diagnose what Britain has, what it lacks, and why the missing piece is alias addressing rather than speed.

On the same Tuesday morning, three people pay the same small amount of money to somebody they can see.

In São Paulo, a woman buys two kilos of tomatoes at a street market. The stallholder turns a laminated card round so she can photograph the square black-and-white pattern on it. She taps twice, the stallholder's phone chimes, and the money is his, finally, irreversibly, at eleven o'clock in the morning. Neither of them paid anything to move it.

In Columbus, Ohio, a man pays his window cleaner. He has the window cleaner's bank routing number and account number written on a slip of paper, and he types them into his bank's website. It tells him the payment will arrive on Thursday. He also has the option to pay a fee to make it arrive today, and there is a third option, which is to write a cheque, and this is 2026.

In Rotterdam, a woman pays her plumber from her phone. Before she confirms, the screen shows her the name registered to the account she has typed, and it matches the name on the invoice. She presses send. The plumber's bank credits him within ten seconds, and by law her bank may not charge her more for this than it would have charged for a payment that took a day.

Three countries, three answers, one question. The question is the same everywhere and it is not a technical question. Moving a number from one ledger to another has been technically trivial for forty years. The question is who decides, who must join, and who pays. Brazil answered it one way, the United States another, and the European Union a third, and the differences between the three explain almost everything about why instant payments are ubiquitous in one place, legislated into existence in another, and still an optional extra in the third.

All figures, dates, limits and cut-off times in this chapter are accurate at the time of writing, August 2026. Instant payment systems are among the fastest-moving infrastructures in finance, and several of the numbers below changed within the last eighteen months.

The plain version

Imagine three towns, each with a river running through it and no bridge.

In the first town, the council decides that a bridge is not something you leave to chance. It builds the bridge itself, out of public money. Then it passes a rule: every haulage company in town above a certain size must use the bridge, and none of them may charge ordinary residents a penny to walk across. Businesses pay a small toll, but people do not. The bridge is open every hour of every day.

In the second town, the council does not build anything. It says that bridges are a matter for private enterprise. Two consortia eventually build two bridges, some distance apart. Neither bridge connects to every road. A haulier who wants to reach every address in town must have an arrangement with both, and many hauliers have an arrangement with neither, because the old ferry still runs and the ferry is cheap and everybody is used to the ferry. Some residents cross the river every day and have never used either bridge.

In the third town, the council also does not build the bridge. But it does something else. It passes a law: within eighteen months, every haulage company must be able to receive goods from every other haulage company across the river, within ten seconds, and may not charge more for the fast crossing than for the slow one. It then says: here is a piece of publicly owned road at the centre of the town that all the bridges must connect to. How you build your end is your business. That the connection exists is not negotiable.

The first town is Brazil. The second is the United States. The third is the European Union. And the striking thing, when you look at the traffic figures, is how well the outcome matches the design.

Brazil, and what happens when the state builds the road

Brazil's system is called Pix. It is run by the country's central bank, the Banco Central do Brasil, and it opened in November 2020.

Two things about Pix matter more than everything else combined.

The first is that if you are a bank or a payment company in Brazil with more than five hundred thousand active customer accounts, you are not asked whether you would like to join. You are required to. There was no launch period during which some banks had it and others did not, no competitive advantage for early movers, no customer ringing up to ask why her bank did not support it. From very close to the beginning, if you had a bank account in Brazil, you could send and receive.

The second is that banks are legally forbidden from charging ordinary people to use it. Not discouraged. Forbidden, in a written rule, with exceptions so narrow that they prove the point. A shop pays to receive. A person does not pay to send.

Then there is a third thing, less dramatic but arguably the cleverest. To pay someone in Brazil you do not need their bank details. You need a *key*: their mobile number, their email address, their national tax number, or a long random string their bank generated for them. The central bank runs a directory that maps every key to an account. You type the key, the directory returns the account, and it also returns the name on that account so you can check you are paying who you think you are paying. Businesses print a square barcode instead, which contains the same thing.

The numbers that follow from those three decisions are difficult to argue with. In 2021, its first full year, Pix carried 9.4 billion transactions. In 2024 it carried 63 billion, moving BRL 26.4 trillion, which the central bank's own deputy governor put at around two and a half times Brazil's entire economic output for that year. At the five-year mark, in November 2025, close to 170 million people were using it, in a country of roughly 215 million, along with more than 20 million businesses. As at 15 August 2026, 889 institutions took part.

For a shop, the cost of taking a Pix payment has been measured at about 0.22 per cent of the sale, against roughly 1.1 per cent for a debit card and 2.2 per cent for a credit card. That is not a marginal saving. That is the difference between a business model working and not working.

The United States, and what happens when nobody is in charge

The United States has the largest economy in the world and, for most of the last decade, some of the slowest ordinary payments in the developed world. The reason is not technical incompetence. It is that the United States has roughly nine thousand banks and credit unions, no legal power to make them all join anything, and a payment system that was already good enough that nobody had to.

The old system is called ACH, for Automated Clearing House. It is a batch system, in the exact sense of the Bacs cycle familiar to British readers: files of instructions are collected, sorted, exchanged and applied, with settlement in between. It carries wages, benefits, tax refunds, mortgage payments and most business-to-business invoicing. It is enormous. In 2025 it moved 35.2 billion payments worth USD 93 trillion.

It also got faster. Since 2016 there has been a same-day option, with three deadlines during the working day. Submit a file by 10:30 in the morning, New York time, and it settles at one o'clock the same afternoon. Submit by 2:45 and it settles at five. Submit by 4:45 and it settles at six. Miss all three and you are on tomorrow. That is not instant, but for a great many purposes it is close enough, and it costs a fraction of a wire transfer, and this is precisely why the United States was slow to build anything faster. The pressure was never quite unbearable.

When faster things were built, they were built twice. A group of the largest banks built one, called RTP, which went live in 2017. The Federal Reserve built another, called FedNow, which went live in July 2023. Both run around the clock. Both are genuinely instant. And there are two of them, which means that a bank must join both to be sure it can reach everyone, and many banks have joined neither.

The volumes tell the story plainly. In its first full year, 2024, FedNow settled 1.5 million payments. In 2025 it settled 8.4 million. That is fast growth from a small base, and it is a rounding error next to 35.2 billion ACH payments. Meanwhile, Americans do in fact send each other money instantly, in very large quantities, using a service called Zelle, which is not a payment rail at all but a directory and a set of rules sitting on top of the banks. In 2024 Zelle carried 3.6 billion payments worth more than a

trillion dollars. It solved the addressing problem, which is the problem people actually experience, and left the plumbing alone.

Europe, and what happens when you legislate

The euro area had the technology from 2017 and almost nobody used it. A pan-European instant payment scheme existed, banks could join it, and joining was voluntary, so a large proportion of banks did not. Money crossing Europe in ten seconds was possible in theory and unavailable in practice, because the bank at the other end had not signed up.

So the European Union passed a law. Under the Instant Payments Regulation, from 9 January 2025 every bank in the euro area had to be able to *receive* an instant euro payment. From 9 October 2025 every bank had to be able to *send* one, and had to offer a free check of the payee's name before you press the button, and was forbidden to charge more for the instant version than for the slow version.

The effect was immediate and visible in the settlement figures. The Eurosystem's instant settlement platform, TIPS, handled 1.35 billion transactions in 2024. In 2025 it handled 2.47 billion, a rise of 82.5 per cent, with the sharpest growth in the closing months of the year, exactly as the sending obligation bit. On 1 December 2025 it settled 5,759,570 payments in a single day.

Note what Europe did *not* do. It did not build a consumer product; there is no European Pix app. It legislated the two things Brazil got by other means, universal reach and price parity, and let the market do the rest.

The pattern

Put the three next to each other and the pattern is not subtle.

	Brazil	United States	Euro area
Who runs the instant rail	Central bank	Federal Reserve and a private bank consortium, in parallel	Eurosystem platform, private schemes on top
Must banks join	Yes, above a size threshold	No	Effectively yes, by law, since 2025
Can consumers be charged	No, by regulation	Yes	Not more than for a slow payment
How you address a payment	Phone, email, tax number or random key	Routing number and account number	IBAN, with a name check
Adoption	Near-universal	Growing from a small base	Rising steeply since the deadlines

Central operation, or at least central compulsion. Mandated reach. A cap on what the customer can be charged. Addressing a human being can remember. Where those four are present, instant payments take over. Where any of them is missing, they do not.

Where the plain version stops being true

The story above is true in outline and misleading in four specific places, and each of them matters if you are building anything.

"Instant" describes the customer's experience, not the banks'

When a Pix payment lands, the two banks involved have genuinely settled with each other, individually, transaction by transaction, in central bank money, at that instant. When a FedNow payment lands, the same is true. That is unusual.

For a great many fast payment systems it is not true at all. In the United Kingdom, a Faster Payment reaches the recipient in seconds, but the obligation between the two banks is settled afterwards, on a net basis, across accounts at the Bank of England, against balances the banks have prefunded. India's UPI, which moves more transactions than any system on earth, likewise gives the user an instant result over a deferred net settlement arrangement underneath. The customer cannot tell the difference and should not have to. The bank's treasurer very much can, because the two models put risk and liquidity in completely different places. Real-time gross settlement means every payment consumes central bank money at the moment it is made, all night and all weekend, which means a bank must fund an account it cannot easily top up out of hours. Deferred net settlement means the banks owe each other for a while, which is cheaper and introduces exposure between them.

So "instant" is a statement about the payee's balance, not about the plumbing. Two systems that look identical to a customer can be entirely different animals.

Public ownership correlates with adoption; it does not cause it

It is tempting to read Brazil's numbers and conclude that the state should build the rail. The evidence for that is real but weaker than the story suggests.

The Bank for International Settlements ran the comparison properly: thirteen jurisdictions, monthly data from April 2001 to December 2023, just over a thousand country-month observations, controlling for country and year. Public ownership of a fast payment system is associated with an increase of about 1.8 per cent in transactions per head. That is a genuine effect and it is the *smallest* of the four they found. Allowing non-bank payment firms to participate is associated with 3.5 per cent. Each additional use case is worth about 2 per cent, and so is each additional cross-border connection.

And there are counter-examples in plain sight. Sweden's Swish, one of the most thoroughly adopted instant payment products anywhere, is owned and operated entirely by private banks. India's UPI is governed jointly by private banks and the central bank through a purpose-built body. What Brazil got from public ownership was not superior engineering. It was the political authority to compel participation and to forbid consumer fees, and those are the levers that moved. A private consortium with the same two powers would have had much the same result. A central bank without them would not.

"Free" means somebody else is paying

Pix is free to individuals because a rule says it must be. It is not free to operate. The central bank charges participating institutions for the settlement infrastructure and for the addressing directory. Merchants pay to receive. The 0.22 per cent that a Brazilian shop pays is small next to a card fee, but it is not zero, and the entire consumer-facing economics of the system rest on it.

The same is true everywhere. The Federal Reserve charges banks 4.5 cents to originate a FedNow customer credit transfer, a dollar for a liquidity transfer and a cent for a request for payment, and in 2026 waived the twenty-five dollar monthly participation fee and rebated the first 2,500 credit transfers a month, which is a subsidy with a purpose. The Eurosystem charges 0.2 of a euro cent per instant payment, split between the sending and receiving bank. In Europe, the Instant Payments Regulation does not say instant payments must be free. It says they may not cost the customer *more* than a non-

instant one, which in most euro-area retail banking means free in practice and is a different legal statement.

If you are modelling a business on a rail because "it's free", find out who is paying and whether they are still going to be paying in five years.

The United States is not behind; it is differently arranged

The comparison that makes the United States look primitive is unit volume on instant rails, and by that measure it is far behind Brazil and India. But the United States has more instant payment infrastructure than most countries, not less. It has two independent real-time networks, one public and one private, both settling in central bank money, both with transaction ceilings far above anything in Europe or the United Kingdom: RTP raised its per-transaction limit to USD 10 million on 9 February 2025, and FedNow raised its network limit from USD 1 million to USD 10 million on 12 November 2025. British readers should sit with that. The Faster Payments scheme maximum is GBP 1 million.

What the United States lacks is not rails. It lacks the three things Brazil legislated: compulsory reach, a price cap at the consumer end, and one directory. It also has something Brazil did not have, which is a card industry so profitable and so entrenched that account-to-account payment at the point of sale has no natural sponsor. Nobody in the American payments industry makes more money when a customer pays a shop directly from their bank account. In Brazil, the central bank did, in the sense that it was pursuing a public objective rather than a margin.

And the volume comparison flatters the wrong number in any case. Same Day ACH alone carried 1.4 billion payments worth USD 3.9 trillion in 2025, up 16.7 per cent on the year, and Zelle carried 2 billion payments worth nearly USD 600 billion in the first half of 2025 alone. Americans are not waiting three days to pay each other. They are doing it over three different systems, none of which reaches everybody.

The technical version

Brazil: Pix, SPI and DICT

Pix is a payment scheme owned by the Banco Central do Brasil, established by BCB Resolution 1 of 12 August 2020 and launched in November 2020. It is important to separate three things that colloquial usage runs together.

Pix is the scheme: the rulebook, the branding, the user experience standards, the participation criteria.

SPI, the *Sistema de Pagamentos Instantâneos*, is the settlement infrastructure. It is a real-time gross settlement system built and operated by the BCB specifically for instant payments, running 24 hours a day, every day of the year. Direct participants hold dedicated Instant Payment accounts, known as PI accounts, at the central bank. Every Pix transaction settles individually across those accounts in central bank money at the moment it is made. Brazil already had a conventional RTGS system, the STR, when Pix was designed; the decision to build a separate platform rather than extend the existing one was a deliberate one about service levels and technology. Legally, SPI was built for Pix, and other payment schemes cannot connect to it for instant settlement.

DICT, the *Diretório de Identificadores de Contas Transacionais*, is the addressing directory. It maps aliases, known as Pix keys, to transaction accounts. Four key types are permitted: the CPF or CNPJ tax identifier, a mobile telephone number, an email address, and a randomly generated key, the EVP. Participants that are direct participants in SPI must access DICT directly.

Participation criteria are set out publicly. An institution licensed by the BCB that holds more than 500,000 active customer accounts, counting demand, savings and prepaid payment accounts together, is obliged to join, and once it crosses that threshold it has 90 days to apply for membership in the transaction account provider category. Smaller institutions may join voluntarily; institutions outside the regulatory perimeter may participate indirectly through a direct participant, subject to a contract, demonstrated technical capacity and a minimum equity requirement.

Pricing to end users is governed by BCB Resolution 19 of 1 October 2020, which is worth quoting because its structure is the model other jurisdictions keep reaching for. Article 3 prohibits an account-holding institution from charging a natural person customer, including individual entrepreneurs, for the remittance of funds for transfer or purchase, or for the receipt of funds for the purpose of transfer. The single paragraph carves out transactions conducted through in-person channels or the call centre where an electronic means was available. Article 4 permits charging a natural person for the receipt of funds *for the purpose of purchase*, which is how a sole trader taking payment for goods ends up paying, and permits charging legal entities for both sending and receiving.

The functional set has widened steadily: billing (Pix Cobrança), cash withdrawal at merchants (Pix Saque), cash back (Pix Troco), scheduled payments (Pix Agendado), contactless payment over NFC (Pix por Aproximação, launched 28 February 2025) and direct debit (Pix Automático, launched June 2025). Pix Automático is the structurally interesting one: any participant may offer it to corporate customers regardless of where the payer's account is held, which breaks the historic pattern in which recurring collection was the preserve of a handful of large banks that the biller had to bank with. Fraud handling runs through a Special Return Mechanism, the MED, which permits funds to be blocked and returned despite transactions being nominally irrevocable, with a further revision announced by the BCB.

Adoption, on the central bank's figures: 9.4 billion transactions worth around BRL 5 trillion in 2021; 63 billion transactions worth BRL 26.4 trillion in 2024; close to 170 million users and more than 20 million businesses at the five-year mark in November 2025. As at 15 August 2026 the BCB listed 889 participants, of which 560 were credit cooperatives and 173 payment institutions, which is itself a comment on how wide the participation rules are drawn.

The United States: ACH, Same Day ACH, FedNow and RTP

The ACH network is governed by the Nacha Operating Rules and cleared by two operators: FedACH, run by the Federal Reserve, and the Electronic Payments Network, run by The Clearing House. It carries credits and debits in batch, with deferred net settlement across Federal Reserve accounts.

In 2025 the network processed 35.2 billion payments valued at USD 93 trillion, the thirteenth consecutive year in which total value rose by at least a trillion dollars. December 2025 was the highest-volume month recorded, at 3.22 billion payments, and November 2025 set an average daily record of 151 million payments.

Same Day ACH is an expedited service within the same network, not a separate rail. The FedACH processing schedule, effective 12 September 2022, defines three same-day windows.

Transmission deadline	Target distribution	Settlement
10:30 ET	12:00 ET	13:00 ET, current day
14:45 ET	16:00 ET	17:00 ET, current day
16:45 ET	17:30 ET	18:00 ET, current day

Items not eligible for same-day settlement, including anything submitted in the later overnight windows, settle at 08:30 ET on the next banking day. The per-transaction limit for same-day eligibility is USD 1,000,000, effective 18 March 2022. Items above that limit submitted in a same-day window are not rejected; the operators assign them to the next available window for next-day settlement. Nacha has approved an increase of that limit to USD 10,000,000 with an effective date of 17 September 2027. In 2025, Same Day ACH carried 1.4 billion payments worth USD 3.9 trillion, up 16.7 per cent by volume and 21.4 per cent by value, averaging 5.8 million payments per day across the year and 7.8 million per day in December.

The FedNow Service launched in July 2023. It settles customer credit transfers individually in real time across participants' Federal Reserve accounts, using ISO 20022 messages: pacs.008 for the credit transfer, pacs.004 for the return, pacs.009 for a liquidity management transfer, pain.013 for a request for payment.

Operating hours are worth stating precisely, because "24/7" conceals a business-day convention. FedNow operates on a 24-hour funds transfer business day every day of the week including weekends and Federal Reserve holidays. The business day rolls over at approximately 19:01 ET each calendar day, with the new day beginning immediately as the previous one ends and no interruption in processing. Liquidity Management Transfers, which let a participant move funds between its Federal Reserve account and the balances supporting instant payments, run 19:00 to 07:00 ET on weekdays and continuously at weekends and on Federal Reserve holidays.

Value limits: the default per-transaction limit is USD 100,000, which a participant may adjust, and the network limit rose from USD 1 million to USD 10 million effective 12 November 2025.

The 2026 fee schedule, effective 1 January 2026:

Item	Fee
Customer credit transfer origination (pacs.008)	USD 0.045 per item
Customer credit transfer origination, on-us	USD 0.000
Return origination (pacs.004)	USD 0.045 per item
Liquidity Management Transfer (pacs.009)	USD 1.00 per item
Request for Payment (pain.013)	USD 0.01 per item
Participation fee	USD 25.00 per routing number per month, discounted to USD 0.00 in 2026
Origination discount	Minus USD 0.045 per item for up to 2,500 credit transfers per month in 2026

Settled volumes, from the Federal Reserve's own statistics, last updated 6 July 2026:

Period	Settled payments	Value (USD)	Average value (USD)	Average daily volume
2023 (part year)	47,262	18,433,363	390	285
2024	1,505,250	38,196,907,431	25,376	4,113
2025	8,413,402	853,411,108,511	101,435	23,050
2026 Q1	2,728,510	271,252,920,121	99,414	30,317
2026 Q2	4,997,811	274,663,803,964	54,957	54,921

The 2026 second quarter is the interesting line: volume up 83.2 per cent on the quarter while value was almost flat, and average payment value more than halved. That is the signature of a wholesale-ish

network beginning to carry retail traffic. Participation was reported by the Federal Reserve at more than 1,500 financial institutions across all fifty states in autumn 2025, with 97 more institutions and four certified service providers joining in the first quarter of 2026.

The RTP network, operated by The Clearing House, launched in 2017 and predates FedNow by six years. It settles from a prefunded joint balance held at the Federal Reserve. Its per-transaction limit rose to USD 10 million effective 9 February 2025, a tenfold increase on the previous USD 1 million cap. At the end of 2024 The Clearing House reported the network averaging about one million payments a day, with some 285,000 businesses a month originating payments through participating institutions; by mid-2025 it reported more than 1,000 banks and credit unions live, a 51 per cent increase year on year.

Zelle is neither of these. It is an overlay operated by Early Warning Services, owned by a group of large banks, which resolves an email address or mobile number to an account and moves the resulting instruction over whichever underlying rail the participating banks use. It is the closest thing the United States has to a DICT. It carried 3.6 billion payments and more than USD 1 trillion in 2024, and 2 billion payments worth nearly USD 600 billion in the first half of 2025.

Why the United States is unusual, stated plainly. There is no legal power to compel roughly nine thousand depository institutions to join a payment system. There are two competing instant networks, so reach is a commercial question rather than a given. Addressing is by routing transit number and account number, credentials which are both hard to remember and, being reusable, unsafe to publish. Consumer pricing is unregulated. Cheques remain in use. And the existing batch rail is cheap, extremely reliable and, since 2016, same-day capable, which removed most of the urgency. The Bank for International Settlements notes a further constraint: FedNow, like Mexico's CoDi and Sweden's Swish, is open only to banks, in some cases for legislative reasons, and non-bank participation is one of the strongest correlates of adoption they measured.

Europe: T2, TIPS and the Instant Payments Regulation

Terminology first, because the old name persists and is now wrong. TARGET2 was replaced on 20 March 2023 by **T2**, following consolidation with the securities settlement platform T2S. TARGET Services today comprise T2 for high-value and wholesale euro payments, T2S for securities, TIPS for instant payments and, since 16 June 2025, the Eurosystem Collateral Management System.

T2 is the euro area's real-time gross settlement system: the direct analogue of CHAPS within the Bank of England's RTGS, at continental scale. In 2025 it settled an average of 431,067 payments per day in euro, worth EUR 1,932.8 billion, across 255 operating days, for a yearly total of 111,910,103 transactions worth EUR 492,859.0 billion. That total exceeded the previous peak of 107,999,982 transactions set in 2024, and the value grew 6.3 per cent, ahead of euro-area nominal GDP growth of 3.8 per cent.

The business day matters for anyone building payment operations against it. The real-time settlement window concludes at 18:00 CET and may be interrupted by an optional maintenance window from 03:00 to 05:00 CET. The cut-off for customer payments is 17:00 CET; all other categories may settle until 18:00 CET. In 2025, 99.7 per cent of volume and 96.3 per cent of value had settled by the 17:00 customer cut-off, with the remaining payments in the final hour typically few in number and very large in value, as banks square positions and refinance on the money market.

TIPS, TARGET Instant Payment Settlement, launched in November 2018. It settles instant credit transfers in central bank money, 24 hours a day, 365 days a year. Participants hold a TIPS Dedicated Cash Account, a DCA, funded from their RTGS position; the alternative structure is an Ancillary System Technical Account, an ASTA, used where a clearing mechanism sits between the participant and the

platform. Pricing is EUR 0.002 per instant payment, split equally between originator and beneficiary at EUR 0.001 each.

TIPS is a settlement platform, not a scheme. The scheme is **SCT Inst**, the SEPA Instant Credit Transfer scheme of the European Payments Council, launched in November 2017. Participation in TIPS, directly or indirectly, is mandatory for payment service providers that have signed up to SCT Inst and are reachable via T2. Until 9 October 2025 the scheme-level default maximum amount was EUR 100,000; that ceiling has been removed.

The **Instant Payments Regulation**, Regulation (EU) 2024/886, was adopted on 13 March 2024. It amends the SEPA Regulation, the cross-border payments regulation, the Settlement Finality Directive and PSD2. Its four substantive obligations are that a payment service provider offering ordinary credit transfers must also offer instant ones (Article 5a); that charges for instant credit transfers may not exceed charges for other credit transfers of corresponding type (Article 5b); that providers must offer a free verification of payee service before the payer authorises, returning a match, close match, no match or other result (Article 5c); and that providers must screen their own customer base against targeted financial sanctions at least daily rather than screening each transaction in flight (Article 5d).

The implementation timetable is staggered and precise.

Requirement	Who	Deadline
Receive instant payments	Euro area Member States	9 January 2025
Receive instant payments	Non-euro area Member States	9 January 2027
Receive instant payments	EMIs and payment institutions, euro and non-euro area	9 April 2027
Send instant payments	Euro area Member States	9 October 2025
Send instant payments	Non-euro area Member States	9 July 2027
Send instant payments	EMIs and payment institutions, euro area	9 April 2027
Send instant payments	EMIs and payment institutions, non-euro area	9 July 2027
Send outside business hours from accounts in national currency	Non-euro area Member States	9 June 2028
Equality of charges	Euro area Member States	9 January 2025
Equality of charges	Non-euro area Member States	9 January 2027
Verification of payee	Euro area Member States	9 October 2025
Verification of payee	Non-euro area Member States	9 July 2027

The Eurosystem offers a verification of payee service built on solutions developed by Banco de Portugal and Latvijas Banka, conforming to the European Payments Council's VoP scheme rulebook, effective 5 October 2025. Separately, from October 2025, non-bank payment service providers meeting the conditions in the TARGET Guideline may access TARGET Services directly, including T2 and TIPS, a change with the same intent as Brazil's wide participation categories.

The effect on volumes was sharp and unambiguous. TIPS transactions rose from 1.35 billion in 2024 to 2.47 billion in 2025, growth of 82.5 per cent. Euro-denominated TIPS activity averaged 2,735,053 payments per day in 2025, worth EUR 2.2 billion, up 110.8 per cent from a daily average of 1,297,241 in 2024. December 2025 averaged 3,845,376 payments a day and the maximum single day was 5,759,570 on 1 December 2025. Germany overtook Italy as the largest contributor by volume, rising from 15.2 to 19.8 per cent of the total, and contributed 27.4 per cent of value.

The stress shows in the settlement ratio. Across 2025, 98.0 per cent of euro traffic settled by volume and 95.8 per cent by value. The value ratio fell to 93.1 per cent in October, coinciding with the sending obligation, as participants adapted to the shorter processing timeline, newly connected participants struggled with genuine 24/7 operation, and much larger payments arrived following the removal of the EUR 100,000 ceiling. By December it had recovered to 96.5 per cent. That is what a well-executed mandate looks like from the inside: a measurable two-month degradation, then recovery.

TARGET Services are also becoming multi-currency. The Danish krone joined T2 and TIPS in April 2025, with T2 settling an average of 11,425 krone payments per day worth DKK 1,641.9 billion in 2025, and the migration of Danish instant payments contributed materially to TIPS growth. Cross-currency capability was implemented in TIPS in June 2025, and the ECB is exploring interlinking TIPS with fast payment systems elsewhere.

What the successful systems have in common

Set Brazil, the euro area, India, Thailand and the United Kingdom side by side and five design choices recur wherever adoption is high. They are not all necessary, but the more of them you have, the steeper the curve.

Central operation or central compulsion. Someone with authority has to be able to say that this exists and everyone connects to it. In Brazil that authority is ownership of the scheme and the infrastructure. In the euro area it is a regulation with dates in it. What matters is not the flag on the building; it is that reachability stops being a commercial decision.

Mandated participation, and therefore guaranteed reach. This is the single largest determinant, and it is upstream of everything else. A payment system a customer cannot rely on reaching the other party is not a payment system; it is a feature. Brazil compelled institutions above 500,000 active accounts. Europe compelled everyone, in two stages. The United States compelled nobody, and the outcome is exactly what you would predict.

Low or zero cost at the consumer end, established by rule. Pix forbids charging individuals. The Instant Payments Regulation forbids charging more than for a slow transfer. Neither leaves it to competition, because competition in retail banking does not reliably drive a per-transaction fee to zero, particularly where the incumbent alternative pays the bank a share of interchange.

Simple addressing, backed by a directory, with a name check. A payment instrument that requires the payer to obtain and correctly transcribe an account credential has a ceiling on its use. Aliases remove that ceiling. Brazil built DICT. India built the virtual payment address. Europe legislated verification of payee, which is a name check rather than an alias directory, and is a weaker version of the same idea. The United States has aliases, but they live in a private overlay rather than in the rail.

A settlement account that is open when the rail is open. An instant system running at three in the morning on a Sunday needs participants to hold funded positions at the central bank outside banking hours, and needs a way to top them up. The Eurosystem gives participants TIPS DCAs. The Federal Reserve gives them Liquidity Management Transfers with defined overnight and weekend windows. The BCB gives direct participants remunerated PI account balances, liquidity lines, and a window after the close of the RTGS system to move funds across. Skip this and the system works beautifully until a large payment arrives on a bank holiday.

There is a sixth, less often stated: **one scheme, not several**. Two competing instant networks in one currency area do not double the reach; they halve the certainty. The United States is the demonstration.

What that predicts for other markets

For the United Kingdom, the diagnosis is uncomfortable, because Britain was early. Faster Payments arrived in 2008, before almost anyone, and being early froze in the design assumptions of 2008. The country has central operation, effective reach, and no consumer charge for retail Faster Payments in practice. What it does not have is alias addressing. A British payment is still addressed by six-digit sort code and eight-digit account number, exactly as it was in 1968, and Confirmation of Payee is a check on that credential rather than a replacement for it. Britain has the Europe answer to fraud without the Brazil answer to addressing. Open banking has partially filled the gap, but as an API layer that initiates payments over Faster Payments it inherits the addressing model rather than replacing it. Any future British improvement will have to solve identity resolution, and the scheme maximum of GBP 1 million per Faster Payments transaction will look increasingly conservative next to the USD 10 million now permitted on both American instant networks.

For the United States, expect convergence on reach without convergence on scheme. Both networks are growing, both have raised limits far beyond international norms, and the FedNow figures for the second quarter of 2026 show average payment value falling as volume climbs, which is retail adoption beginning. What will not happen without legislation is a single directory or a consumer price cap, so Zelle will continue to be the alias layer and card interchange will continue to fund the resistance at the point of sale.

For the euro area, the reach problem is solved and the product problem is not. TIPS settles the payments; it does not give a consumer a reason to use one. That gap is where private schemes layered on SCT Inst are competing, and their success or failure over the next few years will show whether legislated infrastructure is sufficient without a consumer-facing brand. Brazil's answer suggests it is not: Pix succeeded partly because the central bank specified minimum user experience requirements and a single recognisable name rather than leaving presentation to 889 institutions.

For markets that have not yet built, the BIS evidence gives a clear order of operations. Admit non-bank payment service providers, because that was the strongest single correlate they measured. Launch with several use cases rather than one, because each additional use case is worth about as much as public ownership. Plan cross-border connections early. And, though the data were not available to test it directly, mandate participation, because that is the mechanism by which Brazil obtained in one year what voluntary schemes in Europe failed to obtain in seven.

The counter-pressure is fraud, and it is the one thing every fast system has had to retrofit. Irrevocable, instant, push payments to an account you cannot inspect are an excellent instrument for a confidence trick, which is why Brazil built the MED into an otherwise irrevocable scheme, why Europe legislated verification of payee alongside the sending obligation rather than afterwards, and why the United Kingdom has operated a mandatory reimbursement regime for authorised push payment fraud since October 2024. Speed and reversibility trade against each other, and the trade cannot be designed away, only priced and allocated. Every jurisdiction that has built a fast rail has ended up building a slow, human, contested process alongside it for the payments that should not have happened. That process is not a failure of the design. It is part of the design, and the ones that planned for it from the start have had a considerably easier time than the ones that did not.

43.98 Common wrong ideas

Wrong: if a payment lands instantly, the two banks have settled. **Right:** that is true of Pix over SPI and of FedNow, and false of Faster Payments and UPI, which give an instant customer result over deferred net settlement.

Wrong: public ownership is what makes a fast payment system succeed. **Right:** the BIS comparison found public ownership the smallest of four effects at about 1.8 per cent, behind non-bank participation at 3.5 per cent; what Brazil actually used was the authority to compel participation and forbid consumer fees.

Wrong: Pix is free. **Right:** it is free to individuals because BCB Resolution 19 says it must be, while merchants pay about 0.22 per cent of the sale and participating institutions pay the central bank for settlement and for the directory.

Wrong: the Instant Payments Regulation made instant euro payments free. **Right:** it forbids charging more for an instant credit transfer than for a corresponding non-instant one, which usually means free in practice and is a different legal statement.

Wrong: the United States is technologically behind on instant payments. **Right:** it runs two independent real-time networks settling in central bank money, both now with USD 10 million ceilings against the Faster Payments maximum of GBP 1 million; what it lacks is compulsory reach, a consumer price cap and one directory.

Wrong: Americans wait days to pay each other. **Right:** Same Day ACH carried 1.4 billion payments worth USD 3.9 trillion in 2025 and Zelle carried 2 billion in the first half of that year; the traffic is spread across systems, none of which reaches everybody.

Wrong: Zelle is an instant payment rail. **Right:** it is an overlay operated by Early Warning Services that resolves an email address or mobile number to an account and rides whichever underlying rail the banks use; it is the closest thing the United States has to DICT.

Wrong: two competing instant networks give a country better coverage than one. **Right:** they do not double the reach, they halve the certainty, because a bank must join both to be sure of reaching everyone and many have joined neither.

Wrong: TARGET2 is the euro area's real-time gross settlement system. **Right:** TARGET2 was replaced by T2 on 20 March 2023; TIPS is the instant settlement platform, and SCT Inst is the scheme that runs over it.

Wrong: an instant, irrevocable rail removes the need for a dispute process. **Right:** every jurisdiction that has built one has retrofitted a slow, human, contested process alongside it, from Brazil's MED to Europe's verification of payee to Britain's mandatory reimbursement regime.

43.99 Chapter summary in 20 lines

1. Moving a number from one ledger to another has been technically trivial for forty years, so the interesting questions are who decides, who must join and who pays.
2. Brazil built the rail publicly, the United States left it to the market, and the European Union legislated, and the traffic figures match the designs.
3. Pix is a scheme owned by the Banco Central do Brasil, launched in November 2020, settling individually and in real time across dedicated PI accounts on SPI, every hour of every day.
4. Institutions licensed by the BCB with more than 500,000 active accounts are obliged to join, so reachability was never a commercial decision.
5. BCB Resolution 19 forbids charging a natural person to send, while a shop pays about 0.22 per cent against roughly 1.1 per cent for debit and 2.2 per cent for credit cards.
6. DICT maps four kinds of key, tax identifier, mobile number, email address and a random string, to accounts, and returns the account name so the payer can check.

7. Those three decisions produced 63 billion transactions worth BRL 26.4 trillion in 2024 and close to 170 million users by the five-year mark.
8. The United States has roughly nine thousand depository institutions, no legal power to compel any of them to join anything, and a batch rail that was already good enough.
9. ACH carried 35.2 billion payments worth USD 93 trillion in 2025, and has had three same-day windows since 2016, which removed most of the urgency to build anything faster.
10. When faster rails were built they were built twice, RTP in 2017 and FedNow in 2023, so reach became a commercial question rather than a given.
11. FedNow settled 8.4 million payments in 2025, a rounding error next to ACH, while Zelle carried 3.6 billion in 2024 by solving the addressing problem and leaving the plumbing alone.
12. The euro area had SCT Inst from 2017 and almost nobody used it, because joining was voluntary and the bank at the other end had not joined.
13. Regulation (EU) 2024/886 required euro-area providers to receive instant payments from 9 January 2025 and to send them, check payee names and match prices from 9 October 2025.
14. TIPS volumes rose 82.5 per cent to 2.47 billion in 2025, with a measurable settlement-ratio dip in October and recovery by December, which is what a well-executed mandate looks like from the inside.
15. "Instant" is a statement about the payee's balance, not about the plumbing, and real-time gross and deferred net settlement put risk and liquidity in entirely different places.
16. Nothing is free: Brazilian merchants, FedNow's 4.5 cents per credit transfer and the Eurosystem's 0.2 of a euro cent are all somebody paying for the rail.
17. The BIS evidence ranks non-bank participation, additional use cases and cross-border links above public ownership as correlates of adoption.
18. Five things recur wherever adoption is high: central operation or compulsion, mandated reach, a rule-based price cap at the consumer end, alias addressing with a name check, and a settlement account open whenever the rail is open, with a sixth being one scheme rather than several.
19. Britain was early, and being early froze in the assumptions of 2008: it has reach and no consumer charge, but still addresses payments by sort code and account number, with Confirmation of Payee as a check on that credential rather than a replacement for it.
20. Speed and reversibility trade against each other and the trade cannot be designed away, only priced and allocated, which is why the slow human fraud process beside a fast rail is part of the design rather than a failure of it.

Sources: Banco Central do Brasil, on Pix participation criteria, SPI and DICT, BCB Resolution 1 of 12 August 2020, BCB Resolution 19 of 1 October 2020, the Pix Statistics participant tables as at 15 August 2026, and the "Pix at 5" press article of November 2025; Federal Reserve Financial Services, on the FedNow Service volume and value statistics updated 6 July 2026, operating hours, the 2026 fee schedule, the transaction limit increase effective 12 November 2025, and the FedACH processing schedule effective 12 September 2022; Nacha, on 2025 ACH Network volume and value, Same Day ACH limits and the increase to USD 10 million effective 17 September 2027; The Clearing House, on the RTP network transaction limit effective 9 February 2025 and participation figures; Early Warning Services, on Zelle volumes; European Central Bank, TARGET Services Annual Report 2025, and the ECB pages on the Instant Payments Regulation, verification of payee and TIPS; European Payments Council, on the SCT Inst and VoP scheme rulebooks; Regulation (EU) 2024/886; Bank for International Settlements, "Fast payments: design and adoption", BIS Quarterly Review, March 2024, and BIS Bulletin no 52 on Brazil's Pix; International Monetary Fund, "Pix: Brazil's Successful Instant Payment System", 2023.

44

CHAPTER

Open Banking

44.0 What this chapter gives you

1. You will be able to explain why open banking is not a payment system, and name the rail that actually carried the £340.
2. You will be able to say what a payment initiation service provider may and may not do, and why only the account holder can authorise money to move.
3. You will be able to price an open banking payment against a card payment and say why the saving grows with the average transaction value rather than staying constant.
4. You will be able to tell a merchant exactly what they gain and what their customer gives up when a checkout moves from cards to pay-by-bank.
5. You will be able to distinguish the Payment Services Regulations 2017 from the CMA's Retail Banking Market Investigation Order 2017, and say which binds every bank and which produced the specification and binds nine.
6. You will be able to explain why "open banking works with your bank" is a claim about one institution on one day, and build a coverage matrix instead of relying on a principle.
7. You will be able to describe a variable recurring payment's control parameters and say why sweeping is free at the point of use and commercial VRP is not.
8. You will be able to state what the ninety-day rule now requires and why most published explanations of it are out of date.
9. You will be able to trace a payment initiation from consent creation to a terminal status, and say why building a checkout that waits for one is a mistake.
10. You will be able to explain what regulation 79 does, why its application to commercial VRP is unsettled, and why the Direct Debit Guarantee is the standard VRP will be judged against.

At twenty past nine on a Wednesday evening, a woman in Bristol pays a quarterly energy bill of £340. She is on the supplier's website. Next to the card fields there is a button that says "Pay by bank". She presses it, picks her bank from a list of logos, and her phone opens her banking app. The app shows her three lines: the amount, £340.00, the name of the energy supplier, and the account the money will leave. She holds her thumb to the sensor. The app closes, the supplier's page refreshes, and the bill is marked paid. The whole thing takes about eleven seconds. The supplier has the money, finally and completely, before she has put her phone down.

At two o'clock the following morning, while she is asleep, £85 moves out of her current account and into a savings account she holds with a different firm. She set this up seven months ago and has not thought about it since. It happens because the balance in her current account crossed a threshold she chose. No standing order was involved. No direct debit was involved. Nobody at either institution touched anything.

Both of those movements used the same machinery, and the machinery has a name that misleads almost everybody who hears it. It is called open banking, and the most important thing to understand about it is a negative: it is not a payment system. No money moves through it. It is a way of asking a bank to do something the bank could already do, and of asking in a form the bank is legally obliged to answer. The £340 travelled down the Faster Payments Service, the same pipe that has carried British instant retail payments since 2008. The £85 travelled down it too. Open banking supplied the instruction and the permission. It did not supply the road.

That distinction sounds pedantic. It is in fact the source of nearly every important consequence in this chapter, including the one that matters most to anyone building a business on it, which is what happens when the customer wants their money back.

All figures, dates, versions and limits in this chapter are accurate at the time of writing, August 2026. This is the fastest-moving area in the entire book. The commercial arrangement that makes open banking payments viable for ordinary businesses went live ten weeks before this was written, the legal framework that will govern the whole thing from the late 2020s has not yet been made, and the body that will run the standard does not yet exist. Check every date below against a primary source before you rely on it.

The plain version

Think of your bank as a large office building with a counter at the front. You go to the counter, you prove who you are, and the clerk does things for you: shows you your statements, moves your money.

Now suppose you want somebody to help you. Maybe an app that adds up all your accounts across four different banks and tells you what you actually spend on food. Maybe a shop that wants to be paid without the faff of you typing an account number off an invoice.

Before 2018, there was only one way to let a helper do either of those jobs. You gave them your keys. You told the app your online banking username and password, and it logged in pretending to be you, walked around inside your account, and copied down what it found. This was, and this is not too strong a word, a disaster. Your bank could not tell the helper apart from a thief, because there was nothing to tell apart: they were using the same credentials in the same way. If something went wrong nobody could say whose fault it was. And it broke every time the bank redesigned its login page.

Open banking replaced the keys with two service hatches cut into the wall of the building, and a rule that the bank has to keep them open.

Through the first hatch, a helper who holds a **reader's pass** can ask for a copy of your statements. Not your password, not the ability to move anything, just the numbers: balances, transactions, standing orders, the name on the account. That reader's pass is a real licence, issued by the financial regulator, and the helper's name goes on a public list.

Through the second hatch, a helper who holds a **form-filler's pass** can hand in a payment instruction on your behalf. And here is the part that carries all the weight: the form-filler cannot sign the form. They fill in the payee, the amount, the reference. Then they hand it back to you, and you walk to the counter yourself and sign it, with your own thumbprint or your own passcode, in your own bank's app. The bank checks the signature, not the form-filler's word. Then the bank pays.

That is the whole idea. Two hatches, a list of who is allowed to knock, and a hard rule that only the account holder can authorise money to move.

What it costs, and who it costs

Here is why a business cares, using the £340 energy bill.

If our Bristol customer had paid by debit card, a slice of that £340 would have gone to the bank that issued her card. That slice is capped by law at 0.2 per cent for a consumer debit card, which is 68 pence. If she had used a credit card the cap is 0.3 per cent, which is £1.02. Those caps only cover one slice; the card network takes a fee and the supplier's own payment provider takes a margin, and the all-in cost is meaningfully higher than the cap.

The key thing is not that the number is big. The key thing is that it is a *percentage*. Doubling the bill doubles the fee. A business collecting £3,400 pays ten times what a business collecting £340 pays, for exactly the same amount of work.

An open banking payment is not priced that way. The form-filler charges the business a fee per payment, quoted in pence, and it does not move when the amount moves. Collecting £3,400 costs the same as collecting £34. For a business with large average transaction values, an energy supplier, a letting agent, an investment platform, a car dealer, that is not a discount. That is a different order of magnitude.

The second difference is timing. The card payment showed up as approved instantly, but the supplier would not actually have the cash for a day or two. The open banking payment went over Faster Payments, which runs every hour of every day including Christmas morning, and the money was in the supplier's account in seconds. On a Wednesday evening. Which is when people pay bills.

The third difference is the one nobody puts on the marketing page. With the card, the supplier's money can be taken back off them, months later, if the customer complains to her bank and the bank agrees. With the open banking payment, it cannot. Once a Faster Payment has gone, it has gone. If the customer wants a refund, the supplier has to choose to send her one.

That is the trade, in one sentence: **the merchant gets the money faster, cheaper and for keeps, and the customer gives up the right to reach back and take it.**

Standing orders that can change their mind

The £85 that moved at two in the morning is the other half of open banking, and it has a clumsy name: a variable recurring payment, or VRP.

A standing order is a fixed instruction: £85 on the 3rd of every month, until I say stop. It cannot adapt. A direct debit is the opposite: the company you are paying decides the amount, and tells your bank to take it. That is flexible, but you have handed the steering wheel to somebody else.

A VRP sits between the two. You sign one form, once, that says something like: *this named helper may take money from my account, but never more than £200 in a single payment, never more than £600 in a calendar month, only to this one named destination, and only until 5 April next year.* After that, the helper can make payments inside those walls without waking you up, and cannot make one that steps outside them by a penny. Your bank enforces the walls, not the helper.

Right now the overwhelming majority of live VRPs do one job, and it has a name too: **sweeping**. Sweeping means moving your own money between your own accounts. Current account into savings when the balance goes above a threshold. Savings into current account when the balance goes below one, so you do not slide into an overdraft. Spare cash into an account that repays a loan early. The nine largest British banks were required to build sweeping, and they have all built it.

The interesting frontier, and it is only just open, is letting a VRP pay somebody else: your energy company, your council, a charity, a subscription. That is called commercial VRP, and it went live in Britain on 2 June 2026 under a new industry scheme. It is the first serious attempt to build something that does the direct debit's job without the direct debit's three-day cycle and without the direct debit's paper-era limitations.

How big is this

Open banking payments made in the UK in 2025	351 million	up 57 per cent on 2024
Successful requests answered by banks in 2025	24.0 billion	up 27 per cent
User connections at December 2025	16.5 million	up 36 per cent
Successful requests in June 2026 alone	2.8055 billion	99.50 per cent success rate
Average time a bank took to answer, June 2026	349 milliseconds	

Those are Open Banking Limited's own published figures. In mid-2026 the ecosystem passed one billion open banking payments and one hundred billion requests since it started in 2018.

For scale: 31 million open banking payments were made in March 2025, which Open Banking Limited put at roughly one in every thirteen Faster Payments. So it is real, it is growing quickly, and it is still a minority of a minority of how Britain moves money.

The one-line summary

Open banking is a set of doors cut into the wall of your bank, a licensed list of who may knock, a rule that the bank must answer, and an absolute requirement that only you can authorise money to leave. Everything behind the door is the old plumbing.

Where the plain version stops being true

The hatch is not the same size at every bank

The plain version says "the bank has to keep the hatch open", and that is true, but it hides that two entirely different pieces of law are doing two entirely different jobs, and they cover different institutions.

The first is the Payment Services Regulations 2017, which apply to every account servicing payment service provider in the country. They require access. They say almost nothing about what that access looks like. Regulation 69 says a bank must communicate securely with a payment initiation service provider and treat its payment orders exactly as it treats orders coming direct from the customer. Regulation 70 says the same for account information. Neither regulation specifies a data format, a field name, an uptime target, or a response time.

The second is the Competition and Markets Authority's Retail Banking Market Investigation Order 2017, made under the Enterprise Act 2002. That is where the actual specification came from, and it binds nine institutions: Barclays, HSBC, Lloyds Banking Group, Nationwide, NatWest Group, Santander, Danske Bank, Bank of Ireland and Allied Irish Bank. Those nine, universally called the CMA9, were required to fund and implement a common standard against a published roadmap, and to submit to monitoring.

Everybody else built whatever they liked, or built to the standard voluntarily, or built something that technically satisfies the regulation and is miserable to integrate with. Sweeping VRP was mandated on the CMA9 alone. So the sentence "open banking works with your bank" is a claim about one specific institution on one specific day, not a claim about the United Kingdom. Anyone building a product needs a coverage matrix, not a principle.

There is not one consent, there are at least two, and they drift

The plain version has you signing one form. In reality two separate permissions exist, they are held by two different parties, they have different lifetimes, and cancelling one does not reliably cancel the other.

The first is the consent you give the third party. That is a contractual and data-protection permission: you told this app it may look at your accounts, for these purposes, for this long.

The second is the authorisation you give your bank. That is a strong customer authentication event which produces an access token and a refresh token, held by the third party, scoped to a consent resource that lives on the bank's servers.

They come apart routinely. Delete the app from your phone and the tokens keep working. Cancel in the app and the bank may still show a live consent in its "connected services" screen. Cancel at the bank and the app's next call fails, but the app may not tell you why.

The famous ninety-day rule has also changed, and most explanations of it are now wrong. Originally the bank had to re-authenticate you every ninety days for data access to continue, which meant a monthly-budgeting app dragged you through your bank's login four times a year for every bank you had connected. The Financial Conduct Authority changed this in policy statement PS21/19, adding a new Article 10A to the UK version of the technical standards. Since 26 March 2022, the obligation is a *reconfirmation of consent* that the account information provider obtains from you, at least every ninety days, rather than a *re-authentication* the bank performs. The FCA expected banks to adopt the exemption by 30 September 2022. The friction moved from the bank's app to the third party's app, which was the point. The ninety days did not go away.

The money arrives instantly, but that is not the same as knowing it arrived

The plain version says the supplier has the money in seconds. Usually true. What is not true is that the supplier's software has been told so with certainty.

A payment initiation service provider does not receive a settlement confirmation from Pay.UK. It receives a status from the bank about the payment order, which is a different object in a different system. Statuses lag, sometimes by minutes, and a number of banks stop at an intermediate status and never emit the terminal one. Building a checkout that waits for final confirmation before releasing goods is a good way to build a checkout that hangs.

The limits are similarly misleading. The Faster Payments scheme maximum is £1 million per transaction, and you will almost never meet it. Individual banks set their own lower ceilings, and the ceiling applied to payments initiated through open banking is frequently lower than the ceiling the same bank applies inside its own app for the same customer. Some banks step the limit by authentication method, some by account age, some by whether the payee is already known. Never size a product against the scheme limit. Size it against the worst bank on your coverage matrix, and find out what that number is by testing rather than by asking.

And "instant" has a legal floor rather than a legal guarantee. Pay.UK's own description is that payments between direct participants normally arrive almost immediately but can occasionally take up to two hours, with the statutory backstop being end of the following business day.

It is not a cheaper card, and treating it as one is expensive

This is the correction that costs real money.

A card payment is a *pull* with a dispute system bolted to it. Two dispute systems, in fact. Chargeback is a set of private rules written by the card schemes: the customer complains, the issuer reverses the transaction, the merchant has to defend it. It is not law, and it exists because the schemes decided it should. On top of that, for credit cards, section 75 of the Consumer Credit Act 1974 makes the card issuer jointly and severally liable with the retailer for breach of contract or misrepresentation, where the cash price of the item is over £100 and no more than £30,000. Section 75A extends a narrower right above that.

An open banking payment is a *push* with neither. There is no chargeback, because there is no scheme sitting between the parties writing dispute rules. There is no section 75, because there is no credit agreement. A refund is a brand new payment, in the opposite direction, that the merchant elects to send.

The statutory refund right that does exist, regulation 76 of the Payment Services Regulations 2017, requires a payment service provider to refund an *unauthorised* transaction and restore the account, by the end of the business day following the day it becomes aware. But an open banking payment that the customer approved with strong customer authentication is, by definition and by design, authorised. The regulation is not available for a sofa that never arrived.

The authorised push payment reimbursement regime, live since 7 October 2024, is also not the answer. It covers being deceived into sending money to a criminal, up to £85,000 per claim, with the cost shared equally between the sending and receiving firms. It does not cover a merchant dispute. A supplier who took your money and then went quiet is a civil matter, not a scam claim.

So the honest framing is that open banking payments are a *bank transfer with a good user interface*, and they carry exactly the protections of a bank transfer, which is to say almost none beyond the fraud regime. That is fine, and often better than fine, for paying your own energy bill or moving your own money. It is a materially worse deal for the consumer than a credit card when buying a mattress from a company they have never heard of. Any business presenting the two side by side and describing them as equivalent is misrepresenting the product.

The technical version

Two legal instruments, one ecosystem

UK open banking sits on two foundations that are frequently conflated.

The Payment Services Regulations 2017 (SI 2017/752) transposed the second Payment Services Directive, Directive (EU) 2015/2366, and came into force on 13 January 2018. The relevant provisions:

- Regulation 68, confirmation of availability of funds, for card-based payment instrument issuers.
- Regulation 69, access to payment accounts for payment initiation services. The ASPSP must communicate securely with the PISP and treat its payment orders identically to orders received directly from the payer. The PISP must not hold the payer's funds at any time, must not store sensitive payment data, and must not alter the amount, the payee or any other feature of the transaction.

- Regulation 70, access to payment accounts for account information services. The AISP must have the user's explicit consent, must identify itself in every communication session, and must access only the designated accounts and their associated transactions.
- Regulation 71, limits on the use of payment instruments and access to payment accounts. An ASPSP may deny a TPP access for reasonably justified and duly evidenced reasons relating to unauthorised or fraudulent access, but must notify the customer, report the incident to the FCA, and restore access once the justification lapses.
- Regulation 100, authentication, which imposes strong customer authentication.

The Retail Banking Market Investigation Order 2017, made by the CMA under the Enterprise Act 2002, produced the standard. Article 10.5 of Part 2 provided for an Implementation Trustee to drive delivery against an agreed roadmap. Open Banking Limited, originally the Open Banking Implementation Entity, was established by the CMA9 in October 2016 to do that work. The CMA declared the roadmap substantially complete on 12 January 2023, with six providers finished, and confirmed full completion in September 2024 once Danske Bank, Bank of Ireland and Allied Irish Bank had delivered their remaining obligations, including sweeping VRP.

The forward-looking picture is different again. The Joint Regulatory Oversight Committee has been wound up and replaced by a department inside the FCA, which is now lead regulator. HM Treasury is expected to legislate under the **Data (Use and Access) Act 2025**, whose Part 1 provides for smart data schemes, to give the FCA rulemaking powers over open banking. The FCA set out its expectations for a successor to Open Banking Limited, called the Future Entity, in Feedback Statement **FS25/4** (August 2025), and wrote to trade associations on 19 December 2025, published 16 January 2026, inviting industry to coalesce behind a candidate body and offering an independent consultant-led assessment. The FCA expects the Future Entity and commercial scheme operators to be regulated as **interface bodies** under the DUAA, and intends to consult on the long-term regulatory framework by the end of 2026.

The role taxonomy

Term	Meaning
ASPSP	Account servicing payment service provider. The bank or building society holding the account.
AISP	Account information service provider. Reads account data.
PISP	Payment initiation service provider. Constructs and submits payment orders for the customer to authorise.
CBPII	Card-based payment instrument issuer. May ask a yes or no question about whether funds are available.
TPP	Third party provider. Umbrella term for AISPs, PISPs and CBPIIs.
PSU	Payment service user. The customer.

A firm providing account information services only may be a registered account information service provider, a lighter regulatory status. Payment initiation requires full authorisation as a payment institution, or the firm must already be a credit institution or e-money institution, and it must hold professional indemnity insurance or a comparable guarantee. In practice a large proportion of merchants do not hold either permission and reach the rails through a regulated PISP acting as their provider, or as an agent.

Identity in the ecosystem is managed through the **Open Banking Directory**. Participants enrol, obtain certificates, and publish software statements. Third parties obtain a Software Statement Assertion signed by Open Banking, which can be used for dynamic client registration at each ASPSP's authorisation server, avoiding a manual onboarding per bank.

Certificates deserve a paragraph because Brexit changed them. Article 34 of the SCA-RTS originally required eIDAS qualified certificates: a QWAC for transport, a QSeal for signing. UK TPPs' eIDAS certificates were revoked at the end of the transition period. The UK version of the RTS was amended to permit alternatives, and the Open Banking Certificate Authority issues ETSI-profile equivalents: **OBWAC** for website authentication and **OBSeal** for electronic seals. EEA TPPs accessing UK ASPSPs may still present QWACs and QSeals, which can be uploaded to the Directory.

As at June 2026 the Directory carried 327 regulated providers: 88 account providers and 239 third party providers, of which 143 regulated entities had at least one proposition live with customers.

The security profile

The current specification is the Open Banking Standard, currently v4.0.1. Version 4.0 was published on 28 June 2024 and was the first major release since 2018, developed with over forty ecosystem participants.

Its headline change was the security uplift from FAPI 1.0 Implementer's Draft 2 to **FAPI 1.0 Advanced Final**, made mandatory, with the earlier profile deprecated at the end of 2024. Version 4.0 also aligned message content with ISO 20022, including the fields required for Bank of England CHAPS compliance from 1 May 2025, centralised the codeset repository, and improved payment status reporting and error messaging.

The concrete requirements a practitioner will implement:

Transport. TLS 1.2 with mutual authentication. The client presents its transport certificate; the ASPSP validates it against the Directory.

Client authentication. `private_key_jwt`. The TPP signs a JWT assertion with its private key; the ASPSP validates against the registered public key. Shared secrets are not used.

Authorisation. OAuth 2.0 and OpenID Connect. The specification supports the authorisation code grant and identifies the **hybrid grant** as the preferred redirect mechanism under the FAPI profile. Authorisation parameters are packaged in a signed JWT **request object**, which prevents tampering during the redirect.

Intent model. Rather than relying on coarse OAuth scopes, the standard uses dedicated consent resources: `account-access-consent`, `domestic-payment-consent`, `domestic-vrp-consent` and their siblings. An access token binds to exactly one PSU and one intent.

Message signing. Detached JWS per RFC 7515 Appendix F, algorithm PS256, carried in the `x-jws-signature` header. The JOSE header carries critical claims `http://openbanking.org.uk/iat`, `http://openbanking.org.uk/iss` and `http://openbanking.org.uk/tan`, the last identifying the trust anchor domain. This gives non-repudiation independently of the TLS session. Message encryption via JWE is optional.

Correlation headers. `x-fapi-interaction-id`, an RFC 4122 UUID that the ASPSP must echo in its response, is the single most useful field in the entire specification when you are on the phone to a bank's support desk about a failed payment. `x-fapi-auth-date` carries the time of the PSU's last

login with the TPP. `x-fapi-customer-ip-address` carries the PSU's IP when the PSU is present, and its absence signals an unattended call.

Idempotency. POST endpoints that create resources take `x-idempotency-key`, maximum 40 characters. The ASPSP must treat an identical key from the same TPP within 24 hours as a duplicate. The TPP must not modify the request body while reusing a key, and must leave at least one second between a submission and its retry.

Authentication models. The redirect model sends the PSU to the ASPSP's authorisation endpoint, in practice app-to-app on mobile, and back again. The decoupled model uses CIBA, Client Initiated Backchannel Authentication: the TPP initiates, the PSU authenticates on a separate device or channel, and the TPP polls. Decoupled matters for call centres, in-branch journeys and any flow where the PSU is not holding the device that displays the merchant.

A payment initiation, step by step

1. The PISP POSTs to `/domestic-payment-consents` with the payee, amount, currency, reference and, optionally, the debtor account. The ASPSP returns a `ConsentId`. Nothing has been authorised and no money has moved.
2. The PISP constructs a signed request object carrying the `ConsentId` as an intent claim and redirects the PSU to the ASPSP's authorisation endpoint.
3. The ASPSP authenticates the PSU with strong customer authentication. If the debtor account was not specified, the PSU selects it here. The ASPSP plays the consent back to the PSU: amount, payee, reference. The PSU accepts or rejects.
4. The consent resource transitions to authorised. The ASPSP returns an authorisation code to the PISP's redirect URI. The PISP exchanges it at the token endpoint for an access token scoped to that consent.
5. The PISP POSTs to `/domestic-payments`, referencing the `ConsentId`, with a request body that must match the consented payment exactly.
6. The ASPSP submits the payment to a clearing system. For a domestic single immediate payment in sterling this is, in practice, almost always Faster Payments.
7. The PISP polls `GET /domestic-payments/{DomesticPaymentId}` for status.

Version 4.0 aligned payment order statuses to the ISO 20022 external code set. The values a practitioner will encounter include `RCVD` received, `PDNG` pending, `ACTC` accepted technical validation, `ACCP` accepted customer profile, `ACFC` accepted fraud checking, `ACSP` accepted settlement in process, `ACWC` accepted with change, `ACSC` accepted settlement completed, `ACWP` accepted with partial execution, `ACCC` accepted and closed, `BLCK` blocked, `RJCT` rejected, and `CANC` cancelled for future-dated payments and standing orders.

The specification also supports **multi-authorisation** for accounts requiring more than one signatory, through the `MultiAuthorisation` block carrying `Status`, `NumberRequired`, `NumberReceived`, `LastUpdateDateTime` and `ExpirationDateTime`, with the PISP able to request an authorisation type of Single or Any and an optional `CompletionDateTime`. Corporate journeys live or die on this.

There is a `RefundAccount` structure in the domestic and international payment models, which returns the payer's account details to the PISP so that a merchant can send a refund. Note carefully what this is: it is an address, so that the merchant can make a new payment. It is not a reversal mechanism.

Strong customer authentication in the consent flow

Regulation 100 of the PSRs 2017 requires a payment service provider to apply strong customer authentication where a user accesses a payment account online, whether directly or through an AISP; initiates an electronic payment transaction; or carries out any action through a remote channel which may imply a risk of payment fraud or other abuses.

Strong customer authentication means two or more elements from independent categories: knowledge, something only the user knows; possession, something only the user has; and inherence, something the user is. Independence means the breach of one does not compromise the others.

Regulation 100(2) adds **dynamic linking**: where a payer initiates an electronic remote payment transaction, directly or through a PISP, the authentication must include elements which dynamically link the transaction to a specific amount and a specific payee. This is why your bank's app shows you the amount and the payee name on the approval screen and why a PISP cannot alter either after authorisation. It is also why the payment request body must match the consent.

The exemptions live in the SCA-RTS, originally Commission Delegated Regulation (EU) 2018/389, onshored and now maintained by the FCA as technical standards in the Handbook. The ones that matter here:

Article 10 and Article 10A, account information. Article 10 permitted an ASPSP not to apply SCA where a user accesses balance or transaction data, subject to a ninety-day limit. Article 10A, introduced by PS21/19, removed the periodic re-authentication at the ASPSP for AIS access through a TPP and shifted the obligation onto the AISP to obtain the customer's explicit reconfirmation of consent at least every ninety days. The FCA strongly encouraged adoption by 30 September 2022.

Article 13, trusted beneficiaries. An ASPSP need not apply SCA where the payee is on a list of trusted beneficiaries previously created by the payer. This is the mechanism that makes VRP work at all.

Article 11, contactless, and Article 16, low value remote, are card-side and rarely relevant to open banking, though the FCA has publicly encouraged continued support for the contactless exemption for charitable donations.

The FCA has also been explicit that firms must develop SCA solutions that work for all groups of consumers, including those without mobile telephones. A journey that assumes app-to-app on a smartphone is not, on its own, an adequate SCA implementation for a whole customer base.

Variable recurring payments

The VRP API Profile, at v3.1.10 of the read-write specification, defines two resources:

- `domestic-vrp-consents`, a consent between a PSU and a TPP permitting the TPP to create payments subject to **control parameters**.
- `domestic-vrps`, individual payment orders that must fall within those parameters.

The control parameters set the walls. They include the validity window, `ValidFromDateTime` and `ValidToDateTime`; a maximum individual amount; and periodic limits, each with an amount, a period type, and a period alignment. The period type enumerations are Day, Week, Fortnight, Month, Half-year and Year, and an ASPSP must support all of them. Period alignment is either Consent, meaning the window starts when the consent was created, or Calendar. ASPSPs may impose additional control parameters, limits and restrictions on non-sweeping VRPs.

The flow is: the TPP POSTs a `domestic-vrp-consents` resource, the PSU authorises it once with SCA, and thereafter the TPP POSTs `domestic-vrps` against it without further SCA, provided each payment sits inside the parameters. `GET` on `vrp-details` returns funds availability and remaining allowance.

Sweeping is a regulatory category, not a technical one. The CMA approved VRP as the mechanism for implementing sweeping under item A10 of the revised roadmap on 27 July 2021, extended the CMA9 implementation deadline to July 2022 by letter of 15 November 2021, and published a clarification of the definition of sweeping in March 2022. On 3 December 2025 the CMA published a letter responding to an OBL recommendation regarding the sweeping use case of alternative forms of credit that closely compete with overdrafts.

The requirements the CMA9 had to meet for sweeping, as set out in OBL's standards proposition, are worth listing because they explain the consent screens you actually see:

- The sweeping consent must specify the payee account name, the payee account identification, the maximum cumulative amount per time window, and the maximum amount of each individual sweeping payment.
- The ASPSP must tell the PSU that the payee will be added to their trusted beneficiary list, and must add it once consent setup completes.
- The ASPSP must then apply the trusted beneficiary SCA exemption when the PISP initiates a payment inside the consent parameters, or the transfer-to-self exemption where the destination is at the same ASPSP. It may decline to apply the exemption in exceptional circumstances, taking an objective approach consistent with the proportionality requirements of the PSRs.
- The ASPSP must provide a specific error indicator if the payee has been removed from the trusted beneficiary list and the exemption can no longer be applied.
- The PISP must attest to the ASPSP that the consent is a sweeping consent.
- The ASPSP must provide non-discriminatory access both to sweeping consent setup and to the initiation of sweeping payments.

That last requirement is the reason sweeping is free at the point of use and commercial VRP is not. Non-discriminatory access, in the CMA's sense, meant the CMA9 could not charge for it.

Commercial VRP and the UK Payments Initiative

Extending VRP beyond me-to-me sweeping required something the CMA Order did not provide: a commercial model. A bank has no obligation to offer VRP to a merchant and no reason to build the capacity for free.

Under the JROC, the Payment Systems Regulator led work on a commercial model for so-called phase 1 use cases. The FCA took over as lead regulator and continued the work jointly with the PSR, publishing a summary report on development and rollout in December 2025.

In late 2025, 31 firms came together to establish **UK Payments Initiative Ltd** (**UKPI**), an independent industry-owned company to own and operate a commercial VRP scheme with its own rulebook, commercial model and operational standards. The FCA described the first live payments under the scheme as the start of a new era for payments and open banking in the UK, and expected them in the first quarter of 2026.

The scheme was announced as launched on **2 June 2026**, at Money20/20 Europe in Amsterdam. UKPI's own announcement names 24 founding shareholders, including Barclays, HSBC, Lloyds Banking Group, NatWest, Nationwide and Santander, alongside Monzo, Revolut, Starling, GoCardless, Plaid, TrueLayer, Yapily and Token.io. The scheme's first wave covers government payments, utilities,

charities and financial services, with subscriptions and wider e-commerce planned to follow. A launch coverage target of around 75 per cent of UK current accounts is quoted consistently in the guidance scheme participants have issued to merchants, and it is the figure the industry has settled on. It is worth being clear about its standing: it does not appear in the FCA's statement on the launch, nor in the regulators' correspondence with the CMA, and no public scheme document confirms it, so the number rests on industry restatement rather than on a published primary source.

The commercial model itself required a competition-law answer, because a group of competitors agreeing a central price is exactly the arrangement competition law exists to police. On 15 January 2026 the FCA and the PSR wrote to the CMA setting out their position; on 16 January the CMA confirmed it did not intend to take a different view on prioritisation; and on 20 January 2026 the regulators published a joint statement confirming they would not at this stage prioritise a Competition Act 1998 investigation into UKPI's centralised access fee model. That forbearance runs until the government's anticipated legislative framework or July 2027, whichever comes first. The FCA has said it will assess industry-led cVRP growth by the end of 2026 and fold the lessons of the first phase into the long-term framework.

VRPs accounted for roughly 15 to 16 per cent of open banking transaction volume at the time of writing, with sweeping VRP volumes having nearly doubled year on year in 2025.

Why it settles over Faster Payments, and what follows from that

Nothing in the specification requires Faster Payments. The standard describes a payment order; the ASPSP decides how to execute it. `LocalInstrument` can carry a scheme preference and some ASPSPs support CHAPS or scheduled Bacs credits for particular payment types.

In practice, the domestic single immediate payment, which is the overwhelming majority of open banking payment volume, executes over the Faster Payments Service, for the obvious reason: it is the only sterling retail rail that clears in seconds, around the clock, seven days a week. Bacs takes three working days on a submission, processing and entry cycle, and is deferred net settlement, which is fine for payroll and useless for a checkout. CHAPS is same-day, irrevocable real-time gross settlement inside the Bank of England's RTGS system, carrying the large majority of sterling value, with an early-afternoon cut-off for the same-day guarantee, and it is not priced for a £340 energy bill.

Four consequences follow, and they are the practical heart of this chapter.

Irrevocability. Pay.UK's own description is unambiguous: because of their real-time nature, once sent, Faster Payments cannot be cancelled. There is a Credit Payment Recovery process for misdirected payments, which is a best-efforts request to the receiving institution to return funds it has not yet released, not a right. Confirmation of Payee, launched in 2020, checks the payee name against the account before you send, which prevents some errors and some frauds but changes nothing about finality once the payment has gone.

Limits. The scheme maximum is £1 million per transaction. Individual PSPs set their own lower ceilings, and channel-specific ceilings for open banking are frequently lower still.

No chargeback. There is no scheme adjudication layer, because the scheme is Faster Payments and Faster Payments does not adjudicate merchant disputes. Compare the card world, where chargeback rights are written by the card networks into their operating rules and where section 75 of the Consumer Credit Act 1974 imposes joint and several liability on a credit provider for purchases with a cash price over £100 and up to £30,000.

Refunds are new payments. A merchant refunding an open banking payment initiates a fresh credit transfer to the payer. That is why the specification carries a `RefundAccount` structure: not to reverse anything, but to tell the merchant where to send the money. Operationally this means refunds need their own payout capability, their own reconciliation, their own fraud controls and their own audit trail, none of which a merchant migrating off cards will have.

The protection map

	Card (credit)	Card (debit)	Direct Debit	Open banking payment
Direction	Pull	Pull	Pull	Push
Rail	Card scheme	Card scheme	Bacs	Faster Payments
Merchant has funds	After settlement, days	After settlement, days	Three working day cycle	Seconds
Reversal by customer	Chargeback, scheme rules	Chargeback, scheme rules	Direct Debit Guarantee, immediate refund on request	None
Statutory purchase protection	Section 75 CCA 1974, £100 to £30,000	None	None	None
Fraud reimbursement	Unauthorised transaction rules, PSRs reg 76	Unauthorised transaction rules, PSRs reg 76	Guarantee, plus reg 76	APP regime since 7 Oct 2024, up to £85,000; reg 76 for unauthorised
Cost to merchant	Percentage	Percentage	Low, per item	Flat fee, per item

The statutory refund provisions in the PSRs are worth stating precisely, because they are routinely misdescribed:

- **Regulation 74** requires notification without undue delay and in any event within thirteen months of the debit date, otherwise the redress provisions are unavailable.
- **Regulation 76** requires the PSP to refund an unauthorised transaction and restore the account, as soon as practicable and no later than the end of the business day following the day it becomes aware. Where a PISP is liable, regulation 75(2) applies and the PISP must compensate the ASPSP immediately on request.
- **Regulation 77** caps the payer's own liability at £35 for losses from a lost, stolen or misappropriated payment instrument, with the cap disapplied where the payer acted fraudulently or with intent or gross negligence, and with no liability at all where the ASPSP failed to apply SCA that regulation 100 required.
- **Regulation 79** gives a refund right for authorised transactions *initiated by or through the payee*, where the authorisation did not specify the exact amount and the amount exceeded what the payer could reasonably have expected. Regulation 80 requires the request within eight weeks of the debit and a response within ten business days.
- **Regulation 95** gives a right of recourse between PSPs where liability is attributable to another provider or intermediary, including for failure to apply SCA.

Regulation 79 is the interesting one for VRP, and the position is genuinely unsettled. Its trigger is a transaction initiated by or through the payee, which is the direct debit shape. A VRP is initiated by a PISP acting for the payer, within limits the payer set, which is a different shape. Whether and how

the regulation 79 refund right attaches to a commercial VRP is one of the questions the FCA's long-term framework will need to answer, and it is the single largest open consumer-protection question in the area. The direct debit's competitive advantage over VRP is not technical. It is the Direct Debit Guarantee, which promises an immediate refund from your own bank, on request, with no time limit and no argument. VRP as currently specified has no equivalent, and the UKPI rulebook's treatment of disputes will determine whether it needs one.

What changes next

Three things are moving, and a chapter written eighteen months from now will read differently.

In the UK, the Data (Use and Access) Act 2025 provides the vehicle. HM Treasury is expected to legislate to give the FCA rulemaking powers over open banking; the FCA intends to consult on the long-term regulatory framework by the end of 2026; and industry is expected to stand up a Future Entity to succeed Open Banking Limited as the primary API standard-setting body, regulated as an interface body under the DUAA. Beyond payments, the same Act provides for smart data schemes in energy, telecoms, transport and retail, which is what "open finance" and eventually "open data" mean in a British context.

In the EU, the successor to PSD2 has landed. The Parliament and Council reached provisional political agreement on PSD3 and the accompanying Payment Services Regulation on 27 November 2025; the Council published final compromise texts on 23 April 2026 following COREPER endorsement on 22 April; the ECON committee approved the agreed text on 5 May 2026. Formal adoption by both institutions was still outstanding at the time of writing. A separate Regulation on a Framework for Financial Data Access, which is the EU's open finance instrument, remained in trilogue. Among the changes relevant here, the package clarifies that SCA for account information access should be performed by the account provider once, at the outset.

In the rails beneath, the EU's Instant Payments Regulation has made instant euro transfers universal and price-capped, which removes the main structural reason European open banking payments lagged British ones. Britain's advantage in this area was always the early availability of a genuinely instant, genuinely ubiquitous retail rail. That advantage is now shared.

The through-line is worth restating, because it is the thing most easily lost. Open banking did not build a payment system. It built a legally mandated, cryptographically enforced way of asking a bank to use the payment system it already had, and a licensed register of who may ask. Everything good about it, the speed, the price, the finality, comes from Faster Payments. Everything difficult about it, the absence of chargeback, the absence of section 75, the fact that a refund is a fresh payment somebody has to choose to make, comes from Faster Payments too. It is the same fact wearing two faces, and any product built on this layer has to be designed for both of them at once.

44.98 Common wrong ideas

Wrong: open banking is a payment system and money moves through it. **Right:** it is a legally mandated, cryptographically enforced way of asking a bank to use the payment system it already had, and the money travels down Faster Payments.

Wrong: an open banking payment is a cheaper card payment. **Right:** it is a push with no chargeback and no section 75, and a refund is a brand new payment in the opposite direction that the merchant elects to send.

Wrong: regulation 76 will get the customer's money back when the goods never arrive. **Right:** regulation 76 covers unauthorised transactions, and a payment the customer approved with strong customer authentication is authorised by definition and by design.

Wrong: the authorised push payment reimbursement regime covers a merchant dispute. **Right:** it covers being deceived into sending money to a criminal, up to £85,000 per claim; a supplier who took the money and went quiet is a civil matter.

Wrong: every bank must keep the hatch open to the same standard. **Right:** the Payment Services Regulations require access without specifying a data format, an uptime target or a response time, and only the CMA9 were required to implement the common standard and sweeping VRP.

Wrong: deleting the app or cancelling in it ends the connection. **Right:** there are at least two permissions, held by two different parties with different lifetimes, and they come apart routinely.

Wrong: the ninety-day rule drags a customer through their bank's login four times a year. **Right:** since 26 March 2022, Article 10A has made it a reconfirmation of consent that the account information provider obtains, not a re-authentication the bank performs.

Wrong: the Faster Payments scheme maximum of £1 million is the ceiling to build against. **Right:** individual banks set lower limits, and the limit applied to open-banking-initiated payments is frequently lower than the same bank applies in its own app, so size against the worst bank on the coverage matrix and find the number by testing.

Wrong: an accepted payment status means the money has settled. **Right:** a PISP receives a status about the payment order from the bank, not a settlement confirmation from Pay.UK, statuses lag, and some banks stop at an intermediate status and never emit the terminal one.

Wrong: the `RefundAccount` structure is a reversal mechanism. **Right:** it is an address, returned so that the merchant can make a fresh payment, with its own payout capability, reconciliation, fraud controls and audit trail.

44.99 Chapter summary in 20 lines

1. Open banking is not a payment system: it is a way of asking a bank to do something it could already do, in a form the bank is legally obliged to answer.
2. Both the £340 energy bill and the £85 overnight sweep travelled down the Faster Payments Service, which open banking did not build and does not own.
3. It replaced shared passwords and screen-scraping with two regulated interfaces: one that reads account data and one that submits a payment order for the customer to authorise.
4. The payment initiation provider fills in the form but never signs it; only the account holder authorises, with strong customer authentication, in their own bank's app.
5. Two legal instruments underpin the ecosystem: the Payment Services Regulations 2017, which require access from every account provider, and the CMA Order 2017, which produced the specification and bound only the CMA9.
6. Everybody outside the CMA9 built what they liked, so coverage is a matrix of institutions rather than a national fact.
7. Card fees are percentages and open banking fees are flat in pence, which is why the saving grows with the size of the payment rather than staying constant.
8. The money arrives in seconds at twenty past nine on a Wednesday evening, because Faster Payments runs every hour of every day including Christmas morning.

9. It also arrives for keeps: a push payment has no chargeback, no section 75 protection and no reversal mechanism.
10. Regulation 76 refunds unauthorised transactions, and a payment approved with strong customer authentication is authorised, so it is unavailable for a sofa that never arrived.
11. The authorised push payment regime covers deception by a criminal rather than a dispute with a merchant.
12. So an open banking payment is a bank transfer with a good user interface: better than a card for paying your own energy bill, materially worse for buying a mattress from a company you have never heard of.
13. Consent is at least two permissions, one contractual with the third party and one an authorisation event at the bank, and they drift apart in ordinary use.
14. Since March 2022 the ninety-day obligation has been a reconfirmation of consent obtained by the account information provider, which moved the friction rather than removing it.
15. Status is not settlement: the provider sees the bank's view of the payment order, statuses lag, and a checkout that waits for a terminal status is a checkout that hangs.
16. A variable recurring payment is one authorisation plus control parameters, a maximum per payment, periodic limits, a named payee and a validity window, enforced by the bank rather than by the third party.
17. Sweeping, moving your own money between your own accounts, was mandated on the CMA9 with non-discriminatory access, which is precisely why it is free at the point of use.
18. Commercial VRP needed a commercial model the CMA Order never provided, which arrived with the UK Payments Initiative scheme launched on 2 June 2026 and a regulatory forbearance on its centralised access fee running to July 2027.
19. Whether regulation 79's refund right attaches to a commercial VRP is genuinely unsettled, and the Direct Debit Guarantee, an immediate refund on request with no time limit, is the standard against which VRP will be measured.
20. Everything good about open banking payments and everything difficult about them come from the same fact wearing two faces: they run on Faster Payments.

Sources: Open Banking Limited, on the Open Banking Standard v4.0 and v4.0.1, the Read-Write API Profile and VRP API Profile v3.1.10, the Payment Initiation API Profile v4.0, the standards proposition on Variable Recurring Payments, Directory and certificate guidance on OBWAC and OBSeal, the API performance statistics for June 2026, the ecosystem highlights published on openbanking.org.uk, and the year-in-review analysis for 2025; the Competition and Markets Authority, Retail Banking Market Investigation Order 2017, the roadmap completion confirmations of January 2023 and September 2024, the letters on VRP for sweeping of 27 July 2021, 15 November 2021, March 2022 and 3 December 2025; the Payment Services Regulations 2017 (SI 2017/752), regulations 68 to 71, 74 to 80, 95 and 100, as published on legislation.gov.uk; the Financial Conduct Authority, PS21/19 on changes to the SCA-RTS, the strong customer authentication pages, FS25/4 on the design of the Future Entity, the letter to trade associations of 19 December 2025, the open banking progress statement of 2025, the joint statement with the PSR of 20 January 2026 on open banking pricing models, and the statement on the launch of the UK Payments Initiative scheme; the Payment Systems Regulator, PS24/7 on the maximum level of reimbursement for Faster Payments APP scams; Pay.UK, on how Faster Payments work and on Confirmation of Payee; UK Payments Initiative, launch announcement of 2 June 2026; Directive (EU) 2015/2366 and Commission Delegated Regulation (EU) 2018/389; the Data (Use and Access) Act 2025; the Council of the European Union, final compromise texts for PSD3 and the Payment Services Regulation of 23 April 2026; the Consumer Credit Act 1974, sections 75 and 75A, and the Financial Ombudsman Service guidance on section 75 and chargeback.

VOLUME V

Trust, Failure and the Law

Fraud, disputes, reconciliation and regulation. The perishable volume, dated deliberately and revised on its own.

45

CHAPTER

Where Cards Are Stolen

45.0 What this chapter gives you

1. You will be able to name the five places card details are actually taken — the stripe, the shim, the radio, the breach and the guess — and say which of them still costs anybody real money.
2. You will be able to explain why watching a chip talk to a terminal gets you a used ticket, while reading a magnetic stripe gets you a key that works for ever.
3. You will be able to say why the chip did not make card fraud go away, and point at £423.5 million of remote purchase losses in the United Kingdom in 2025 as the evidence of where it went instead.
4. You will be able to answer somebody worried about contactless in a queue with the actual proportion: £46.8 million of £703.4 million of unauthorised losses, under seven per cent.
5. You will be able to explain why a shim yields far less than a skimmer ever did, and name the issuer-side checks — iCVV, service code, transaction counter, fallback constraint — that decide whether it yields anything at all.
6. You will be able to distinguish a server-side breach from client-side e-skimming, and say why a merchant's own logs stay clean throughout the second one.
7. You will be able to cite PCI DSS requirements 6.4.3 and 11.6.1, mandatory since 31 March 2025, as the two controls written specifically for payment-page script theft.
8. You will be able to do the enumeration arithmetic — ten million accounts under an eight-digit issuer identification number, sixty expiry dates, a thousand three-digit codes — and explain why centralised counting rather than merchant rate limiting is the fix.
9. You will be able to read the Visa Acquirer Monitoring Program thresholds and say why the enumeration ratio deliberately counts declined authorisations on both sides of the fraction.
10. You will be able to state a United Kingdom cardholder's legal position under regulations 76 and 77 of the Payment Services Regulations 2017, including the £35 cap and the circumstances that disapply it.

Everything in the four volumes before this one described a system working. This volume describes the same system under attack, and the first thing to establish is that "card fraud" is not one crime. It is at least five, they have almost nothing in common technically, and the defences that stopped the first one did nothing at all about the fifth.

This chapter is the threat model. It is written to be useful to somebody who has to defend a payment system, not to somebody who wants to attack one, and that distinction shapes what is on the page: how each attack works and why each control exists, without the operational detail that would make any of it a recipe. Where a control is mandated by a rulebook or a regulator, the figure and the date are given, and both are marked as accurate at the time of writing, because in this volume more than any other they will move.

The plain version

Think of your card as a key to a door.

For the first forty years of the card industry, it was a very simple key: a shape stamped into a piece of metal. Anybody who could get a good look at the key could file an identical one. The magnetic stripe on the back of your card is exactly that. It is a strip of iron oxide with your card number, expiry date and a few other details written into it, in the clear, and it never changes. A machine that reads the stripe learns everything it needs to make a stripe that behaves identically. That is why, for decades, the standard card crime was to read stripes and write copies.

Then the industry replaced the key with a locksmith.

The chip in your card is a very small computer with a secret inside it that it will never tell anyone, including you. When you put the card into a machine, the machine does not ask "what is your key". It asks a question, and the chip writes a short note in reply, sealed using the secret. Crucially, the question is different every time, because the machine throws in a fresh random number, and the chip adds a counter that ticks up on every single transaction it has ever done. So the reply is different every time too.

Copy that reply and you have a note that answers one question that will never be asked again. Watching the conversation between a chip and a machine gets you a used ticket. Watching a magnetic stripe gets you a key.

That single change is why the crime moved.

Where it moved to

Here are the five places card details are actually taken, in the order they became important.

The first is the stripe. Put a reader over the slot on a cash machine, or run a card through a second reader behind the counter, and you get a perfect copy. This is skimming, and where chips are used properly it is close to dead. The industry's own numbers show why: EMVCo, the body that owns the chip specification, reports 15.6 billion cards in circulation using its specifications, and that 97 per cent of card-present transactions worldwide are chip transactions, most recently reported as of the fourth quarter of 2025. There is very little stripe left to skim.

The second is a thin device slipped into the slot of a chip reader, sitting between the card's metal contacts and the machine's. It is called a shim, and it listens to the conversation. It gets the card number, the expiry date, and a great deal else. What it does not get is the secret. The reply the chip wrote is a used ticket, and so a shim is worth far less than a skimmer ever was.

The third is the radio. A contactless card answers anyone who asks, from a few centimetres away, without a PIN. So the obvious idea is to put one machine near a real card and another machine near a real terminal and pipe the conversation between them, so that a card in a pocket in one place pays for something in another. That is a relay attack. It works in laboratories. It is limited in the world by two things: the radio only reaches centimetres, so somebody has to physically get close to the card, and the terminal times the conversation and gets suspicious if the answers come back late.

The fourth is the one that actually matters now, and it is not clever at all. Break into the shop's computer, or the payment company's computer, and take the list. In November and December 2013, attackers who had obtained a supplier's credentials got inside the American retailer Target and took details for up to 40 million payment cards, along with contact details for up to 70 million more people. No skimmer, no shim, no radio. One break-in, forty million cards.

The modern version of the same idea does not even need to get into the shop. It gets into the shop's website. A few lines of code slipped into a checkout page will quietly copy what you type into the card fields and send it somewhere else, before it ever reaches the shop. This happened to British Airways in 2018. The Information Commissioner's Office found the attacker was in a position to access the personal data of approximately 429,612 customers and staff, including the payment card details of around 244,000 customers, and issued a penalty notice of £20 million on 16 October 2020.

The fifth is guessing. A card number is sixteen digits, but the first eight identify the bank, and the last is a check digit calculated from the others. So under one bank's range there are about ten million possible accounts. An expiry date within the next five years is one of sixty possibilities. The three digits on the back are one of a thousand. Nobody can guess a whole card. But if you already have the number from a breach, you only need the expiry and the three digits, which is sixty thousand combinations, and a machine can try them at a website far faster than a person can. This is called card testing, and the small charity donation page with no shipping address and a five-pound minimum is its favourite target.

What the totals look like

The Nilson Report, which counts this for the industry, published in January 2026 that worldwide payment card fraud losses were $33.41 billion in 2024, against global card volume of $51.920 trillion. That is 6.43 cents of fraud per $100 spent, down from 6.58 cents the year before.

Six and a half pence in a hundred pounds. Card fraud is enormous in absolute terms and small in proportional terms, and both facts are true at once, which is why arguments about it are so bad-tempered.

In the United Kingdom, UK Finance reported on 15 June 2026 that criminals stole £1.28 billion through payment fraud in 2025. Of that, £703.4 million was unauthorised fraud, meaning transactions the customer never made at all. The breakdown is the point:

Type of unauthorised fraud, UK, 2025	Losses	Change on 2024
Remote purchase (card details used online)	£423.5m	up 3%
Lost and stolen cards	£109.8m	down 2%
Remote banking	£104.4m	down 27%
Contactless	£46.8m	up 8%
Card ID theft	£54m	down

Remote purchase fraud, on its own, is more than sixty per cent of all unauthorised card losses. That is card details typed into a website. Contactless, the thing people worry about in queues, is £46.8 million out of £703.4 million: under seven per cent.

The lesson of the plain version is this. The chip did not make card fraud go away. It made one kind of card fraud, the counterfeit card, extremely hard, and everything else moved to the internet, where there is no chip to consult.

Where the plain version stops being true

"The chip ended counterfeit fraud" is a claim about a control, and controls only work where they are switched on. The Federal Reserve Bank of Kansas City published a research briefing on 12 February 2025, using the Federal Reserve Board's biennial debit card data, which found that the United States migration to chip cards did not produce a uniform fall in card-present fraud. On dual-message

networks the counterfeit fraud rate was 9.0 basis points in 2015, fell to 6.6 in 2017, and was back to 9.1 by 2021. Over the same period the lost-or-stolen rate on those networks rose from 1.6 to 4.5 basis points, and overall card-present fraud rose from 10.1 basis points in 2017 to 14.9 in 2021. Only single-message networks showed the expected pattern, with counterfeit falling from 3.2 basis points in 2015 to 1.7 in 2021. The chip is sound. Its deployment was uneven, its fallback paths were exploitable, and fraud is a fluid: squeeze it and it goes somewhere else in the same container.

The chip protects the reply, not the number. I said the chip never reveals its secret, and that is true. What the chip does reveal, in the clear, on every transaction, is the card number, the expiry date and a block of track-equivalent data. Reading a chip conversation does not let you counterfeit a chip card. It does let you obtain the details that a website will accept. The plain version implies the chip made your card details secret. It did not. It made them non-replayable at a terminal. On a website, where there is no terminal and no cryptogram, they are as useful as they ever were, which is exactly why remote purchase fraud is the largest category in the table above.

Most stolen card data is now not stolen from cards at all, and a growing share is not stolen at all. The dominant modern source is a breached merchant, processor or checkout page, which touches no physical card and no physical terminal. Alongside it, and rising fast, is a category that involves no theft of card data in the conventional sense: the criminal already has the card number and telephones the customer to obtain the one-time passcode, then uses it to provision that card into a digital wallet on their own device. UK Finance's report of 15 June 2026 names this explicitly as a driver of 2025 losses. The result is a wallet credential that passes every check the network can make, on a device that is not yours. No skimmer, no shim, no breach. An authentication defeat dressed as a card theft.

Relay attacks are famous out of all proportion to their share of losses, and the reason is worth understanding. A relay does not break the cryptography. It borrows it. The issuer receives a genuine cryptogram from a genuine card, correctly computed, correctly verified. Nothing in the message says the card was three hundred miles away. That makes a relay attack unfalsifiable from the issuer's side and turns the dispute into an evidential argument rather than a technical one, which is what makes it interesting to researchers and to lawyers. It does not make it common. The £46.8 million of contactless fraud UK Finance reported for 2025 is dominated by ordinary use of ordinary stolen cards, not by relays.

"Card data" is not one asset. A PAN alone, a PAN with expiry, a PAN with expiry and CVV2, full track data, a PIN block, and a network token are six different things with six different resale values, six different attack paths and six different controls. Treating them as one category is the single most common analytical error in this field, and it is the reason people build defences in the wrong place.

The technical version

What lives where on a card

Three credentials coexist on a typical card, and each fails differently.

The magnetic stripe follows ISO/IEC 7811. Track 1, the IATA track, carries up to 79 alphanumeric characters including the primary account number, cardholder name, expiry date, a three-digit service code and discretionary data. Track 2, the ABA track, carries up to 40 numeric characters: PAN, expiry, service code and discretionary data. The discretionary data contains the card verification value written at personalisation, CVV1 on Visa or CVC1 on Mastercard. The service code matters more than its size suggests: its first digit encodes interchange rules, with values in the 2xx range signalling that the card

carries a chip and that the chip must be used where the terminal is capable, and its third digit encodes cardholder verification requirements.

Nothing on the stripe changes. It is a bearer credential in the purest sense.

The chip follows ISO/IEC 7816 for the contact interface and holds an EMV payment application, a set of records read by the terminal, a card private key or shared symmetric keys it never discloses, and an application transaction counter. Among the records is tag 57, track-2 equivalent data, which looks like Track 2 but carries a different verification value in the discretionary field: the integrated circuit card verification value, iCVV. It is deliberately different from CVV1. That difference is the entire defence against turning chip data into a stripe.

The contactless interface follows ISO/IEC 14443, operating at 13.56 MHz with a baseline data rate of 106 kbit/s in each direction, half duplex, over a designed range of a few centimetres. In full EMV contactless mode it produces a cryptogram exactly as the contact interface does. The older magnetic-stripe emulation modes, which produced a dynamic CVV or CVC3 instead, are the modes that most published contactless attacks targeted, and their retirement is a large part of why those attacks have narrowed.

The card-not-present credential is different again: PAN, expiry, and the CVV2, CVC2, CID or CAV2 printed on the card but written into neither the stripe nor the chip. It exists precisely so that stripe or chip data alone is insufficient online.

Finally, the tokenised credential. Under the EMV Payment Tokenisation Specification, a device token or DPAN is provisioned in place of the funding PAN, restricted by domain to a particular device, channel or merchant, and accompanied by a cryptogram of its own. A stolen DPAN outside its domain is inert.

Magnetic stripe cloning, and why the chip ended it

A stripe clone is a replay attack against a credential with no freshness. The attacker acquires the track data by any means, writes it to a stripe, and presents it. The issuer sees identical data to the genuine card and has no basis on which to distinguish them. CVV1 in the discretionary data defends against a specific and narrower threat: card details harvested from receipts, embossing or a manual imprint, which do not include it. Against a full track read it is no defence at all, because it was read too.

EMV removes replay by three mechanisms working together. The terminal supplies an unpredictable number, tag 9F37, which is fresh per transaction. The card supplies an application transaction counter, tag 9F36, which increments monotonically and never repeats. The card computes an authorisation request cryptogram over transaction data including both, using a key derived from an issuer master key that the card holds and never emits. The issuer recomputes the cryptogram and compares. A captured cryptogram authenticates one transaction, with one nonce and one counter value, and is worthless afterwards.

The migration was driven by liability rather than by technology, and the dates are worth having exactly. In the United Kingdom the liability shift took effect on 1 January 2005, moving responsibility for fraudulent magnetic swipe transactions from issuers to retailers who could not accept chip, and the national rollout completed on 14 February 2006. Counterfeit card fraud losses in the United Kingdom, reported by what was then the UK Cards Association, fell from £169.8 million in 2008 to £80.9 million in 2009. In the United States the equivalent shift for most point-of-sale acceptance took effect on 1 October 2015, roughly a decade later, with automated fuel dispensers deferred beyond that.

The stripe itself is now being withdrawn. Mastercard announced on 12 August 2021 that newly issued Mastercard credit and debit cards would not be required to carry a stripe from 2024 in most markets,

that United States banks would no longer be required to issue chip cards with a stripe from 2027, that no new Mastercard credit or debit card would be issued with a stripe by 2029, and that no Mastercard credit or debit card would have a stripe by 2033, with prepaid cards in the United States and Canada exempt at the time of the announcement. Those dates are the scheme's published plan as of writing in August 2026 and are subject to change by the scheme.

Until then, the residual controls are issuer-side and they are the ones that determine whether a shim or a chip-derived clone is worth anything. They are: validating CVV1 on any stripe-read transaction; validating iCVV separately on chip transactions and refusing to accept one where the other is expected; checking the service code against the card's actual profile so that a stripe claiming no chip is present on a card that has one is treated as suspicious; monitoring the application transaction counter for non-monotonic or duplicated values; and constraining technical fallback, where a chip read fails and the terminal reverts to the stripe, by volume, geography and merchant category.

Shimming, and why it yields far less

A shim is a passive device placed in the card slot between the card's contacts and the reader's, recording the application protocol data units exchanged under ISO/IEC 7816-3. The reason it exists is that the chip's dialogue with the terminal is not encrypted; confidentiality was never its security property. Authenticity was.

What the exchange discloses is the static record content the card returns during read records: the PAN, the expiry date, the cardholder name where present, the track-2 equivalent data of tag 57 including the iCVV, the application transaction counter, and the cryptogram computed for that one transaction. What it does not disclose is any key. The card never transmits its private key or its derived session keys, and the cryptogram is a one-way function of data including a nonce the attacker cannot control.

So a shim produces two products, and both are constrained.

The first is card-not-present abuse. PAN and expiry are enough at any merchant that does not require CVV2, and are a strong starting point at any merchant that does, because they reduce the guessing problem to the remaining fields.

The second is what Recorded Future named EMV-bypass cloning in 2020: writing chip-derived track-2 equivalent data onto a magnetic stripe and presenting it as a stripe transaction. This should fail, because the iCVV in the chip record differs from the CVV1 the issuer expects from a stripe. It succeeds only where the issuer does not validate the value, or validates the wrong one, or does not check the service code. Krebs on Security, reporting Royal Canadian Mounted Police material in January 2017, made the same point when shimmers first appeared publicly: shimmers exist because some issuers did not implement the standard correctly. The vulnerability is not in EMV. It is in a small number of issuer authorisation stacks, and it is closable by configuration.

Defences are correspondingly split between the issuer and the device. On the issuer side, the controls listed at the end of the previous subsection are the whole answer, and an issuer that applies all of them reduces a shim to a source of card-not-present data. On the device side, cash machine and unattended terminal manufacturers have added physical countermeasures to the card path, and terminals are certified against the PCI PIN Transaction Security Point of Interaction requirements, currently at version 7.0 as of writing in August 2026, which cover tamper detection, tamper responsiveness and physical security of the card reader and PIN entry path. What no device countermeasure can do is make the chip dialogue secret, because the terminal must be able to read it.

Relay attacks on contactless, and the timing that limits them

A relay attack is a proxy. One device presents itself to the genuine card as a terminal; a second presents itself to the genuine terminal as a card; the two are linked. Nothing is decrypted, forged or replayed. The cryptogram is genuine, the counter is correct, and the issuer's verification succeeds, because from the protocol's point of view the card really did participate.

Three things constrain it.

The first is physics. ISO/IEC 14443 is a proximity standard. The reader field falls away over centimetres, so one end of the relay must be brought very close to the victim's card. That is a physical-access problem, and it is why relay attacks appear in demonstrations and disputes far more often than in loss statistics.

The second is the protocol's own patience. ISO/IEC 14443-4 defines a frame waiting time, and EMV contactless kernels enforce an overall transaction time budget, commonly cited in the research literature at around 500 milliseconds, because the entire commercial proposition of contactless is that the customer does not wait. A relay adds latency at both radio interfaces and across whatever link joins them, and that latency competes against a budget that was set for user experience rather than for security but happens to serve both.

The third is an explicit countermeasure. EMV specifies a Relay Resistance Protocol, defined in the EMV Contactless Specifications for Payment Systems, Book C-2, the Mastercard kernel, and carried forward into EMVCo's Kernel 8. It works by adding a timed exchange. The terminal issues an EXCHANGE RELAY RESISTANCE DATA command carrying terminal entropy and starts a timer. The card replies with its own entropy and with its Device Estimated Transmission Time For Relay Resistance R-APDU, a two-byte value in units of hundreds of microseconds stating how long it expects its own response to take. The terminal records a Measured Relay Resistance Time in the same units and compares the measurement against the card's declaration, its own Terminal Expected Transmission Time For Relay Resistance R-APDU, and configured Relay Resistance Minimum and Maximum Time Difference Limits. If the measurement falls outside the window, the terminal can decline or force the transaction online.

It is important to be precise about what this achieves, because it is routinely overstated. RRP is a latency bound, not a cryptographic distance-bounding protocol in the academic sense. Its granularity is hundreds of microseconds; one hundred microseconds is thirty kilometres of light travel in free space. The protocol therefore cannot localise a card geometrically at all. What it does is impose a ceiling on the additional processing and network latency an attacker can introduce, which is a real and useful constraint against a relay built over general-purpose networks, and a much weaker one against a purpose-built low-latency link. Published analysis, including the systematisation-of-knowledge literature on EMV contactless security, notes that measured RRP timings vary with card position and orientation, which forces the acceptance window wider than a security engineer would like.

The academic record is worth citing accurately, because it is what drove the fixes. Francis, Hancke, Mayes and Markantonakis demonstrated practical relay of contactless transactions using ordinary NFC-capable mobile telephones in 2010. Radu, Chothia, Newton, Boureanu and Chen published "Practical EMV Relay Protection" at the IEEE Symposium on Security and Privacy in 2022, demonstrating a relay against Apple Pay in Express Transit mode with a Visa credential, exploiting the fact that transit modes deliberately relax cardholder verification so that a locked phone can pay at a gate; Apple and Visa subsequently addressed it. A cluster of work between 2019 and 2021, much of it by Galloway, Yunusov and colleagues, showed cardholder verification and limit bypasses achieved by

manipulating terminal transaction qualifiers and card transaction qualifiers in the contactless dialogue rather than by attacking cryptography at all.

The economic constraint deserves its own paragraph because it is the control most people actually experience. Contactless transactions are bounded by the strong customer authentication regime. In the European Economic Area, Article 11 of the PSD2 regulatory technical standards permits an exemption for contactless payments at point of sale up to €50 per transaction, with cumulative limits of €150 or five consecutive transactions since the last application of strong customer authentication. In the United Kingdom, the corresponding Article 11 of the UK SCA-RTS was set at £100 per transaction and £300 cumulative, following the Financial Conduct Authority's policy statement PS21/2, which raised those figures from £45 and £130.

That position changed recently and the change is the kind of thing this volume exists to track. On 19 December 2025 the Financial Conduct Authority announced, through Handbook Notice 136, rule changes taking effect in March 2026 which allow firms with strong fraud controls to set their own contactless limits rather than being bound by the regulatory figures. The FCA stated it did not expect immediate changes to the £100 single-transaction limit, and that firms remain obliged to reimburse consumers for unauthorised transactions and to communicate limit changes under the Consumer Duty. As of writing in August 2026, the £100 and £300 figures therefore persist as industry practice rather than as a hard regulatory cap in the United Kingdom, and any reader relying on this paragraph after 2026 must re-check it.

For a relay attacker, those limits are the binding constraint. A relayed contactless transaction below the floor limit yields a small amount, exhausts the cumulative counter quickly, and then triggers a PIN or biometric prompt that the relay cannot satisfy, because a device-based wallet requires consumer device cardholder verification on the genuine device.

Breaches at merchants and processors: the dominant modern source

Everything above concerns single cards. Breaches concern lists, and lists are what the economics reward.

There are two families and they need separating.

Server-side compromise means card data taken from inside a merchant's or processor's environment: from a database, from a log file, from application memory, or from network traffic. The canonical era was point-of-sale memory-scraping malware, which exploited the window in which track data existed in the clear in a till's memory between being read and being encrypted. TJX in 2007, Heartland Payment Systems disclosed in January 2009, Target in 2013 and Home Depot in 2014 are the reference cases; Target alone was up to 40 million payment card records taken between 27 November and 15 December 2013, using credentials obtained from a supplier and BlackPOS malware deployed on till systems.

The structural answer to that family is point-to-point encryption. If the reading device encrypts card data inside its own secure hardware, using keys the merchant never holds, then the merchant's environment never contains card data in a usable form and the memory-scraping window closes. This is why PCI Point-to-Point Encryption solutions are built on PCI PTS POI devices with the Secure Reading and Exchange of Data capability, and why a validated P2PE deployment collapses a merchant's PCI DSS scope so dramatically. It is also why tokenisation matters: replacing stored PANs with tokens means a stolen database of tokens is a stolen database of nothing.

Client-side compromise, usually called e-skimming or by the threat-actor label Magecart, is the modern replacement. The attacker does not enter the merchant's environment at all. They compromise a

script that the merchant's checkout page loads into the customer's browser, often a third-party script for analytics, chat or tag management, and that script copies the card fields as they are typed and sends them to an attacker-controlled endpoint. The merchant's servers never see the theft. The merchant's own logs are clean. The card data is taken between the customer's keyboard and the merchant's form handler.

British Airways in 2018 is the reference case in the United Kingdom, and the regulatory outcome is instructive. The Information Commissioner's Office issued a notice of intent on 4 July 2019 proposing a penalty of £183.39 million, and after representations issued a penalty notice on 16 October 2020 for £20 million, finding infringements of Articles 5(1)(f) and 32 of the General Data Protection Regulation as it then applied. The attacker was in a position to access the personal data of approximately 429,612 customers and staff, including the payment card details of around 244,000 customers. The reduction from the notice of intent to the final penalty is itself worth noting by anyone modelling regulatory risk: it was not a finding that the breach was less serious, but the product of representations and of the Commissioner's five-step penalty methodology.

The controls that address e-skimming are now explicit in the standard rather than implied. PCI DSS version 4.0.1 is the current version of the standard as of writing in August 2026. Its future-dated requirements ceased to be best practice and became mandatory on 31 March 2025. Two of them exist specifically for this attack:

Requirement	What it obliges	Status as of writing, August 2026
6.4.3	Maintain an inventory of all scripts executing on payment pages, authorise each one, justify its necessity, and assure its integrity	Mandatory since 31 March 2025
11.6.1	Deploy a change and tamper detection mechanism that alerts on unauthorised modification of the HTTP headers and content of payment pages as received by the consumer browser	Mandatory since 31 March 2025

The PCI Security Standards Council published an information supplement, "Payment Page Security and Preventing E-Skimming — Guidance for PCI DSS Requirements 6.4.3 and 11.6.1", on 10 March 2025, which is the primary reference for how to satisfy them.

The rest of PCI DSS matters here too and is frequently misremembered, so it is worth stating plainly. Sensitive authentication data — full track data from stripe or chip, the CVV2 family of codes, and PINs or PIN blocks — must not be retained after authorisation, by anyone, under any circumstances, even encrypted. The PAN may be stored but must be rendered unreadable wherever it is stored. Those two rules between them determine what a breach can actually yield, and they are the reason a well-run breached merchant loses PANs and expiry dates while a badly run one loses everything needed to transact.

Two reporting obligations attach in the United Kingdom, and they run on different clocks. Under UK GDPR Article 33, a personal data breach must be notified to the Information Commissioner's Office without undue delay and, where feasible, not later than 72 hours after becoming aware of it. Separately, under the Payment Services Regulations 2017, a payment service provider must report major

operational or security incidents to the Financial Conduct Authority. Card scheme rules add a third: an account data compromise event triggers scheme notification and, above thresholds set by each scheme, a PCI Forensic Investigator engagement. All three are live as of writing in August 2026.

Card testing and enumeration attacks

The last category is aimed at issuers, executed through merchants, and paid for by both.

The mechanism is straightforward. An attacker sends authorisation requests through a merchant's payment endpoint using candidate card credentials and reads the response. An approval, or in some configurations a decline code that distinguishes "wrong CVV2" from "no such account", tells the attacker which candidates are live. The merchant is not the target. The merchant is the oracle.

The arithmetic explains why this is worth doing and where the defences must sit. Under ISO/IEC 7812:2017 an issuer identification number may be eight digits within a PAN of up to nineteen, and the final digit is a Luhn check digit. For a sixteen-digit PAN under an eight-digit IIN, that leaves seven free digits: ten million candidate accounts, of which some fraction is issued. Expiry dates within a typical five-year validity window give around sixty combinations. CVV2 gives one thousand. Guessing a complete credential from nothing is not economic. Completing a partial credential is: given a PAN from a breach, expiry and CVV2 together are sixty thousand combinations, and an attacker who can make unlimited attempts will find them.

That "unlimited" is the vulnerability, and it was documented precisely. Ali, Arief, Emms and van Moorsel of Newcastle University published "Does the Online Card Payment Landscape Unwittingly Facilitate Fraud?" in IEEE Security and Privacy in 2017, describing what they called the distributed guessing attack. Their finding was structural rather than cryptographic: because merchants require different combinations of fields, and because no party was counting failed attempts across merchants, guesses could be spread across many sites so that no single site saw enough failures to react. They reported that Mastercard's centralised network detected the pattern within fewer than ten attempts even when distributed, and that Visa's did not. The fix is not at the merchant. It is centralised counting.

Enumeration has a variant that is particularly awkward: account verification messages, zero-value or nominal-value authorisations whose purpose is to check a card is live. They produce no clearing record, no settlement and no cardholder-visible charge, which makes them ideal for testing and invisible to the person whose card is being tested.

The scheme response, and the figures a practitioner will be measured against, is the Visa Acquirer Monitoring Program. Visa's own published material states that VAMP took effect on 1 April 2025 with an advisory period from 1 April 2025 to 30 September 2025, that it consolidates five existing fraud and dispute programmes into a single acquirer programme and thirty-eight remediation processes into one, that it incorporates enumeration criteria based on the count of enumerated authorisation transactions and an enumeration rate, and that enumeration attacks account for $1.1 billion annually in fraud losses.

The numeric thresholds do not have to be taken on trust from acquirers and risk vendors, because Visa publishes them itself. Its fact sheet "Visa Acquirer Monitoring Program Overview", hosted on the corporate Visa site, sets them out as follows, and they stand as below at the time of writing in August 2026:

Measure	Definition	Threshold
VAMP ratio	(TC40 fraud reports + TC15 disputes) ÷ TC05 settled transactions, card-not-present	Acquirer: 50 bps Above Standard, 70 bps Excessive
VAMP ratio, merchant	As above	Excessive, reduced from 220 bps to 150 bps from 1 April 2026 in the United States, Canada, the European Union and Asia-Pacific
Minimum count	Combined fraud and dispute events per month to enter the programme	1,500, on a different basis in the CEMEA region
Enumeration ratio	Enumerated authorisations, approved and declined, ÷ total authorisations	2,000 bps, that is 20 per cent
Minimum enumeration count	Enumerated authorisations per month to enter the programme	300,000

Two features of that table are worth dwelling on. The enumeration ratio counts declined authorisations in both numerator and denominator, which is correct, because a card-testing run is overwhelmingly declines and a ratio built only on approvals would be blind to it. And the VAMP ratio counts a TC40 fraud report and a TC15 dispute arising from the same transaction as two events, so one bad transaction can move the ratio twice.

Defences against enumeration are layered, and each layer catches a different thing.

At the merchant, the controls are rate limiting by internet protocol address, device and session, bot management, a minimum transaction amount, removal of free-text amount fields on donation pages, cardholder verification challenges on low-value flows, and monitoring of the approval ratio and the mix of decline codes as a leading indicator. A merchant whose approval rate collapses overnight is usually being used as an oracle, not losing customers.

At the gateway and acquirer, the controls are velocity rules by card, by BIN range and by merchant, pre-authorisation screening that never lets the request reach the network, and specific treatment of zero-value account verification traffic.

At the issuer, the controls are CVV2 attempt counters that block a card after a small number of failures, velocity monitoring across the issuer's own BIN ranges regardless of which merchant the attempts arrive from, and deliberate parsimony in decline codes. This last point connects directly to the response codes discussed in Volume III: "do not honour" is uninformative on purpose. Every extra bit of information in a decline is a bit of information for an enumeration attacker, and the industry's decision to be vague is a security decision, not laziness.

At the scheme, the controls are network-level enumeration detection of the kind Visa markets as Account Attack Intelligence, the monitoring programmes above, and financial penalties on acquirers that transmit the traffic.

The threat model in one table

Attack	What is obtained	Primary control	Where the control lives
Stripe skimming	Full track data, replayable	EMV chip; stripe withdrawal; CVV1 validation	Card and issuer
Shimming	PAN, expiry, track-2 equivalent with iCVV, one spent cryptogram	iCVV and service code validation; ATC monitoring; PTS POI tamper resistance	Issuer and device
Contactless relay	A genuine cryptogram, at a distance	RRP timing bound; SCA contactless limits; CDCVM on wallets	Terminal, kernel and regulator
Merchant or processor breach	Bulk PAN and expiry, sometimes more	P2PE; tokenisation; PCI DSS storage rules	Merchant and acquirer
E-skimming	Full card-not-present credential, at scale	PCI DSS 6.4.3 and 11.6.1; script control	Merchant
Enumeration and card testing	Validated credentials from partial data	Centralised velocity counting; CVV2 attempt limits; scheme monitoring	Issuer and scheme

The pattern in the right-hand column is the argument of this chapter. Every control that worked, worked because it was placed where the attacker could not choose to avoid it. EMV worked because the card is in every transaction. P2PE works because the reading device is in every transaction. Centralised velocity counting works because the network is in every transaction. Every control that failed, failed because it was placed at a merchant, and there are millions of merchants, and the attacker picks which one to use.

What the law does when it fails

The consumer position in the United Kingdom is governed by the Payment Services Regulations 2017, and it is more favourable than most people assume. Under regulation 76, where a payment transaction was unauthorised, the payment service provider must refund the amount and restore the account to the state it would have been in, as soon as practicable and in any event no later than the end of the business day following the day on which it becomes aware of the unauthorised transaction, unless it has reasonable grounds to suspect fraudulent behaviour by the payment service user and notifies the appropriate person under section 333A(2) of the Proceeds of Crime Act 2002.

Under regulation 77, the payer may be made liable for a maximum of £35 for losses arising from the use of a lost, stolen or misappropriated payment instrument, and even that does not apply where the loss was not detectable by the payer before the payment, where it resulted from an act of the provider's employee or agent, where the payer had already notified the provider, or where the provider did not require strong customer authentication. The £35 figure and both regulations are as they stand at the time of writing in August 2026. In practice UK Finance reports that customers are fully refunded in more than 98 per cent of unauthorised fraud cases, which is why the losses in the table at the start of this chapter fall on banks and merchants rather than on cardholders.

One structural change is pending and worth flagging for anyone reading this after 2026. On 11 March 2025 the United Kingdom government announced its intention to abolish the Payment Systems Regulator and consolidate its functions primarily within the Financial Conduct Authority. As of writing in August 2026 the PSR continues to exist and to exercise its functions, and the legislation required to complete the consolidation is awaited. The National Payments Vision has also committed the government to revoking the strong customer authentication provisions from the Payment Services Regula-

tions 2017 in favour of an outcomes-based approach, which, if delivered, will change the regulatory footing of several paragraphs in this chapter.

That is the honest state of this material. The mechanisms in this chapter — replay, interception, relay, bulk theft, enumeration — are permanent, because they follow from what a credential is. The numbers, the caps, the thresholds and the regulators are not, and this chapter will need rewriting before the rest of the book does.

45.98 Common wrong ideas

Wrong: The chip ended counterfeit card fraud. **Right:** The chip is sound, but its deployment was uneven and its fallback paths exploitable; on United States dual-message networks the counterfeit fraud rate was 9.0 basis points in 2015, fell to 6.6 in 2017, and was back to 9.1 by 2021.

Wrong: The chip keeps your card number secret. **Right:** The chip never reveals its key, but it reveals the primary account number, the expiry date and track-equivalent data in the clear on every transaction; what it removed was replay at a terminal, not disclosure.

Wrong: Most stolen card details come from stolen or copied cards. **Right:** The dominant source is a breached merchant, processor or checkout page that touches no physical card at all, and a fast-growing share involves no card-data theft whatsoever, only a telephoned customer reading out a one-time passcode so the criminal can load the card into their own wallet.

Wrong: Relay attacks are the main contactless threat. **Right:** A relay borrows genuine cryptography rather than breaking it, which makes it interesting to researchers and lawyers; the £46.8 million of contactless fraud reported for 2025 is dominated by ordinary use of ordinary stolen cards.

Wrong: The Relay Resistance Protocol proves the card was physically at the terminal. **Right:** It is a latency bound with a granularity of hundreds of microseconds, and one hundred microseconds is thirty kilometres of light in free space, so it cannot localise a card geometrically at all.

Wrong: "Card data" is a single asset to be protected. **Right:** A PAN alone, a PAN with expiry, a PAN with expiry and CVV2, full track data, a PIN block and a network token are six different things with six different resale values, attack paths and controls, and merging them is the commonest analytical error in the field.

Wrong: A successful shim-derived clone proves EMV is broken. **Right:** It proves an issuer failed to validate iCVV, validated the wrong verification value, or ignored the service code; the defect sits in a small number of authorisation stacks and is closable by configuration.

Wrong: Card testing is a merchant problem, fixed by rate limiting at the merchant. **Right:** The merchant is the oracle rather than the target, and because attempts can be spread across many merchants the only counting that works is centralised at the network or the issuer.

Wrong: Decline codes should tell the merchant precisely what went wrong. **Right:** Every extra bit of detail in a decline is a bit of information for an enumeration attacker, so the vagueness of "do not honour" is a deliberate security decision rather than laziness.

Wrong: Cardholders end up carrying the losses in the fraud tables. **Right:** Regulation 76 requires refund by the end of the following business day, regulation 77 caps the payer at £35 and disapplies even that in most cases, and more than 98 per cent of unauthorised fraud cases are fully refunded, so the losses fall on banks and merchants.

45.99 Chapter summary in 20 lines

1. Card fraud is not one crime but at least five, and they have almost nothing in common technically.
2. The magnetic stripe is a bearer credential in the purest sense: static data that a reader learns in full and a writer can reproduce exactly.
3. The chip replaced that key with a locksmith — a secret it never discloses, a terminal nonce and a counter that ticks on every transaction — so every reply is fresh.
4. A captured cryptogram is therefore a used ticket, which is why copying a chip conversation does not produce a working counterfeit card.
5. Skimming is close to dead where chips are used properly, with EMVCo reporting 15.6 billion cards and 97 per cent of card-present transactions on chip as of the fourth quarter of 2025.
6. A shim can read the chip dialogue because that dialogue was never confidential; authenticity, not secrecy, was its security property.
7. What a shim yields is the account number, the expiry and track-2 equivalent data carrying iCVV, which is useful on a website and useless at a terminal.
8. EMV-bypass cloning succeeds only against issuers that fail to check iCVV or the service code, so the weakness lies in a few authorisation stacks rather than in EMV.
9. A relay attack borrows genuine cryptography rather than breaking it, which makes it unfalsifiable from the issuer's side and famous out of all proportion to its losses.
10. Three things constrain relays: centimetre-range radio, a transaction time budget commonly cited at around 500 milliseconds, and EMV's Relay Resistance Protocol.
11. That protocol is a latency bound and not academic distance bounding, and its granularity leaves the acceptance window wider than a security engineer would like.
12. In practice the binding constraint on contactless is the authentication regime, with the United Kingdom's £100 and £300 figures now industry practice rather than a hard cap after Handbook Notice 136.
13. The dominant modern source of card data is the breach: Target lost up to 40 million payment card records in three weeks of 2013 with no skimmer, shim or radio involved.
14. Point-to-point encryption and tokenisation answer server-side theft by making the merchant's environment hold nothing worth taking.
15. E-skimming moved the theft into the shopper's browser, where the merchant's servers never see it, and British Airways in 2018 is the United Kingdom reference case, penalised £20 million on 16 October 2020.
16. PCI DSS requirements 6.4.3 and 11.6.1, mandatory since 31 March 2025, exist for exactly that attack, while the storage rules on sensitive authentication data determine what any breach can yield.
17. Enumeration completes partial credentials rather than guessing whole ones: sixty expiry dates against a thousand codes is sixty thousand tries, which only unlimited attempts make economic.
18. The merchant is the oracle rather than the target, and the documented fix is centralised counting at the network, as the Newcastle work on distributed guessing showed in 2017.
19. The Visa Acquirer Monitoring Program supplies the discipline, with a combined fraud and dispute ratio, an enumeration ratio of 2,000 basis points that counts declines on both sides, and penalties landing on acquirers.
20. Every control that worked was placed where the attacker could not choose to avoid it — the card, the reading device, the network — and every control that failed was placed at a merchant, and the attacker picks the merchant.

Sources: UK Finance Annual Fraud Report 2026 and 2025 press releases; Home Office call for evidence on unauthorised fraud, July 2026; The Nilson Report, January 2026; EMVCo worldwide deployment statistics, Q4

2025; Federal Reserve Bank of Kansas City payments research briefing, February 2025; PCI Security Standards Council document library and e-skimming information supplement, March 2025; Visa corporate publications on the Acquirer Monitoring Program; FCA press release of 19 December 2025 and Handbook Notice 136; Payment Services Regulations 2017 regulations 76 and 77 as published on legislation.gov.uk; ICO penalty notice against British Airways, October 2020; Mastercard magnetic stripe announcement, August 2021; EMV Contactless Book C-2 and CPACE terminal kernel specification for the Relay Resistance Protocol; Radu et al., IEEE S&P 2022; Ali et al., IEEE Security and Privacy, 2017; Krebs on Security and RCMP material on shimmers, January 2017.

46

CHAPTER

Card Not Present

46.0 What this chapter gives you

1. You will be able to explain why a chip transaction cannot be replayed while an online one is nothing but replayable facts, and why that single difference decides the rest of the chapter.
2. You will be able to say what "card not present" actually means — a combination of indicators in an authorisation message, not a fact about where the customer was standing.
3. You will be able to describe exactly what the address verification service compares, and why a criminal holding a delivery label defeats it without ever knowing the street name.
4. You will be able to read the address verification response codes and explain why `G`, `U`, `S` and `R` mean the merchant learned nothing, and what a decline-on-mismatch rule costs in conversion.
5. You will be able to explain why the shop rather than the bank carries an online fraud loss by default, and why the acquirer is the party of last resort behind it.
6. You will be able to state the effect of regulation 77(4)(d) of the Payment Services Regulations 2017 and say why a United Kingdom consumer's exposure to unauthorised online card fraud is zero rather than £35.
7. You will be able to explain why quoting £328.4 million of 2008 card-not-present fraud without the £41.2 billion of online spending behind it is campaigning rather than reporting.
8. You will be able to map the three e-commerce indicator outcomes to their liability positions, and name the three limits that cause most disappointment about the liability shift.
9. You will be able to say why first-party misuse is invisible at the message layer, and what Compelling Evidence 3.0 does instead of arguing about it.
10. You will be able to explain why a billing descriptor that does not resemble the trading name is a fraud problem the merchant manufactured itself, and the cheapest one on the list to fix.

The plain version

Imagine a members' club with a very good doorman.

When you arrive in person you hand him your membership card. The card has a small computer inside it. The doorman's reader asks the card a question, the card does a sum using a secret number that only it and the club's office know, and hands back an answer. The answer is different every single time. If somebody photographed everything about your card and turned up tomorrow with a perfect copy of the picture, they would still be stuck at the door, because the doorman would ask his question and a picture cannot do a sum.

Now imagine the same club takes bookings by post. You write a letter containing your membership number, the date your membership runs out, and the three-digit code on the back of your card. The club opens it, checks all three against its file, and books you a table.

Everything about that letter is copyable. The membership number does not change. The expiry date does not change. The three-digit code does not change. Anybody who has seen your card, or received one of your letters, or broken into a filing cabinet full of other people's letters, can write out an identical letter and send it in. The club will read it and it will agree with itself, because all the club can do is check the writing on the page against the writing in its file.

That is the whole problem of buying things on the internet. In a shop the chip does a fresh sum every time. Online there is no chip, no reader, no sum. There is only a page of facts about a card, and facts can be copied.

Nadia buys a pair of running shoes for £48.99 from a British website. The checkout page asks her for five things: the sixteen digits on the front of her card, the expiry date, the three-digit code on the back, her name, and her billing address, which is Flat 4, 27 Bramble Road, Leeds, LS6 2QD.

The shop sends all of that to its bank, which sends it to Nadia's bank, which does three checks and answers all three in the same breath. Is there £48.99 available and is the card in good standing? Yes. Does the three-digit code match the one we generated? Yes. Does the address match the one we hold on file?

That third check is much weaker than it sounds. The bank does not compare the words. It does not know or care that the street is Bramble Road or the town is Leeds. It pulls out the numbers, the house number and the digits of the postcode, compares them with the digits it holds, and reports whether both agree, one does, or neither does.

So a criminal who has stolen Nadia's card details does not need to know she lives on Bramble Road. He needs a house number and a postcode, the two least secret things about anybody. They are printed on delivery labels, typed into hundreds of websites, and usually stolen in the same breath as the card number, because the same shopping basket that took the card number also took the address to send the shoes to.

Now the part that decides everything else: who loses the money.

Suppose the transaction was not Nadia. Somebody in another country bought £48.99 of shoes with her card. She sees the line on her statement, does not recognise it, and rings her bank, which puts the £48.99 back into her account. That is not a favour; in the United Kingdom it is the law, and it has to happen quickly. Her bank then tells the shop's bank it wants the money back. The shop's bank takes the £48.99 out of the shop's account and charges the shop a fee for the trouble on top. So the shop has lost the shoes, the £48.99, the postage and the fee. The bank has lost nothing. Nadia has lost nothing.

That asymmetry is the single most important fact in this chapter. In a shop, when the chip does its sum properly, the bank carries the fraud. Online, by default, the shop carries it. Which means everything a merchant does about online fraud, every awkward extra screen, every declined order, every "please confirm this purchase in your banking app", is a merchant spending its own money to protect its own money.

For decades the easiest way to steal with a card in Britain was to copy the magnetic stripe and make a fake card, or to steal a real card and forge a signature. Then, over 2004, 2005 and 2006, the country replaced signatures with chips and PINs. It worked: between 2004 and 2008 the value of counterfeit card fraud committed in the United Kingdom fell by sixty-eight per cent. The criminals did not retire. They moved. Over the same stretch card-not-present fraud on British cards rose from £72.9 million in 2000 to £328.4 million in 2008, and by 2008 total losses on British cards hit £609.9 million, the highest since 1998, of which fifty-four per cent was card-not-present and only twenty-eight per cent counterfeit.

Squeeze a balloon at one end and the air goes to the other. Security measures rarely destroy fraud. They relocate it, towards whichever door is now cheapest to open.

Finally, there is a kind of card-not-present loss with no criminal in it, and it is the one merchants find hardest to talk about. Sometimes the person who says "I did not buy that" really did buy it. Sometimes a twelve-year-old bought it on a parent's card. Sometimes a subscription renewed and the customer had forgotten signing up. Sometimes the statement line reads "SP*ORD9931" and the customer genuinely does not recognise their own purchase. And sometimes somebody has worked out that saying "I did not buy that" is a quick way to get free shoes.

At the level of the message that arrives at the shop, every one of those looks identical. They arrive as the same dispute, with the same code, saying the same thing. The industry calls this friendly fraud, or first-party misuse, and nothing in the payment system separates it from the real thing. That is the awkward part.

Where the plain version stops being true

"Card not present" is not a description of the shopper. It is a set of codes in a message. Whether a card is present is not a physical fact about the room. It is an assertion the acquiring side puts into the authorisation, and the issuer prices, risk-scores and disputes the transaction on the basis of that assertion rather than on reality. A customer can stand at a counter with the card in their hand and still generate a card-not-present transaction, because the shop typed the number into a virtual terminal or the chip reader was broken. Conversely, a subscription renewal at three in the morning with no human near it is card-not-present in the same category as a live checkout, yet treated differently again because it carries an indicator saying the merchant initiated it. The question is never "was the card there?" It is always "what did the message claim, and does the issuer believe it?"

Address verification does not verify an address, and often does not run at all. Two things are hidden. The first is that the check is numeric and partial, so it is defeated by knowing two facts that are barely secret. The second, and more practically important, is coverage: it depends on the issuer supporting it and on the acquiring chain passing the data through. British and American issuers generally do; a great many issuers elsewhere do not, and return a code meaning "not checked" or "not supported". A merchant selling internationally gets a real answer on some of its traffic and a shrug on the rest, so a rule that declines on mismatch will refuse many honest foreign customers and almost no informed criminals. There is also the mundane failure mode: a cardholder who moved house and never told their bank fails the check on every purchase for years, while being entirely genuine.

The liability shift did not move money. It moved a right to argue, and it did not move it to a bank. What the shift changes is who carries the risk of one category of dispute, under scheme rules that are private contracts, not law. The party carrying the merchant's risk in practice is the acquirer, because the acquirer is the one the scheme takes the money from. If a merchant with a large fraud problem goes out of business, the chargebacks keep arriving for months and the acquirer pays them from its own capital. That single fact explains behaviour that looks arbitrary from the merchant's side: rolling reserves, delayed settlement for new merchants, onboarding refusals for high-risk sectors, and the acquirer's intense interest in a dispute ratio. The acquirer is not being difficult. It is the party of last resort.

"Friendly fraud" is a commercial term, not a legal one, and a meaningful share of it is the merchant's own fault. No regulation anywhere contains the phrase. In United Kingdom law the customer's claim is simply that a transaction was unauthorised, and the burden of proving otherwise sits by statute on the payment service provider. The law goes further than most merchants realise: the fact that the

instrument was used, as recorded by the bank's own systems, is expressly stated not to be sufficient in itself to prove the customer authorised it. The merchant who says "the transaction went through, so obviously they authorised it" is making an argument the legislation has already anticipated and rejected. Separately, an unglamorous share of these disputes are caused by the merchant: a descriptor that does not resemble the trading name, a free trial that converts silently, a delivery date that slipped. That is not fraud in any direction. It is bad merchandising producing a fraud-shaped message.

One caution applies to every number below. Published fraud figures are not comparable with each other and often not with themselves. Some count cards or accounts rather than people; some count issuer reports of suspected fraud, which are not confirmed loss; and year-on-year percentages are routinely computed against restated prior-year figures. Every figure below therefore carries its source and its date.

The technical version

What the message actually says

A card-not-present transaction is not a distinct protocol. It is an ordinary authorisation request carrying a particular combination of indicators, and everything downstream, from the interchange rate to the dispute rights, keys off them.

The primary indicator is the point-of-service entry mode, carried in ISO 8583 data element 22. The values that matter here are those asserting the card details were not read from the card: manually keyed entry, and electronic commerce. Acquirer specifications commonly assign `01` to manual key entry, `05` to chip, `07` to contactless chip and `81` to electronic commerce, but this is precisely the area where, as Chapter 24 set out, the standard fixes the box number and the parties negotiate the contents. An integrator must read the specification in front of them rather than a table on the internet.

Alongside it sit several other assertions, each changing the commercial and legal treatment:

Assertion	What it claims	What it changes
Entry mode	How the card data reached the terminal	Interchange, fraud scoring, dispute rights
E-commerce indicator (ECI)	Whether the cardholder was authenticated, and to what degree	Whether fraud liability sits with issuer or merchant
MOTO indicator	The order was taken by mail or telephone	Excludes most authentication routes entirely
CIT/MIT flag	Whether the cardholder or the merchant initiated this specific transaction	Whether authentication was required at all
Recurring/instalment indicator	Part of an agreed series	Availability of an authentication exemption

A merchant's fraud position is therefore decided partly by the accuracy of its own integration. One that flags genuine cardholder-initiated e-commerce as merchant-initiated, or sends recurring transactions without the series indicator, will find a liability shift it believed it had does not exist when a dispute arrives. That is among the most common and least discussed causes of lost disputes.

What the merchant can check, and what each check is actually worth

The Luhn check on the primary account number, described in Chapter 42, involves no network at all: it proves only that the digits are internally consistent, and catches typing errors. The expiry date is likewise static and printed on the card. The issuer identification number is more useful, because it names the issuing institution, country and product, and the mix of card types in an order stream is stable for a real business and unstable for an attacked one.

The card verification code printed on the signature panel, called CVV2 by Visa and CVC2 by Mastercard, is the one static check with real teeth, and the reason is archival rather than cryptographic. The code is not in the magnetic stripe and not in the chip, and PCI DSS prohibits merchants and processors from retaining it after authorisation. A criminal who obtained card data from a merchant's stored records should therefore not have it; one who copied the physical card, or persuaded the cardholder to read it out, will. The code does not distinguish honest from dishonest. It distinguishes one class of data theft from another, and it is worth exactly that much.

Everything beyond address verification, treated below, is inference rather than verification: network address, device, email address, shipping address, time, basket composition, the velocity of orders sharing any of those, and how it all compares with the merchant's own history. That is the domain of Chapter 51.

Address verification and its limits

The address verification service is an issuer-side check invoked as part of authorisation. The acquiring side sends the billing address supplied at checkout; the issuer compares it with the address on the account and returns a single-character result code alongside the authorisation response.

The comparison is not textual. It compares the numeric portion of the street address and the numeric portion of the postcode or ZIP code. Street, town and county names play no part. That was a deliberate decision from an era of noisy address data, where "27 Bramble Rd" and "Flat 4, 27 Bramble Road" had to be treated as the same place, and it has never been revisited.

Each scheme publishes its own code set and they do not agree. The following are the codes a British merchant will meet most often, as published by the schemes and by acquirers at the time of writing in August 2026:

Code	Meaning
`Y`	Address and postcode both match (five-digit ZIP in the United States)
`A`	Address matches, postcode does not
`Z`	Postcode matches, address does not
`N`	Neither matches
`F`	Visa only: address and postcode both match, United Kingdom domestic
`G`	Non-participating issuer outside the United States; not verified
`U`	Issuer holds no data or the service is unavailable
`S`	Issuer does not support address verification
`R`	System unavailable, retry

Four properties of this list matter more than the list itself.

First, most of the codes are not answers. `G`, `U`, `S` and `R` all mean the merchant learned nothing. A rule expressed as "decline on anything other than a full match" therefore declines a large fraction of legitimate international orders, and merchants who write that rule discover it in their conversion figures rather than their fraud figures.

Second, the result arrives with, or after, the authorisation decision, not before it. The issuer may approve a transaction whose address check failed completely, because the two decisions are separate: approval is about funds and card status, the address code is advice. A merchant that rejects on the strength of the code must then reverse or void the authorisation, and if it does not it leaves a hold on a genuine customer's balance for days. Failing to reverse abandoned authorisations is among the most common defects in a young integration, and it generates complaints that look like fraud disputes and are not.

Third, the check is defeated by the same breach that supplies the card number, because retail data theft almost always yields the billing address in the same record. It is not a defence against an attacker holding a full record; it is a defence against one holding a partial record, which is a real and common situation.

Fourth, it does nothing for digital goods, where there is no shipping address, and is easily neutralised for physical goods by shipping somewhere other than the billing address, which honest customers do constantly. The correct use is therefore not as a gate but as one weighted input, where a mismatch on a high-value first order from an unrecognised device is worth a great deal and a mismatch on a small repeat order from a two-year customer is worth almost nothing.

Who bears the loss, in law and in practice

In the United Kingdom, the statutory position for consumer card payments is set by the Payment Services Regulations 2017 (S.I. 2017/752), and it is far more favourable to the cardholder than most people working in payments assume. What follows describes the regulations as in force at the time of writing in August 2026; legislation.gov.uk recorded regulation 77 as up to date to 14 August 2026.

Regulation 76(1) requires the payment service provider to refund the amount of an unauthorised transaction and restore the account. Regulation 76(2) requires that refund "as soon as practicable, and in any event no later than the end of the business day following the day on which it becomes aware of the unauthorised transaction", with a single exception in 76(3) where the provider suspects fraud by its own customer and reports it in writing under the Proceeds of Crime Act 2002.

Regulation 77(1) allows the provider to hold the payer liable for up to £35 of losses arising from a lost, stolen or misappropriated instrument, and 77(3) removes all protection where the payer has acted fraudulently or has failed, with intent or gross negligence, to keep credentials safe.

Then comes regulation 77(4), where card-not-present becomes a special case. Except where the payer has acted fraudulently, the payer is not liable for any losses at all in four situations. Two concern notification. The third is where strong customer authentication was required but the payer's own provider did not apply it. The fourth, at regulation 77(4)(d), is where "the payment instrument has been used in connection with a distance contract", subject to narrow exceptions.

An ordinary online purchase is a distance contract. The practical effect is that for a consumer card used to buy something online in the United Kingdom, the £35 excess does not apply at all. The cardholder's exposure to unauthorised online card fraud is, absent their own fraud, zero. Regulation 77(6) completes the circuit: where strong customer authentication was required but the payee or the payee's provider did not accept it, that party must compensate the payer's provider for what it paid out.

Two evidential provisions finish the picture. Under regulation 75(1) the burden of proof is on the payment service provider to show the transaction was authenticated, accurately recorded and unaffected by technical failure. Under 75(3) the use of the instrument as recorded by the provider "is not in itself necessarily sufficient to prove" either that the payer authorised it or that the payer was fraudulent or grossly negligent, and under 75(4) a provider alleging either must give the payer supporting evidence.

Read together, these provisions leave the issuer under a next-business-day refund obligation with a burden of proof it usually cannot discharge cheaply, and therefore with one economically rational response: refund the customer and push the loss down the chain by raising a chargeback. The scheme rules then determine whether the acquirer, and through it the merchant, can push it back. That is how a consumer protection statute becomes a merchant cost line, and it is why the merchant's true cost of a single fraudulent order is never the order value alone. It is the order value, plus goods and shipping already dispatched, plus the chargeback fee, plus the labour of responding, plus that dispute's contribution to the ratios described next, plus, over time, the controls bought to prevent it and the revenue lost to honest customers those controls turned away.

The price of the channel

Card-not-present acceptance is also more expensive to run before any fraud occurs, and the clearest recent United Kingdom illustration is a matter of public regulatory record. Following the United Kingdom's withdrawal from the European Union, Mastercard and Visa increased interchange fees for card-not-present transactions between the United Kingdom and the European Economic Area on consumer cards, from 0.2 per cent to 1.15 per cent for debit and from 0.3 per cent to 1.5 per cent for credit. The Payment Systems Regulator opened a market review, and its final report concluded that the two schemes were not subject to effective competitive constraint, that the increases were made without regard to the effect on businesses, that no justification for them was identified, and that they were costing United Kingdom businesses an additional £150 million to £200 million a year.

The state of that remedy at the time of writing is worth recording precisely, because it will not stay true. As of the regulator's page last updated in October 2025, it had decided not to proceed with the interim cap it had consulted on, citing litigation about its powers, and had instead launched a consultation on the methodology for a longer-term cap. No cap was in force. Separately, in March 2025 the government announced its intention to abolish the Payment Systems Regulator and consolidate its functions primarily into the Financial Conduct Authority, with a Treasury consultation closing in October 2025 and legislation expected when parliamentary time allows. Any reader in a later year must check both.

The principle underneath survives regulatory change: card-not-present interchange is higher than card-present interchange, in every scheme and region, because the issuer carries a higher expected fraud cost and prices accordingly. Anything that reduces that cost, above all authentication, tends to be rewarded with a lower rate.

Scheme monitoring: the thresholds that actually discipline a merchant

Statutory liability determines who pays for a given transaction. Scheme monitoring programmes determine whether a merchant may keep trading at all, and they shape day-to-day behaviour in a fraud team. Visa consolidated its previous Visa Dispute Monitoring Program and Visa Fraud Monitoring Program into a single Visa Acquirer Monitoring Program, VAMP, effective 1 April 2025. It combines fraud reports and disputes into one ratio rather than tracking them separately, and applies at both acquirer portfolio and individual merchant level.

The thresholds below are as reported by acquirers, processors and the Merchant Risk Council at the time of writing in August 2026, and are marked as such deliberately: the Visa Rules are a private document distributed to participants and not published, so no figure here can be cited to a primary public source, and the thresholds have already changed twice since launch.

Level	Ratio threshold	Minimum volume to enter the programme
Acquirer, above standard	0.5 per cent	1,500 combined fraud and dispute events per month
Acquirer, excessive	0.7 per cent	1,500 combined fraud and dispute events per month
Merchant, excessive (from 1 April 2026)	1.5 per cent, reduced from 2.2 per cent	1,500 combined fraud and dispute events per month

The design consequence is worth stating plainly. Because fraud reports and disputes are combined into one ratio, a merchant with almost no fraud but many "goods not received" disputes can be caught by a fraud programme, and a single large merchant can drag an entire acquirer's portfolio over a threshold. That is why acquirers monitor individual merchants far more closely than their dispute volumes would seem to justify, and why a merchant that quietly refunds every complaint to keep its ratio down may be acting rationally even while being taken advantage of.

Alongside this, PCI DSS v4.0.1, published in June 2024 and the active version of the standard at the time of writing, brought two requirements out of their future-dated grace period on 31 March 2025. Requirement 6.4.3 requires every script executing on a payment page to be authorised, integrity-assured and inventoried. Requirement 11.6.1 requires a mechanism alerting staff to unauthorised changes to payment page scripts and HTTP headers, evaluated at least weekly. Both exist because the modern route to bulk card data theft is not the merchant's database but the shopper's browser, and the control is inventory and integrity rather than detection after the fact, precisely because the merchant's own server never sees the theft happen. Chapter 48 covers the standard in full.

Displacement: why the internet got worse as the shop got better

The British chip and PIN migration is the cleanest natural experiment in payment security, because it happened quickly, in one country, with published statistics either side. The figures given in the plain version come from the Office for National Statistics, drawing on data from the UK Cards Association. Of the £609.9 million lost on United Kingdom-issued cards in 2008, £379.7 million was incurred domestically and £230.1 million abroad; the composition was fifty-four per cent card-not-present, twenty-eight per cent counterfeit, nine per cent lost and stolen, eight per cent card identity theft and two per cent mail non-receipt.

Two refinements stop this being a simple morality tale.

The first is that displacement also happened geographically. Counterfeit fraud accounted for fifty-eight per cent of losses incurred abroad on British cards in 2008, because criminals took cloned British magnetic stripes to countries that had not yet migrated to chip. The stripe stayed on the card for years after the chip arrived, precisely so British cards would work abroad, and that compatibility was the vulnerability. Displacement follows the gradient of the weakest reader anywhere in the world that will accept your card.

The second is arithmetic honesty. Card-not-present fraud rose about three hundred and fifty per cent between 2000 and 2008, while the value of online shopping in the United Kingdom rose from £3.5

billion to £41.2 billion, more than tenfold. The fraud rate per pound spent therefore fell substantially while the absolute loss more than tripled. Both statements are true and only one is usually quoted. A book that reports the £328.4 million without the £41.2 billion is not reporting; it is campaigning.

The same pattern is visible today at European scale. The joint European Banking Authority and European Central Bank report on payment fraud published on 15 December 2025, covering data from 2022 to 2024, found card payment fraud seventeen times higher when the payee was outside the European Economic Area, where strong customer authentication is not legally required and frequently is not applied. Total reported payment fraud across the Area rose from €3.4 billion in 2022 to €4.2 billion in 2024, of which fraud on cards issued in the European Union and Economic Area was €1.329 billion, a twenty-nine per cent increase. The report concluded that strong customer authentication remains effective against the fraud types it was designed to defeat, while new types, chiefly the manipulation of legitimate payers into authenticating fraudulent transactions themselves, are rising.

That is confirmed on the British side. UK Finance's Annual Fraud Report 2026, published on 15 June 2026 and covering 2025, reported remote purchase card fraud losses of £423.5 million on 3.2 million cases, up three per cent by value and thirteen per cent by case count, and explicitly identified criminals compromising one-time passcodes in order to authenticate fraudulent transactions or to enrol stolen card credentials into digital wallets. The preceding edition, published on 28 May 2025, reported 2024 losses up eleven per cent to just under £400 million across nearly 2.6 million cases, and made the same observation. The two do not reconcile arithmetically, because figures are restated between editions, which is routine and legitimate and is exactly why this chapter dates every number. The 2025 edition is consistently reported, in the professional summaries written of it by firms such as TLT and circulated through Lexology, as having found that seventy-five per cent of United Kingdom e-commerce card-not-present fraud occurred at merchants acquired outside the United Kingdom, alongside £154.2 million of card fraud incurred abroad. UK Finance places the full report behind registration rather than publishing it openly, so that particular figure cannot be checked against the primary text, and it is given here on the authority of those summaries rather than of the report itself.

The mechanism to carry away does not date. Authentication closed the route that consisted of replaying stolen static credentials. It did not close the route that consists of persuading the legitimate human to complete the authentication. The defence against the second is not a stronger credential; it is behavioural detection at the issuer, out-of-band confirmation for high-risk actions such as wallet enrolment, and consumer messaging that treats a one-time passcode as something never to be repeated to another person under any circumstances, including to somebody claiming to be the bank.

Liability shift, and what it did to merchant behaviour

There are two distinct liability shifts in cards and they are routinely confused. The first is the chip liability shift, in the card-present world: where a counterfeit card is used at a terminal that could not read a chip, the party that failed to support the technology bears the fraud. That is the shift that drove terminal replacement.

The second is the authentication liability shift, which applies in the card-not-present world and is the one that matters here. Where a merchant submits a transaction successfully authenticated through 3-D Secure, fraud liability moves from the merchant to the issuer, and the issuer loses the right to raise a chargeback under the fraud reason codes. The mechanism is the e-commerce indicator carried in the authorisation, together with the cryptographic authentication value the authentication produced. The mapping, as documented by acquirers including Worldpay at the time of writing in August 2026, is:

Visa ECI	Mastercard ECI	Meaning	Liability
05	02	Cardholder fully authenticated	Shifts to issuer
06	01	Authentication attempted; card not enrolled or issuer unavailable	Generally shifts, subject to scheme rules
07	00	Authentication not performed	Remains with merchant

Three limits on this shift are load-bearing and are the source of most disappointment.

It covers fraud disputes only. It does not cover "goods not received", "goods not as described", "credit not processed" or any other non-fraud condition. A merchant that authenticates every transaction still receives chargebacks, and for many merchants those are the majority.

It does not apply to data-only authentication, where risk data is passed to the issuer to improve approval rates without a full authentication taking place. That is a conversion optimisation, not a liability instrument, and is regularly mis-sold as both.

It does not apply where the merchant, or its acquirer, requested an exemption. If you asked not to authenticate, you cannot claim the protection authenticating would have given you. That is the central trade-off of the modern checkout, and it is a commercial decision made per transaction, at speed, by software.

The behavioural effect over two decades was what an economist would predict. Merchants who could adopt authentication cheaply did so and stopped caring about fraud disputes; merchants who could not, because they took orders by telephone or sold digital goods into markets with poor coverage, stayed exposed and built fraud teams instead. The shift did not reduce fraud by itself. It reallocated the cost of preventing it onto whichever party could prevent it most cheaply, which is what a well-designed liability rule is supposed to do.

Strong customer authentication changed the calculus again by making authentication a legal requirement rather than a competitive option. The rules, set in the Payment Services Regulations 2017 and the associated technical standards, applied from 14 September 2019, but the Financial Conduct Authority granted successive forbearance for card-not-present e-commerce: an industry plan to deliver by 14 March 2021, a six-month extension for the coronavirus crisis, and a further six months announced in 2021 taking the final date to 14 March 2022, which the Authority described as the latest at which it expected full compliance.

Once authentication became compulsory, competition moved to the exemptions, which Chapter 47 treats in full. One belongs here because it prices the whole channel. Under Article 18 of the technical standards, a provider may skip authentication on a remote transaction it identifies as low risk, provided the amount is below an exemption threshold value and its own fraud rate for that transaction type sits at or below a reference rate tied to that threshold. The Annex to Commission Delegated Regulation (EU) 2018/389 sets those out as follows:

Exemption threshold value	Reference fraud rate, remote card payments	Reference fraud rate, remote credit transfers
€500	0.01 per cent	0.005 per cent
€250	0.06 per cent	0.01 per cent

Exemption threshold value	Reference fraud rate, remote card payments	Reference fraud rate, remote credit transfers
€100	0.13 per cent	0.015 per cent

The United Kingdom version restates the same three-tier structure in sterling. Article 18 as it stands in the Financial Conduct Authority's Handbook points to a table set out in an Appendix, and that table, reproduced in the Authority's own Policy Statement PS19/26 of October 2019, gives exemption threshold values of £440, £220 and £85 against reference fraud rates of 0.01, 0.06 and 0.13 per cent for remote card payments and 0.005, 0.01 and 0.015 per cent for remote credit transfers. The sterling figures look untidy because they are the euro ones converted at the Bank of England's spot rate on exit day and rounded down to the nearest five pounds, the same method that turned the €30 low-value exemption into £25.

Article 19 computes that fraud rate as the total value of unauthorised or fraudulent remote transactions over the total value of all remote transactions of that type, on a rolling ninety-day basis, covering authenticated and exempted transactions alike. Article 20 requires the exemption to cease if the rate exceeds the reference rate for two consecutive quarters. That is the discipline: a provider using transaction risk analysis aggressively to remove friction is buying conversion with a fraud budget it must account for quarterly.

One dating caution is unavoidable. The retained version of Commission Delegated Regulation (EU) 2018/389 carries, on the United Kingdom statute book at the time of writing, a recorded future effect: revocation by Schedule 1, Part 3 of the Financial Services and Markets Act 2023, as part of the programme replacing assimilated European Union financial services law with rules made by the regulators. The substance is expected to be restated in the Financial Conduct Authority's own rulebook. Any reader after 2026 must confirm the current instrument before quoting an article number.

First-party misuse: the category the system cannot see

The industry's terms are friendly fraud, first-party misuse, first-party fraud and chargeback abuse. Visa's own documentation uses "friendly fraud or first party misuse" interchangeably; Mastercard's programme is called First-Party Trust. There is no agreed definition, and no regulatory definition at all.

The reason it is structurally hard rather than merely annoying is that it is invisible at the message layer. A dispute raised because a stranger stole a card and a dispute raised because a customer changed their mind arrive under the same reason code: Visa dispute condition 10.4, "Fraud — Card-Absent Environment", and Mastercard message reason code 4837, "No Cardholder Authorization". Nothing in the message distinguishes them, and nothing can, because the distinguishing fact is a state of mind.

The economics finish the job. The issuer is under a next-business-day refund obligation, carries the burden of proof, and is told by regulation 75(3) that its own record of the instrument's use does not by itself discharge it. Investigating a £48.99 dispute properly costs more than £48.99. So it refunds and charges back, and the merchant, the only party with evidence about what actually happened, is the party furthest from the decision.

Both major schemes have responded with the same architectural idea: replace argument with a deterministic data match.

Visa's Compelling Evidence 3.0, which took effect on 15 April 2023 as an update to dispute condition 10.4, lets a merchant establish a historical footprint by supplying two prior transactions on the same

credential. Per Visa's merchant readiness documentation of March 2023, those transactions must be at least 120 calendar days old and no more than 365 days old measured from the dispute date, must carry no active fraud report and no active fraud dispute, and must be from the same merchant. At least two of four core data elements must match across all three: user identifier, network address, shipping address, and device identifier or fingerprint; at least one of the two matches must be the network address or the device identifier. Where the criteria are met, liability shifts back away from the merchant. Merchants may respond before a dispute exists, through Verifi's Order Insight service, or afterwards through their acquirer as a pre-arbitration submission in Visa Resolve Online; the submission may be attempted only once, and the response window is thirty days. Mastercard's First-Party Trust programme applies the same principle, requiring a match of one element from each of three categories.

Two consequences deserve naming. The first is that it works, because it removes human judgement: a merchant meeting the criteria wins automatically, while one that does not falls back to ordinary representment where the issuer weighs the evidence and the outcome is uncertain. That difference in expected value has turned qualification into a product requirement rather than a dispute-team concern.

The second is that it quietly converts device fingerprinting and client-side network address capture from optional analytics into commercial necessities, because those are the two elements a merchant cannot reconstruct after the event and one must be present to qualify. A server-side log of the address that received the checkout request often records a load balancer or content delivery network edge rather than the customer. The privacy consequence is real and rarely discussed: a scheme dispute rule has, in effect, mandated persistent device identification across the retail internet.

The scale is not precisely measurable, for the definitional reasons given, but the direction is not in doubt. The Merchant Risk Council's 2026 Global eCommerce Payments and Fraud Report, released on 18 March 2026 and drawn from 1,278 merchant professionals in 37 countries, found sixty-four per cent of merchants reporting increasing rates of first-party misuse, a quarter of those reporting increases of twenty-five per cent or more, while the average number of distinct fraud attack types fell from 4.2 to 3.7. More attacks of one kind, fewer kinds of attack: the profile of a maturing threat.

And it must be said, because merchants rarely say it, that a substantial share of first-party misuse is manufactured by the merchant. A billing descriptor that does not match the trading name produces disputes from customers who genuinely do not recognise the charge, and descriptor drift between the initial charge and the renewal also breaks the scheme's own merchant-matching logic and disqualifies the merchant from the remedy. A free trial that converts without a reminder produces disputes. A dispatch date that slips produces disputes filed under fraud codes, because that is what the banking app offers first. None of that is fraud. It is a merchandising problem wearing a fraud problem's clothes, and it is the cheapest thing on this list to fix.

What a competent card-not-present programme actually looks like

The working model is a constrained optimisation rather than a defence. The merchant minimises fraud losses plus dispute handling cost plus revenue lost to declined honest customers, subject to keeping its combined fraud and dispute ratio below the scheme threshold and its fraud rate low enough that its acquirer can keep claiming risk-based exemptions on its behalf. Every control buys one of those terms at the price of another.

Velocity limits and per-issuer throttling belong in that stack for a reason worth naming: a decline is not a neutral outcome. It is an answer, and an endpoint that answers quickly and often is itself a source of

information to somebody testing stolen numbers in bulk, which is why scheme monitoring now treats that testing traffic as a category of its own.

None of these controls is optional in a serious operation and none is sufficient. The internet remains the hard case not because any single defence is weak, but because the underlying credential, sixteen digits and an expiry date, is a static secret that must be shared with every party the cardholder transacts with, and a secret shared with thousands of parties is not a secret. Everything in this chapter is a compensating control for that one architectural fact. The two developments that attack the fact itself are network tokenisation, which replaces the shared number with a merchant-specific or device-specific one, and authentication, which adds a dynamic element to a static exchange. Chapter 45 covered the first. Chapter 47 covers the second.

46.98 Common wrong ideas

Wrong: "Card not present" describes a shopper who was not in the shop. **Right:** It is an assertion the acquiring side puts into the authorisation message, so a customer standing at the counter can generate one when the chip reader is broken and the number is typed into a virtual terminal.

Wrong: Address verification verifies the address. **Right:** It compares only the numeric portion of the street address and the numeric portion of the postcode, so street, town and county names play no part and two barely secret facts defeat it.

Wrong: A full address match is a meaningful gate, so decline everything else. **Right:** Most of the code set means the check did not run, and a rule expressed that way declines a large fraction of legitimate international orders while stopping almost no informed criminal.

Wrong: The address result arrives in time to prevent the authorisation. **Right:** It arrives with or after the authorisation decision, so a merchant rejecting on it must reverse or void the hold, and failing to do so generates complaints that look like fraud disputes and are not.

Wrong: The liability shift moves the loss to a bank. **Right:** It moves a right to argue under private scheme rules, and the party actually carrying the merchant's risk is the acquirer, which is why rolling reserves, delayed settlement and onboarding refusals exist.

Wrong: "Friendly fraud" is a recognised legal category. **Right:** No regulation anywhere contains the phrase; in United Kingdom law the claim is simply that a transaction was unauthorised, and the burden of proving otherwise sits by statute on the payment service provider.

Wrong: The transaction went through, so obviously the customer authorised it. **Right:** Regulation 75(3) expressly states that use of the instrument as recorded by the provider is not in itself necessarily sufficient to prove authorisation, so that argument has already been anticipated and rejected.

Wrong: Authenticating every transaction ends chargebacks. **Right:** The authentication liability shift covers fraud disputes only, and for many merchants "goods not received", "not as described" and "credit not processed" are the majority.

Wrong: Data-only authentication buys the liability shift more cheaply. **Right:** Passing risk data to the issuer to improve approval rates is a conversion optimisation, not a liability instrument, and it is regularly mis-sold as both.

Wrong: Published fraud figures can be compared year on year straight out of the report. **Right:** Some count cards rather than people, some count suspected rather than confirmed fraud, and percentages are routinely computed against restated prior-year figures, which is why every number in this chapter carries its date.

46.99 Chapter summary in 20 lines

1. In a shop the chip does a fresh sum every time; online there is no chip, no reader and no sum, only a page of facts about a card, and facts can be copied.
2. Every element the checkout asks for — number, expiry, code, name, billing address — is static, and a criminal who has any complete record has all of them.
3. "Card not present" is not a physical condition but a set of indicators in the authorisation, chiefly the point-of-service entry mode in ISO 8583 data element 22.
4. Entry mode, the e-commerce indicator, the mail-order flag, the cardholder-or-merchant-initiated flag and the recurring indicator each alter interchange, fraud scoring and dispute rights.
5. A merchant that mis-flags its own traffic discovers a liability shift it believed it had does not exist when the dispute arrives, which is among the commonest causes of lost disputes.
6. The address verification service compares numbers only, so a house number and a postcode — the two least secret things about anybody — are enough to pass it.
7. Most of its response codes are not answers at all, and the ones that are get defeated by the same breach that supplied the card number in the first place.
8. The card verification code has real teeth for an archival reason: PCI DSS forbids its retention, so a thief working from stored records should not have it while one working from the physical card will.
9. In the United Kingdom regulation 76 requires the issuer to refund an unauthorised transaction by the end of the business day following the day it becomes aware of it.
10. Regulation 77(4)(d) removes even the £35 excess where the instrument was used in connection with a distance contract, so consumer exposure to unauthorised online card fraud is zero.
11. Regulation 75 puts the burden of proof on the provider and denies that its own record of use discharges it, which leaves the issuer one economically rational move: refund and charge back.
12. The loss therefore lands on the merchant through the acquirer, and the acquirer is the party of last resort when a failed merchant's chargebacks keep arriving.
13. The merchant's true cost of one fraudulent order is the order, the goods, the shipping, the fee, the labour, the ratio, the controls bought afterwards and the honest customers those controls turn away.
14. The channel is dearer before any fraud occurs, and the United Kingdom to European interchange increases were found to cost businesses an additional £150 million to £200 million a year.
15. Scheme monitoring, consolidated into the Visa Acquirer Monitoring Program from 1 April 2025, combines fraud reports and disputes into one ratio, so a merchant with no fraud can be caught by a fraud programme.
16. The British chip and PIN migration is the cleanest natural experiment available: counterfeit fraud fell sixty-eight per cent between 2004 and 2008 while card-not-present fraud rose to £328.4 million.
17. Displacement was geographic as well as channel-wise, following the gradient of the weakest reader anywhere in the world that will accept the card.
18. Arithmetic honesty requires the £41.2 billion of online spending alongside the £328.4 million, because the rate per pound fell while the absolute loss more than tripled.
19. Authentication closed the route that replays stolen static credentials and not the route that persuades the legitimate human to authenticate, which is why passcode compromise and wallet enrolment now drive the numbers.
20. Every control in the chapter is a compensating measure for one architectural fact — a static secret shared with thousands of parties is not a secret — and only tokenisation and authentication attack that fact itself.

Chapter sources: Payment Services Regulations 2017 (S.I. 2017/752), regulations 75 to 77 on legisla-

tion.gov.uk, current to August 2026; Commission Delegated Regulation (EU) 2018/389, Articles 16 and 18 to 20 and the Annex, with the FCA Handbook Technical Standards on Strong Customer Authentication, Chapter 3, last updated 19 March 2026; the FCA's Strong Customer Authentication page, its Dear CEO letter of 20 August 2019, and its statement extending the e-commerce deadline to 14 March 2022; UK Finance Annual Fraud Report press releases of 15 June 2026 and 28 May 2025; the joint EBA and ECB payment fraud reports of 15 December 2025 and 1 August 2024; ONS Social Trends, sourced to the UK Cards Association, for the 2000 to 2008 card fraud series; the Payment Systems Regulator's cross-border interchange market review, page last updated October 2025; HM Treasury's March 2025 announcement on consolidating the PSR into the FCA; Visa's Compelling Evidence 3.0 Merchant Readiness document of March 2023; Mastercard Developers documentation for First-Party Trust; PCI Security Standards Council material on PCI DSS v4.0.1 and Requirements 6.4.3 and 11.6.1; the Merchant Risk Council press release of 18 March 2026; and acquirer documentation from Adyen, Worldpay and Stripe for address verification codes and e-commerce indicator mappings.

47

CHAPTER

Three-D Secure and Strong Customer Authentication

47.0 What this chapter gives you

1. You will be able to name the three domains, the component that sits in each — 3DS Server, Directory Server, Access Control Server — and say which one actually authenticates the cardholder.
2. You will be able to explain why the protocol is called three-domain rather than three-dimensional, and why the second wire exists at all.
3. You will be able to distinguish authentication from authorisation, and explain how a cardholder can pass one and be declined by the other ninety milliseconds later inside the same bank.
4. You will be able to read a `transStatus` value and say whether the exchange ended frictionless, went to challenge, or produced only evidence of an attempt.
5. You will be able to name the three artefacts an authentication produces — the authentication value, the e-commerce indicator and the directory transaction identifier — and say what happens when they do not reach the authorisation intact.
6. You will be able to state the counting rule of strong customer authentication and explain why a registered phone plus a fingerprint on it satisfies it while a passcode plus another passcode does not.
7. You will be able to explain dynamic linking and why changing the amount by a penny makes the approval worthless.
8. You will be able to list the exemptions with their figures as they stood in August 2026, and say which of them the United Kingdom has already diverged on or deleted.
9. You will be able to explain why an acquirer-claimed exemption and an issuer-granted one look identical to the shopper and carry opposite consequences when the money turns out to be stolen.
10. You will be able to run the merchant's arithmetic — roughly eight abandoned sales per fraudulent transaction prevented on a £420 bicycle at twelve per cent margin — and say why it inverts completely on a £6 digital item.

The plain version

Two strangers are trying to agree about a third person who is not in the room.

The shop does not know you. It knows a card number you typed in. Your bank knows you well, but has never heard of the shop and cannot reach you through its checkout page. There is no wire between them carrying a question like "is this really her?" The only wire carries money, and money messages are not conversations. They are one-line demands with one-word answers. So the industry built a second wire. Not for money. For a question.

Picture a market stall and a bank on opposite sides of a town, with a switchboard operator between them. The stallholder cannot ring the bank directly: he does not know which of the town's forty banks issued your cheque book, and none of them would take his call anyway. What he can do is hand a sealed note to the switchboard, which reads the first few digits printed on your cheque, identifies the bank, and passes the note across. Your bank opens it, decides, and hands a stamped answer back the same way.

That is the shape of Three-D Secure. Three domains: the shop's side, the bank's side, and the switchboard in the middle belonging to neither. Hence three-domain, hence 3-D. It has nothing to do with three dimensions, and the number of people who assume otherwise is one of the small ongoing embarrassments of the payments industry.

Now, what is in the note?

Priya is buying a bicycle for £420 from a British website at ten past eight on a Thursday evening. When she clicks pay, the shop's software quietly assembles a description of the moment. Not just the card number and the amount. The make and model of her phone, its screen size, its language setting, its time zone. Whether she is in the shop's own app or in a browser, and which browser. The address she wants the bicycle delivered to, and whether it matches the address her card is registered to. Her email address. How many times she has bought from this shop before. Whether this is a one-off purchase or the first payment of a subscription.

All of that goes into the sealed note, and Priya's bank now has something it never had in the old world: context. This is the same phone she used to buy petrol on Tuesday. This is her home broadband. This is her actual delivery address. The amount is large but not absurd for her. Nothing here looks wrong.

So the bank says yes. It stamps the note "authenticated", hands it back, and Priya never sees a thing. The bicycle checkout carries on as though nothing happened. From her side the button just worked. This is the frictionless flow, and on a well-implemented system it is what happens to the large majority of purchases. The check happened. She simply was not asked to participate in it.

Now change one detail. Suppose the phone is one the bank has never seen, the delivery address is in a city Priya has never shopped in, and it is four in the morning. The bank does not feel comfortable. It cannot telephone her, but it can do something better: reach her through the banking app on the phone it does trust. Her checkout pauses, a panel appears, her banking app buzzes. It shows her the merchant's name and the exact amount, £420, and asks her to approve with her fingerprint. She does. The app tells the bank, the bank tells the switchboard, the switchboard tells the shop, and the checkout resumes. That is the challenge flow. It cost her about eight seconds and it is the reason the whole system exists.

Here is the second half of the chapter, and it is a law rather than a piece of software.

In Europe and in the United Kingdom it is not optional for your bank to check that it is really you. There is a rule called strong customer authentication, and it is a counting rule. There are three kinds of evidence a bank can use: something you know, like a passcode; something you have, like a specific phone the bank has tied to your account; something you are, like a fingerprint or a face.

The bank must use at least two of the three, from different families, and they must be independent, meaning a thief who defeats one has not thereby defeated the other. A passcode plus a second passcode is not two things. A password plus a fingerprint is. A registered phone plus a fingerprint on that phone is, which is why almost every bank in Britain settled on exactly that: the phone proves possession, the

fingerprint proves inherence, and the app shows you the amount and the payee so you know what you are approving.

That last part is not decoration. The rule requires the code your phone produces to be welded to that specific amount and that specific merchant. Change the amount by a penny and the approval becomes worthless. This is called dynamic linking, and it exists so nobody can persuade you to approve one thing while a different thing is actually happening.

Now the awkward part, and the reason this chapter has a second half at all.

Checking is expensive. Not in money, in customers. Every time you interrupt a person in the middle of buying something, some of them stop buying. They lose the thread, the app does not open, the phone is upstairs, the passcode does not work, the panel looks like a scam. A rule that challenged everybody for everything would make online shopping meaningfully worse, and regulators knew it when they wrote it.

So the rule comes with holes deliberately cut in it. They are called exemptions, and each names a situation where the risk is low enough that the interruption is not worth it.

If the purchase is small, you can skip the check. In the United Kingdom, as the rules stand at the time of writing in August 2026, small means £25 or less, and only a few times in a row before the bank has to check you properly again. If you are paying the same amount to the same company every month, the bank checks you once at set-up and then leaves you alone. If you have told your bank a particular shop is one you trust, it can stop asking. And the largest hole of all: if the bank runs its own fraud analysis, decides this payment is genuinely low risk, and can prove its overall fraud losses are below a very low threshold, it may skip the check entirely, up to a value ceiling that rises the cleaner its record is.

That last one is worth pausing on, because it is a beautiful piece of regulatory design. It does not tell banks to be safe. It gives them a commercial reward — less friction, more completed sales, happier customers — that they only get to keep while their fraud numbers stay low. Let the fraud rate rise and the reward is taken away automatically.

Finally, the money. Why would a shop volunteer for any of this? Because of who pays when it goes wrong. As the previous chapter showed, when somebody uses a stolen card number online and the real cardholder complains, the default is that the shop loses the goods and the money. But if the shop asked for authentication, and the bank said yes, and the bank was wrong, then in most cases the loss belongs to the bank. The industry calls this the liability shift, and it is the entire commercial engine of 3-D Secure. The shop is not buying security. It is buying a receipt that says: your customer, your call, your loss.

So the shop is caught between two costs. Ask everybody and lose sales to friction. Ask nobody and eat the fraud. The whole of modern e-commerce payments is an argument about where between those two the answer sits, conducted transaction by transaction, in about four hundred milliseconds.

Where the plain version stops being true

Authentication and authorisation are two different messages, on two different networks, with two different answers, and one does not imply the other. The plain version makes it sound like one continuous conversation. It is not. The 3-D Secure exchange happens first, over the internet, between the shop's authentication provider and the bank's authentication server, and it produces a verdict about identity. Only then does the shop send a completely separate authorisation request over the card

network, which produces a verdict about money. A cardholder can pass authentication perfectly and be declined ninety milliseconds later for insufficient funds, for exceeding a velocity limit, or because the fraud engine on the authorisation side reached a different conclusion from the one on the authentication side. Those two systems inside the same issuer frequently disagree, because they are usually different products, sometimes from different vendors, sometimes bought a decade apart. Merchants who have never seen the inside of an issuer assume the bank is one mind. It is at least two, and they are not always on speaking terms.

3-D Secure is not strong customer authentication, and strong customer authentication does not require 3-D Secure. These are constantly conflated and they are different categories of thing. Strong customer authentication is a legal obligation on payment service providers, defined in regulation, that says how confident a firm must be about who it is dealing with. Three-D Secure is a messaging protocol published by EMVCo, a body owned by the card schemes, that carries data between merchants and issuers. The protocol is a delivery van; the regulation is the law about what may be in the van. The practical consequence is that 3-D Secure by itself authenticates nobody. The European Banking Authority made this explicit in its opinion of 21 June 2019: information transmitted using a communication protocol such as EMV 3-D Secure does not, on its own, constitute an inherence element, because the protocol carries no biometric measurement. What authenticates the cardholder is whatever the issuer's access control server does behind the protocol — the push to the banking app, the biometric, the one-time passcode. The van is not the cargo. And in jurisdictions with no authentication law at all, most obviously the United States, 3-D Secure is used enthusiastically anyway, purely for the liability shift.

The liability shift is a private contract, it is narrower than merchants believe, and it sits alongside a statutory allocation that works differently. The shift is not in any statute. It lives in the card schemes' operating rules, which are commercial documents between the schemes and their licensed members, revised at least annually, and which a merchant is bound by only indirectly through its acquirer's contract. It covers fraud-type disputes. It does not cover a customer who says the bicycle never arrived, or arrived buckled, or that they cancelled the order, or that the subscription was misdescribed. Those are processing and consumer disputes and the merchant carries them regardless of how beautifully the transaction was authenticated. Running underneath all of this, and quite separate from it, is the statutory position. In the United Kingdom the Payment Services Regulations 2017 provide that where strong customer authentication is required but the payee or the payee's payment service provider does not accept it, that party must compensate the payer's payment service provider for the resulting loss. That is law, it runs between institutions rather than between a scheme and a merchant, and it does not care what the scheme rulebook says.

Exemptions are requested by one party and granted by another, and the merchant is rarely the one who decides. The plain version implies a merchant can choose to skip the check. It cannot. A merchant, through its acquirer, can flag a transaction as one it believes qualifies for an exemption, and it can express a preference about whether it wants a challenge. The issuer decides. An issuer that disagrees will challenge anyway, or will decline the authorisation with a response code that means, in effect, go back and authenticate properly. Worse, the two sides can be applying different exemptions with different consequences: an exemption claimed by the acquirer generally leaves the fraud liability with the merchant, whereas the same transaction let through unchallenged on the issuer's own risk analysis generally does not. Two transactions that look identical to the shopper, both frictionless, can therefore have opposite consequences if the money later turns out to be stolen. Merchants who optimise for "frictionless rate" without knowing which side granted the exemption are optimising a number that does not mean what they think it means.

The technical version

The three domains and the components in them

The three-domain model divides the world into the acquirer domain, the issuer domain, and the interoperability domain that connects them.

In the acquirer domain sits the 3DS Requestor, which is the merchant or whoever acts for it, and the 3DS Server, the component that actually speaks the protocol. Historically this was the Merchant Plug-In, and the abbreviation MPI survives in most gateway APIs. In app-based flows there is also a 3DS SDK embedded in the merchant's mobile application, collecting device information natively rather than through a browser.

In the issuer domain sits the Access Control Server, universally the ACS. It holds the issuer's authentication logic: its risk model, its decision about frictionless versus challenge, its challenge user interface, and its cryptographic signing. Most issuers do not build one. They buy or rent one from a specialist vendor, which is why the challenge screens of unrelated banks in the same country often look suspiciously alike.

In the interoperability domain sits the Directory Server, the DS, operated by the card scheme. It is the switchboard: it maintains the mapping from card ranges to participating issuers' access control servers, enforces protocol version compatibility, applies the scheme's own rules, and routes every message. Each scheme runs its own and brands the consumer-facing product separately — Visa Secure, Mastercard Identity Check, American Express SafeKey, Discover ProtectBuy, JCB J/Secure. The protocol underneath is EMVCo's and common to all of them; the rules layered on top are each scheme's own and are not. EMVCo operates none of this. It publishes the specification and runs the approval processes attesting that a given ACS, DS, 3DS Server or SDK conforms to it.

From 3DS1 to 3DS2, and the death of the redirect

The first generation, formally 3-D Secure 1.0.2, was designed for a web that no longer exists. Its flow was a redirect. The merchant's plug-in asked the directory whether the card was enrolled, using a verify enrolment request and response pair. If the answer was yes, the shopper's browser was thrown out of the checkout onto a page hosted by the issuer, where they were asked for a static password. That page was frequently unstyled, frequently unbranded, and frequently arrived without warning. A meaningful proportion of issuers used activation during shopping, which meant a shopper who had never enrolled was asked to invent a password mid-purchase, on a page they had never seen, that looked exactly like the phishing attack it was training them to fall for. Abandonment was severe, mobile support was poor to non-existent, in-app purchases had no sensible flow at all, and the static password was a knowledge factor of low quality that could be captured and reused.

EMVCo's replacement, EMV 3-D Secure, was published in its first form in 2016, with version 2.1.0 following in 2017 and version 2.2.0 in December 2018. Version 2.3 was published by EMVCo in October 2021, followed by 2.3.1.0 in 2022 and 2.3.1.1 in May 2023. At the time of writing in August 2026, a draft of version 2.4.0.0 had been published by EMVCo on 3 June 2026 with a public comment period closing on 1 July 2026, which means the next revision of the protocol is in flight as this book goes to press and any version-specific detail below should be checked against the current specification.

The first generation was retired on a scheme-by-scheme timetable rather than by EMVCo. Visa's published notice, article AI10745 dated 4 February 2021, stated that effective 16 October 2021 it would discontinue support of the 3DS 1.0.2 Attempts Server for non-participating issuers, and that effective 15 October 2022 it would discontinue support of 3DS 1.0.2 and all related technology. Mastercard's timetable ended in the same month. Industry documentation and gateway migration guidance con-

sistently give 14 October 2022 as the date its support for 3DS 1.0.2 ceased, but Mastercard circulates its announcement bulletins to its customers rather than publishing them, so there is no public primary document to cite for it. For practical purposes, from late 2022 onwards, 3DS1 is gone.

The message flow in detail

Every message in EMV 3DS is a JSON structure with a defined element set and a version number. The sequence, in the order a practitioner sees it in logs, runs as follows.

The 3DS Server may first send a Preparation Request, PReq, to the Directory Server, and receive a PRes. This is not part of any individual purchase; it retrieves the protocol versions supported for a card range and, importantly, the 3DS Method URL if the issuer's ACS publishes one.

If a 3DS Method URL exists in a browser flow, the 3DS Server loads it in a hidden iframe before authentication begins, letting the ACS run its own device profiling directly against the browser and associate the result with a transaction identifier. The specification bounds how long the 3DS Server waits — ten seconds in the current versions — after which it proceeds regardless. In app flows the equivalent work is done by the 3DS SDK, which gathers operating-system-level device information and encrypts it so only the intended recipient can read it.

Then the substantive message. The 3DS Server sends an Authentication Request, AReq, to the Directory Server, which routes it to the ACS. This carries the context described in the plain version, organised into groups: cardholder data including billing and shipping address, email and telephone; account information such as how long the account has existed with the merchant and whether the card was added recently; purchase data including amount, currency, and whether it is a one-off, a recurring series or an instalment plan; merchant data including the merchant category code and country; browser or SDK device data; and indicators covering the merchant's challenge preference, any exemption claimed, and whether the request is merchant-initiated.

The ACS replies with an Authentication Response, ARes, and the single most important field in it is transStatus.

transStatus	Meaning
Y	Authentication or account verification successful
N	Not authenticated or verified; transaction denied
U	Authentication or verification could not be performed; technical or other problem
A	Attempts processing performed; not authenticated, but proof of an attempt is provided
C	Challenge required; further authentication is needed
D	Challenge required; decoupled authentication confirmed (from version 2.2)
R	Authentication or verification rejected; the issuer is rejecting the request
I	Informational only; the requestor's challenge preference has been acknowledged (from version 2.2)

A transStatus of Y or A ends the exchange there. That is the frictionless flow. Anything requiring cardholder interaction returns C, and the second phase begins.

In the challenge phase the 3DS Server or SDK exchanges Challenge Request and Challenge Response messages, CReq and CRes, with the ACS. In a browser this renders in an iframe using one of a fixed set of standard challenge window sizes, so the ACS knows in advance how much space it has. In an app the SDK renders the challenge natively using a small vocabulary of interface types — single select, multi select, text entry, out-of-band, and in later versions HTML — so it looks like part of the merchant's own app rather than a web page bolted into it.

Out-of-band is the important one, and it is the pattern almost all European issuers use. The challenge screen does not ask for anything. It says: approve this in your banking app. The actual authentication happens on a separate channel, on a device the issuer already trusts, and version 2.3 added automated transitions between the merchant app and the banking app so the shopper need not navigate manually and come back. Decoupled authentication, added in version 2.2, goes further: authentication happens entirely outside the payment session, with no challenge screen at all.

When the challenge concludes, the ACS sends a Results Request, RReq, to the Directory Server, which forwards it to the 3DS Server, which acknowledges with an RRes. This is the authoritative result. A merchant that reads the outcome from the browser rather than from the RReq is trusting the client, which is a category of mistake that needs no further explanation.

Version 2.2 also introduced 3DS Requestor Initiated messages, 3RI, which allow an authentication with no cardholder session at all — for a recurring payment, a split shipment or an account verification.

What the authentication produces, and how it reaches the money

Authentication produces three artefacts that must be carried into the authorisation message, and the carriage is where implementations go wrong.

The first is the authentication value: the Cardholder Authentication Verification Value, CAVV, in Visa's vocabulary, or the Accountholder Authentication Value, AAV, in Mastercard's. It is a cryptogram generated by the ACS which the issuer's authorisation system can verify. It is the proof.

The second is the Electronic Commerce Indicator, ECI, a two-digit field that states what kind of authentication occurred. The schemes number them differently, which is a permanent source of confusion:

Visa ECI	Mastercard ECI	Meaning
05	02	Fully authenticated
06	01	Attempted; authentication not completed but an attempt is evidenced
07	00	Not authenticated; no 3-D Secure performed or it failed

The third is the transaction identifier assigned by the Directory Server, dsTransID, which ties the authentication and the authorisation together for dispute purposes.

If these are absent, malformed, or attached to an authorisation whose amount does not match the amount authenticated, the merchant has paid the friction cost and received nothing for it. A substantial share of merchant complaints that the liability shift "did not work" resolve to exactly this.

Strong customer authentication as a legal instrument

The obligation originates in Directive (EU) 2015/2366, the second Payment Services Directive, Article 97. The operative detail is in Commission Delegated Regulation (EU) 2018/389, the regulatory technical standards on strong customer authentication and common and secure open standards of communication, adopted on 27 November 2017. Article 38 of that regulation provided that it applied from 14 September 2019, save for certain interface documentation provisions which applied from 14 March 2019. That is the date every practitioner means when they say "SCA came in".

In the United Kingdom the obligation is in regulation 100 of the Payment Services Regulations 2017, SI 2017/752, which requires strong customer authentication when a payer accesses a payment account online, initiates an electronic payment transaction, or carries out any action through a remote channel which may imply a risk of payment fraud or other abuse, and which additionally requires dynamic linking to a specific amount and payee for remote electronic payments. The detailed standards are the United Kingdom's own version of the technical standards, made by the FCA and reproduced in the FCA Handbook; all UK figures in this chapter are taken from that instrument as it stood in August 2026.

The United Kingdom did not enforce the e-commerce element on the European timetable. The FCA agreed a phased implementation, extended it twice, and stated that the deadline for full compliance for e-commerce card transactions was 14 March 2022. That date, and not September 2019, is when British online checkouts changed.

The mechanics inside the regulation constrain what any authentication product may do. Article 4 requires that authentication produce a single-use authentication code, that no information about the underlying elements can be derived from it, that a failed attempt must not reveal which element was wrong, that consecutive failed attempts be limited to no more than five before the action is blocked, and that a session of online account access must not survive more than five minutes of inactivity. Article 5 sets out dynamic linking: the payer must be made aware of the amount and the payee, the code must be specific to both, and any change to either must invalidate it. Articles 6 to 8 govern the three categories of element and Article 9 requires their independence, including specific mitigations where a single multi-purpose device — that is, a phone — carries more than one element.

What counts as each element was clarified by the European Banking Authority in Opinion EBA-Op-2019-06 of 21 June 2019, which remains the reference document even where it is no longer formally binding. Its conclusions were unwelcome to parts of the industry. Card details printed on the card are neither a knowledge element nor evidence of possession. A one-time passcode is not a knowledge element, because it is not something the user knows in advance. An application installed on a device is not evidence of possession unless it has been bound to that device. A dynamic card security code does evidence possession, and an app or browser bound to a device through a security chip or a private key does too. On the inherence side, fingerprint, iris, face and hand geometry, voice, vein recognition, keystroke dynamics and the angle at which a device is held all qualify; information transmitted using EMV 3-D Secure does not.

The exemptions, with figures, dated

Every figure below is stated as it stood at the time of writing in August 2026, and every one is capable of changing without the underlying mechanism changing at all.

Article	Exemption	EU position (Regulation (EU) 2018/389, as at August 2026)	UK position (FCA technical standards, as at August 2026)
10	Account information accessed directly	Balance and last 90 days of transactions; SCA on first access and at least every 90 days	Same, with Article 10 applying where no account information service provider is involved
10A	Account information accessed via an AISP	No equivalent article	SCA not required provided it has been applied on at least one previous occasion when the AISP accessed the data
11	Contactless at point of sale	EUR 50 single, EUR 150 cumulative, or five consecutive transactions since the last SCA	No monetary limits; exemption available where the transaction is identified by the payment service provider as posing a low level of risk
12	Unattended terminals	Transport fares and parking fees only	Same
13	Trusted beneficiaries	SCA required to create or amend the list; not required for payments to those on it	Same
14	Recurring transactions	SCA on creation, amendment or first initiation of a series of the same amount to the same payee; not on subsequent ones	Same
15	Transfers between a person's own accounts	Where both accounts are at the same account servicing provider	Same
16	Low-value remote payments	EUR 30 single, EUR 100 cumulative, or five consecutive transactions since the last SCA	£25 single, £85 cumulative, or five consecutive transactions since the last SCA
17	Secure corporate processes	Non-consumer payers only, where the competent authority is satisfied as to equivalent security	Same, with the FCA as the satisfied authority
18	Transaction risk analysis	Real-time risk analysis plus a fraud rate at or below the reference rate for the value band	Same mechanism, sterling bands

Three of these deserve expansion.

The contactless exemption is the clearest illustration of why this volume of the book is the perishable one. The European figures of EUR 50 and EUR 150 are as they appear in Regulation (EU) 2018/389

at the time of writing. The United Kingdom diverged progressively, ending with limits of £100 for a single transaction and £300 cumulative, and then abolished the numbers entirely. By the Technical Standards (Strong Customer Authentication and Common and Secure Methods of Communication) (Amendment) Instrument 2025, FCA 2025/62, made under Handbook Notice 136 and announced by the FCA on 19 December 2025, Article 11 was rewritten with effect from 19 March 2026 to remove the monetary caps and replace them with a purely risk-based test: the exemption is available where the payer initiates a contactless transaction identified by the payment service provider as posing a low level of risk. The FCA's announcement noted that most firms were expected to keep their existing limits for the foreseeable future, so the number a British shopper meets in a shop may not have moved even though the rule behind it has been deleted. Any reader consulting this chapter after 2026 should assume the position has moved again and check the FCA's technical standards directly.

The recurring transactions exemption is narrower than the phrase suggests. Article 14 covers a series of transactions of the same amount to the same payee. A subscription that varies in amount — metered usage, a variable utility bill, a plan that changes tier — is not within it. Such payments are usually handled instead as merchant-initiated transactions, which are outside the scope of the obligation altogether rather than exempt from it, provided the mandate itself was established with strong customer authentication. The distinction between exempt and out of scope is not pedantry: exemptions are counted, monitored and can be withdrawn, whereas out-of-scope transactions are simply not the subject of the obligation.

The transaction risk analysis exemption is the commercially important one, and it is a bargain rather than a permission. Article 18 allows a provider to skip authentication on a remote electronic payment it has identified as low risk, but only where the value is at or below an exemption threshold value, and only where that provider's own fraud rate for the relevant transaction type is at or below the reference rate for that band. The rate is calculated under Article 19 as the total value of unauthorised or fraudulent remote transactions divided by the total value of all remote transactions of that type, on a rolling quarterly basis of ninety days, measured across all of that provider's traffic including transactions it authenticated properly.

EU annex (as at August 2026)	Reference fraud rate, remote card	Reference fraud rate, remote credit transfer
ETV EUR 500	0.01%	0.005%
ETV EUR 250	0.06%	0.01%
ETV EUR 100	0.13%	0.015%

UK appendix (as at August 2026)	Reference fraud rate, remote card	Reference fraud rate, remote credit transfer
ETV £440	0.01%	0.005%
ETV £220	0.06%	0.01%
ETV £85	0.13%	0.015%

The sterling figures come from the FCA's instrument FCA 2020/70, which converted the euro amounts at a stated rate of £0.8876 to the euro, and they are corroborated by the FCA's own notification form NOT004 in SUP 15 of the Handbook, which is laid out in the three bands of £440, £220 and £85.

The enforcement mechanism is automatic and is set out in Article 20. A provider whose monitored fraud rate exceeds the applicable reference rate must report that immediately to the competent authority — in the United Kingdom, using form NOT004 — with a description of how it intends to restore

compliance. If the rate exceeds the reference rate for two consecutive quarters, the provider must immediately stop using the exemption in that threshold band, and may not resume until its calculated rate has been at or below the reference rate for a quarter, having notified the authority and evidenced the restoration first. Article 21 separately requires quarterly recording of transaction values, fraud values, resulting fraud rates, average values and the count and percentage of transactions under each exemption, broken down between remote and non-remote.

That is the whole design in one sentence: the reward for good fraud control is less friction, and it is withdrawn on a timetable the firm does not control.

Who claims an exemption and who grants it

There are two independent routes and the distinction determines who carries the loss.

The issuer route is the ordinary one: the merchant sends an AReq, the ACS applies its own risk analysis, decides the transaction is low risk, and returns transStatus Y without a challenge. The issuer has exercised its own exemption, and for liability purposes the transaction is generally treated as authenticated.

The acquirer route runs the other way. The merchant, through its acquirer, marks the authorisation as exempt — commonly under transaction risk analysis, low value, or trusted beneficiary — and performs no authentication at all. This uses the acquirer's fraud rate rather than the issuer's and generally leaves fraud liability with the merchant. It is nonetheless widely used, because a merchant with excellent fraud performance may rationally prefer to keep both the liability and the conversion.

Either way the issuer has the final word, and its instrument is the soft decline: an authorisation response that does not mean "no" but "not like this". Visa's response code 1A, additional customer authentication required, is the one most often seen in Europe, and code 65, whose older meaning was exceeds withdrawal frequency limit, carries the equivalent sense on Mastercard. Both are recorded in that sense throughout the published response-code documentation of acquirers and gateways, Worldpay's and Adyen's among them; the schemes' own authorisation specifications are not public documents, so the mapping cannot be cited to a primary source here. The correct merchant behaviour is to run a 3-D Secure authentication and retry, not to show the customer a failure. Merchants who did not implement that retry loop discovered in the spring of 2022 that they had built a checkout which declined a substantial share of perfectly good British customers and told them their cards had been refused.

Version 2.2 added the fields that make this expressible: an exemption indicator, a challenge indicator, and support for whitelisting so a cardholder can add a merchant to their trusted beneficiary list during the challenge itself. Delegated authentication, where an issuer contracts out the authentication act to a merchant, wallet or gateway meeting its standards, sits alongside rather than inside the protocol.

Liability, precisely

The card schemes' rules provide, in substance, that a merchant obtaining a fully authenticated result is protected against the fraud-type dispute categories: Visa dispute condition 10.4, other fraud, card-absent environment, and Mastercard reason code 4837, no cardholder authorisation, are the two a European merchant meets most often. Attempted authentication, ECI 06 or 01, may also carry protection, but the circumstances are scheme-specific and have narrowed over time. The authoritative statement in every case is the current edition of the relevant scheme's rules, revised at least annually and reaching the merchant only through its acquirer's contract. No book is a substitute for them, and this one is not.

What the shift does not do bears repeating. It does not apply to non-fraud dispute categories, nor where the authentication values were not correctly presented in authorisation, nor where the authenticated and authorised amounts differ. It does not protect against a first-party misuse claim dressed as a fraud claim, because at the level of the dispute message those look identical, as the previous chapter set out.

Underneath the scheme rules, the statutory allocation continues to operate. In the United Kingdom, regulation 75 of the Payment Services Regulations 2017 puts the burden of proving authentication and correct execution on the payment service provider and expressly provides that use of the instrument as recorded by the provider is not in itself necessarily sufficient. Regulation 76 requires refund of an unauthorised transaction by the end of the business day following the day the provider becomes aware of it. Regulation 77 sets the payer's own liability at a maximum of £35 for losses arising from a lost, stolen or misappropriated instrument, disapplies even that where the provider failed to apply strong customer authentication as required, and provides that where the payee or the payee's payment service provider does not accept strong customer authentication, that party must compensate the payer's payment service provider for the resulting loss. All of these provisions are as they stood at the time of writing in August 2026.

The cost of friction, in numbers

Vendor abandonment statistics vary wildly and are rarely reproducible. The supervisory data is better.

The European Banking Authority and the European Central Bank published the 2025 edition of their joint report on payment fraud on 15 December 2025, covering data to the end of 2024. It found that of electronically initiated card payments made with cards issued in the European Economic Area, around 64% of the total value was authenticated with strong customer authentication, while only about 40% of transactions by volume were, the rest travelling under exemptions. Contactless accounted for around 79% of the volume of non-SCA card payments. Within the EEA, the fraud rate on SCA-authenticated card payments was about 0.016% by value, roughly half the rate on non-authenticated ones. Card payment fraud was around seventeen times higher where the counterpart was outside the EEA, where strong customer authentication is not legally required and frequently not applied. Remote card payments carried a fraud rate about thirteen times that of non-remote payments by value, 0.091% against 0.007%. Card fraud on EEA-issued cards totalled EUR 1.329 billion in 2024, within total fraud of about EUR 4.2 billion across all instruments. All of these figures are as published in that report; the next edition will supersede them.

The British picture from UK Finance's Annual Fraud Report 2026, published on 15 June 2026 and covering 2025, is that remote purchase card fraud losses rose 3% to £423.5 million across 3.2 million cases, an increase of 13% in case numbers, within total payment fraud of £1.28 billion. Unauthorised fraud losses overall fell 5% to £703.4 million. UK Finance specifically noted criminals compromising one-time passcodes in order to register digital wallets or make fraudulent transactions.

That last point matters for anyone designing controls, and it can be stated without giving anybody a manual. The weakest widely deployed possession factor is a one-time passcode delivered over a channel the customer can be talked into repeating aloud. The regulation's own logic — dynamic linking, so the customer sees the amount and payee they are approving; independence, so compromising one element does not compromise another; device binding, so possession means a specific device rather than knowledge of a code — exists precisely because social engineering attacks the human rather than the cryptography. The defensive conclusions follow: prefer app-based approval bound to a registered device over a passcode sent as text; display amount and payee in the approval itself; treat enrolment of a new device or wallet as a high-risk event deserving its own authentication rather than an administra-

tive step; and monitor for the pattern where authentication succeeds but the surrounding behaviour does not match the customer.

The commercial arithmetic merchants actually run compares two expected costs. Challenge a cohort and you lose some proportion of it to abandonment, immediately and certainly, at the full margin. Do not challenge and you keep those sales but carry the fraud on whatever fraction turns out to be stolen, plus dispute fees, plus the effect on the dispute ratio, which at sufficient volume attracts scheme monitoring programmes and their own charges. On a £420 bicycle at a 12% margin the merchant makes £50.40, while a single fraudulent transaction costs the £420, the goods and a dispute fee: the merchant can therefore afford to challenge and lose roughly eight sales for every fraudulent transaction prevented. On a £6 digital item at a 90% margin the arithmetic inverts completely, which is why the low-value exemption exists and why nobody is asked to fingerprint a newspaper subscription.

What is going to change, and when

This is the chapter most likely to be wrong first, so it should say plainly where the ground is moving.

In the European Union, the second Payment Services Directive is being replaced by a package comprising a third directive, PSD3, and a directly applicable Payment Services Regulation. Parliament and Council reached provisional political agreement on 27 November 2025, COREPER endorsed the trilogue texts on 22 April 2026, and final compromise texts were published by the Council on 23 April 2026. At the time of writing in August 2026 the author has not verified formal adoption by both institutions or publication in the Official Journal, so the application date should be treated as unsettled. The authentication changes in the agreed texts include an explicit right of access to strong customer authentication for customers without smartphones, with disabilities or with low digital skills; clarified treatment of merchant-initiated transactions, requiring authentication at mandate set-up rather than on each subsequent payment; explicit classification of delegated authentication as outsourcing, with the delegating provider retaining liability; and new technical standards from the European Banking Authority governing exemptions and transaction risk analysis. Reports of the final text also indicate a redefinition permitting two elements from the same category in the specific case of inherence; verify that against the adopted text rather than against this paragraph.

In the United Kingdom, the technical standards themselves are on borrowed time. Legislation.gov.uk records the United Kingdom's version of Regulation (EU) 2018/389 as subject to revocation by Schedule 1 Part 3 of the Financial Services and Markets Act 2023, listed as a change not yet applied as at 17 August 2026, with firm-facing requirements in assimilated law expected to move into FCA rules. When that happens the article numbers used throughout this chapter will cease to exist even where the substance survives. The contactless amendment of March 2026 previews the pattern: a numerical rule replaced by a risk-based standard, the supervisory burden shifting from compliance with a threshold to demonstrating the quality of a firm's own monitoring.

And in the protocol, EMVCo's draft version 2.4.0.0 was out for public comment until 1 July 2026. Version 2.3's direction of travel — FIDO and WebAuthn data carried inside the authentication message, Secure Payment Confirmation, device binding, automated transitions to and from the banking app — points at where this ends: authentication that is cryptographic, bound to a device, invisible in the common case, and no longer recognisable to the customer as a separate step at all. The regulation will still be counting to two.

47.98 Common wrong ideas

Wrong: A successful authentication means the payment will go through. **Right:** Authentication and authorisation are separate messages on separate networks with separate answers, and a cardholder can authenticate perfectly and be declined moments later for funds, velocity or a fraud engine that reached a different conclusion.

Wrong: Three-D Secure is strong customer authentication. **Right:** One is a messaging protocol published by EMVCo and the other is a legal obligation on payment service providers; the European Banking Authority stated in June 2019 that data carried over EMV 3-D Secure is not itself an inherence element, because the protocol carries no biometric measurement.

Wrong: The liability shift is law. **Right:** It lives in the card schemes' operating rules, which are commercial documents revised at least annually and reaching a merchant only through its acquirer's contract, while the statutory allocation runs underneath it and does not care what the rulebook says.

Wrong: The merchant can decide to skip the check. **Right:** A merchant can flag a transaction as one it believes qualifies and express a preference, but the issuer decides, and its instrument for disagreeing is the soft decline.

Wrong: A high frictionless rate is a good number on its own. **Right:** Two identical-looking frictionless transactions can carry opposite liability depending on whether the acquirer claimed the exemption or the issuer granted it, so the rate means nothing without that split.

Wrong: A soft decline means the card was refused and the customer should be told so. **Right:** Codes such as Visa's 1A mean "not like this" rather than "no", and the correct behaviour is to run an authentication and retry, which is precisely what merchants who lost good British customers in spring 2022 had failed to build.

Wrong: A one-time passcode is something the customer knows, so it is a knowledge element. **Right:** The European Banking Authority's opinion is that it is not, because it is not something the user knows in advance, and an app is not evidence of possession unless it has been bound to the device.

Wrong: The recurring exemption covers subscriptions. **Right:** Article 14 covers a series of the same amount to the same payee, so a metered or tier-changing subscription is handled as a merchant-initiated transaction, which is out of scope rather than exempt, and the difference matters because exemptions are counted, monitored and withdrawable.

Wrong: The authentication result can be read from the browser when the challenge finishes. **Right:** The authoritative result is the Results Request sent by the access control server through the directory to the 3DS Server; reading it from the client is trusting the client.

Wrong: Strong customer authentication arrived in British e-commerce in September 2019. **Right:** The Financial Conduct Authority granted successive forbearance and set full compliance for card-not-present e-commerce at 14 March 2022, and that is when British checkouts actually changed.

47.99 Chapter summary in 20 lines

1. Three-D Secure exists because the money wire carries one-line demands and one-word answers, not questions about who is buying, so the industry built a second wire for the question.
2. The three domains are the acquirer's, the issuer's and the interoperability domain between them, and the name has nothing whatever to do with three dimensions.

3. The components are the 3DS Server acting for the merchant, the Access Control Server holding the issuer's authentication logic, and the scheme's Directory Server acting as switchboard.
4. The authentication request carries context — device, browser, addresses, account age, purchase type, merchant data — which is what allows an issuer to decide without asking the cardholder anything.
5. A frictionless flow is not the absence of a check; it is a check the cardholder was not asked to participate in, and on a good implementation it covers the large majority of purchases.
6. The challenge flow reaches the cardholder through a channel the issuer already trusts, and out-of-band approval in the banking app is the pattern almost all European issuers use.
7. Authentication and authorisation are two messages on two networks, and the two systems inside one issuer are frequently different products that do not always agree.
8. The field that matters is `transStatus`: Y or A ends the exchange, C begins the challenge, and the Results Request rather than the browser carries the authoritative outcome.
9. Authentication produces an authentication value, an e-commerce indicator and a directory transaction identifier, and a merchant that fails to carry all three into the authorisation has paid the friction and bought nothing.
10. Strong customer authentication is a legal obligation defined in regulation, while 3-D Secure is a delivery van; the van is not the cargo, and in jurisdictions with no authentication law the protocol is used anyway purely for the liability shift.
11. The rule is a counting rule: at least two elements from different families, independent of one another, which is why a registered phone plus a fingerprint on it became the British default.
12. Dynamic linking welds the resulting code to a specific amount and payee, so an approval obtained for one thing cannot pay for another.
13. Checking costs customers rather than money, so the regulation cuts deliberate holes in itself: low value, recurring, trusted beneficiaries, contactless, unattended terminals, corporate processes and transaction risk analysis.
14. Transaction risk analysis is a bargain rather than a permission, because the reward of less friction is withdrawn automatically once the firm's own fraud rate exceeds the reference rate for two consecutive quarters.
15. The United Kingdom's figures diverged and then partly vanished: £25 low value, sterling bands of £440, £220 and £85, and Article 11's contactless caps deleted with effect from 19 March 2026 in favour of a risk-based test.
16. The merchant requests an exemption and the issuer grants it, and the issuer's way of disagreeing is a soft decline that means "not like this" rather than "no".
17. An acquirer-claimed exemption generally leaves fraud liability with the merchant while an issuer-granted one generally does not, so two identical frictionless transactions can end in opposite places.
18. The liability shift covers fraud dispute categories only, dies where the authentication values were mispresented or the amounts differ, and cannot separate a first-party misuse claim dressed as a fraud claim.
19. The supervisory data shows authentication working where it is applied and absent where it is not, with about 0.016 per cent fraud by value on authenticated card payments and roughly seventeen times more fraud where the counterpart sits outside the European Economic Area.
20. The direction of travel is cryptographic, device-bound authentication the customer stops noticing, with PSD3 and the migration of the United Kingdom's technical standards into FCA rules about to renumber almost every citation in this chapter.

Sources: Regulation (EU) 2018/389 and Directive (EU) 2015/2366 via EUR-Lex and legislation.gov.uk; the FCA Handbook Technical Standards on Strong Customer Authentication, instruments FCA 2020/70 and FCA

2025/62, Handbook Notice 136, SUP 15, and FCA statements of 19 December 2025 and on the 14 March 2022 e-commerce deadline; the Payment Services Regulations 2017; EBA Opinion EBA-Op-2019-06; the joint EBA-ECB Report on Payment Fraud of 15 December 2025; UK Finance's Annual Fraud Report 2026; EMVCo 3-D Secure materials; and Visa's notice discontinuing 3-D Secure 1.0.2.

48

CHAPTER

PCI DSS and Its Relatives

48.0 What this chapter gives you

1. You will be able to say who writes PCI DSS, who enforces it and why the Council that owns the standard cannot fine anybody for breaking it.
2. You will be able to explain why your validation level follows transaction volume while the work you must actually do follows how close the card number gets to your own computers.
3. You will be able to explain why the same coffee chain answers 27 requirements or 139 depending on a decision a developer made in an afternoon.
4. You will be able to describe what point-to-point encryption and tokenisation do to scope, and name the conditions that silently forfeit the reduction.
5. You will be able to state the storage rules exactly: sensitive authentication data never retained after authorisation even encrypted, the account number rendered unreadable, display limited to the leading digits and the last four.
6. You will be able to say what an Attestation of Compliance proves — a date, a version, a scope and a sample — and what it therefore cannot tell a buyer.
7. You will be able to describe requirements 6.4.3 and 11.6.1, explain the browser-side attack they exist for, and say what the January 2025 revision of SAQ A did to them.
8. You will be able to state the version position as it stood in August 2026: v4.0.1 the only active version, every future-dated requirement effective since 31 March 2025, and a Request for Comments run from 3 June to 20 July 2026.
9. You will be able to place PCI DSS alongside the UK GDPR and explain how an organisation can be compliant with one and in breach of the other on the same day.
10. You will be able to tell a supplier questionnaire that Direct Debit data creates no cardholder data environment, because the account number is the defining factor and there is no account number.

The plain version

Imagine a card number is a key to a house with the address stamped on it. Anyone who copies the key can walk in. The person who loses the key is not the person who gets burgled, and the person who gets burgled is not the person who pays for the new locks. That gap between who is careless and who suffers is the whole reason the rules in this chapter exist.

Six card companies sit at the centre of the world's card payments: American Express, Discover, JCB, Mastercard, UnionPay and Visa. In the early 2000s each wrote its own list of safe-handling rules for card numbers, so a shop taking several brands had to satisfy several overlapping lists. In 2006 they gave up on that and set up a joint body to write one list instead. It is called the PCI Security Standards Council, and the list it maintains is the Payment Card Industry Data Security Standard, or PCI DSS.

Here is the part almost everybody gets wrong. The Council that writes the rules cannot punish you for breaking them. It is not a government; it has no inspectors, no courts and no power to fine anybody, and it is closer to the committee that writes the specification for a plug socket. What gives PCI DSS teeth is that when your shop signed a contract with the bank that pays you — the acquirer — that contract said you would follow PCI DSS. Break it and the person who comes after you is your bank, holding your own signature.

So how much of the rulebook applies to you? Not, as you might expect, according to how big you are. Size decides how you have to *prove* it. What you actually have to *do* depends on something much less obvious: how close the card number gets to your own computers.

Take a real-shaped example. A coffee chain has forty shops and a website, and puts 2.4 million Visa card payments through in a year. Under Visa's published programme at the time of writing, a merchant doing between 1 million and 6 million Visa transactions a year across all channels is a Level 2 merchant, which means it fills in a self-assessment questionnaire once a year rather than paying an outside assessor to write a full report. Good news, so far.

Now look at the website. The Council publishes different questionnaires for different shapes of business, and the one you may use depends on where the payment form comes from. The two numbers below are the Council's own, published in the April 2025 revision of its questionnaire guidance and accurate at time of writing.

If the checkout page drops in a little window supplied whole by the payment provider — an iframe, or a redirect off to the provider's own page — then every part of the page that touches the card number came from the provider, not the coffee chain. The chain uses the shortest questionnaire, SAQ A, and answers 27 requirements. If instead the chain's own website builds the payment form, so the card number is typed into a box the chain's own code drew on the screen before being sent on, the chain must use a different questionnaire, SAQ A-EP, and answer 139.

Same company. Same turnover. Same forty shops. Twenty-seven items or 139, decided years earlier by a developer choosing how to build a checkout page, probably in an afternoon, probably without knowing that was the decision being made.

The shops themselves can be shrunk the same way, and there are only two tricks that genuinely work.

The first is point-to-point encryption, and it is a locked post box. A card reader approved for it scrambles the card number inside the reader itself, in hardware, the instant the card is read, using a key the shop does not have and cannot get. What travels through the shop's tills, network and broadband is unreadable, unscrambled only much later inside the payment provider's own protected building. The shop is carrying sealed post it cannot open. Break into the till and you get nothing worth stealing, and the questionnaire the shop fills in gets dramatically shorter.

The second is tokenisation, and it is a cloakroom ticket. When a customer saves a card for next time, the shop keeps not the card number but a meaningless reference — say `tok_8813f2` — that only the payment provider can turn back into a real number, and which the shop can hand back to charge the customer again next month. A thief who steals the shop's entire customer database steals a pile of tickets for a cloakroom they cannot enter.

What is in the rulebook is less exotic than people expect: twelve numbered sections under six goals, mostly the security housekeeping any competent organisation should be doing anyway. Know what computers you have, change the default passwords, encrypt card numbers at rest and in transit, patch things, restrict access, keep logs and read them, test yourself, write it down. Three specifics are unusu-

ally concrete, and these are the figures in force at the time of writing: passwords in scope must be at least twelve characters, or eight if the system genuinely cannot manage twelve; external vulnerability scans must be run at least once every three months by a scanning company the Council has approved; and patches for critical vulnerabilities go on within one month of release.

And when it goes wrong, the money moves in a direction most people do not expect. The card schemes do not bill the breached shop. They assess the shop's acquiring bank, and the bank then comes after the shop under the contract; Visa's published rules go further and forbid the bank from telling the shop that Visa imposed the charge on it. Separately and quite independently, the Information Commissioner's Office can fine a UK organisation for losing personal data up to £17.5 million or four per cent of total annual worldwide turnover, whichever is higher — the statutory maximum in force at the time of writing. Card rules and data protection law are two different animals arriving from two directions on the same bad day.

Where the plain version stops being true

"Certified" is the wrong word, and the annual pass is not a state of being. Nothing in PCI DSS produces a certificate. What it produces is an Attestation of Compliance, a signed declaration about a single assessment covering a defined scope on a defined date. It is a photograph, not a passport. The standard's own position, and the schemes', is that compliance is a continuous obligation: Visa's rules state that a service provider and merchant must always maintain full compliance, and the standard bakes this in by requiring service providers to review at least once every three months that staff are actually performing security tasks as documented, and to reconfirm their own scope at least once every six months. An organisation can hold a perfectly valid attestation dated four months ago and be flagrantly non-compliant today, and frequently is. Passing the test is not the obligation; meeting the standard every day is, and the test is only evidence.

The club does not fine you, and nobody publishes the tariff. The members'-club analogy is wrong in two ways that matter commercially. First, the direction: the schemes assess the acquirer or issuer, who then recovers under contract, which is why "the PCI fine" is a phrase used by people who have never seen the paperwork. Second, the amounts. Circulating figures for per-card penalties and monthly non-compliance charges are not published by the Council, which does not set them, and the schemes do not put numbers on them either — the Council states plainly that any fines or penalties are a matter for each participating payment brand's own programme. Anyone quoting a precise penalty schedule is quoting marketing, a leaked contract, or an invention. One published nuance is revealing, though: Visa states that assessments may be waived where a forensic investigation demonstrates there was no evidence of PCI DSS non-compliance prior to and at the time of the breach. The attestation is not what protects you. The evidence trail behind it is.

Tokenisation and point-to-point encryption reduce scope. They do not remove you from the standard, and the reduction is conditional. This is where the cloakroom-ticket and locked-post-box analogies mislead most expensively. A merchant using a validated, PCI-listed P2PE solution still fills in a questionnaire and still has requirements to meet, and eligibility for the short version depends on conditions the merchant can silently fail: all payment processing must run through the validated solution, no account data may be received or stored electronically by any other route, retained data must be on paper not received electronically, and — the one that catches people — the merchant must have implemented all the controls in the P2PE Instruction Manual supplied by the solution provider. That manual is a set of obligations, not a leaflet. Listings also expire, and the Council is explicit that solutions on its list of expired validations are no longer considered validated. Scope reduction is a thing you maintain, not a thing you buy. Tokenisation has the mirror-image trap: the tokens may be

harmless, but the vault mapping them back to card numbers is squarely in scope for somebody, and if the merchant can retrieve the original number from the token, the merchant has escaped nothing.

You do not choose your questionnaire, and PCI DSS does not cover fraud. The Council writes the questionnaires but does not decide who may use which. That decision belongs to what the Council calls the compliance-accepting entity — in practice the acquirer, the payment facilitator or the brand. The confusion is live enough that the Council issued a bulletin on 4 August 2026 revising its FAQ 1331 specifically to clarify that merchants should always consult their compliance-accepting entities to confirm validation and reporting requirements, because an earlier May 2025 version of that guidance was being misread. Separately, the whole standard concerns the confidentiality of stored and transmitted account data. It is not about fraud. It will not tell you whether the person holding the card is the cardholder, it does not govern chargebacks, and it does not deliver strong customer authentication. Those live elsewhere, including in 3-D Secure and the regulatory authentication regime, treated below as relatives rather than parts of the same thing.

The technical version

What PCI DSS is, and who owns it

The Payment Card Industry Data Security Standard is a private technical standard owned and maintained by the PCI Security Standards Council, LLC, based in Wakefield, Massachusetts. The Council was formed in 2006 and is led by a policy-setting Executive Committee drawn from its Founding Members and Strategic Members: American Express, Discover Financial Services, JCB International, Mastercard, UnionPay and Visa Inc. Participating Organisation membership is open to merchants, banks, processors, hardware and software developers and point-of-sale vendors, who gain access to Request for Comments periods and drafts.

The division of labour, in the Council's own words, is the single most load-bearing statement in the subject. The Council is responsible for developing and managing the PCI Security Standards and the related qualification and listing programmes; each participating payment brand maintains its own separate compliance enforcement programme, including which entities must validate compliance, at what validation level, whether an entity may complete a Self-Assessment Questionnaire or must complete a Report on Compliance, and any fines or penalties. Visa states the same split from the other side: the Council owns, maintains and manages PCI DSS and its supporting documents, while Visa manages all data security compliance enforcement and validation initiatives.

PCI DSS is therefore not legislation anywhere. It reaches you through a chain of private contracts: scheme rules bind the acquirer, the acquirer's merchant agreement binds the merchant, and the merchant's contracts bind its suppliers. That chain is not weaker than a statute in practice — it can terminate your ability to accept cards, worse for most businesses than a fine — but it is legally a different creature, and confusing the two produces bad advice in both directions.

The enforcement chain, concretely

Visa's programme is Account Information Security, and the governing instrument is the Visa Core Rules and Visa Product and Service Rules. Visa's published summary, accurate at time of writing, states that issuers and acquirers are responsible for ensuring the PCI DSS compliance of their service providers and merchants, including service providers the merchant is using, and that a service provider and merchant must always maintain full compliance, citing Visa Core Rules section IDs 0002228 and 0008031. Where a service provider or merchant does not comply or fails to rectify a security issue, Visa may assess a non-compliance assessment to the issuer or acquirer, who is responsible

for paying all assessments and must not represent that Visa has imposed any assessment on the service provider or merchant, citing section ID 0001054. Visa also states that assessments may be waived where there is no evidence of PCI DSS non-compliance prior to and at the time of a data breach, as demonstrated during a forensic investigation.

Mastercard's programme is Site Data Protection. Its published process, at time of writing, has the merchant determine its level using Mastercard transaction volume from the most recent 52-week period, confirm the applicable validation requirement, engage an approved assessor where required, and submit validation documents to the acquirer, who submits an SDP Acquirer Submission and Compliance Status Form semi-annually reporting progress for Level 1 and Level 2 merchants. Level 3 and Level 4 merchants need not validate compliance to Mastercard, but the acquirer must validate to Mastercard that it has a risk management programme identifying and managing payment security risk within those portfolios. Those merchants are still required to comply; only validation is excused.

Visa also operates the Technology Innovation Programme, which at time of writing eliminates the requirement to validate PCI DSS compliance for eligible merchants where at least 75 per cent of yearly transactions originate through EMV chip-enabled terminals, a validated point-to-point encryption solution, or an integrated tokenisation solution meeting the EMVCo tokenisation specification. Note the precise scope of that relief: it removes the obligation to *validate*, not to *comply*.

Validation levels

Levels determine how you prove compliance, not what you must do. The two largest schemes' published criteria, accurate at time of writing, are below. They are not identical, and a merchant that is Level 2 under one may be Level 1 under another because each scheme also picks up the other's Level 1 designations.

Scheme	Level	Criteria as published at time of writing	Validation
Visa	1	Over 6 million Visa transactions annually across all channels, or global merchants identified as Level 1 by any Visa region	Annual ROC by a QSA, or by an internal resource if signed by an officer of the company, plus an AOC
Visa	2	1 to 6 million Visa transactions annually across all channels	Annual SAQ
Visa	3	Fewer than 1 million Visa e-commerce transactions annually across all channels	Annual SAQ
Mastercard	1	More than 6 million combined Mastercard and Maestro transactions annually; or meeting Visa's Level 1 criteria; or designated by Mastercard at its sole discretion	Annual ROC, with the AOC signed by a QSA, a certified ISA, or, unless prohibited by law, an executive officer of the merchant

Scheme	Level	Criteria as published at time of writing	Validation
Mastercard	2	More than 1 million but not more than 6 million combined Mastercard and Maestro transactions annually, or meeting Visa's Level 2 criteria	Annual SAQ; where the SAQ is A, A-EP or D, a QSA or certified ISA must also be engaged for validation
Mastercard	3	More than 20,000 but not more than 1 million combined Mastercard and Maestro e-commerce transactions annually, or meeting Visa's Level 3 criteria	Annual SAQ
Mastercard	4	All other merchants	Annual SAQ; validation to Mastercard not required

Visa notes that level identification is based on the corporate entity's total volume of Visa transactions, inclusive of credit, debit and prepaid, in one country or with one acquirer per year, and that volume from independently owned and operated locations such as franchisees may be excluded if the corporate entity does not process it.

Service providers are treated more strictly. Visa's published criteria at time of writing put VisaNet processors and any service provider storing, processing or transmitting over 300,000 Visa transactions annually at Level 1, requiring an annual on-site assessment and an AOC signed by both the service provider and the QSA; those below 300,000 are Level 2 and submit a signed SAQ D or an AOC including QSA signature, and QSA validation is required before any service provider can be listed on the Visa Global Registry of Service Providers. Mastercard's service provider levels are defined by role rather than volume: at time of writing all Third-Party Processors, Staged Digital Wallet Operators, Digital Activity Service Providers, Business Payment Service Providers, Token Service Providers, 3-D Secure Service Providers, Installment Service Providers and Merchant Payment Gateways are Level 1, together with AML and sanctions service providers, Data Storage Entities and Payment Facilitators above 300,000 combined Mastercard and Maestro transactions annually; those below that threshold, plus Terminal Servicers, are Level 2. Mastercard also recommends at time of writing that Level 1 and Level 2 service providers demonstrate compliance with the Designated Entities Supplemental Validation appendix. For any service provider eligible to self-assess, SAQ D for Service Providers is the only questionnaire; there is no short form.

Scope: account data, the CDE, and where the boundary is drawn

PCI DSS applies to all entities that store, process or transmit cardholder data or sensitive authentication data, or that could impact the security of either. Cardholder data comprises the Primary Account Number, cardholder name, expiry date and service code. Sensitive authentication data comprises full track data — magnetic stripe data or the equivalent on a chip — the card verification code, and PINs and PIN blocks. Together these are account data. The PAN is the defining factor: the Council's position is that if an entity stores, processes or transmits PAN, a cardholder data environment exists and PCI DSS requirements apply to it. The storage rules that follow are unforgiving, and are worth stating exactly as they stand at time of writing.

Element	Storage	Must be rendered unreadable when stored
PAN	Kept to a minimum per Requirement 3.2	Yes, per Requirement 3.5
Cardholder name, service code, expiry date	Kept to a minimum per Requirement 3.2 where in the same environment as PAN	No
Full track data, card verification code, PIN and PIN block	Cannot be stored after authorisation per Requirement 3.3.1	Data held until authorisation completes must be protected with strong cryptography per Requirement 3.3.2

Requirement 3.3.1 as published states that SAD is not stored after authorisation, even if encrypted, and that all SAD received is rendered unrecoverable upon completion of the authorisation process; the narrow exception for issuers and companies supporting issuing services is defined separately at Requirement 3.3.3. Requirement 3.4.1 as published states that PAN is masked when displayed, with the BIN and last four digits the maximum number of digits to be displayed, such that only personnel with a legitimate business need can see more.

The cardholder data environment is not only the systems that touch account data. As the Council defines it, the CDE comprises system components, people and processes that store, process or transmit account data, plus components that do not but have unrestricted connectivity to those that do; and PCI DSS additionally applies to components, people and processes that could impact the security of account data. That third category is where most scoping arguments happen, because it pulls in authentication servers, remote access infrastructure, logging servers, jump boxes, monitoring agents and increasingly the entire identity plane.

Scope reduction by segmentation is permitted, and the Council's test is strict: to be out of scope, a system component must be isolated from the CDE such that it could not impact the security of the CDE even if it were compromised. Requirement 12.5.2 obliges the assessed entity — not its assessor — to confirm scope accuracy at least once every 12 months, identifying all locations and flows of account data and all systems connected to or capable of impacting the CDE, and to retain the documentation; Requirement 12.5.2.1 tightens this for service providers to at least once every six months and upon significant change. Where segmentation is relied upon, Requirement 11.4.5 requires penetration testing of the segmentation controls at least once every 12 months and after any changes, and Requirement 11.4.6 requires it at least once every six months for service providers.

Tokenisation and P2PE as scope-reduction mechanisms

Tokenisation replaces the PAN with a surrogate value, and its scope effect turns on one question: can the entity holding tokens retrieve the PAN? If it can, the token is a key to cardholder data and the retrieval path is in scope. If it cannot, the token is not cardholder data and the systems holding it may fall outside the CDE, subject as always to whether they could impact the CDE's security by other routes. The vault performing the mapping is in scope for whoever operates it, in practice usually a service provider assessed at Level 1. Where the tokens are EMV payment tokens issued under the EMVCo Payment Tokenisation Specification Technical Framework, the issuing entity is covered by a separate PCI standard, the Token Service Provider Standard, and Mastercard classes all TSPs as Level 1 service providers at time of writing.

Point-to-point encryption is a more absolute mechanism, because it removes cleartext PAN from the merchant environment rather than substituting for it. The governing standard is PCI P2PE, current

at time of writing at v3.2 published June 2025, supported by the P2PE Program Guide v3.2 dated July 2026. A P2PE solution encrypts account data within a PCI-approved point-of-interaction device using keys the merchant never holds, and decrypts it only within the solution provider's or component provider's validated environment. The merchant's own systems carry ciphertext they cannot open.

The reporting consequence is SAQ P2PE, whose eligibility criteria as published at time of writing require that all payment processing is via a validated PCI-listed P2PE solution; that the only systems in the merchant environment storing, processing or transmitting account data are payment terminals from that solution; that the merchant does not otherwise receive, transmit or store account data electronically; that any retained account data is on paper not received electronically; and that the merchant has implemented all controls in the P2PE Instruction Manual provided by the solution provider. It applies to neither e-commerce channels nor service providers. The Council's footnote on validity matters as much as the criteria: solutions on the PCI list with expired validations are no longer considered validated per the P2PE Program Guide, and a merchant using one should ask its acquirer whether the questionnaire is still acceptable.

Encryption solutions that are not PCI-listed are a much weaker position. The Council published Assessment Guidance for Non-listed Encryption Solutions in June 2020 for exactly this case; such solutions confer no SAQ P2PE eligibility, and any scope reduction is a matter for the acquirer's judgement on the evidence, not an entitlement.

E-commerce scope: the SAQ A / SAQ A-EP boundary

For card-not-present channels the decisive question is where each element of the payment page originates. SAQ A is for card-not-present merchants — e-commerce or mail and telephone order — that completely outsource all account data functions to PCI DSS validated and compliant third parties, with no electronic storage, processing or transmission of account data on their systems or premises, and any retained account data on paper not received electronically. SAQ A-EP is for e-commerce merchants that partially outsource, with a website that does not itself receive account data but does affect the security of the transaction or the integrity of the page accepting the customer's account data.

The Council's own mapping, in the questionnaire guidance revision of April 2025 and accurate at time of writing:

E-commerce method	Questionnaire	Applicable PCI DSS v4.x requirements
Fully outsourced, merchant has no access to its own webpage	SAQ A	14
Fully outsourced, merchant webpage redirects customers to a compliant TPSP, for example a URL redirect	SAQ A	27
Fully outsourced, merchant webpage includes a compliant TPSP's embedded payment page or form, for example an iframe	SAQ A	27

E-commerce method	Questionnaire	Applicable PCI DSS v4.x requirements
Merchant website creates the payment form and payment data goes directly from the consumer browser to the TPSP, often called a Direct Post; or the merchant website loads or delivers scripts running in the consumer browser that support creation of the payment page or how data is transmitted	SAQ A-EP	139
All other e-commerce methods and implementations	SAQ D for Merchants	All requirements

The Council's rule of thumb is blunt: if any element of a payment page delivered to consumers' browsers originates from the merchant's own website, SAQ A does not apply.

The complete questionnaire set at time of writing is SAQ A, SAQ A-EP, SAQ B, SAQ B-IP, SAQ C, SAQ C-VT, SAQ P2PE, SAQ SPoC and SAQ D, the last in separate merchant and service provider forms. The card-present ones divide by connectivity: SAQ B for imprint machines or standalone dial-out terminals with no internet connection, SAQ B-IP for standalone PCI-listed approved PTS point-of-interaction devices with an IP connection to the processor and isolated from all other system types, SAQ C-VT for manual entry into a validated third-party virtual terminal from an isolated device, SAQ C for payment application systems connected to the internet at a single location, and SAQ SPoC for merchants using a commercial off-the-shelf mobile device with a secure card reader from the Council's list of validated SPoC solutions.

The SAQ A revision of January 2025

This episode is the clearest illustration of how the standard and the reporting tools can diverge, and it is dated precisely.

PCI DSS v4.x introduced two requirements aimed squarely at e-commerce script attacks — the pattern in which an attacker modifies or adds a script that a checkout page loads into the customer's browser, so that card details are copied as they are typed, before any of the merchant's own encryption applies. The merchant's servers are never touched and its logs show nothing wrong, which is precisely why the controls exist. Requirement 6.4.3 as published requires that all payment page scripts loaded and executed in the consumer's browser are managed so that a method confirms each script is authorised, a method assures the integrity of each script, and an inventory of all scripts is maintained with written business or technical justification for each. Requirement 11.6.1 as published requires a change- and tamper-detection mechanism to alert personnel to unauthorised modification — including indicators of compromise, changes, additions and deletions — to the security-impacting HTTP headers and script contents of payment pages as received by the consumer browser, evaluated at least weekly or at a frequency set by the entity's targeted risk analysis under Requirement 12.3.1.

On 30 January 2025 the Council announced that, in response to stakeholder feedback about the complexity of implementing these controls, Requirements 6.4.3 and 11.6.1 were being removed from SAQ A, along with Requirement 12.3.1 in so far as it supported 11.6.1. In their place SAQ A gained an eligibility criterion requiring the merchant to confirm that its site is not susceptible to attacks from scripts that could affect the merchant's e-commerce systems. Two versions of SAQ A were live at once for a period: the October 2024 version was retired on 31 March 2025, and the January 2025 version

took effect the same day, which was also the day the future-dated requirements became effective. Requirements 6.4.3, 11.6.1 and 12.3.1 themselves became effective on 31 March 2025 and remain in the standard, continuing to apply to merchants reporting on SAQ D and to merchants assessed on-site with a Report on Compliance. All those dates are as published and accurate at time of writing.

Read the substitution carefully: the reporting burden moved, not the security problem. A merchant attesting to the new eligibility criterion has asserted something about its own susceptibility to script attacks with no prescribed method of verifying it.

The version in force, and the migration position, dated

As at the time of writing, August 2026, the position is as follows, and every element of it is dated. PCI DSS v4.0 was published in March 2022. PCI DSS v3.2.1 was retired on 31 March 2024. PCI DSS v4.0.1, a limited revision containing corrections and clarifications and, in the Council's words, no additional or deleted requirements, was published in June 2024. PCI DSS v4.0 was retired on 31 December 2024, after which v4.0.1 became the only active version supported by the Council, and the Council's document library still lists it as the current standard at time of writing.

Version 4.x introduced 64 new requirements, of which 51 were future-dated. Future-dated requirements are treated as best practice until the specified date and are not validated before it; once reached, they become effective and applicable. That date was 31 March 2025, it was not changed by the v4.0.1 revision, and it has passed. There is therefore no remaining phase-in: at the time of writing every requirement of PCI DSS v4.0.1 is fully in force. The requirements below carried the "best practice until 31 March 2025" designation and are now effective; the wording is as published in v4.0.1 and accurate at time of writing.

Requirement	Substance
3.4.2	Technical controls prevent copy or relocation of PAN over remote-access technologies, except for documented, explicitly authorised personnel with a legitimate business need
3.5.1.2	Disk- or partition-level encryption renders PAN unreadable only on removable media, or on non-removable media only where PAN is also rendered unreadable by another mechanism meeting 3.5.1
4.2.1.1	An inventory of trusted keys and certificates protecting PAN in transmission is maintained
5.4.1	Processes and automated mechanisms detect and protect personnel against phishing attacks
6.4.2	Public-facing web applications are fronted by an automated technical solution continually detecting and preventing web-based attacks, actively running, up to date, generating audit logs, and configured to block or alert for immediate investigation
6.4.3	Payment page scripts authorised, integrity-assured and inventoried with justification
8.3.6	Passwords used as authentication factors are at least 12 characters, or eight where the system cannot support 12, and contain both numeric and alphabetic characters

Requirement	Substance
8.4.2	MFA is implemented for all non-console access into the CDE
8.6.2	Passwords for application and system accounts usable for interactive login are not hard-coded in scripts, configuration files or source code
10.4.1.1	Automated mechanisms are used to perform audit log reviews
11.5.1.1	Service providers only: intrusion detection or prevention techniques detect, alert on or prevent, and address covert malware communication channels
11.6.1	Change- and tamper-detection mechanism for payment page headers and scripts, at least weekly or per targeted risk analysis
A1.1.1, A1.1.4	Multi-tenant service providers: logical separation such that neither provider nor customer can reach the other's environment without authorisation, confirmed at least once every six months by penetration testing

Others in the same set cover encryption of SAD held before authorisation completes, keyed hashing of the full PAN, software and component inventories, targeted-risk-analysis frequencies for malware scanning and log review, management of lower-ranked vulnerabilities, awareness training on acceptable use of end-user technologies, and multi-tenant support for customer penetration testing.

The forward-looking position, also dated, is that the Council has begun the next iteration. From 3 June to 20 July 2026 it ran a six-week Request for Comments on the currently published v4.0.1, expressly to gather feedback shaping the standard's evolution and specifically inviting comment on opportunities supporting future technology and artificial intelligence innovation. As at the time of writing no successor version has been published and no publication date has been announced. Anything predicting the content or timing of a next version is speculation; the Council's pattern with v4.0 was a multi-year development cycle, then a transition period with two versions simultaneously active, then a separate phase-in for future-dated requirements, but it has not committed to repeating that shape.

The relatives

PCI DSS is one member of a family; the others govern devices, keys, software, mobile acceptance, card manufacture and authentication. The versions below are those current in the Council's document library at time of writing.

Standard	Current version at time of writing	Governs
PCI DSS	v4.0.1, June 2024	Environments where account data is stored, processed or transmitted
PCI PTS POI Modular Security Requirements	v7.0, May 2025	Point-of-interaction devices: PIN terminals, POS devices, encrypting PIN pads, unattended payment terminals
PCI PTS HSM Modular Security Requirements	v5.0, May 2026	Hardware security modules across their lifecycle

Standard	Current version at time of writing	Governs
PCI PIN Security Requirements and Testing Procedures	v3.1, March 2021	Management, processing and transmission of PIN data at ATMs and attended and unattended POS
PCI P2PE	v3.2, June 2025	P2PE solutions, components and applications
PCI 3DS Core Security Standard	v1.0, October 2017	Environments where specified 3-D Secure functions are performed
PCI 3DS SDK Security Standard	v1.1, December 2018	3-D Secure software development kits
PCI MPoC	v1.1, November 2024	Mobile payment acceptance on commercial off-the-shelf devices
PCI Secure Software and Secure SLC	Software Security Framework	Secure design and maintenance of payment software
PCI Card Production and Provisioning, Logical and Physical	Two complementary standards	Manufacture, transport and personalisation of payment cards
PCI Token Service Provider Standard	Current	TSPs issuing EMV payment tokens

Three sunsets are in progress at time of writing, and they matter because they change what a supplier can validly claim. The Council announced a formal sunset period of 1 May to 31 October 2026 for the PCI 3DS Software Development Kit Standard, and the same window for both the Software-based PIN Entry on COTS and the Contactless Payments on COTS Standards, the latter two superseded in substance by MPoC. The Payment Application Data Security Standard was retired earlier, on 28 October 2022, superseded by the Secure Software and Secure Software Lifecycle Standards; a vendor still advertising PA-DSS validation is advertising something that has not existed for years.

Two of these deserve a paragraph rather than a table row.

PCI PTS POI makes P2PE and PIN entry possible at all, because it defines what a trustworthy card reader is. Approved devices are evaluated against physical and logical requirements including tamper responsiveness: the device is built so that interference with its enclosure or internal signals causes it to erase its cryptographic keys and stop working, which is why the correct response to a suspected-tampered terminal is to take it out of service rather than open it up. This is also why PCI DSS Requirement 9.5.1 exists — periodic inspection of POI devices for tampering and unauthorised substitution — and why Requirement 9.5.1.2.1 lets the entity set inspection frequency and type by targeted risk analysis rather than imposing one interval. Approvals expire; a lapsed device does not stop working but does stop counting for eligibility purposes.

PCI PIN is a separate compliance regime with its own assessor qualification, the Qualified PIN Assessor, whose Qualification Requirements stood at v1.2 in March 2025 and Program Guide at v1.2 in May 2025, with the PIN Attestation of Compliance template at v3.3 dated February 2026 — all as published at time of writing. One wrinkle trips people up: Visa announced the sunset of its own PIN Security compliance programme effective 1 October 2023, while stating that clients, processors and service providers would still be required to comply with PCI PIN security requirements. The programme that collected the paperwork went away; the obligation did not.

What certification does and does not prove

There is no PCI DSS certificate. There are two reporting outputs and one attestation form that accompanies either. A Report on Compliance is the long form, produced against the Council's ROC Template — at time of writing v4.0.1 revision 3 dated January 2025, and mandatory for QSAs documenting any assessment as a ROC — recording the entity's environment, the samples the assessor selected, and how each requirement was assessed. A Self-Assessment Questionnaire is the short form, available only where the entity meets the eligibility criteria in that questionnaire and its compliance-accepting entity permits it. Either way, the Attestation of Compliance is the covering declaration: a statement of the assessment's results, signed by the assessed entity and, where one was involved, the QSA company.

What an AOC proves is narrow, and the narrowness is the point. It proves that on a stated date, against a stated version of the standard, for a stated scope, on a stated sample, an assessor or the entity itself concluded the applicable requirements were met. Change any one of those variables and the document tells you nothing. The commonest failure in third-party due diligence is not reading the scope section: an AOC covering a supplier's hosting platform in one region does not cover the product you are buying, and one covering "the tokenisation service" does not cover the support tooling your account managers will actually use.

It does not prove the entity is compliant now, because the obligation is continuous and the document is not. Nor does it prove every requirement was met by the defined method. Requirements may be met by the defined approach, by a compensating control documented on a worksheet, or by the customised approach, in which the entity designs its own control to meet a stated Customized Approach Objective and the assessor writes bespoke testing procedures for it. The Council's guidance for compensating controls and the customised approach, most recently revised in June 2026 at time of writing, exists because these are the two mechanisms through which an assessment can pass while looking nothing like the standard's text. Neither is illegitimate; both require you to read further before relying on the attestation.

It also does not discharge your own third-party risk, and PCI DSS pushes that back onto you explicitly. Requirement 12.8.4 requires a programme to monitor third-party service providers' compliance status at least once every 12 months, and Requirement 12.9.2 requires TPSPs to support their customers' requests for information to meet 12.8.4 and 12.8.5 by providing compliance status information and stating which requirements are whose responsibility. The Council also notes that a service provider holding an AOC is expected to provide it to customers on request. If a supplier will not hand over an AOC and a responsibility matrix, that refusal is itself the finding — and two scheme-published registries give an independent cross-check, the Visa Global Registry of Service Providers and the Mastercard SDP Compliant Registered Service Provider List.

Finally, there is the programme nobody wants to meet. The PCI Forensic Investigator programme qualifies firms and individuals, who must already be QSAs, to investigate actual or suspected compromises affecting card transactions or cardholder data, under a single set of qualification requirements accepted across the participating brands rather than the brand-by-brand arrangements that preceded it. A PFI investigation is where the attestation and the reality are compared in writing by someone paid to find the difference, and Visa's waiver language quoted earlier is why that matters: the forensic report decides whether there was evidence of non-compliance before and at the time of the breach.

Where PCI stops and the law starts

PCI DSS is contract. Data protection is statute, and the two run on separate rails to separate destinations.

In the United Kingdom, a card number combined with anything identifying a person is personal data, and losing it engages the UK GDPR and the Data Protection Act 2018 regardless of PCI DSS status. The Information Commissioner's Office can impose fines up to the higher maximum of £17.5 million or four per cent of the undertaking's total annual worldwide turnover in the preceding financial year, whichever is higher — the figures in force at time of writing. A valid AOC is evidence of care that may bear on the ICO's assessment; it is not a defence, and the ICO is not bound by the schemes' conclusions or vice versa. An organisation can be compliant with PCI DSS and in breach of data protection law, or the reverse.

The boundary also runs the other way. Bank-transfer and direct debit data — sort code and account number, mandate references, payer details — is not account data as PCI DSS defines it, because the PAN is the defining factor for cardholder data and there is no PAN. Handling UK Direct Debit therefore does not create a cardholder data environment; the obligations that do apply come from the Bacs scheme rules operated by Pay.UK and from the sponsoring bank, separate instruments with separate accreditation routes. Treating one regime as evidence for the other is a category error that turns up regularly in supplier questionnaires.

Strong customer authentication is a third rail again. PCI 3DS Core governs the security of environments where specified 3-D Secure functions are performed; it does not mandate the use of 3-D Secure, and says nothing about when authentication is legally required or what exemptions apply. Three separate documents therefore govern the same checkout page: PCI DSS for the confidentiality of the card number, PCI 3DS Core for the security of the authentication infrastructure where it is in scope, and financial regulation for whether the customer must be authenticated at all.

The final observation is the one to take into a negotiation. PCI DSS is unusually explicit about the limits of its own authority. The Council states that it does not define compliance requirements for any organisation and does not set compliance validation responsibilities; that those are set by brands, acquirers and payment facilitators; and that organisations with questions must consult their compliance-accepting entity — a clarification it repeated as recently as its bulletin of 4 August 2026. When someone tells you PCI DSS requires a particular outcome for your business, the accurate response is that PCI DSS says what it says, and your acquirer decides what you must prove. Those are two different conversations, and only one can be settled by reading the standard.

48.98 Common wrong ideas

Wrong: We are PCI certified. **Right:** Nothing in PCI DSS produces a certificate; what it produces is an Attestation of Compliance covering one scope on one date against one version, which is a photograph rather than a passport.

Wrong: Last year's attestation means we are compliant. **Right:** Compliance is a continuous obligation, which is why service providers must review quarterly that staff are performing the documented tasks and reconfirm their scope at least every six months.

Wrong: The PCI Security Standards Council fined us, and there is a published tariff of per-card penalties. **Right:** The Council sets no compliance requirements and levies nothing, the schemes assess the acquirer who then recovers under contract, Visa's rules forbid the acquirer from representing that Visa

imposed the assessment, and nobody publishes a penalty schedule, so a precise figure is marketing, a leaked contract or an invention.

Wrong: Buying a tokenisation or point-to-point encryption product takes us out of the standard. **Right:** It reduces scope conditionally, and eligibility for the short questionnaire is lost by an expired listing, by receiving account data through any other route, or by not implementing every control in the P2PE Instruction Manual.

Wrong: Tokens are harmless, so the whole token estate is out of scope. **Right:** If the entity holding tokens can retrieve the account number, the retrieval path is in scope, and the vault performing the mapping is in scope for whoever operates it.

Wrong: We choose which questionnaire we complete. **Right:** The Council writes the questionnaires but the compliance-accepting entity decides who may use which, a point it restated by revising FAQ 1331 on 4 August 2026.

Wrong: PCI DSS makes us safe from card fraud. **Right:** The standard concerns the confidentiality of stored and transmitted account data; it will not tell you whether the person holding the card is the cardholder, does not govern chargebacks and does not deliver strong customer authentication.

Wrong: Removing 6.4.3 and 11.6.1 from SAQ A removed the script-attack problem. **Right:** The reporting burden moved and the requirements remain in the standard for merchants on SAQ D and on-site assessment, while SAQ A merchants now assert their own non-susceptibility with no prescribed method of verifying it.

Wrong: Visa's Technology Innovation Programme, or being a Mastercard Level 4 merchant, means we need not comply. **Right:** Both remove the obligation to validate, not the obligation to comply, and Mastercard still requires the acquirer to run a risk management programme over those portfolios.

Wrong: A supplier's Attestation of Compliance discharges our third-party risk. **Right:** Requirement 12.8.4 obliges you to monitor supplier compliance status at least annually and 12.9.2 obliges the supplier to tell you which requirements are whose, and the commonest due-diligence failure is not reading the scope section.

48.99 Chapter summary in 20 lines

1. A card number is a key with the address stamped on it, and the person who loses it is not the person who gets burgled or pays for the locks, which is the gap the whole chapter addresses.
2. Six card companies wrote six overlapping rulebooks until 2006, when they founded the PCI Security Standards Council to maintain one.
3. The Council develops the standards and runs the qualification and listing programmes, and each payment brand separately runs its own enforcement, validation and penalty programme.
4. The standard is therefore not legislation anywhere; it reaches a business through a chain of private contracts and its ultimate sanction is the loss of card acceptance.
5. Validation level follows transaction volume, and the two largest schemes' criteria do not agree, so a merchant can be Level 2 under one and Level 1 under another.
6. What you must actually do follows how close the card number gets to your own systems, not how big you are.
7. On a website that difference is 27 requirements under SAQ A or 139 under SAQ A-EP, decided by whether the payment form came whole from the provider or was drawn by the merchant's own code.

8. The Council's rule of thumb is blunt: if any element of the payment page delivered to the browser originates from the merchant's own website, SAQ A does not apply.
9. Point-to-point encryption removes cleartext account numbers from the merchant environment using keys the merchant never holds, and tokenisation substitutes a value only the vault can reverse.
10. Both reduce scope conditionally, and the reduction is something you maintain rather than something you buy.
11. Scope covers account data, everything with unrestricted connectivity to it, and everything that could impact its security, which is the category that pulls in identity, logging and remote-access infrastructure.
12. Segmentation removes a component only where it could not impact the cardholder data environment even if compromised, and the segmentation controls must be penetration tested annually, or six-monthly for service providers.
13. The storage rules are unforgiving: sensitive authentication data must not survive authorisation even encrypted, the account number must be rendered unreadable at rest, and display is limited to the leading digits and the last four.
14. Requirements 6.4.3 and 11.6.1 exist for script attacks that copy card details in the shopper's browser, where the merchant's servers are untouched and its logs show nothing wrong.
15. On 30 January 2025 the Council removed both from SAQ A in favour of a self-asserted eligibility criterion, which moved the reporting burden and not the security problem.
16. As at August 2026, v4.0.1 is the only active version, all 51 future-dated requirements became effective on 31 March 2025, and the Council ran a Request for Comments on the successor from 3 June to 20 July 2026.
17. There is no certificate: there is a Report on Compliance or a Self-Assessment Questionnaire, and an Attestation of Compliance covering one date, one version, one scope and one sample.
18. A requirement may have been met by the defined approach, by a compensating control or by the customised approach, so the attestation alone does not tell you what the control actually is.
19. PCI DSS is one of a family covering devices, hardware security modules, PINs, encryption solutions, 3-D Secure, mobile acceptance, software and card production, with sunsets in progress that change what a supplier may validly claim.
20. PCI DSS is contract and data protection is statute, running on separate rails to separate destinations: the Information Commissioner's Office can fine up to £17.5 million or four per cent of worldwide turnover, an attestation is evidence of care rather than a defence, and your acquirer decides what you must prove.

Sources: PCI SSC document library, standards and blog, including PCI DSS v4.0.1, the v4.x Quick Reference Guide and Prioritized Approach, the SAQ Instructions and Guidelines v4.0.1 r1, and the FAQ 1331 bulletin of 4 August 2026; Visa Account Information Security and the Visa Core Rules; Mastercard Site Data Protection; ICO data protection fining guidance. All verified as published in August 2026.

49

CHAPTER

Chargebacks

49.0 What this chapter gives you

1. You will be able to explain why the money leaves a merchant's account before anybody has decided who was right, and why that ordering is deliberate rather than a defect.
2. You will be able to name the four parties in a card dispute and say which of them the cardholder actually has a claim against.
3. You will be able to read a dispute's reason code as the question being asked, and choose evidence that answers that question rather than the customer's grievance.
4. You will be able to work out, for a given sale, whether contesting a dispute pays, using the dispute fee, the countered fee, staff time and a realistic win rate.
5. You will be able to explain why a merchant refunds a case it would almost certainly win, and why that is arithmetic rather than cowardice.
6. You will be able to compute a VAMP ratio and a Mastercard Excessive Chargeback Programme ratio, and say why the two produce different answers for the same business.
7. You will be able to distinguish a fraud report, filed as TC40 or through SAFE, from a dispute, and explain why an event that moved no money can still end a merchant account.
8. You will be able to say what chargeback gives a United Kingdom consumer that section 75 does not, and what section 75 gives that chargeback cannot.
9. You will be able to explain why a disputed sale can come back at a different amount from the one that settled, and name at least three causes.
10. You will be able to state what a merchant must record at the moment of sale in order to use Compelling Evidence 3.0 eighteen months later.

The plain version

Picture a school summer fete where nobody carries cash. Twenty stalls, and every one is paid the same way: you tell the desk by the gate what you bought and for how much, the desk writes it down, and at the end of the week the desk pays each stall what it is owed. You settle up with the desk separately.

The desk is not a bystander. It is the thing that actually moved the money. So if you come back on Monday and say the cake stall took four pounds off you and handed you an empty box, the desk has a lever nobody else at the fete has. It can decline to pay the cake stall, or, having already paid, take four pounds out of next week's takings.

That is a chargeback, and your bank is the desk. When you tap a card, your bank pays the shop and bills you afterwards. Your money never travels to the shop; your bank's money does. Your bank therefore has both the standing and the plumbing to reverse the payment.

The part that surprises people is the order of events. The money moves first and the argument happens afterwards. You ring your bank, your bank forms a view in a few minutes, and the shop's money is taken. It is not a hearing followed by a verdict; it is a seizure followed by an appeal.

Here is what that looks like with real numbers. Dee runs a small online shop in Leicester selling headphones and turntables. In March she sells a pair of headphones for £145 to a customer in Swansea and posts them the next morning. Her cost was £85, so on that sale she made £60 before overheads.

Eleven weeks later, in the last week of May, an email arrives from her payment provider. The £145 has been disputed. The reason given is that the cardholder says he did not make the payment. Dee's account has already been debited: £145 for the sale, and £20 on top as a fee for having received a dispute at all. That £20 figure is what a mainstream UK payment provider charged at the time of writing in August 2026, and it does not come back whatever happens next.

So before Dee has read the second paragraph of the email, she is £165 down and the headphones are in Swansea.

Now she has a decision to make, and it is not the one most people expect.

She can fight. The industry word for fighting is **representment**, which literally means presenting the transaction a second time: here is the sale again, and here is why it was good. To do that she gathers what she has. The order, the delivery address, the courier's tracking record, the time and date the parcel was signed for, the customer's email address, and the fact that the same email address and the same card bought a turntable from her in January and never disputed it. She uploads all of it. Filing costs another £20 at the time of writing, refundable only if she wins.

And then she waits. Not days. The bank that raised the dispute typically takes two to three months to read the file and decide.

Two things about that wait matter. Dee never speaks to her customer again; she is submitting a file to a stranger at a bank she has no relationship with, who will read it alongside a few hundred others. And the question that stranger answers is not "was Dee treated fairly?" It is a much narrower question, fixed at the very beginning by the box the customer's bank ticked.

That box has a number, and the number is the most important thing in this chapter.

If the box says "I did not make this payment", the only question is whether the cardholder made the payment. Delivery proof helps only insofar as it suggests the real cardholder received the goods at his own address. If the box says "the goods never arrived", the only question is delivery, and a signature at the door ends the argument in a sentence. If the box says "the goods were not as described", a signature at the door is worth nothing at all, because nobody disputes that the parcel arrived; they dispute what was in it.

Merchants lose disputes they should have won by answering the grievance instead of the box.

Suppose Dee wins. The £145 comes back, and so does the £20 filing fee. The £20 dispute fee does not. She has spent three hours and waited eleven weeks to end up £20 and three hours worse off than if the sale had gone through.

Suppose she loses. She is out £145, out both £20 fees, and out the £85 the headphones cost her: £270 in total. On a net margin of four and a half per cent, which is realistic for a small online retailer, £270 is the entire profit on about £6,000 of sales. One disputed pair of headphones eats a fortnight.

If the story ended there, chargebacks would be an expensive nuisance. It does not end there, because the card companies count them.

Every month, Visa and Mastercard divide the number of disputes a shop received by the number of sales it made, and compare the answer with a threshold. Cross it and the shop is put into a monitoring programme. That means monthly fines starting at around a thousand US dollars and climbing, at the top of the schedules in force in August 2026, to two hundred thousand US dollars a month. It means a written remediation plan. And if the numbers do not come down, the shop's bank ends the relationship, at which point its name goes onto a shared list that other banks consult before taking on a new customer. For a business that sells online, being unable to accept cards is not a setback. It is the end of the business.

Dee sells about 9,000 items a month. On Mastercard's threshold in force at the time of writing, a shop of that size is at risk once it takes 135 chargebacks in a month and its ratio reaches 1.5 per cent. Not 135 lost chargebacks. 135 chargebacks. Whether she wins them makes no difference whatsoever to the count, because the count is taken when the dispute is filed, months before anyone knows who was right.

Which produces the most counter-intuitive behaviour in the business, and the thing this chapter exists to explain. A shop with a strong, defensible, almost certainly winnable case will very often refund the customer instead of fighting. Not because it thinks it would lose, but because winning costs a non-refundable fee, several hours of somebody's week, an eleven-week wait, and does nothing at all to the number that can get its ability to take payments withdrawn. Refunding the customer before a dispute is raised costs the sale price and keeps that number down. The shop is not conceding the argument. It is paying to keep a counter from ticking.

That is why the industry has quietly built an entire layer in front of the dispute process: services that let a bank ask the shop about a suspicious transaction before raising anything, and services that let the shop pre-authorise an automatic refund the instant a dispute is about to be filed. Both exist for the same reason. A refund that arrives before the dispute is a refund. A refund that arrives afterwards is a chargeback, and chargebacks are counted.

Where the plain version stops being true

Chargeback is not your legal right, and you are not a party to it. The plain version made it sound as though Dee's customer was pursuing a claim. He was not. He asked his bank to pursue one, and his bank chose to. In the United Kingdom, at the time of writing in August 2026, chargeback has no statutory basis at all: it is a private contractual mechanism created by the card schemes and binding only on their licensees. UK Finance states it in exactly those terms: unlike section 75, chargeback is not a legal right. The Financial Ombudsman Service is as plain, saying a bank does not have to raise a chargeback, though where valid reasons exist it can be good practice, and that what it examines is whether it was fair not to raise one or to discontinue one. That is a duty owed by the bank to its customer, not a right of the customer against the shop. The legal instruments sit alongside chargeback and behave differently: section 75 of the Consumer Credit Act 1974 makes a credit card issuer jointly and severally liable with the supplier for breach of contract or misrepresentation, applying only to credit cards, only where the cash price of the item is over £100 and no more than £30,000, and with a six-year window rather than a scheme deadline; and regulation 76 of the Payment Services Regulations 2017 requires a payment service provider to refund an unauthorised transaction no later than the end of the business day following the day it becomes aware of it, which is a statutory obligation on the bank that exists whether or not any chargeback is ever raised or ever succeeds. The bank refunds because the law tells it to. The chargeback is how the bank goes looking for the money afterwards.

A chargeback is not the reversal of the original payment, and it does not always come back the same size. The fete analogy invites you to picture the same four pounds travelling backwards. It is not the same four pounds and it is often not four pounds. A chargeback is a new financial transaction with its own message, its own clearing cycle, its own value date and its own currency conversion, and any system built on the assumption that the disputed amount equals the original amount will eventually produce a reconciliation break nobody can explain. The commonest cause is foreign exchange: a payment converted at January's rate and disputed in April comes back at April's rate, so a sale that settled as 100 euros can return as 107.86 euros. Three other causes are routine. An issuer can bundle several months of a disputed subscription into one chargeback filed against a single charge. A cardholder can dispute part of an order. And a cardholder can dispute the full amount of a charge already partly refunded, leaving the merchant out the refund and the chargeback at once and having to prove the overlap.

Winning is neither final nor free, and it does not undo the count. The plain version implied a scoreboard with two outcomes. There are more. A successful representment can be met by the issuer filing a further stage, so cases marked won are routinely reversed weeks later and the money leaves the merchant's account twice. Money also moves outside the lifecycle altogether when an issuer credits or debits to correct an amount, which is why a lost case occasionally turns into a late win months after everyone stopped looking. The fee for receiving a dispute is normally non-refundable however the case ends. And the monitoring counters are indifferent to outcomes by design, not by oversight: waiting months for every outcome would make monitoring useless, and what the schemes measure is how well a merchant prevents disputes rather than how well it litigates them. Refunds do not remove a dispute from the count either, so a merchant who refunds after the dispute exists has paid twice for nothing.

Most disputes filed as fraud are not fraud, and the reason code is frequently the wrong one. The plain version treated the box as a neutral label chosen by an informed person. It is neither. A dispute filed as card-absent fraud asserts that the cardholder did not authorise the transaction, and that is the assertion the merchant must answer, regardless of what the customer's actual grievance was. Very often the actual grievance is that the subscription was harder to cancel than to start, or that a family member used the card, or that the statement descriptor was a holding-company name nobody recognised. The industry calls this first-party misuse, and it is a large enough share of card-absent disputes that both major schemes have built dedicated programmes to attack it. Two consequences follow. A merchant holding perfect delivery proof can lose a dispute where delivery was never in question, because it answered the customer instead of the code. And merchants get undeserved wins in the other direction, when a case really about quality is filed as non-delivery and defeated by a tracking number, leaving the customer's genuine complaint unadjudicated by anybody.

The technical version

Everything in this section is stated as accurate at the time of writing, in August 2026, and this is the most perishable chapter in the most perishable volume of this book. Scheme rulebooks are revised at least twice a year, one threshold below moved in April 2026 having moved in 2025, and the fee schedules are contractual documents not published in full. Check the current rulebook and your own acquirer's schedule before relying on any figure here.

Vocabulary, and who the parties actually are

Visa's rulebook calls the process a dispute; Mastercard's calls it a chargeback. Most acquirers treat the words as synonyms, and so does this chapter, while keeping one distinction sharp: a **pre-dispute** event is one where no money has moved and no counter has incremented, and a **dispute** proper is one where funds have been debited from the acquirer and passed to the issuer.

The parties are those of the four-party model. The **cardholder** complains. The **issuer** decides whether to raise the case and carries the statutory obligations towards its customer. The **scheme** provides the message formats, the deadlines, the reason codes and, at the end, the adjudication. The **acquirer** is debited by the scheme and passes the debit on to the **merchant**. That last link matters more than merchants realise: the acquirer's right to take money out of a merchant's account comes from the contract between them, not from the scheme, and the acquirer carries the credit risk if the merchant has no money left. That single fact explains rolling reserves, delayed settlement for high-risk sectors, and why an acquirer exits a merchant long before the scheme forces it to.

Two data feeds run alongside the dispute system and are constantly confused with it. Visa's **TC40** and Mastercard's **SAFE** are fraud reports: an issuer that believes a transaction was fraudulent must report it, and the report reaches the acquirer as what most gateways call an early fraud warning. A fraud report is not a dispute and no money moves, but it counts in Visa's monitoring ratio, which is the most misunderstood fact in merchant risk management. Analysis published by one large processor at the time of writing indicates that around 80 per cent of early fraud warnings become a fraud dispute if the merchant does nothing.

Stage zero: the pre-dispute layer

Before a dispute exists there is a layer designed to stop one existing, and it explains most of the merchant behaviour that looks irrational from outside. Visa's is operated by Verifi. **Order Insight** pushes enriched transaction detail, the merchant name a human would recognise, the item, the delivery address, into the issuer's call-centre screen and banking app at the moment the cardholder queries the charge, so that "I don't recognise this" is answered before it becomes anything; an April 2026 update lets merchants share Compelling Evidence 3.0 data at this stage. **Rapid Dispute Resolution** goes further: the merchant writes rules in advance, up to ten scenarios keyed on attributes such as amount and dispute condition, and a matching case about to be raised is refunded automatically in near real time. Mastercard's equivalent alerting network is **Ethoca**.

The commercial logic is one sentence: cases resolved through these pre-dispute products are excluded from the merchant's dispute count for monitoring purposes, as Visa's own programme documentation states. A merchant paying an alert fee and refunding a sale in full is buying the removal of a tick from a counter, and that is often worth several times the value of the sale.

American Express and Discover retain a formal **inquiry** stage, sometimes called a retrieval or request for information, where the issuer asks before it takes; Visa and Mastercard no longer use inquiries this way. Failure to answer an inquiry is treated as implicit acceptance and produces a chargeback that is, in practice, unwinnable.

The lifecycle, scheme by scheme

Visa has run two workflows since Visa Claims Resolution took effect in April 2018. The **allocation** workflow covers fraud, dispute condition group 10, and authorisation, group 11: Visa allocates liability up front from data it already holds, such as whether the transaction was authenticated or authorised, so there is no classic representment and the merchant's evidence is carried into a pre-arbitration case filed by the acquirer. The **collaboration** workflow covers processing errors, group 12, and con-

sumer disputes, group 13, and works the way most people picture it: chargeback, representment by the acquirer, then, if the issuer is unpersuaded, an issuer-initiated pre-arbitration, then arbitration.

Mastercard runs one flow: first chargeback, second presentment by the acquirer, pre-arbitration by the issuer, arbitration case filed with Mastercard's Dispute Resolution Management team. The mechanics are executed in Mastercom, and a party that fails to respond within its window is treated as having accepted liability.

American Express is a three-party scheme and its own issuer, so there is no inter-bank adjudication to have: inquiry, chargeback, a single opportunity to respond, and the decision is final. There is no arbitration stage.

Reason codes, and what each one actually demands

The reason code is the question. Evidence that does not answer it is not weak evidence but irrelevant evidence, and submitting a great deal of it is the commonest self-inflicted loss in the trade.

Visa's conditions and issuer time limits, as published at the time of writing in August 2026:

Code	Name	Issuer window	Workflow
10.1	EMV liability shift, counterfeit fraud	120 days	Allocation
10.2	EMV liability shift, non-counterfeit fraud	120 days	Allocation
10.3	Other fraud, card-present environment	120 days	Allocation
10.4	Other fraud, card-absent environment	120 days	Allocation
10.5	Visa Fraud Monitoring Programme	120 days	Allocation
11.1	Card recovery bulletin	75 days	Allocation
11.2	Declined authorisation	75 days	Allocation
11.3	No authorisation	75 days	Allocation
12.2 to 12.5	Incorrect transaction code, currency, account number or amount	120 days	Collaboration
12.6.1	Duplicate processing	120 days	Collaboration
12.6.2	Paid by other means	120 days	Collaboration
12.7	Invalid data	75 days	Collaboration
13.1	Merchandise or services not received	120 days	Collaboration
13.2	Cancelled recurring transaction	120 days	Collaboration
13.3	Not as described or defective	120 days	Collaboration
13.4	Counterfeit merchandise	120 days	Collaboration
13.5	Misrepresentation	120 days	Collaboration
13.6	Credit not processed	120 days	Collaboration
13.7	Cancelled merchandise or services	120 days	Collaboration

Code	Name	Issuer window	Workflow
13.8	Original credit transaction not accepted	120 days	Collaboration
13.9	Non-receipt of cash at an ATM	120 days	Collaboration

Mastercard's principal codes, with the windows published at the time of writing:

Code	Name	Issuer window
4808	Authorisation-related chargeback	90 days
4834	Point-of-interaction error	90 days
4837	No cardholder authorisation	120 days
4841	Cancelled recurring or digital goods transaction	120 days
4849	Questionable merchant activity	Programme-driven
4853	Cardholder dispute	120 days, start date varies
4863	Cardholder does not recognise, potential fraud	120 days
4870	Chip liability shift	120 days
4871	Chip and PIN liability shift	120 days

The evidence that answers each family is different in kind.

For **fraud in a card-absent environment**, Visa 10.4 and Mastercard 4837 and 4863, the merchant must show that the person who benefited was the cardholder: delivery to the cardholder's own billing address, a signature, an account with a login history predating the order, prior undisputed purchases from the same device or address, and any usage log showing the account was used after the purchase. Nothing about product quality is relevant, and a successful 3-D Secure authentication usually disposes of the case before any of it matters.

For **non-delivery**, Visa 13.1 and Mastercard 4853, the merchant must show delivery, or that the promised delivery date has not yet passed. A courier record placing the parcel at the cardholder's address, ideally with a signature or a photograph and a geolocation, ends it; for digital goods, an access log with timestamps and IP address.

For **not as described or defective**, Visa 13.3, the burden is descriptive: the listing as it stood on the day, photographs, specifications, the terms the customer accepted, and evidence that what shipped matched the listing. Proof of delivery is worthless here.

For **cancelled recurring or subscription** disputes, Visa 13.2 and Mastercard 4841, the merchant must produce the cancellation policy as displayed, the record that the customer agreed to it, the absence of a cancellation request before the billing date, and, if a trial converted, evidence of the reminder sent before conversion. For **credit not processed**, Visa 13.6, the answer is either the refund record with its date and amount or the published policy showing no refund was due. For **processing errors**, Visa group 12 and Mastercard 4834, the merchant is defending its own systems: authorisation and settlement records, the currency indicator submitted, and proof that a supposed duplicate was two genuine purchases.

The clock

Deadlines are calendar days, and they are enforced by software rather than by people. A missed deadline is a loss with no appeal.

Stage	Visa	Mastercard	American Express
Issuer raises dispute	120 days for most conditions, 75 for authorisation conditions	120 days for most codes, 90 for 4808 and 4834	120 days
Where goods or services are supplied later	Clock runs from the expected delivery or performance date	Clock runs from the delivery, cancellation or service date depending on code	Extended for certain conditions
Merchant or acquirer responds	30 days	45 days	20 days
Issuer escalates	30 days to pre-arbitration	45 days to pre-arbitration	Not applicable
Acquirer answers escalation	30 days	30 days	Not applicable
Arbitration	Filed after the escalation window	10 days to escalate after rejection	No arbitration stage

Two corrections to that table. The merchant almost never gets the scheme's window, because the acquirer or gateway must reserve time to format and transmit the response, so the internal deadline a merchant sees is typically 7 to 21 days by network and provider. And the issuer's own decision takes far longer than any response window: 60 to 75 days is normal, and the full lifecycle commonly runs two to three months, with no way to accelerate it other than to give up.

Compelling Evidence 3.0, and its Mastercard counterpart

The single most useful development for merchants in this decade is Visa's Compelling Evidence 3.0, effective 15 April 2023 and still in force at the time of writing. It applies to dispute condition 10.4, fraud in a card-absent environment, and it lets a merchant defeat the dispute with history rather than argument.

To qualify, the merchant must identify two previous transactions by the same cardholder that were not disputed, that are at least 120 days old and no more than 365 days old measured from the dispute date, and that share with the disputed transaction at least two of four data elements: user or account identifier, IP address, delivery address, and device identifier or fingerprint. At least one of the two matching elements must be the IP address or the device identifier. Meet that and liability moves back to the issuer.

Two design points matter. This is a rule about instrumentation rather than advocacy, so a merchant that does not record device fingerprints and IP addresses against orders cannot use it however honest its business; and Visa excludes qualifying fraud reports from the monitoring ratio, so the mechanism protects the counter as well as the money.

Mastercard's answer is the **First-Party Trust** programme, launched in the United States and, per Mastercard's announcement of June 2025, expanding to Canada, Latin America and the Caribbean, and Asia Pacific. Merchants supply enriched evidence either during authorisation or afterwards during the dispute, including purchase history, device details, delivery information, identity elements and location, so that an issuer can test a cardholder's claim at the point it is made. Mastercard's framing

of the problem, citing its 2025 State of Chargebacks report, is that the global cost of chargebacks is forecast to reach 42 billion US dollars by 2028, with nearly half reported as fraudulent.

Who pays, at each stage

Follow the money for a single £145 card-absent dispute at a UK merchant, on the published fee schedules in force at the time of writing in August 2026.

Event	Merchant	Acquirer or PSP	Issuer
Original sale	+£145 less scheme and acquirer fees	Earns processing margin	Funds the purchase, bills cardholder
Dispute raised	−£145 and −£20 dispute received fee	Debited by scheme, passes on	Credits cardholder, often already required to by law
Representment filed	−£20 dispute countered fee	Transmits and formats the case	Reviews for 60 to 75 days
Merchant wins	+£145 and +£20 countered fee	Fee income retained	Bears the loss or pursues the cardholder
Merchant loses	Nothing further moves; total cost £185 plus £85 of stock	Fee income retained	Retains the funds
Pre-arbitration	Further acquirer handling fee	Charges per case	Files at its own cost
Arbitration	Scheme fees on the losing side	Passes them through	Scheme fees on the losing side

The £20 received and £20 countered figures are the published UK dispute fees of one large payment provider, effective 17 June 2025 and unchanged at the time of writing; the same provider charges 15 US dollars for each in the United States and 20 euros in the euro area. Traditional UK acquirers typically charge £15 to £25 per chargeback. The countered fee is returned on a win; the received fee, at that provider and most others, is not.

Arbitration is where the economics become absurd on purpose. The schemes' fee schedules are contractual documents and are not published, which is easy to demonstrate: Visa's own public rulebook, the Visa Core Rules and Visa Product and Service Rules in the edition of 18 April 2026, sets out the whole dispute resolution machinery, arbitration and compliance included, without stating a single amount, referring the reader instead to the applicable fee schedule, and Mastercard's Chargeback Guide reaches members through its technical resource centre rather than open publication. The shape of the charging is nonetheless consistent across every account of it: a filing fee paid by the party that escalates, a substantially larger fee borne by the party that loses, and a separate penalty for technical violations such as filing outside the rules or in the wrong format. Industry sources consistently put the minimum exposure on a losing party at several hundred US dollars, with case filing and ruling amounts commonly quoted between 500 and 600 US dollars and technical violation penalties of 100 to 250 US dollars, and those sources disagree with one another at the edges. Treat the range as the settled view of the trade rather than as a verified figure, because no publicly available primary specification confirms it.

The consequence is not in dispute even where the numbers are. Spending six hundred dollars to contest £145 is irrational, so arbitration is almost never used at ordinary retail values. It exists as a threat that disciplines the earlier stages, and as a remedy for large-ticket disputes: travel, freight, professional services, luxury goods.

Ratios, and the programmes that end merchant accounts

This is the part that decides whether a business survives, and the arithmetic differs between schemes in ways that catch people out.

Visa replaced the old Visa Dispute Monitoring Programme and Visa Fraud Monitoring Programme with a single **Visa Acquirer Monitoring Programme**, VAMP, which took effect on 1 April 2025 with an advisory period that Visa's published fact sheet ended on 30 September 2025.

The VAMP ratio is the count of TC40 fraud reports plus the count of TC15 disputes, divided by the count of settled transactions, measured monthly on card-absent volume. Three features of that formula matter enormously. Fraud reports count even though no money moved and no dispute was ever raised. A transaction appearing in both the TC40 and TC15 feeds is counted twice. And the denominator is settled transactions, so declining suspicious authorisations does not flatter the ratio, though it does keep fraud out of the numerator. Excluded from the count are cases resolved through pre-dispute products and fraud reports qualifying for Compelling Evidence 3.0.

On the thresholds in force at the time of writing in August 2026, a merchant is identified as excessive at a VAMP ratio of 1.5 per cent in Asia Pacific, Canada, the European region, the United States and Latin America and the Caribbean, and at 2.2 per cent in Central Europe, the Middle East and Africa. The 1.5 per cent figure is recent: Visa cut it from 2.2 per cent on 1 April 2026. A minimum count applies before a merchant is assessed at all, 1,500 combined items a month in most regions, or 150 items and 75,000 US dollars in CEMEA. At acquirer portfolio level the thresholds are far tighter: 0.5 per cent above standard, 0.7 per cent excessive.

Visa monitors enumeration, meaning card-testing traffic, separately: excessive monitoring applies at an enumeration count of 300,000 and a ratio of 20 per cent, with no fines attached.

Visa assesses a per-item fee on identified merchants but does not publish the schedule in its public fact sheet. Industry reporting at the time of writing puts the fee at 8 US dollars per fraud report or dispute at the excessive tier and 4 US dollars at the above-standard tier, with a three-month grace period for a first identification in a rolling twelve months. Treat those as industry reports rather than primary citations.

Mastercard operates the Excessive Chargeback Programme in two tiers, and calculates its ratio differently: chargebacks received in the current month divided by the transaction count of the **preceding** month. That definition punishes sharply seasonal merchants, whose January disputes are divided by a December that has ended.

Programme	Chargeback count	Ratio	Effect
Excessive Chargeback Merchant	100 to 299 in the month	1.5 to 2.99 per cent	Escalating monthly fines from month two
High Excessive Chargeback Merchant	300 or more in the month	3 per cent or more	Steeper fines, plus issuer recovery

The assessment schedules published at the time of writing, in US dollars or euros per month, count from the first month above threshold:

Months above threshold	Excessive Chargeback Merchant	High Excessive Chargeback Merchant
1	none	none
2	1,000	1,000

Months above threshold	Excessive Chargeback Merchant	High Excessive Chargeback Merchant
3	1,000	2,000
4 to 6	5,000	10,000
7 to 11	25,000	50,000
12 to 18	50,000	100,000
19 and beyond	100,000	200,000

From month four an issuer recovery assessment is added: 5 US dollars or euros for each chargeback above 300 in the month, so a merchant identified in month four with 400 chargebacks pays 5,500 on top of nothing else. Exit requires three consecutive months below the threshold, and the schedule then resets to month one rather than resuming where it stopped. Mastercard counts a chargeback regardless of any prior refund, of liability shift, and of the outcome.

Mastercard's separate **Excessive Fraud Merchant** programme targets a different failure: high fraud combined with an unwillingness to authenticate. On the criteria published at the time of writing, all of the following must hold in a month: at least 1,000 Mastercard e-commerce payments; net fraud chargeback volume above 50,000 US dollars, or 15,000 Australian dollars in Australia; a fraud chargeback rate above 0.5 per cent, or 0.2 per cent in Australia; and 3-D Secure used on no more than 10 per cent of Mastercard payments in non-regulated countries or 50 per cent in regulated ones. Fines run from 500 US dollars in month two to 100,000 US dollars from month nineteen. Where both are breached, this programme takes precedence but the Excessive Chargeback Programme clock keeps running underneath.

Two regional regimes deserve naming. In the United States only, Visa's **Secure Excessive Fraud Programme** captures merchants whose fraud on domestic 3-D Secure authenticated Visa transactions reaches 75,000 US dollars and 0.9 per cent in a month; there is no fine, but the merchant loses the liability shift on domestic authenticated transactions until it fully exits, which is heavier than any fine. In Australia, the AusPayNet card-not-present framework works quarterly from 50,000 Australian dollars of fraud chargebacks and a 0.2 per cent ratio, escalating to a mandate that every card-not-present transaction be sent for strong customer authentication.

The lists, which are the real sanction

Fines are survivable. The terminated merchant files are not, and they operate on thresholds far below the monitoring programmes.

Mastercard's **MATCH**, the Mastercard Alert to Control High-risk Merchants system, is a shared database of merchants whose acquiring relationship was terminated. Every processor must check it when onboarding, and must add a terminated merchant meeting the criteria within one business day. Eleven of the reason codes are qualitative, covering laundering, collusion, illegal transactions, PCI DSS non-compliance and the rest. Two are quantitative, and as published at the time of writing they are stark. Reason code 4, **excessive chargebacks**, means Mastercard chargebacks in any single calendar month exceeding 1 per cent of Mastercard sales transactions in that month and totalling 5,000 US dollars or more, with no minimum chargeback count. Reason code 5, **excessive fraud**, means a fraud-to-sales dollar volume ratio of 8 per cent or more in a calendar month with ten or more fraudulent transactions totalling 5,000 US dollars or more. Note the 1 per cent: below every monitoring threshold in this chapter, and indifferent to whether the chargebacks were later reversed or won. Records remain for five years and are then purged automatically; a processor can remove an entry only if it

listed the merchant in error, or where the listing was for PCI DSS non-compliance since remedied. Mastercard does not assess the accuracy of listings.

Visa's equivalent is the **Visa Merchant Screening Service**. Its two quantitative reason codes, as published at the time of writing, are excessive disputes, meaning a dispute count of 1,000 and a 1.8 per cent dispute-to-sales amount ratio in any single month, and excessive fraud, meaning 250,000 US dollars of fraud at the same 1.8 per cent ratio; both are qualified by the merchant having failed to remediate adequately.

Neither list is supposed to be a blacklist. Both function as one, because most acquirers decline an application that produces a hit.

Why a merchant refunds a case it would win

Everything above assembles into two decisions taken at different moments, and only the second is about winning. Confusing them is why merchant behaviour looks like cowardice from outside.

The first decision comes before any dispute exists, when an early fraud warning or a pre-dispute alert arrives. Refunding costs the whole sale, £145, with certainty. Doing nothing risks the dispute, and on the processor analysis cited earlier roughly 80 per cent of ignored early fraud warnings become disputes: a non-refundable £20 fee, the £145 if lost, the stock, and one tick on a counter that no later victory removes. Hence the published guidance of one large processor at the time of writing, which is to refund proactively where the charge is roughly at or below your dispute fee and generally not to bother above about 135 per cent of it. For a UK merchant paying £20, that means refunding the small ones and standing and fighting on the rest.

The second decision comes after the dispute exists, and by then the counter has already moved. Refunding now achieves nothing: the merchant cannot refund outside the dispute process while it is open, and a refund would not remove the dispute from any monitoring count. So the only remaining question is money. Fighting the £145 case costs the £20 countered fee, returned on a win, plus perhaps £30 of loaded staff time to assemble the evidence properly. At a 40 per cent win rate, realistic for a well-evidenced delivery case, the expected recovery is £66 against about £42 of expected cost, and fighting pays. At a 15 per cent win rate, realistic for a subscription dispute with no usage logs, the expected recovery is £25 against £47 of cost, and it does not.

The behaviour looks irrational because observers watch the second decision while the merchant is making the first. A business inside a monitoring programme, or near one, is not optimising individual cases at all. Its objective is the three consecutive months below threshold that end the programme, and the only lever on the numerator is stopping disputes before they are filed. Hence alerts that cost more than the sale they cancel.

Three circumstances justify fighting regardless. Where the value is large enough that the fee stack is immaterial. Where the case is a template, a cancellation policy or a delivery process that will generate hundreds of identical disputes, so winning once fixes evidence practice for all of them. And where the merchant is comfortably below every threshold, so the counter is not binding and money is the only consideration.

The United Kingdom layer

British merchants and issuers operate under scheme rules and statute simultaneously, and the two do not align.

Regulation 76 of the Payment Services Regulations 2017, quoted earlier for its next-business-day deadline, also requires the account to be restored to the state it would have been in, with a value date no

later than the original debit. That obligation is owed by the bank to its customer and is independent of the chargeback. Regulation 77(6) allocates the loss further along: where strong customer authentication was required under regulation 100 but the payee or the payee's payment service provider did not accept it, that party must compensate the payer's payment service provider for the sums it paid out under regulation 76. That is the statutory backbone of what merchants experience as the 3-D Secure liability shift, and it is why a UK merchant that suppresses authentication is not merely accepting scheme liability but a regulatory one.

Section 75 of the Consumer Credit Act 1974 is the other statutory route, and often the better one: joint and several liability of the credit card issuer with the supplier, on the terms set out earlier, with a six-year limitation period rather than a 120-day scheme deadline. Chargeback reaches debit cards and small values that section 75 does not; section 75 reaches consequential loss and long-dead purchases that chargeback cannot. A bank handling a complaint properly considers both.

The Financial Ombudsman Service sits above all of it. Its approach for consumers is described earlier in this chapter; for a merchant complaining about its acquirer, its stated position is that where the acquirer acted in accordance with the merchant services agreement the complaint is unlikely to be upheld, which returns the merchant to the contract it signed.

For scale, UK Finance's Annual Fraud Report published on 15 June 2026 recorded £1.28 billion of payment fraud losses in 2025, up 4 per cent, of which £703.4 million was unauthorised fraud, down 5 per cent. Remote purchase fraud, the card-absent category that generates most fraud disputes, was £423.5 million across 3.2 million cases: a 3 per cent rise in value on a 13 per cent rise in cases. Falling average values on rising case counts is precisely the pattern that pushes dispute counts up while losses hold steady, and it is counts that the monitoring programmes measure.

What is moving under this chapter

Three things will date this chapter first, and a reader in 2028 should check all three before quoting any of it.

The thresholds are moving downwards. Visa's merchant excessive threshold fell from 2.2 per cent to 1.5 per cent on 1 April 2026, having been set only a year earlier, and the acquirer-level figures moved on their own schedule. There is no reason to expect that to be the last revision.

The adjudication is being automated. On 1 April 2026 Visa announced six dispute services: the Visa Dispute Resolution Network for pre-dispute handling, in pilot with general availability planned for late 2026; Visa Dispute Recovery Manager, which automates representment with generative responses and win-prediction scoring; Dispute Intelligence and Dispute Doc Analyzer, which summarise and score cases for issuers and acquirers; Visa Dispute Case Manager; and the Order Insight update that carries Compelling Evidence 3.0 into the pre-dispute conversation. Visa's stated context was 106 million disputes processed globally in 2025, a 35 per cent increase since 2019. When both sides of a dispute are assembled and assessed by models, the evidence that wins will be structured data captured at the time of the order rather than prose written afterwards, which is a change in what merchants must build, not merely in what they must write.

And the ground is shifting under the word fraud. Compelling Evidence 3.0 and First-Party Trust both let history speak for the merchant, and both reward instrumentation over argument. The merchant that survives the next revision of these rules is not the one with the best dispute writer. It is the one that recorded the device, the address, the login and the delivery at the moment of sale, and can prove eighteen months later that the same person came back.

49.98 Common wrong ideas

Wrong: A chargeback is a legal right the cardholder exercises against the shop. **Right:** In the United Kingdom it has no statutory basis at all; it is a private contractual mechanism of the card schemes, the cardholder asks their bank to raise one, and the bank decides whether to.

Wrong: A chargeback reverses the original payment, so the same amount comes back. **Right:** It is a new financial transaction with its own clearing cycle, value date and conversion, so a sale that settled as 100 euros can return as 107.86 euros, and issuers can also bundle, split, or overlap with a refund already given.

Wrong: Winning the dispute puts the merchant back where it started. **Right:** The fee for receiving a dispute is normally non-refundable, the issuer can escalate and take the money a second time, and no victory removes the tick from the monitoring counter.

Wrong: Good delivery proof wins most disputes. **Right:** Proof of delivery decides non-delivery cases, helps in card-absent fraud only insofar as it places the goods with the cardholder, and is worth nothing at all under "not as described", where nobody disputes that the parcel arrived.

Wrong: Refunding the customer once the dispute exists gets the chargeback off the books. **Right:** The count is taken when the dispute is filed, the merchant cannot refund outside the process while it is open, and a refund at that stage pays twice for nothing.

Wrong: Disputes filed as fraud are fraud. **Right:** A large share of card-absent fraud disputes are first-party misuse, a subscription harder to cancel than to start, a family member, an unrecognised statement descriptor, which is why both schemes have built programmes aimed squarely at it.

Wrong: Declining suspicious authorisations improves the VAMP ratio. **Right:** The denominator is settled transactions, so declines do not flatter the ratio; they only keep fraud out of the numerator.

Wrong: The fines are the thing to fear. **Right:** Fines are survivable; the terminated merchant files are not, and MATCH reason code 4 bites at 1 per cent of monthly Mastercard sales totalling 5,000 US dollars, below every monitoring threshold and indifferent to whether the chargebacks were later won.

Wrong: Arbitration is where a merchant with a strong case gets justice. **Right:** With case filing and ruling amounts commonly quoted between 500 and 600 US dollars, it is irrational at ordinary retail values and exists as a threat that disciplines the earlier stages.

Wrong: The merchant gets the scheme's full response window. **Right:** The acquirer or gateway must reserve time to format and transmit the file, so the internal deadline is typically 7 to 21 days, and a missed deadline is a loss with no appeal.

49.99 Chapter summary in 20 lines

1. When you tap a card your bank pays the shop and bills you afterwards, so it is the bank's money that moved and the bank that has both the standing and the plumbing to reverse it.
2. A chargeback therefore runs backwards from the ordinary intuition: the money is seized first and the argument happens afterwards.
3. For the merchant the first news is a debit of the sale plus a non-refundable dispute fee, arriving weeks or months after the goods were posted.
4. Contesting a dispute is called representment, costs a further fee refundable only on a win, and is decided by a stranger at the issuing bank two to three months later.

5. The question that stranger answers is fixed by the reason code the issuer chose, not by the merchant's account of what happened.
6. Evidence that does not answer the reason code is irrelevant rather than weak, and submitting a great deal of it is the commonest self-inflicted loss in the trade.
7. Visa runs allocation and collaboration workflows, Mastercard a single flow ending in arbitration, and American Express, being its own issuer, decides once and finally.
8. Deadlines are calendar days enforced by software, and the merchant's internal window is shorter than the scheme's because the acquirer must reserve time to transmit.
9. Chargeback in the United Kingdom is a scheme mechanism with no statutory basis; section 75 of the Consumer Credit Act 1974 and regulation 76 of the Payment Services Regulations 2017 sit alongside it and behave differently.
10. Because a chargeback is a new transaction, foreign exchange, bundled subscriptions, partial disputes and earlier refunds routinely make the returning amount differ from the original.
11. Winning is neither final nor free: issuers escalate, fees stay spent, and money moves outside the lifecycle through issuer credits and debits months later.
12. Every month the schemes divide a merchant's disputes by its transactions and compare the answer with a threshold.
13. Visa's VAMP ratio adds fraud reports to disputes over settled card-absent transactions, so events that moved no money count, and a transaction appearing in both feeds counts twice.
14. Mastercard divides this month's chargebacks by last month's transaction count, which punishes sharply seasonal merchants.
15. The counters are indifferent to outcomes by design, because what the schemes measure is prevention and waiting months for every result would make monitoring useless.
16. Crossing a threshold brings escalating monthly fines, a written remediation plan, and eventually the end of the acquiring relationship.
17. The real sanction is the terminated merchant lists, MATCH and the Visa Merchant Screening Service, whose quantitative triggers sit below every monitoring threshold and whose entries last five years.
18. That is why a merchant refunds a case it would win: a refund arriving before the dispute is a refund, and only one arriving afterwards is counted.
19. Hence the pre-dispute layer, Order Insight, Rapid Dispute Resolution and Ethoca, whose commercial value is the removal of a tick from a counter rather than the rescue of a sale.
20. And hence the direction of travel: Compelling Evidence 3.0 and First-Party Trust reward data captured at the moment of the order rather than prose written afterwards.

Sources, all consulted in August 2026: Visa's Acquirer Monitoring Programme fact sheet, its Compelling Evidence 3.0 merchant readiness paper and its press release of 1 April 2026 on dispute resolution services; Verifi on Order Insight and Rapid Dispute Resolution; Mastercard's Chargebacks Made Simple guide, its Excessive Chargeback Programme guide and its June 2025 First-Party Trust announcement; Stripe's public documentation on disputes, monitoring programmes and high-risk merchant lists, which quotes the MATCH and VMSS standards verbatim, and its June 2025 dispute pricing notice; regulations 76, 77 and 100 of the Payment Services Regulations 2017 and section 75 of the Consumer Credit Act 1974 on legislation.gov.uk; UK Finance on chargeback and section 75 and its Annual Fraud Report of 15 June 2026; Financial Ombudsman Service pages on disputed transactions and goods bought on credit; and trade references from Chargeback Gurus, Chargeflow, Braintree and the Merchant Risk Council where scheme documents are not public, used only where consistent and flagged where not.

50

CHAPTER

Reconciliation

50.0 What this chapter gives you

1. You will be able to explain why three records of the same week disagree, and why forcing them to agree is not the job.
2. You will be able to take a till total, a processor payout and a bank statement and account for every penny of the difference, naming fees, refunds, timing and breaks separately.
3. You will be able to tell a reason from a question, so that an explained difference is closed and an unexplained one is opened as a break with a date of birth and an owner.
4. You will be able to age a break register by value as well as by count, and say why the age of the oldest open item is the most diagnostic number in it.
5. You will be able to name the five assertions a reconciliation makes, completeness, existence, accuracy, cut-off and valuation, and show why equal totals test only the weakest of them.
6. You will be able to design a ledger that hands you the break list rather than making you hunt for it, using an immutable event log and clearing accounts held per counterparty and per currency.
7. You will be able to list the five identifiers that must be persisted at the moment of the event, because none of them can be reconstructed afterwards.
8. You will be able to assign an unexplained item to a known category quickly, whether it is fee netting, an authorisation never captured, a partial capture, a retranslation movement or a Direct Debit return.
9. You will be able to state what CASS 15 requires of a United Kingdom safeguarding institution on each reconciliation day, and what must happen by the end of the day a shortfall is found.
10. You will be able to explain why young payments firms die of being unable to prove where the money is rather than of losing it.

The plain version

Ravi runs a bakery on a side street in Coventry. Hardly anybody pays him in cash any more, so there are three separate written records of what happened in his shop last week, kept by three parties who have never spoken to one another.

The first is Ravi's own. His till logs every sale as it happens: the time, the amount, the item. That is a record of what he believes he sold.

The second belongs to the company that supplies his card machine. Each day they total what they collected on his behalf, subtract what they charge for the service, and send the remainder onward, publishing all of it in a weekly report. That is a record of what they believe they processed.

The third is his bank statement. It shows money landing in his account: a figure and a date, nothing more. That is a record of what actually arrived.

Three notebooks describing one week. Here is the thing that startles everybody the first time they meet it. The three notebooks will not agree. They will not agree this week, they will not agree next week, and no amount of diligence will make them agree on the first pass. The job is not to force them to agree. The job is to explain, line by line, precisely why they differ.

That job has a name. It is reconciliation, and it is simultaneously the least glamorous and the most load-bearing activity in finance.

Let us walk through Ravi's week, because the shape of the answer matters more than the numbers. His till says he took 386 card payments between Monday and Sunday, adding up to £4,812.60.

Day	Card sales	Value taken
Monday	49	£612.40
Tuesday	47	£588.15
Wednesday	56	£701.90
Thursday	52	£655.25
Friday	73	£912.60
Saturday	88	£1,104.30
Sunday	21	£238.00
Total	386	£4,812.60

The card company's report says three things the till does not. First, it charges him: one and a half per cent of each sale plus five pence per transaction. Second, it records a refund of £38.50, handed back on Wednesday to a customer whose wedding cake order was cancelled. That customer paid three weeks earlier, so the £38.50 appears nowhere in this week's sales but comes out of this week's money. Third, and this is the smallest and most interesting difference, the report shows 385 payments, not 386. One sale, £18.75 on Saturday afternoon, is simply not there.

So the report reads: £4,793.85 processed across 385 sales, less a refund of £38.50, less fees of £91.16, leaving £4,664.19 to be paid over.

Finally, the bank statement. Money from a card machine does not arrive the instant the card is tapped. It arrives a couple of working days later, and a weekend's takings are bundled into the Monday run. These are the deposits that landed.

Day money arrived	Amount	What it was for
Monday	£498.11	The previous week's Thursday
Tuesday	£2,043.77	The previous week's Friday, Saturday and Sunday
Wednesday	£600.76	This Monday
Thursday	£576.98	This Tuesday
Friday	£650.07	This Wednesday, less the £38.50 refund
Saturday	none	
Sunday	none	
Total	£4,369.69	

Three numbers for one week: £4,812.60, £4,664.19 and £4,369.69. All three are correct. Reconciling means turning that spread into arithmetic that closes.

Take the first gap. £4,812.60 minus £4,664.19 is £148.41, and Ravi can account for every penny: £91.16 of fees, £38.50 of refund, £18.75 for the sale that vanished. The gap is now explained. It is not a problem. It is a set of reasons, each with a name.

The second gap needs one more piece of information: what each day was worth after the card company took its cut. Monday's £612.40, less £11.64 of fees, is £600.76, which is precisely what landed on Wednesday. Do the same for the rest of the week and the seven days come to £4,702.69 net, which less the £38.50 refund is £4,664.19, exactly what the card company says it is paying.

Now the second gap closes. The card company paid £4,664.19; the bank shows £4,369.69 arriving; the difference is £294.50, and it is pure timing. Four days of this week's money had not landed when Sunday ended: Thursday at £642.82 net, Friday at £895.26, Saturday at £1,064.92 and Sunday at £233.38, which is £2,836.38 in transit. Against that, £2,541.88 of last week's money landed on Monday and Tuesday. Subtract the second from the first and you get £294.50. Explained.

That is the entire discipline in miniature. Ravi did not make the numbers match. He wrote down a reason for every difference, and the reasons added up.

Except for one. The fees are a reason. The refund is a reason. The timing is a reason. The missing £18.75 is not a reason; it is a question. A customer bought two loaves and a coffee cake, the machine said approved, and no money ever moved. The likeliest explanation is that the payment was reserved against the card but never claimed before the machine closed its day, so the reservation expired and the card released the money back. Ravi is out the bread and out the cash.

An unexplained difference of this kind is called a break, and the most important fact about a break is not what caused it but how old it is.

A break one day old is normal. Money is in transit, files arrive late, somebody will look at it tomorrow. A break ninety days old is a different animal: a standing statement in Ravi's accounts that he owns £18.75 which does not exist, quietly making him richer on paper than in life. Do that a thousand times and the gap between the books and the bank stops being a rounding error and becomes a hole.

This is why serious operations do not merely count their breaks; they age them. Every unexplained difference gets a date of birth, an owner with a name, and a bucket: under two days, under a week, under a month, over a month. A business with a flat ageing profile knows where its money is; a business with a lengthening tail does not.

Then scale it. Ravi had one break in 386 payments, roughly a quarter of one per cent, which is an unremarkable rate. At a hundred thousand payments a day, the same rate produces around 260 new breaks daily and 1,800 a week, arriving forever. At ten minutes of attention each, that is more than a full working week of somebody's time every week just to stand still.

Which brings us to the sentence this chapter exists to deliver. Young financial technology companies do not usually die of fraud, and they do not usually die of being hacked. They die because they could not prove where the money was. Not because they lost it. Because they could not prove it. The two are different, and the second is fatal on its own.

Where the plain version stops being true

There are not three notebooks, and "the processor report" is not one document. A real payments business reconciles between six and fifteen sources at once: an order system, a gateway, one or more acquirers, the card schemes, a wallet provider, a foreign exchange counterparty, a Bacs bureau or sponsoring bank for Direct Debits, an operating bank account, and, if the firm is regulated, a safeguarding account whose contents are legally not its own. Worse, a single processor issues several reports covering the same period, deliberately keyed on different dates, which will never tie to one another without work. Stripe's public documentation, at the time of writing in August 2026, shows the seam plainly: a balance transaction carries an `available_on` date, when its net funds become available in the Stripe balance, and separately an `automatic_payout_effective_at` date, when the payout is expected to reach your bank account. Reconcile on the first and you get one population; on the second, another, both correct. Adyen's settlement details report, again at the time of writing, changes shape with the acquirer: where the acquirer supplies transaction information at interchange level, the Commission column is empty and the Markup, Scheme Fees and Interchange columns are populated instead; where it does not, Commission carries the total and those three are blank. Identical economics, different structure, and a reconciliation written against one shape will silently produce nonsense against the other.

Matching is not the control. Explanation is. Two systems can match on amount and reference and both still be wrong, because both were fed by the same faulty upstream event. Two errors of opposite sign can cancel, producing a difference of zero across a population containing a five-hundred-pound overstatement and a five-hundred-pound understatement. This is why mature reconciliation runs on gross, signed values and never on the net, why a reported difference of nil is the beginning of the check rather than the end, and why the useful metric is not "what proportion of lines matched" but "what is the total absolute value of items I cannot yet explain, and how old is the oldest one".

"As at close of business" describes a moment that never existed. The bakery assumed a single instant at which all three records can be photographed. There is none: the United Kingdom's payment systems settle at fixed and different times of day. On the Bank of England's published summary of the Real-Time Gross Settlement daily timetable, as it stood at the time of writing in August 2026, CHAPS settlement starts at 6am, Faster Payments settles three times each working day at 7am, 1pm and 5pm, Bacs settles at 9.30am, LINK at 11am, Mastercard at 11.15am, Visa at 12.15pm and the Image Clearing System at 4.30pm, with CHAPS closing for customer payments at 5.40pm and for financial institution payments at 6pm. A payment can therefore be irrevocable in one system, provisional in another and invisible in a third at the same wall-clock second. A bank statement line also carries at least two dates, booking and value, frequently not the same day. A reconciliation that does not define in writing which clock and which date field it cuts on produces a different answer depending on when it is run, which is the same as producing no answer.

Once the money belongs to somebody else, this stops being bookkeeping and becomes law. In the bakery the money was Ravi's, so a break was his problem alone. The moment a firm holds funds for customers, the arithmetic acquires a statutory deadline and a regulator at the other end of it. In the United Kingdom, at the time of writing in August 2026, authorised payment institutions and electronic money institutions must safeguard relevant funds under regulation 23 of the Payment Services Regulations 2017 and regulations 20 to 22 of the Electronic Money Regulations 2011, and since 7 May 2026 those requirements have been supplemented by a new chapter of the FCA's Client Assets sourcebook, CASS 15, which converts reconciliation from good practice into a rule with a same-day remedy attached. Under CASS 15 as it stands at the time of writing, a shortfall found by an internal safeguarding reconciliation must be paid in by the end of the day on which the reconciliation was performed,

out of the firm's own money if customer money cannot be reached. No version of that obligation can be met with a spreadsheet somebody updates on Fridays.

The technical version

On dating

Everything below is stated as accurate at the time of writing, in August 2026, and this corner of the subject moves fast. The FCA's Supplementary Regime for safeguarding came into force on 7 May 2026, three months before this was written, following Policy Statement PS25/12 of 7 August 2025. In that statement the FCA declined, for now, to proceed with the end-state proposals it had consulted on, rebranding them the Post-Repeal Regime, so the rule references below have a known revision ahead of them. The Payment Systems Regulator still existed as a separate body as this was written: the Government announced in March 2025 that it intended to abolish the PSR and consolidate its functions primarily within the FCA, consulted in September 2025, and on 21 April 2026 published a response confirming full consolidation and noting that this requires primary legislation, to be brought forward as soon as Parliamentary time allows. Check the Handbook, the Bank of England's timetable and your own scheme documentation before relying on any figure here.

What a reconciliation actually asserts

A reconciliation is not a comparison. It is an assertion, and most weak processes fail by never having stated the assertion at all. It has five parts. Completeness: every event that occurred appears in the record. Existence: every event in the record occurred, and nothing has been invented or duplicated. Accuracy: each event is recorded at the right amount, in the right currency, against the right counterparty. Cut-off: each event falls in the right period, on a defined and documented clock. Valuation: where an amount has been translated between currencies, or is subject to later adjustment, the value carried is the right one on the right basis.

Notice that "the two numbers are the same" tests a weak form of accuracy and nothing else. A process producing a matching total while a hundred transactions are absent from both sides has passed a test never worth taking, which is why a processor restatement of a closed period must be posted as a versioned, dated correction referencing the original rather than an edit to it.

The three legs, named properly

Leg one is the internal ledger. In a serious build this is not the order table. It is a double-entry subledger fed by an append-only, immutable event log, in which a payment is a state machine with explicit transitions: initiated, authorised, captured in full or in part, settled, refunded in full or in part, disputed, represented, written off. Each transition posts a balanced journal. Critically, the design includes a clearing account, sometimes called cash-in-transit or settlement suspense, held per counterparty and per currency, debited when a payment is captured and credited when the cash lands. Its balance is, by construction, the exact population of open items. Designed this way, you do not hunt for breaks; the ledger hands you the list.

Leg two is the processor or acquirer record. It arrives as files or API objects, and it is where the commercial complexity lives. Under an interchange-plus-plus arrangement it decomposes the cost of a transaction into interchange, paid to the card issuer; scheme fees, paid to Visa or Mastercard; and the acquirer's own markup. In the United Kingdom, at the time of writing, the UK Interchange Fee Regulation caps interchange on consumer card transactions where the merchant, the acquirer and the card issuer are all within the UK, at 0.2 per cent of transaction value for consumer debit cards including prepaid cards and 0.3 per cent for consumer credit cards, with some American Express cards

outside the regime. Commercial cards and most cross-border transactions fall outside those caps, and UK to European Economic Area card-not-present interchange rose substantially after the UK left the European Union, prompting a Payment Systems Regulator market review of cross-border interchange fees. That is why a fee reconciliation built on one blended assumed rate drifts the moment a business acquires foreign or corporate customers. On a blended contract the same economics arrive as one undifferentiated deduction: simpler to reconcile, impossible to audit.

Leg three is the bank record. In modern form that means ISO 20022: a camt.053 end-of-day statement, a camt.052 intraday report, a camt.054 debit or credit notification. Legacy estates still consume SWIFT MT940 and MT942, or BAI2 in North American systems. All distinguish booking date from value date, and all carry remittance information of wildly varying quality, which is why bank-side matching is harder than processor-side matching. A processor gives you its own reference on every line. A bank frequently gives you eighteen characters of free text typed by a stranger.

A reconciliation is only as strong as the weakest identifier in the chain, so five keys must be persisted as each event happens: the internal payment identifier and idempotency key, the processor's reference, the modification or capture reference for anything amending an earlier event, the payout or settlement batch identifier, and the bank's end-to-end reference on the credit. None can be reconstructed later, and the break a missing one causes will be unclosable rather than merely open.

Why they never match: a taxonomy

These causes account for the overwhelming majority of differences in a card-and-bank-transfer business, and they are worth learning as a taxonomy, because break investigation is mostly the work of assigning an unexplained item to a known category quickly.

Cause	Where it shows	Typical age	How it clears
Fees netted at settlement	Ledger gross, payout net	Same day	Post fee journal from itemised fee data
Settlement timing	Processor paid, bank silent	1 to 3 days	Clears on arrival; monitor the tail
Authorised but never captured	Ledger sale, no processor line	Days to weeks	Write off or re-attempt; fix capture logic
Partial capture	Amounts differ on one reference	Same day	Model capture separately from authorisation
Refund crossing a period	Payout smaller than sales	1 to 30 days	Link refund to original charge, not to the day
Chargeback and representment	Money out, then possibly back	30 to 180 days	Separate dispute sub-ledger
Foreign exchange	Amounts differ by a few per cent	Ongoing	Retranslate; recognise the difference explicitly
Duplicate submission	Two ledger events, one payout line	Same day	Idempotency keys; deduplicate at source
Processor restatement	A closed period changes	Weeks	Versioned correction, never an overwrite
Lump-sum bank credit	One credit, many payouts	Same day	Match many-to-one on batch reference
Direct Debit unpaid	Collection made, then returned	2 to 10 days	Consume the returns file as a first-class event
Missing or partial file	A whole day absent	Hours	File-level completeness control

Four deserve expansion.

Fee netting. Fees break reconciliations not because they exist but because they are deducted at a different point from where they are incurred. Adyen documents that per-transaction payment fees are booked when captured funds are settled, while its own processing fees are invoiced and adjusted monthly, with the adjustment booked separately. A business reconciling on one monthly total therefore finds two fee populations arriving on two cadences, and the difference between them is not an error. Accrue from itemised data, not an assumed rate.

Authorisation versus capture. These are separate events with separate lifetimes, and treating them as one is probably the commonest structural defect in an early-stage ledger. An authorisation places a hold on a customer's available balance; a capture claims it. Authorisations expire. Captures can be partial, so an authorisation for £120 may be captured at £97.40 when an item is out of stock, and a system recognising revenue on authorisation is permanently overstated by the difference. Some schemes and acquirers support incremental authorisation and over-capture within a tolerance, so a captured amount can legitimately exceed the amount authorised, and a validation rule forbidding that will reject good data.

Foreign exchange. A single cross-border card payment can involve four amounts: the transaction currency the customer sees, the presentment currency in which the scheme presents it to the acquirer, the settlement currency in which the acquirer pays, and the functional currency of the books. Each conversion carries its own rate and its own rounding to the minor unit, struck at different moments. Under IAS 21, or Section 30 of FRS 102 for most UK small and medium-sized companies, monetary balances are retranslated at the closing rate with the resulting exchange differences taken to profit or loss. So an unreconciled receivable in euros changes value every month without anybody touching it, and a reconciliation that does not separate genuine breaks from retranslation movement produces an unreadable break register within a quarter. Refunds are worse: a sale converted at January's rate and refunded in April is refunded at April's rate, so a full refund of a full sale does not net to zero and never will.

Direct Debits. For anyone operating on Bacs there is a whole reconciliation layer with no card equivalent, and it is the one most often bolted on late. Bacs runs a fixed three-working-day cycle: day one is input day, when the service user submits the file, the transmission window closing at 22.30 for processing on the following processing day; day two is processing day, when Bacs validates and distributes; day three is entry day, when the payer's account is debited and the service user's credited. Nothing has moved on days one or two, so a ledger recognising cash on submission is wrong for two days on every collection. Then the returns arrive. ARUDD, the Automated Return of Unpaid Direct Debits, reports collections that failed and are being returned, with a reason code attached; ARUCS does the equivalent for unapplied credits; ADDACS carries amendments to and cancellations of instructions; AWACS advises that account details used for a credit were wrong; and indemnity claims, notified through DDICA, reverse a collection under the Direct Debit Guarantee at the payer's bank's initiative, long after the event. Each is a file, each is a financial event, and each must be consumed as a first-class ledger entry rather than read by a human and typed into a spreadsheet. A firm that reconciles Bacs collections but not Bacs returns will show cash it has already given back.

Break investigation and ageing

A break register is a small database and should be treated as one, not as a tab in a workbook. Every open item carries a unique identifier that never changes, the date first detected, the source systems on both sides, the signed amount and currency, a category from a controlled list, a named individual owner, an expected clearing date, a timestamped investigation trail, and a resolution code recorded when it closes.

Three rules separate a register that works from one that is theatre. First, a break with no named owner is not a break, it is a rumour: ownership sits with an individual rather than a team, and the register reports the count of unowned items, which should be zero. Second, ageing is measured gross, and by value as well as by count, because ten thousand two-pence breaks and one four-hundred-thousand-pound break are different problems and a register reporting only counts hides the second behind the first; workable buckets are nought to one day, two to five days, six to thirty days, thirty-one to ninety days and over ninety days, in reconciliation days rather than calendar days. Third, explained is not the same as cleared. An item can be fully understood and still open, because the cash has not arrived. An item can be cleared by a manual journal and still not understood, which is far more dangerous, because the journal removes the symptom and leaves the defect. Every clearing journal should require a second approver and should be counted, and a rising count is among the earliest reliable signs that a reconciliation process has begun to fail.

Ageing thresholds should trigger action, not merely reporting. A serviceable ladder runs: over five days, the owner records a root cause hypothesis; over thirty days, the item escalates to the finance lead and a provision is considered; over ninety days, the item is written off to profit or loss but deliberately kept open in the register, because writing an item off is an accounting decision and closing it is an operational lie. Items written off but held open are the raw material for root cause analysis. Items closed on write-off vanish, and the defect that generated them runs forever.

Reconciliation is also a fraud control, and quietly one of the best available, because internal misappropriation almost always leaves an anomaly in exactly this data: a payout corresponding to no settled transaction, a refund with no matching original, a clearing account that never returns to a plausible balance. The defensive value comes from the segregation of duties around it rather than from the matching itself. Whoever can create a payment must not be whoever can approve a reconciliation adjustment, and neither should be able to amend the break register without leaving a record.

Where the law arrives

For a UK firm holding customer money this is a regulatory obligation with named rules, deadlines and notification duties. What follows is the position at the time of writing, in August 2026.

Authorised payment institutions, small payment institutions that elect to safeguard, electronic money institutions and credit unions issuing electronic money are, as safeguarding institutions, subject to CASS 15, in force since 7 May 2026. Firms providing only payment initiation or account information services do not hold relevant funds and are outside it. CASS 15.8 is the records, accounts and reconciliations chapter, and its architecture is worth knowing precisely.

An internal safeguarding reconciliation compares what the firm's own records say it owes customers, the safeguarding requirement, against what those records say it holds, the safeguarding resource. CASS 15.8.19R requires this as frequently as necessary and no less than once each reconciliation day. That term was a late and practically significant change during the consultation: the Handbook glossary defines a reconciliation day, at the time of writing, as a business day that is not a Saturday or Sunday, Christmas Day, Good Friday or a bank holiday in any part of the United Kingdom, and, where a

reconciliation requires anything to be done by reference to a market outside the United Kingdom, not a day on which that market is not normally open for business. The standard method also includes a forward-looking comparison of the firm's D+1 segregation requirement against its D+1 segregation resource. A firm relying on an insurance policy or guarantee unlimited in cover need not perform an internal safeguarding reconciliation at all, but must record a daily calculation of its safeguarding requirement.

An external safeguarding reconciliation compares the firm's internal records against statements or confirmations from the third parties actually holding the money. Under CASS 15.8.42R it too must be performed as frequently as necessary and no less than once each reconciliation day, and as soon as reasonably practicable after the date to which it relates.

The consequences of a discrepancy are where the rules bite hardest. Where an internal reconciliation reveals a difference between resource and requirement, CASS 15.8.50R requires the firm to determine the reason and then ensure either that any shortfall is paid into a relevant funds bank account or invested in relevant assets as soon as possible and in any case by the end of the day on which the reconciliation is performed, or that any excess is withdrawn. CASS 15.8.51R applies the same end-of-day deadline to a deficiency in the D+1 segregation resource and goes further: where relevant funds cannot remedy it, the firm must use its own, even where that produces a surplus, with the position corrected by later reconciliations. Where an external reconciliation reveals a discrepancy, CASS 15.8.56R requires investigation and all reasonable steps to resolve it without undue delay, unless it arises solely from timing differences between the two parties' accounting systems. And CASS 15.8.57R provides that where the firm cannot immediately resolve such a discrepancy, and one set of records indicates a need for a greater amount of relevant funds or assets than another, it must assume until the matter is finally resolved that the record showing the greater amount is accurate, and pay its own funds in accordingly. The regulatory default is to assume you are short.

Then the notifications. CASS 15.8 requires a safeguarding institution to inform the FCA in writing without delay if its internal records and accounts of relevant funds are materially out of date, inaccurate or invalid; if it will be unable to, or materially fails to, conduct an internal or an external safeguarding reconciliation in compliance with the rules; if it will be unable to, or materially fails to, pay in a shortfall or withdraw an excess after an internal reconciliation; if it will be unable to, or materially fails to, identify and resolve discrepancies after an external reconciliation; or if it becomes aware that at any time in the preceding twelve months the amount of relevant funds safeguarded was materially different from the total aggregate amount it was required to safeguard. That last limb deserves reading twice. Discovering historic under-safeguarding is itself notifiable, so an honest clean-up of a legacy break population is a conversation with the regulator, not a quiet weekend of journals.

Around the reconciliation sit the monitoring obligations. Under SUP 16.14A a safeguarding institution must submit a safeguarding return to the FCA within fifteen business days of each month end, and that return asks directly whether internal safeguarding reconciliations were carried out every reconciliation day during the period. An annual safeguarding audit by a qualified auditor is required, with an exemption at the time of writing for firms not required to safeguard more than £100,000 of relevant funds at any point over a period of at least 53 weeks; the report is due within four months of the period covered, extended to six months for the first report under the new regime. Under SUP 3A.9 the auditor must confirm whether the firm has maintained systems adequate to enable compliance with the relevant funds regime. Oversight must be allocated to a single named director or senior manager.

Investment firms holding client money live under the older parallel regime in CASS 7.15, which requires an internal client money reconciliation each business day under CASS 7.15.15R, based on the

records as at close of business on the previous business day. CASS 15 borrows heavily from CASS 7, which is useful rather than merely tidy: the enforcement history and audit methodology built up around CASS 7 over more than a decade are largely transferable.

The direction of travel is visible elsewhere. The Bank of England's draft Code of Practice for systemic sterling-denominated stablecoin issuers, published in 2026, requires at paragraph 3.5 that a systemic issuer carry out an internal safeguarding reconciliation as regularly as necessary and at least once each business day. Where money is held for others, daily reconciliation is becoming the floor across UK financial regulation. And the Payment and Electronic Money Institution Insolvency Regulations 2021 created a special administration regime whose objectives include returning relevant funds to customers as soon as is reasonably practicable, exactly the objective unreconciled records defeat.

Why young firms actually die here

This part of the chapter is an argument rather than a description, and it should be made without hedging.

The FCA's own published evidence is stark. In consultation paper CP24/20, published in September 2024, it reported that twelve payments firms which safeguarded relevant funds became insolvent between the first quarter of 2018 and the second quarter of 2023, that the weighted average shortfall in client funds across those cases was 65 per cent, and that seven of the twelve showed shortfalls over 50 per cent. Its table of selected averages put mean client funds owed at £11.3m against mean total assets available of £4.0m, a shortfall of £7.3m; for electronic money institutions alone, £16.3m owed against £3.3m available, a shortfall of £13.0m or 80 per cent. Of the six cases in which some funds had been distributed, a first distribution took on average over two years. The FCA was careful to say these shortfall figures are likely underestimates, because the data available did not record safeguarded funds and total assets at insolvency had to serve as a proxy. The same paper recorded that in 2023 it opened supervisory cases about safeguarding arrangements at approximately 15 per cent of the firms that safeguard.

Read those numbers with the reconciliation lens on. A weighted average shortfall of 65 per cent is not embezzlement; embezzlement on that scale across twelve firms would have been a national scandal. It is what happens when firms do not know, at any given moment, what they owe. The requirement side drifted from the resource side over months and years, nobody computed the difference daily, and the gap was first measured on the day an administrator arrived to measure it. The two years to a first distribution is the same failure from the other end: an insolvency practitioner cannot pay customers until they know who is owed what, and the records did not say.

The proximate causes are almost always mundane. Corporate and customer money commingled in one operating account, because opening a safeguarding account was slow and the founders were busy. Revenue recognised on authorisation rather than capture, so the profit and loss account told a happier story than the bank did. Fees accrued at a blended estimate rather than from itemised data, so the estimate drifted into a suspense account nobody owned. A break register that was a spreadsheet, so its oldest item was whatever the last person to open the file had not deleted. Customer float treated as working capital, because float behaves exactly like money right up until the customers ask for it back at the same time.

The legal machinery makes the consequences harder rather than easier. In the administration of the electronic money institution Ipagoo, the Court of Appeal held in 2022 that the Electronic Money Regulations do not impose a statutory trust over funds received from e-money holders, while also holding that a shortfall in the customer asset pool should be topped up. A subsequent judgment following the insolvency of Allied Wallet, [2022] EWHC 1877 (Ch), extended that reasoning to the Payment Services

Regulations, so relevant funds held with authorised payment institutions are likewise not subject to a statutory trust. The FCA's assessment in CP24/20 was that this leaves practitioners seeking directions from the court on the treatment of the safeguarded asset pool, raising costs and delaying returns. All of that litigation is downstream of one fact: the records did not make it obvious which money was whose. Wirecard illustrates the same point at scale. When it collapsed in June 2020 the missing €1.9 billion, said to be held in trustee accounts in Asia, turned out not to exist: an asset asserted on a balance sheet that no independent third-party record ever confirmed.

So why do founders under-invest in this, consistently, across an industry that has watched other firms die of it?

Because reconciliation has no demo. It cannot be shown to an investor, and when it works perfectly its entire visible output is a report saying nothing is wrong. Every hour spent on it is an hour not spent on something a customer can see.

Because its cost curve is deceptive. At a hundred transactions a day, reconciliation genuinely is a spreadsheet and half an hour on a Monday, and the founder who says "we will do this properly later" is right that it is not urgent today. At a hundred thousand transactions a day it is a team, a data pipeline and a rules engine, and the transition does not happen gradually. It happens in the eight weeks after a large partner goes live, on top of a data model designed when none of it mattered. Migrating from "our ledger is the payments table" to an immutable double-entry event log with clearing accounts per counterparty and currency is a rewrite of the most sensitive code in the business, invariably attempted while that code is under maximum load.

Because the early signals look benign. A small unexplained surplus reads as a pleasant surprise rather than a defect, and a growing business generates surpluses constantly through timing alone. Inflows exceed outflows, so the safeguarding account never looks empty, and the difference between what is held and what is owed is masked by growth for as long as growth continues. When growth stops the hole is instantly visible, and it has been there for two years.

Because the first person who would notice has usually not been hired. Reconciliation sits in the seam between engineering, which owns the data and not the balance sheet, and finance, which owns the balance sheet and cannot query the data. It is nobody's specialism until somebody is made responsible by name, which under CASS 15, at the time of writing, is a rule.

And because the consequences are back-loaded and then arrive at once: a qualified audit opinion; a due diligence process that stalls because the acquirer's accountants cannot tie revenue to cash; a notification to the FCA under CASS 15.8 that a reconciliation was not performed; a monthly safeguarding return that has to answer "no"; or a report from a skilled person, required under section 166 of the Financial Services and Markets Act 2000 and paid for by the firm. Any one of those can consume a management team for six months. All are cheaper to prevent than to survive.

What competence looks like

The automatic match rate at T+1, by count and by value, is the headline health metric; in a well-instrumented card business it should sit comfortably above 99 per cent by value on the day after settlement. The absolute value of unexplained items in each ageing bucket matters far more than the net figure. The age of the oldest open item is the most diagnostic number in the register and belongs in the board pack. Mean time to clear, by break category, tells you which defect to fix next. The count of manual clearing journals, and their aggregate value, tells you how much of the match rate is real. And the number of reconciliation days on which the reconciliation was not completed should be zero, because for a safeguarding institution a failure to perform it is itself notifiable.

Around those numbers sit four practices. Persist every identifier at the moment of the event, because none can be reconstructed afterwards. Use idempotency keys on every mutating call, because duplicate submission is the cheapest break to prevent and among the most expensive to unwind. Reconcile file-level completeness before line-level content, because a missing file produces a thousand phantom breaks and one real problem. And write the reconciliation before the product ships, because the data model that makes reconciliation possible is the same one that makes the product correct.

What will date this chapter

Three things, and a reader in 2029 should check all of them before quoting any of the above.

The CASS 15 rule set is explicitly staged. The Supplementary Regime in force since 7 May 2026 is, in the FCA's own framing, an intermediate step. The end-state proposals consulted on in CP24/20, including a statutory trust over relevant funds and a requirement to receive funds directly into a designated safeguarding account, were not taken forward in that form; renamed the Post-Repeal Regime, they await further consultation once HM Treasury has repealed and replaced the Payment Services Regulations 2017 and the Electronic Money Regulations 2011. Rule numbers will move, and the substance may move further.

The regulatory perimeter is being redrawn. Abolition of the Payment Systems Regulator and transfer of its functions to the FCA was confirmed as government policy in the April 2026 consultation response, but requires primary legislation not enacted at the time of writing, so any sentence here naming the PSR as a live regulator has a foreseeable expiry date.

And the plumbing is changing underneath. The Bank of England confirmed in February 2026 that it intends to move the start of CHAPS settlement from 6am to 1.30am in September 2027, and has since consulted on further steps towards near twenty-four-hour, seven-day settlement on its renewed RTGS platform, with Bacs, Faster Payments, the Image Clearing System, Visa, Mastercard and LINK expected to keep their existing timelines under that first extension. The timetable quoted earlier is therefore a snapshot of a moving object. Migration to ISO 20022 continues to widen the structured data on the bank leg, which will make bank-side matching materially easier than the account given here. When the timetable moves, every cut-off assumption built on it moves too, which is a useful reminder that a reconciliation is not a piece of software but a standing relationship with external clocks you do not control.

50.98 Common wrong ideas

Wrong: Reconciliation means making the numbers match. **Right:** It means explaining, line by line, exactly why they differ; Ravi's three totals were all correct, and the discipline is writing a reason against each difference until the reasons add up.

Wrong: There are three records to reconcile. **Right:** A real payments business reconciles between six and fifteen sources, and a single processor issues several reports covering the same period deliberately keyed on different dates, which will never tie to one another without work.

Wrong: A reported difference of nil means the reconciliation passed. **Right:** Two errors of opposite sign cancel, so nil is the beginning of the check; mature processes run on gross signed values and measure the total absolute value of items they cannot yet explain.

Wrong: "As at close of business" is a single moment at which all the records can be photographed. **Right:** CHAPS, Bacs, Faster Payments, LINK, Visa, Mastercard and the Image Clearing System settle at fixed and different times of day, and a bank line carries booking and value dates, so a reconciliation that does not define its clock in writing gives a different answer depending on when it is run.

Wrong: Authorisation and capture are one event. **Right:** They have separate lifetimes; authorisations expire, captures can be partial, and some schemes permit over-capture within a tolerance, so recognising revenue on authorisation overstates permanently and a rule forbidding over-capture rejects good data.

Wrong: A full refund of a full sale nets to zero. **Right:** Across currencies the sale converts at one date's rate and the refund at another's, so it does not net to zero and never will.

Wrong: An item cleared by a manual journal is an item resolved. **Right:** Explained is not the same as cleared and cleared is not the same as understood; the journal removes the symptom and leaves the defect, which is why every clearing journal needs a second approver and a rising count of them is an early sign of failure.

Wrong: A break that has been written off can be closed. **Right:** Writing an item off is an accounting decision and closing it is an operational lie; items written off but held open are the raw material for root cause analysis.

Wrong: A small unexplained surplus is a pleasant surprise. **Right:** A growing business generates surpluses constantly through timing alone, which masks the gap between what is held and what is owed for exactly as long as growth continues.

Wrong: A legacy safeguarding shortfall can be tidied up quietly over a weekend. **Right:** CASS 15.8 makes it notifiable if at any time in the preceding twelve months the amount safeguarded was materially different from the amount required, so an honest clean-up is a conversation with the regulator.

50.99 Chapter summary in 20 lines

1. A card-accepting business has several separate records of the same week, kept by parties who have never spoken to one another, and they will not agree on the first pass.
2. Reconciliation is not the work of making them agree; it is the work of explaining every difference by name.
3. Ravi's till, his card company's report and his bank statement differ by fees, a refund from an earlier period, settlement timing and one sale that never became money, and all three totals are correct.
4. A difference with a name is a reason and is closed; a difference without one is a break and is opened.
5. The most important fact about a break is not its cause but its age, which is why serious operations age their register by value as well as by count.
6. The plain picture understates the problem, because real firms reconcile between six and fifteen sources and one processor issues several reports of a period keyed on different dates.
7. Matching is not the control, since two systems fed by the same faulty upstream event can agree while both are wrong, and errors of opposite sign cancel.
8. There is no single moment called close of business, because the United Kingdom's payment systems settle at fixed and different times and bank lines carry two dates.
9. A reconciliation asserts completeness, existence, accuracy, cut-off and valuation, and equal totals test only a weak form of accuracy.
10. Done properly, the internal leg is a double-entry sub-ledger over an immutable event log, with clearing accounts per counterparty and currency whose balance is by construction the population of open items.
11. The processor leg carries the commercial complexity, interchange, scheme fees and markup, which is why a fee accrual built on one blended assumed rate drifts the moment foreign or corporate cards arrive.

12. The bank leg is the hardest to match, because a processor gives you its own reference on every line and a bank frequently gives you free text typed by a stranger.
13. Five identifiers must be persisted as each event happens, because a missing one makes the resulting break unclosable rather than merely open.
14. Most differences fall into a known taxonomy, of which fee netting, authorisation versus capture, foreign exchange and the Bacs returns files deserve particular attention.
15. Once the money belongs to somebody else this stops being bookkeeping and becomes law, and United Kingdom safeguarding institutions have fallen under CASS 15 since 7 May 2026.
16. CASS 15.8 requires internal and external safeguarding reconciliations on each reconciliation day, with any shortfall paid in by the end of the day the reconciliation was performed, from the firm's own money if customer money cannot be reached.
17. Where a discrepancy cannot be resolved the firm must assume the record showing the greater amount is accurate, and a whole list of failures, including historic under-safeguarding, is notifiable to the FCA.
18. The FCA's own evidence records twelve insolvent safeguarding firms with a weighted average shortfall of 65 per cent and a first distribution taking on average over two years, which is what not knowing what you owe looks like from the outside.
19. Founders under-invest because reconciliation has no demo, its cost curve jumps rather than climbs, its early signals look benign, and nobody owns it until somebody is named.
20. Young financial technology firms rarely die of fraud or of being hacked; they die because they could not prove where the money was, and being unable to prove it is fatal on its own.

Sources, all consulted in August 2026: the FCA Handbook, CASS 15.8, CASS 7.15, SUP 3A.9, SUP 16.14A and the glossary definition of reconciliation day; FCA Policy Statement PS25/12 and Consultation Paper CP24/20, including Table 3 and its methodology caveat; the Payment Services Regulations 2017, the Electronic Money Regulations 2011 and the Payment and Electronic Money Institution Insolvency Regulations 2021; the Bank of England's summary of the RTGS daily timetable, its 2026 papers on extending RTGS and CHAPS settlement hours, and its 2026 draft Code of Practice for systemic sterling stablecoin issuers; Bacs Payment Schemes Limited and Pay.UK on the Bacs cycle, the 22.30 cut-off and the ARUDD, ARUCS, ADDACS, AWACS and DDICA services; the Payment Systems Regulator on the UK Interchange Fee Regulation; HM Treasury's April 2026 response on payment systems regulation; Ipagoo LLP in the Court of Appeal and Allied Wallet [2022] *EWHC 1877 (Ch); Stripe and Adyen developer documentation; and IAS 21 and Section 30 of FRS 102.*

CHAPTER 51

Fraud Detection

51.0 What this chapter gives you

1. You will be able to explain why a fraud system makes two kinds of mistake and why only one of them ever generates an incident report.
2. You will be able to compute the decline threshold for a given basket from the loss given fraud and the margin forgone, and say why it lands near one in three rather than nine in ten.
3. You will be able to show how a rule with seventy-five per cent recall and five per cent precision can make a business nearly eight times worse off while appearing all the while to work.
4. You will be able to distinguish precision, recall and specificity, and explain why at a fraud rate of 0.4 per cent almost all the leverage sits on the false positive rate.
5. You will be able to name the four different problems the word fraud covers, and say which of them no device signal will ever help with.
6. You will be able to separate signals supplied by the party being judged from evidence that party does not produce, and layer the two accordingly.
7. You will be able to say who bears the loss at each of the five decision points in a card transaction, and how United Kingdom strong customer authentication moves it.
8. You will be able to explain why a fraud rate below the reference rate is a commercial asset under the transaction risk analysis exemption, and what happens after two consecutive quarters above it.
9. You will be able to size a manual review queue by economics rather than headcount, and say when a human is worth roughly two pounds an order.
10. You will be able to explain why a model trained only on approved traffic grows confident about a world it has stopped observing, and what a holdout costs and buys.

The plain version

Imagine a doorman on a busy Friday night. He has three answers, not two. He can wave you in, he can turn you away, or he can say “hang on a second” and check something. That third answer is the interesting one, and we will come back to it.

He cannot ask everybody for identification, because the queue would stretch round the block and half of it would go somewhere else. He cannot let everybody in, because after enough bad nights the venue loses its licence. So he reads people. He notices that you arrived alone but keep glancing back at a group, that this is the fourth time in twenty minutes somebody has come to the door in the same brand-new jacket, that you have been here on eleven previous Saturdays and never caused trouble. None of those observations is proof. Every one shifts the odds.

Now here is the part everybody gets wrong. The doorman makes two completely different kinds of mistake and only one is visible. If he lets a troublemaker in, there is a fight and everyone remembers.

If he turns away an ordinary person who came out for a quiet drink, nothing happens at all. No incident, no report, no smashed glass. That person goes somewhere else, tells their friends the place is unwelcoming, and never comes back. The mistake is real, expensive and completely silent.

Fraud detection is that doorman, wired into a computer, running thousands of times a second. Now put money on it.

Nadia sells running shoes online from a unit outside Leicester, about ten thousand orders a month at an average of eighty pounds. The shoes cost her forty-eight pounds a pair, so she keeps thirty-two pounds of gross margin on a typical order.

About forty orders a month are placed with stolen card details. She ships them, and six to ten weeks later the real cardholder notices the charge and disputes it. Three things then go wrong at once. The eighty pounds is taken back out of her account. The shoes are gone. And her payment provider charges her an administration fee for handling the dispute, somewhere around fifteen pounds, set by her contract rather than by any regulator.

So one successful fraud costs Nadia forty-eight pounds of stock plus fifteen pounds of fee, which is sixty-three pounds. Forty of those a month is two thousand five hundred and twenty pounds. That is her fraud problem, whole and entire. One wrongly refused genuine customer costs her thirty-two pounds, the margin she did not make.

Those two numbers are the whole chapter. Sixty-three against thirty-two. Stopping one fraud is worth roughly two lost customers. Not two hundred. Two.

Nadia's first instinct is the one everybody has. Most of the fraudulent orders go to a delivery address that does not match the billing address on the card, so she writes a rule: if the two differ, refuse the order. It works, in the sense that it catches fraud. Thirty of the forty frauds are stopped, one thousand eight hundred and ninety pounds of loss prevented, and if that were the only number on the report she would think she had had a very good month.

But the rule does not only flag fraudsters. It flags the man buying trainers for his nephew's birthday, the student whose card is registered to her parents' house, everybody who has moved recently, everybody shipping to an office. Six hundred genuine orders a month trip the same rule. Six hundred times thirty-two pounds is nineteen thousand two hundred pounds of margin, destroyed silently, with no incident report anywhere.

Add it up. Before the rule, fraud cost her two thousand five hundred and twenty pounds a month. After it, she still loses six hundred and thirty pounds to the ten frauds that got through and has additionally thrown away nineteen thousand two hundred pounds of good business. The rule made her situation nearly eight times worse, and it felt the whole time as though it were working, because the only number anybody looked at was the fraud she stopped.

There are two words for what went wrong and a business reader should know both. **Precision** asks: of everything I stopped, how much was actually bad? Nadia's rule stopped six hundred and thirty orders and thirty were fraud. Precision of about five per cent, meaning nineteen out of every twenty people she refused had done nothing wrong. **Recall** asks: of all the bad things out there, how much did I stop? She stopped thirty of forty, so recall of seventy-five per cent. Her rule had good recall and catastrophic precision, and precision was where the money was.

Now watch what happens when she asks four questions at once. She keeps the address mismatch, but only acts on it when three other things are also true: the account was created less than ten minutes before the order; the same card number was submitted more than once in the session with different

expiry dates; and that delivery postcode has been used by three or more different accounts in the last thirty days.

Each is innocent alone. New accounts are made constantly, people mistype expiry dates, blocks of flats share postcodes. It is the coincidence that is not innocent. The combination fires on forty-five orders a month and twenty-eight are fraud.

Precision has gone from five per cent to sixty-two, recall has slipped from seventy-five to seventy. She now prevents one thousand seven hundred and sixty-four pounds of fraud while destroying five hundred and forty-four pounds of good business, so her total monthly cost is one thousand three hundred pounds against two thousand five hundred and twenty if she did nothing. She is making money on the arrangement for the first time, and she is also letting twelve frauds through every month on purpose.

That is not failure. That is the answer.

Because there is a way to get fraud to exactly zero and everybody knows what it is: refuse every order. Nadia would lose three hundred and twenty thousand pounds of margin a month and never be defrauded again. Zero fraud is trivially available and nobody wants it. So the real question was never how to stop fraud. It was where, on the dial between letting everything through and letting nothing through, the total cost bottoms out.

The dial has a setting, it falls out of those two numbers, and it is genuinely surprising. Nadia should refuse an order when she thinks there is more than about a **one in three** chance it is fraudulent. Not a nine in ten chance. Not "probably fraud". Thirty-four per cent. At that point the sixty-three pounds she saves two-thirds of the time exactly balances the thirty-two pounds she throws away one-third of the time. Above it she should refuse; below it she should ship the shoes and accept the losses, because refusing costs more than being defrauded. Cheap shoes on a fat margin move the dial one way and expensive shoes on a thin margin move it the other. It is a business decision wearing a technical costume.

So what does the machine look at, if not the address?

Velocity. Not "how many", but "how many, in how long, from the same thing". One card tried once is a customer; one card tried nineteen times in four minutes across eleven accounts is not. One delivery address on two orders is a household; one delivery address on sixty orders in a week, all to different names, is a drop. Velocity is the oldest and cheapest signal there is and still one of the most useful, because criminals have to repeat themselves to make money and honest people mostly do not.

Device fingerprinting. When your browser talks to Nadia's shop it says a great deal about itself without being asked: which fonts are installed, how big the screen is, which time zone, how the graphics chip draws a particular curve. Combine forty of those and you get something often unique. It is not you. It is the machine you are sitting at, which matters because a fraudster with a thousand stolen cards is often sitting at one machine.

Behavioural signals. How you type, not what you type. Whether you typed your own name fluently, the way somebody does who has written it ten thousand times, or slowly, the way somebody does who is copying it off a screen. Whether you were on a call the whole time you were paying, which is one of the strongest warnings in the business, because someone talking a stranger through a bank transfer is usually stealing from them.

The network. Nadia alone sees one strange order. Nadia plus four hundred other shops on the same platform sees the same device try eleven of them in ninety seconds, a pattern invisible to any one of

them. The biggest advantage a large processor has over a small one is not better mathematics. It is that it has seen the card before.

And finally the doorman's third answer. Some orders go into a pile for a person to look at. A trained reviewer costs an employer something like forty thousand pounds a year once you count national insurance, pension, a desk and a supervisor, about two pounds per order at a dozen an hour. Two pounds is not nothing when the order only makes thirty-two, so the rule for the review pile is simple and almost never followed: only put an order in front of a human if a human might change the answer.

Where the plain version stops being true

The doorman can see the person. The machine can only see what the person chooses to send it. Every signal above arrives over a wire from the party being judged. The device fingerprint, the browser headers, the claimed time zone, the typing rhythm: all of it is client-supplied, and a determined attacker can influence any of it. This is not a reason to abandon the signals, which remain useful because spoofing consistently is much harder than spoofing once. It is a reason to notice which signals do not come from the client at all. The issuer's view of the account, the network's view of the card across thousands of merchants, a cryptogram generated inside a chip, a passkey bound to hardware: these are expensive to forge precisely because the attacker is not producing them. The defence is layering: cheap client signals to sort the ordinary from the odd, expensive non-client evidence to decide the odd cases.

"Fraud" is at least four different problems and the arithmetic above fits only one. Nadia's case is third-party fraud: a criminal using a real cardholder's details. Beside it sit three others with different physics. First-party misuse, sometimes called friendly fraud, is where the genuine cardholder makes a genuine purchase and then disputes it, and no device signal will ever help because everything about the transaction is authentic. Account takeover is where the criminal is inside the real customer's account, so the history that normally exonerates now vouches for the attacker. And authorised push payment scams invert the problem: the victim makes the payment themselves, correctly authenticated, fully intending to, while being manipulated by someone on the phone. In the United Kingdom that fourth category is now comparable in scale to the first. UK Finance's Annual Fraud Report 2026, published on 15 June 2026 and covering calendar year 2025, reported unauthorised fraud losses of £703.4 million against authorised push payment losses of £576.4 million, the second figure rising nineteen per cent year on year. In APP fraud the model is not looking for a criminal in the session. It is looking for a victim. The signal set, the intervention and the legal consequences are all different.

The plain version put the whole loss on the merchant, and in the United Kingdom that is frequently wrong. Who pays depends on who authenticated, and it moved substantially with strong customer authentication. Under the Payment Services Regulations 2017, as they stood at the time of writing in August 2026, regulation 76 requires a payment service provider to refund an unauthorised transaction no later than the end of the business day following the day it becomes aware of it, unless it has reasonable grounds to suspect fraud by the customer and has notified the authorities under the tipping-off provisions of the Proceeds of Crime Act 2002. Regulation 77, again as at August 2026, caps the payer's own liability at a maximum of thirty-five pounds for losses arising from a lost, stolen or misappropriated payment instrument, and removes even that where strong customer authentication was required but the payer's provider did not apply it, or where the instrument was used in connection with a distance contract. So a false negative and a false positive have different owners at every point in the chain, and "the loss ratio" means one thing to a merchant, another to an acquirer and another to a card issuer. Any conversation about tuning that has not established whose money is at stake is a conversation about nothing.

A score is not a probability, and the threshold is not an engineering decision. Nadia's thirty-four per cent rule only works if the model's output is a genuine probability, meaning that among all orders scored at thirty-four per cent, roughly thirty-four per cent really do turn out to be fraud. Most models out of training do not have that property, and a raw score of eight hundred and fifty on a scale to a thousand means nothing until somebody has mapped it onto reality. Worse, the ground truth arrives late and incomplete: a dispute may not surface for months, and blocked orders never generate any outcome at all, so the system learns only from traffic it approved and becomes confident about a world it has stopped observing.

The technical version

On dating

Everything below is stated as accurate at the time of writing, in August 2026, and this is among the fastest-moving corners of the subject. Three things shifted in the eight months before it was written and are named in place: the FCA's removal of the prescribed contactless limits, effective March 2026; the replacement of Article 22 of the UK GDPR with new Articles 22A to 22D on 5 February 2026; and the continuing revision of Visa's acquirer monitoring thresholds. Scheme programmes are distributed to acquirers rather than published, so treat every scheme figure here as a prompt to check your acquirer's current bulletin.

Where the decision actually gets made

There is no single fraud check. There are five decision points in a card transaction, each with different data, a different latency budget and a different party carrying the loss.

Point	Who decides	Data available	Latency	Who bears the loss
Pre-authorisation screening	Merchant or gateway	Basket, history, device, session behaviour	Tens to hundreds of milliseconds	Merchant, in lost margin
Authentication	Issuer, via EMV 3-D Secure	Merchant-supplied 3DS data plus issuer's own view	Seconds, with a challenge	Shifts on successful authentication
Authorisation	Issuer	Card, account, balance, issuer models, network signals	Milliseconds in the network round trip	Issuer, or merchant if unauthenticated
Fulfilment hold	Merchant	Everything above, plus time and manual work	Minutes to hours	Merchant
Dispute	Issuer, then scheme arbitration	The full record, months later	Days	Per scheme rules and liability shift

EMVCo publishes the authentication specification, the EMV 3-D Secure Protocol and Core Functions Specification, which has run through version 2.3.1.1, covered by EMVCo Specification Bulletin 279 of 11 August 2025. The point of 3DS 2.x is that it carries merchant-supplied risk data into the issuer's decision, permitting a frictionless outcome in most cases rather than a password screen for everybody.

Rules, scorecards and learned models

Practitioners sometimes talk as though machine learning replaced rules. It did not, and a stack with no rules layer usually cannot implement policy. Rules carry what are not statistical judgements: sanctions and blocklist hits, hard regulatory constraints, contractual prohibitions, an instruction from an investigator to freeze a specific device. These must be deterministic, auditable and changeable at three in the morning by somebody who is not a data scientist. A rule engine edited under change control, with every version retained, is a compliance artefact as much as a detection tool.

Gradient-boosted decision tree ensembles trained on tabular features remain the workhorse for scoring the grey zone. They handle mixed data types and missing values, they are fast enough at inference, and their feature attributions are legible enough to survive a model validation review.

Sequence and graph models sit above both. A sequence model treats activity as an ordered series rather than a bag of counters, which lets it notice that a session is shaped wrongly rather than that a single field is wrong. A graph model works over the entity network, with cards, devices, addresses and beneficiaries as nodes and transactions as edges, and earns its keep on organised fraud and mule networks, where the individual transactions are unremarkable and only the topology gives it away.

They fail differently: rules by going stale, tree ensembles by drifting as the population moves, graph models by being expensive and lagging real time. Running all three, with the rules able to override the models, is defence in depth.

Velocity, entity resolution and the feature store

A velocity feature is a count or sum over a sliding window, keyed on an entity. The design questions are which entity, which window, which aggregate. Cards per device in the last hour. Distinct billing postcodes per card in the last day. Declined authorisations per merchant per BIN range in the last five minutes.

Two engineering problems dominate. The first is entity resolution: a counter is only as good as the identity it is keyed on, and identities are messy, since the same card is a different token at every merchant and the same device changes after an update. Real systems maintain an identity graph stitching these together with explicit confidence, and the quality of that graph sets a ceiling on everything downstream.

The second is point-in-time correctness. If a training feature is computed using data that would not have been available at the moment of the decision, the model learns from the future and looks superb in backtest and mediocre in production. This is leakage, and it is the commonest reason a fraud model performs worse live than on paper.

Card testing, also called enumeration, is the canonical velocity problem. From the defender's side it presents as a burst of low-value authorisation attempts with an abnormally high decline rate, often against a narrow range of card numbers, at merchants with a low-friction checkout. The controls exist because of it. Visa's Account Attack Intelligence produces a VAAI Score used to confirm enumerated authorisations, and Visa's Acquirer Monitoring Program carries a separate enumeration ratio. Mastercard's Security Rules and Procedures, Merchant Edition, in the edition dated 11 February 2025 which was current at the time of writing, provides at section 3.12.4 that in the case of a BIN attack at a card-not-present merchant or payment facilitator, the acquirer, its service providers or the affected merchant must begin to capture and transmit CVC 2 values in all card-not-present authorisation request messages within seventy-two hours of detection or notification.

Device and session signals, and their legal wrapper

Device identification in a browser draws on the properties a client exposes: user agent and header ordering, characteristics of the TLS handshake, screen and viewport metrics, font enumeration, canvas and WebGL rendering differences, declared time zone and language, and the coherence between them. On mobile the signal set is richer and more stable, because an app can read hardware and operating system attributes a browser cannot.

Two caveats belong here. First, the entropy available in browsers is deliberately shrinking as privacy changes across the major engines reduce the distinguishing power of these signals, so fingerprint-based identification is a depreciating asset. Second, this is regulated processing. In the United Kingdom, at the time of writing in August 2026, storing information on or gaining access to information stored on a user's terminal equipment engages regulation 6 of the Privacy and Electronic Communications (EC Directive) Regulations 2003, which requires consent subject to a narrow exemption where the access is strictly necessary to provide a service the subscriber has explicitly requested. Firms generally rely on that exemption for payment fraud checks, and on legitimate interests as the lawful basis under Article 6(1)(f) of the UK GDPR, supported by recital 47, which states expressly that processing strictly necessary for the purposes of preventing fraud constitutes a legitimate interest. Neither position survives inspection if it has not been documented, scoped and assessed.

Behavioural signals are a distinct family. Passive behavioural biometrics observe keystroke dynamics, pointer or touch trajectories, hesitation and correction patterns, and device motion from the accelerometer and gyroscope. The interpretive value is comparative: does this session look like this account's previous sessions, and like a person performing a familiar task or an unfamiliar one. In APP scam detection this family carries the load, because there is nothing wrong with the credentials or the device, and the signals that matter indicate coercion or coaching: a long dwell on the payee details, evidence of an active call during the session, hesitation inconsistent with the customer's baseline. UK Finance's 2026 report quoted BioCatch's global advisory director to this effect, that detection must move to identifying social engineering in real time before funds leave the account. One caution fraud teams underweight: behavioural biometrics can incidentally reveal information about a person's health, motor control or disability, and where processing reveals data concerning health it becomes special category data under Article 9 of the UK GDPR, which matters both for the lawful basis and for the automated decision rules below.

The network effect and the mechanics of sharing

The largest structural advantage in fraud detection is population. A model trained on one merchant's traffic sees that merchant's fraud. A model trained across a consortium sees the same device, card or beneficiary as it moves between merchants, which is where organised fraud is visible. Sharing runs through four mechanisms in the United Kingdom and they are not interchangeable.

Scheme-level merchant risk lists are the oldest. Mastercard operates MATCH, the Mastercard Alert to Control High-risk Merchants system, now MATCH Pro, into which acquirers must add terminated merchants under defined reason codes and against which they must query prospective ones; Visa operates an equivalent. These are compliance obligations under the schemes' security rules, not optional intelligence products, and being listed is close to fatal to a merchant's ability to obtain acquiring.

Industry fraud databases operate at consumer and account level. Cifas maintains the National Fraud Database, into which members file confirmed fraud markers and against which they screen, and UK Finance operates fraud intelligence sharing among its members. Pay.UK's Confirmation of Payee, mandated by the Payment Systems Regulator in phases, checks the name a payer types against the

name on the destination account before the payment is sent, a control aimed squarely at APP fraud and invoice redirection.

Statutory gateways sit underneath. The Economic Crime and Corporate Transparency Act 2023 created a direct disclosure gateway allowing certain businesses to share customer information to prevent economic crime, with protection from civil liability for breach of confidence where the conditions are met. That addresses the confidentiality problem, not the data protection one, which still requires a lawful basis, purpose limitation analysis, transparency and an impact assessment. Commercial consortium models are what most merchants buy, and the question usually left unasked in procurement is what happens to the merchant's own data on exit.

Precision and recall, properly

The confusion matrix has four cells and every one has a price.

	Model says fraud	Model says legitimate
Actually fraud	True positive: loss avoided	False negative: chargeback, goods gone, fee
Actually legitimate	False positive: lost margin and customer	True negative: normal business

Precision is true positives divided by everything flagged. Recall, also called the detection rate, is true positives divided by all actual fraud. Specificity, the true negative rate, is rarely quoted and is where the money hides.

The reason is the base rate, and it deserves a worked example because business readers are routinely misled by accuracy figures. Take ten thousand orders with forty frauds, a fraud rate of 0.4 per cent, and a detector with ninety per cent recall and a one per cent false positive rate. It flags thirty-six real frauds and one hundred false alarms, giving precision of twenty-six per cent: three out of four people you refuse have done nothing wrong. Now improve only the false positive rate, from one per cent to 0.2 per cent, changing nothing about recall. The flagged set becomes thirty-six real frauds and twenty false alarms, and precision jumps to sixty-four per cent.

Nothing about the model's ability to detect fraud changed. All the leverage was on the negative class, because the negative class is two hundred and fifty times bigger. At low base rates, small movements in the false positive rate dominate everything else, and any metric averaging over both classes will hide it. That is also why area under the receiver operating characteristic curve is a poor headline metric for fraud, since it is computed across the full range of false positive rates, almost all of which are commercially unusable. Precision-recall curves, or simply recall at a fixed operating point such as detection rate at a one per cent flag rate, tell you what you need to know.

Two refinements separate professional measurement from amateur. Count-weighted and value-weighted metrics diverge, so a system catching ninety per cent of fraud attempts by count while missing the three largest by value has failed; value detection rate belongs beside the count figures on every report. And the labels are wrong. Chargebacks arrive with a lag of weeks to months, and blocked transactions produce no outcome at all, so the false positive rate cannot be measured without deliberately letting some flagged traffic through as a holdout. A team that has never run one does not know its own false positive rate and should not claim to.

Tuning to a loss ratio

The industry expresses fraud performance in basis points of processed volume: fraud losses divided by transaction value, times ten thousand. The operating discipline hangs off that ratio rather than off an absolute number of incidents.

The context for what counts as a normal ratio comes from the regulators. The joint EBA and ECB Report on Payment Fraud published on 1 August 2024, the first in that series, found EEA payment fraud of €4.3 billion in 2022 and €2.0 billion in the first half of 2023, with card fraud on EEA-issued cards running at 0.031 per cent of total value and 0.015 per cent of total number in the first half of 2023. It also found fraud rates on card payments around ten times higher where the counterparty was outside the EEA, where strong customer authentication is not legally required, and that seventy-one per cent of card fraud by value in that period was cross-border. Later editions report the same direction with a wider extra-EEA gap.

That ratio is not merely a management preference. It is a regulatory threshold in two directions. The first is the transaction risk analysis exemption. Under Article 18 of the UK version of the strong customer authentication regulatory technical standards, as they stood at the time of writing in August 2026 and as reflected in the FCA's Technical Standards chapter last updated on 19 March 2026, a payment service provider may decline to apply strong customer authentication to a remote payment it has identified as low risk, provided its fraud rate for that transaction type is at or below the reference fraud rate in the Appendix, the value is within the corresponding exemption threshold value, and a real-time risk analysis has not identified abnormal spending or behaviour, unusual device or software information, malware in the authentication session, a known fraud scenario, an abnormal payer location or a high-risk payee location. That clause turns a fraud team's performance into a commercial asset: below the reference rate you may waive authentication on low-risk payments, which materially improves conversion; above it you may not.

The reference fraud rates in the equivalent EU instrument, Commission Delegated Regulation (EU) 2018/389, remain as follows at the time of writing, with the UK Appendix carrying sterling exemption threshold values rather than euro ones.

Exemption threshold value	Reference fraud rate, remote card	Reference fraud rate, remote credit transfer
500	0.01%	0.005%
250	0.06%	0.01%
100	0.13%	0.015%

Article 19, at the time of writing, defines the calculation precisely: the total value of unauthorised or fraudulent remote transactions, whether or not funds were recovered, divided by the total value of all remote transactions of that type, whether authenticated or executed under any exemption, on a rolling ninety-day basis, subject to the audit review required by Article 3(2). Article 20 sets the consequence: a provider must report immediately to the FCA when a monitored fraud rate exceeds the applicable reference rate, must cease using the exemption for that transaction type and threshold band if the rate is exceeded for two consecutive quarters, and may not resume until the rate has been at or below the reference rate for one quarter and it has notified the FCA with evidence. Article 21 requires quarterly monitoring of fraud values, fraud rates, average transaction values and exemption usage.

Two other UK figures under the same instrument are worth stating with their dates. The low-value remote transaction exemption in Article 16, at the time of writing in August 2026, applies where the

individual transaction does not exceed £25 and either the cumulative value since the last strong customer authentication does not exceed £85 or there have been no more than five consecutive such transactions. And the contactless exemption in Article 11 no longer carries a prescribed limit at all: the FCA announced on 19 December 2025, in a press release accompanying Handbook Notice 136, that it was removing the fixed contactless limits and allowing providers with strong fraud controls to set their own, with the rule changes taking effect in March 2026. The Article 11 text as it stood in August 2026 simply permits the exemption where the provider identifies the transaction as posing a low level of risk. The FCA expected most providers to keep their existing limits, so a firm's operational limit and the regulatory limit are now separate facts to be checked separately. UK Finance reported contactless fraud losses of £46.8 million in 2025, up eight per cent.

The second regulatory direction is the schemes' own monitoring, which binds merchants and acquirers rather than issuers.

Visa's Acquirer Monitoring Program consolidated the older fraud and dispute programmes into a single count-based ratio. The numerator combines card-not-present fraud reports, filed as TC40 records, with non-fraud disputes filed as TC15 records; the denominator is settled card-not-present transactions. A separate enumeration ratio measures enumerated authorisation attempts, approved and declined, against the total count of authorisation attempts, approved and declined. The programme identifies both acquirer portfolios and individual merchants, applies minimum monthly counts before a party enters it, and assesses fees per fraudulent or disputed transaction levied on the acquirer, which passes them on by contract. The thresholds themselves are published by Visa rather than confined to acquirer bulletins: its fact sheet "Visa Acquirer Monitoring Program Overview", on the corporate Visa site, puts an acquirer portfolio above standard at a VAMP ratio of fifty basis points and excessive at seventy, sets the excessive merchant threshold in Asia-Pacific, Canada, the European Union and the United States at two hundred and twenty basis points falling to one hundred and fifty on 1 April 2026, requires fifteen hundred combined fraud and dispute events in a month before a party is identified at all, and puts the enumeration criteria at a ratio of two thousand basis points on a minimum of three hundred thousand enumerated authorisations. Those figures have been revised more than once since the programme launched and are reported inconsistently by third parties, so check them against Visa's current fact sheet and against your acquirer's own bulletin before relying on any of them.

Mastercard operates an Excessive Chargeback Program and an Excessive Fraud Merchant programme of similar structure, keyed on fraud chargeback basis points, absolute fraud value and, distinctively, the merchant's take-up of EMV 3-D Secure. Separately and verifiably, the Mastercard Security Rules and Procedures, Merchant Edition, in the 11 February 2025 edition current at the time of writing, provides at section 3.12.4 that an acquirer must ensure that each merchant which has exceeded one hundred gross basis points in fraudulent card-not-present transactions for two consecutive calendar months captures and transmits the CVC 2 value for all mail order and telephone order transactions, and for e-commerce either captures and transmits CVC 2 or becomes Mastercard Identity Check enabled, within two months following the second trigger month.

So there are two floors under the loss ratio. There is the commercial floor, where fraud losses plus lost good business plus operating cost are minimised, and a regulatory or scheme floor, above which you lose exemptions, lose the frictionless authentication path and start paying per-transaction penalties. The second is often higher, which means a merchant optimising purely on expected value can find itself out of compliance while making money.

The expected-value calculation is the one from the plain version, stated properly. Let the calibrated probability of fraud be p, the loss given fraud be L, and the contribution margin forgone on a wrongly

declined order be M. Decline when pL exceeds (1 - p)M, that is when p exceeds M divided by (M + L). With the shoe retailer's L of £63 and M of £32 the optimal threshold is 32 divided by 95, approximately 33.7 per cent. Three consequences follow and all three are routinely missed. The threshold moves with the basket, not just the customer: a high-margin, low-cost digital good has a small L and a large M, so the threshold rises. Calibration is not optional, because if the score is not a probability the threshold is meaningless, and calibration decays faster than discrimination. And L is not just the goods; it includes the dispute fee, the operational cost of the case, the effect on the scheme monitoring ratio and any downstream penalty. Firms modelling L as the transaction value alone systematically under-block.

Manual review queues and their economics

A review queue is a third decision bucket and should be sized by economics rather than headcount, which is the reverse of how it usually happens. A reviewer on a thirty-thousand-pound salary costs an employer roughly forty thousand fully loaded once employer national insurance, pension, workspace, tooling and supervision are counted. Against roughly seventeen hundred productive hours a year that is about twenty-four pounds an hour, and at twelve cases an hour the marginal cost of a review is about two pounds.

Now test whether the queue earns it. Take three hundred orders a month in the grey band, of which twenty are fraud and two hundred and eighty genuine, on the shoe retailer's numbers. Auto-approving the band costs twenty times sixty-three pounds, or £1,260. Auto-declining it saves that but destroys two hundred and eighty times thirty-two pounds, a net loss of £7,700, so declining is clearly wrong. Reviewing all three hundred at a reviewer accuracy of ninety per cent on fraud and ninety-seven per cent on genuine orders costs six hundred pounds of review time, plus two missed frauds at £126, plus about eight wrongly declined genuine orders at £269, totalling roughly £995. The queue saves about two hundred and sixty-five pounds a month. It is worth running, barely. Raise the average order value to four hundred pounds on the same margin percentage and the same queue saves thousands, which is why review is standard in travel, jewellery and electronics and marginal in low-value retail: the economics are driven overwhelmingly by order value.

Four operating principles follow. Queue only where a human has information the model lacks: a reviewer seeing the same features on the same screen is an expensive, less consistent copy of the model, whereas one who can make an outbound call or verify a document earns their cost. Measure reviewer precision and recall as you would a model's, by seeding the queue with blind duplicates of resolved cases; reviewers drift, tire and develop idiosyncratic heuristics, and on cases where the model was already confident they are frequently worse than it is. Treat latency as a cost, because conversion decays measurably with delay. And do not let capacity set the threshold: the commonest failure in real operations is a threshold chosen so the queue is exactly as long as the team can clear, which means the risk appetite of the business is set by its recruitment budget.

The legal position on automated decisions

Fraud decisions refuse people service, sometimes close accounts, and sometimes place a marker other institutions will see. That engages data protection law directly, and the UK position changed recently enough that many internal policies had not caught up at the time of writing.

Section 80 of the Data (Use and Access) Act 2025 replaced Article 22 of the UK GDPR with new Articles 22A to 22D, brought into force on 5 February 2026 by the Data (Use and Access) Act 2025 (Commencement No. 6 and Transitional and Saving Provisions) Regulations 2026. The old general prohibition on solely automated decisions with legal or similarly significant effects has become a safeguards regime. Under Article 22A a decision is based solely on automated processing where there is no meaningful human involvement, and is significant where it produces a legal effect or a similarly significant one.

Under Article 22B a significant decision based entirely or partly on special category data may not be taken solely automatically unless explicit consent applies, or the decision is necessary for a contract or required or authorised by law and the substantial public interest condition in Article 9(2)(g) is met. Under Article 22C, where a significant decision is taken solely automatically, the controller must have safeguards giving the individual information about the decision and enabling them to make representations, to obtain human intervention and to contest it.

For a fraud team this converts into four obligations: know which decisions are significant, know which are solely automated, demonstrate that any human in the loop is meaningfully involved rather than clicking through, and have a working route for a customer to contest an outcome. A rubber-stamp review queue is not a compliance control; after these provisions it is arguably worse than no queue at all, because it asserts a human involvement that does not exist.

Two adjacent points. In the European Union the AI Act classifies AI systems used to evaluate the creditworthiness of natural persons as high risk under Annex III, but with an express exception: systems used for the purpose of detecting financial fraud are carved out. The carve-out attaches to the fraud detection purpose, not to any model a fraud team happens to run. And in the United Kingdom, banks, building societies and PRA-designated investment firms holding internal model approval for regulatory capital have been subject to the PRA's Supervisory Statement SS1/23 on model risk management principles since it took effect on 17 May 2024, whose five principles on identification and tiering, governance, development and use, independent validation and mitigants apply across model types rather than only to capital models. Firms outside that scope adopt the framework voluntarily, because the alternative when a fraud model misbehaves is having no answer to who validated it.

What breaks

Fraud models degrade for reasons ordinary models do not. The data-generating process is an adversary who observes your decisions and adapts, so performance decay is not drift but response. Champion-challenger deployment, where a shadow model scores the same traffic without acting on it, is the minimum; holdout groups are the only way to measure true precision, and their cost is the price of knowing.

Two failure modes are structural rather than technical. The first is the feedback loop: a model trained on approved traffic loses sight of the population it declines, grows more confident about a shrinking world, and drifts without any metric moving. The second is metric substitution: a team measured on fraud losses will reduce fraud losses, and the only thing stopping it destroying the business is that somebody else is measured on approval rate. Fraud detection is one of the few disciplines where the correct governance structure is two people with opposed incentives and a shared number.

It is worth closing on what all this exists to change. The Payment Systems Regulator's dashboard updated on 30 July 2026 recorded that in the eighteen months from 7 October 2024 to 31 March 2026, eighty-eight per cent of in-scope APP scam losses, some £316 million, were reimbursed to victims, with eighty-two per cent of claims closed within five business days and the maximum level of reimbursement set at £85,000 per claim under Specific Direction 20, following the PSR's Policy Statement PS24/7. Those figures are current at the time of writing in August 2026 and will be superseded. The structure they describe, in which detection quality decides both how much is stolen and how much a firm must hand back, will not be.

51.98 Common wrong ideas

Wrong: The goal is to stop fraud. **Right:** Zero fraud is trivially available by refusing every order, and nobody wants it; the goal is the setting of the dial at which fraud losses plus destroyed good business plus operating cost are lowest.

Wrong: A rule that catches most of the fraud is a good rule. **Right:** Nadia's address-mismatch rule had seventy-five per cent recall and about five per cent precision, and it destroyed nineteen thousand two hundred pounds of margin to prevent one thousand eight hundred and ninety pounds of loss.

Wrong: You should decline an order when it is probably fraudulent. **Right:** Decline when the calibrated probability exceeds M divided by (M + L), about thirty-four per cent on the shoe retailer's numbers, because refusing a good order costs money too.

Wrong: The loss given fraud is the value of the transaction. **Right:** It is the stock, the dispute fee, the operational cost of the case, the effect on the scheme monitoring ratio and any downstream penalty, and firms modelling it as transaction value alone systematically under-block.

Wrong: A score of eight hundred and fifty out of a thousand means the order is very likely fraud. **Right:** A score is not a probability until somebody has mapped it onto outcomes, and calibration decays faster than discrimination.

Wrong: Machine learning replaced rules. **Right:** Rules carry what are not statistical judgements, sanctions and blocklist hits, hard regulatory constraints, an investigator's freeze, and they must stay deterministic, auditable and changeable at three in the morning by somebody who is not a data scientist.

Wrong: Device fingerprinting identifies the person. **Right:** It identifies the machine, its available entropy is deliberately shrinking as browser privacy changes land, and in the United Kingdom it engages regulation 6 of the Privacy and Electronic Communications Regulations and needs a documented lawful basis.

Wrong: The merchant bears the fraud loss. **Right:** Who pays depends on who authenticated, and regulations 76 and 77 of the Payment Services Regulations 2017 place much of it on the payer's provider, so any tuning conversation that has not established whose money is at stake is a conversation about nothing.

Wrong: You know your false positive rate. **Right:** Blocked orders generate no outcome at all, so it cannot be measured without deliberately letting some flagged traffic through as a holdout, and a team that has never run one does not know its own rate.

Wrong: A manual review queue is a safety net worth having. **Right:** It earns its cost only where the human holds information the model lacks, and since Articles 22A to 22D took effect a rubber-stamp queue is arguably worse than none, because it asserts a human involvement that does not exist.

51.99 Chapter summary in 20 lines

1. A fraud system is a doorman with three answers, wave in, turn away, or hold for a look, running thousands of times a second with money on it.
2. It makes two kinds of mistake, and only the fraud it lets through leaves a trace; the good customer it refuses simply goes elsewhere and says nothing.
3. For Nadia the two numbers are sixty-three pounds lost to a successful fraud and thirty-two pounds of margin lost to a wrongly refused order.

4. Stopping one fraud is therefore worth about two lost customers, not two hundred, and every design decision in the chapter follows from that ratio.
5. Her address-mismatch rule caught three-quarters of the fraud and left her nearly eight times worse off, because nineteen of every twenty orders it stopped were innocent.
6. Precision asks how much of what you stopped was actually bad, recall asks how much of the bad you stopped, and the money was in precision.
7. Combining four signals that are each innocent alone lifted precision from five per cent to sixty-two and turned a loss into a gain, while deliberately letting twelve frauds a month through.
8. Zero fraud is available to anyone willing to refuse every order, so the real question was always where the total cost bottoms out.
9. The optimal decline threshold is M divided by (M + L), about thirty-four per cent here, and it moves with the basket rather than only with the customer.
10. The workhorse signals are velocity, device fingerprinting, behavioural patterns and the network, and the largest structural advantage in the field is simply having seen the card before.
11. Every client-supplied signal can be influenced by the party being judged, so the defence is layering cheap client signals against expensive evidence the attacker does not produce.
12. Fraud is at least four problems, third-party fraud, first-party misuse, account takeover and authorised push payment scams, the last now comparable in United Kingdom scale to the first.
13. There are five decision points in a card transaction, each with different data, latency and loss-bearer, and United Kingdom law decides where the loss lands at each.
14. Rules, tree ensembles and graph models fail differently, so running all three with the rules able to override the models is defence in depth.
15. Velocity features are only as good as the entity resolution beneath them, and point-in-time correctness is what separates a model that works live from one that merely backtests well.
16. At a fraud rate of 0.4 per cent, moving the false positive rate from one per cent to 0.2 per cent lifts precision from twenty-six to sixty-four while changing nothing about detection.
17. Labels arrive late and incomplete, so value-weighted metrics belong beside count-weighted ones and a holdout is the only honest measure of false positives.
18. The loss ratio has two floors, the commercial one where total cost is minimised and a regulatory and scheme one below which exemptions, frictionless authentication and freedom from per-transaction penalties are retained.
19. A review queue is a third bucket to be sized by arithmetic, and the commonest operational failure is letting the recruitment budget set the risk appetite.
20. Fraud models decay because the data-generating process is an adversary that watches your decisions, and the correct governance is two people with opposed incentives and a shared number.

Sources, all consulted in August 2026: the FCA Handbook Technical Standards on Strong Customer Authentication, Chapter 3, as updated 19 March 2026; the FCA press release of 19 December 2025 on contactless limits and Handbook Notice 136; Commission Delegated Regulation (EU) 2018/389; the Payment Services Regulations 2017 and the Data (Use and Access) Act 2025 section 80 with SI 2026/82; the PSR's APP scams reimbursement dashboard of 30 July 2026, PS24/7 and Specific Direction 20; UK Finance's Annual Fraud Report 2026; the joint EBA and ECB Report on Payment Fraud of 1 August 2024; Mastercard Security Rules and Procedures, Merchant Edition, 11 February 2025; EMVCo Specification Bulletin 279; PRA Supervisory Statement SS1/23; and the EU AI Act Annex III.

52

CHAPTER

Anti-Money-Laundering

52.0 What this chapter gives you

1. You will be able to explain why what gets a criminal caught is almost never the money itself but the story attached to it.
2. You will be able to work through how a cash business launders by over-reporting its takings, and say why paying income tax on money nobody earned is a cost of laundering rather than a barrier to it.
3. You will be able to say what job each onboarding question does, so that an eleven-day account opening reads as a set of purposes rather than as obstruction.
4. You will be able to keep sanctions and money laundering apart in your head: suspicion and reasonableness on one side, absolute prohibition and strict liability on the other.
5. You will be able to explain why a screening run producing 120 possible matches and two real ones is working correctly, and why better software does not fix it.
6. You will be able to say why the risk-based approach governs due diligence but never sanctions screening.
7. You will be able to explain the "more than 25 per cent" beneficial ownership threshold and at least three structures that defeat it while leaving control untouched.
8. You will be able to describe what happens to a customer's money after a refused Defence Against Money Laundering request, and how long it can lawfully stay immobilised.
9. You will be able to explain why a bank exits a whole category of customers whose individual members are fine, and why that makes the underlying problem worse rather than better.
10. You will be able to read the SARs and enforcement figures and say what the regime actually is: a privately funded intelligence-gathering exercise measured on process rather than on outcomes.

The plain version

Take a fifty pound note out of your pocket and look at it. There is nothing on it that says where it came from. It does not know whether it was handed over for a week's wages, given at a birthday, or taken out of a till at knifepoint. A note is a note.

Hold on to that, because everything else follows from it. What gets a criminal caught is almost never the money itself. It is the story attached to the money.

If you turn up at a car showroom with £30,000 in a sports bag, nobody has to test the notes. They just have to ask one question: where did that come from? And you have to have an answer that survives being asked twice. Money laundering, stripped of every long word ever attached to it, is the manufacture of believable answers to that question. Anti-money-laundering is the business of asking it properly.

Here is how the manufacturing works, with real numbers, because the arithmetic is the trick. Imagine a hand car wash on a busy road: five people with sponges and a jet hose, £8 a wash, open seven days. Genuinely, they wash about ninety cars a week. That is £720 of honest takings, nearly all in cash, which is completely ordinary for a car wash.

Now suppose the owner also has £1,680 a week in cash from something illegal. They do not need to hide it. They need to *explain* it. So each week they write down not ninety washes but three hundred, and bank £2,400 instead of £720. The bank sees a car wash on a busy road banking £2,400 a week. That is believable. Nobody counts cars.

Over a year that is £87,360 of criminal cash with a birth certificate. It has been taxed. It appears in accounts. It can buy a flat. And this is the part people find hardest to accept: the owner will happily pay income tax on money they never earned, because a tax bill is the receipt that proves the story. Paying tax is a cost of laundering, not a barrier to it.

People who do this professionally break it into three moves. First, get the cash into the system at all, the risky part, because cash is heavy, conspicuous, and the one moment where a person and a pile of money are in the same room. Second, move it around enough that the trail becomes hard to follow: car wash to supplier, supplier to a company abroad, company to a loan, loan to a property. Third, spend it on something ordinary, so from then on the money looks like the proceeds of a house sale rather than of a crime.

Every anti-money-laundering rule you have ever been annoyed by exists to interfere with one of those three moves. Which brings us to why opening a bank account takes so long.

Anaya is starting a small import business. The bank asks her for what feels like an absurd amount. Passport. Proof of address. What the business actually does. Expected turnover. Which countries she will trade with. Who else owns shares. Where her opening deposit came from. It takes eleven days, and twice she is asked for something she has already sent.

None of that is the bank being difficult for sport. Each question does a job. Who are you is the easy one, and the least important. What does the business do gives the bank a shape to compare against later: an importer of textiles moving £40,000 a month is unremarkable, and the identical account moving £40,000 a month with no goods attached is not. Who else owns it matters because the person who signs the forms is often not the person who benefits, and the whole point of a shell company is that it puts a stranger's name on the door. And where the money comes from matters because a plausible source is exactly what a launderer cannot produce without lying, and a lie written down is evidence.

Then the bank runs her name against a list. Not a list of criminals: a list of people and organisations governments have ordered everyone to freeze out entirely. This is a different thing from money laundering and worth separating in your head now. Money-laundering rules say be careful and tell someone if it looks wrong. Sanctions say do not, under any circumstances, move this person's money, and there is no version of that where being careful is good enough.

The list check is where the system's most human failure lives, and the arithmetic is unkind. Say the bank takes on 4,000 new customers a week. Its screening software does not look for exact matches, because anyone wanting to evade an exact-match check could do it by dropping a letter. So it looks for near matches: same name spelled differently, name written back to front, a name that sounds the same aloud, a name transliterated from another alphabet where there was never one correct English spelling to begin with. Set the software to be sensitive and it catches more; set it to be strict and it misses things.

Set sensibly, it produces roughly 120 possible matches out of those 4,000. Perhaps two are real. The other 118 are people who share a name, or half a name, with somebody the software has heard of.

That is not a bug you can fix by buying better software. It is a property of names. Hundreds of millions of people share a few thousand very common names, and the software is deliberately built to over-report because the cost of missing one is enormous and the cost of stopping an innocent person for ten minutes is small. So a person with a common name will be stopped, and stopped again at the next bank, and will reasonably conclude the system has something against them personally. It does not. It cannot tell them apart from someone else, and it is not allowed to guess. At eight minutes each, those 118 wrong matches cost about sixteen hours a week to find two real ones. Multiply that across every bank in the country and you have an industry.

Once Anaya's account is open, the watching starts. Software compares what she actually does against what she said she would do, and against what businesses like hers normally do. A hundred thousand customers might generate two thousand alerts a month. The great majority are nothing: a bonus, an inheritance, a wedding, a house sale, a business that grew. Perhaps forty end up reported to the authorities. And when a bank does decide something looks wrong, it must not tell the customer, because warning someone that they have been reported is itself a criminal offence in the United Kingdom. So the customer gets a delay with no explanation, from staff legally forbidden to explain it. Almost every furious story you have heard about a bank refusing to say why has this at the bottom of it.

The last idea, and the one that governs the rest, is that you cannot do all of this equally to everybody. There is not enough time in the world. So you spend effort where the risk is: light checks on a teenager opening a savings account, heavy checks on a company with owners in four countries and a bank account in a fifth. That is the risk-based approach, and it sounds like common sense because it is. It is also the hardest thing in this chapter to do honestly, because it requires a firm to write down, in advance, which of its own customers it thinks are dangerous.

Where the plain version stops being true

The system is not primarily a net for catching launderers, and its own numbers say so. The car wash story implies that reports lead to arrests. Some do. But the regime measures firms on process, not outcomes: a firm complies by having systems, applying them and reporting suspicion, and it can do all three impeccably while the money still gets away. The United Nations Office on Drugs and Crime has long estimated that 2 to 5 per cent of global GDP is laundered annually, roughly $800 billion to $2 trillion, with its 2011 study putting the 2009 figure at around $1.6 trillion, or 2.7 per cent. Against that, the United Kingdom Financial Intelligence Unit reported in its SARs Annual Report 2025, covering April 2024 to March 2025, that £382.6 million was denied to suspected criminals through Defence Against Money Laundering requests, a record and a 59 per cent rise on the previous year's £240.1 million. In the United States, the Bank Policy Institute's 2018 study *Getting to Effectiveness* found that surveyed institutions reviewed approximately 16 million alerts in 2017 and filed over 640,000 suspicious activity reports and more than 5.2 million currency transaction reports, and that among institutions recording the data, a median of 4 per cent of those reports drew any follow-up inquiry from law enforcement. Whatever the regime is, it is not a filter with a high hit rate. It is a very large intelligence-gathering exercise, paid for privately, from which a comparatively small number of interventions are extracted.

Identity is the easy half; ownership is the hard half, and the analogy skips it entirely. Verifying that Anaya is Anaya is close to a solved problem. Establishing who actually benefits from a company is not. At the time of writing in August 2026, regulation 5 of the Money Laundering, Terrorist Financing and Transfer of Funds (Information on the Payer) Regulations 2017 defines the beneficial owner of a body

corporate as an individual who exercises ultimate control over its management, or who ultimately owns or controls, directly or indirectly and including through bearer share holdings or by other means, more than 25 per cent of its shares or voting rights, or who otherwise controls it. That "more than 25 per cent" is a bright line, and bright lines are designed around. Four people at 25 per cent each disclose nobody. A chain of five companies across three jurisdictions dilutes an owner below the threshold at every layer while leaving control untouched. Nominee shareholders, discretionary trusts and companies limited by guarantee each break the arithmetic differently. This is why practitioners say ultimate beneficial ownership rather than ownership, and why the honest answer to who owns this company is often a judgement rather than a fact.

"False positive" is the wrong frame for screening, and screening is not part of the risk-based approach at all. A wrong name match is not an error. It is the intended output of a system deliberately tuned to over-report, because the two kinds of mistake are not symmetrical: missing a genuine sanctions target is a strict-liability breach, while stopping an innocent customer is an inconvenience. Every screening system therefore sits on a dial between precision and recall, and the dial is set where the penalties push it. More importantly, the risk-based approach does not apply here. A firm may decide a low-risk customer needs less due diligence. No firm may decide a low-risk customer needs less sanctions screening, because an asset freeze is an absolute prohibition. At the time of writing, section 146 of the Policing and Crime Act 2017, as amended by the Economic Crime (Transparency and Enforcement) Act 2022, allows the Office of Financial Sanctions Implementation to impose a civil monetary penalty where it is satisfied on the balance of probabilities that a breach occurred, and for breaches on or after 15 June 2022 any requirement that the person knew, suspected or believed anything is disregarded. Intention is irrelevant. That is a different legal world from money laundering, where suspicion and reasonableness are everything.

De-risking is not a failure of the risk-based approach; it is a rational response to how the penalties are shaped. A bank does weigh a customer's risk against their value, but the two sides of that scale are measured in different units. The revenue from a small money-transfer business serving one remittance corridor is bounded and known. The downside is an enforcement action, a skilled person's report, a criminal investigation and a number in the press: the Financial Conduct Authority stated when fining Nationwide Building Society £44,078,500 in December 2025 that since 2021 it had imposed thirteen fines totalling £300,767,526 on banks for anti-money-laundering systems and controls failings. Against an unbounded and asymmetric downside, exiting an entire customer category is the cheapest available answer even when the individual customers are fine. The consequence lands hardest on those with least ability to complain, and it is a policy problem rather than a compliance one.

The technical version

On dating

Every figure, threshold, list and rule below is stated as accurate at the time of writing, in August 2026, and this is the fastest-moving corner of regulation covered in this book. Three things were in motion as it was written: the United Kingdom's Money Laundering and Terrorist Financing (Amendment) Regulations 2026 came into force on 30 June 2026 with two later commencement dates still ahead of it, the European Union's single rulebook does not apply until 10 July 2027, and OFSI had announced an intention to double its maximum civil penalty without the statutory cap yet changing. Check the primary sources before relying on any number here.

The offences that drive the machine

In the United Kingdom the regulatory architecture rests on criminal law, and the criminal law came first. Part 7 of the Proceeds of Crime Act 2002 creates the principal offences: concealing, disguising, converting, transferring or removing criminal property under section 327; entering into or becoming concerned in an arrangement facilitating another's acquisition, retention, use or control of criminal property under section 328; and acquiring, using or possessing criminal property under section 329. Criminal property is defined by reference to a person's benefit from criminal conduct, with no minimum amount and no separate list of predicate offences: if the underlying conduct would be an offence in the United Kingdom, its proceeds are criminal property, however small.

Two further offences shape day-to-day behaviour more than the principal ones. Sections 330 and 331 create the offence of failing to disclose knowledge or suspicion arising in the course of regulated-sector business, with equivalents at sections 19 and 21A of the Terrorism Act 2000. Section 333A creates the tipping-off offence and section 342 the offence of prejudicing an investigation, with equivalents at sections 21D and 39 of that Act. The National Crime Agency states, at the time of writing, that the penalties on conviction on indictment for failing to disclose are up to five years' imprisonment, a fine, or both. This is why "we are unable to discuss the reason for this decision" is a legal formula rather than rudeness.

The defence matters as much as the offence. A person who would otherwise commit a principal offence can obtain a defence by making an authorised disclosure and obtaining appropriate consent: in practice, the Defence Against Money Laundering request, or DAML. Under section 335 the National Crime Agency has seven working days beginning with the first working day after disclosure in which to refuse consent. If it refuses, a moratorium period of 31 days runs, during which the prohibited act must not be done. Section 336A allows a court, on application by a senior officer, to extend that moratorium by up to 31 days at a time where an investigation is being conducted diligently and expeditiously, but subsection (7) caps further extensions so the total cannot exceed 186 days beginning with the day after the end of the initial 31-day period. A customer's funds can therefore be lawfully immobilised for over half a year on suspicion, with no obligation on anyone to explain why.

Section 339A sets a threshold below which deposit-taking bodies, electronic money institutions and payment institutions may operate an account without seeking a DAML. That threshold was raised from £250 to £1,000 by legislation introduced in January 2023, and the UKFIU attributes the subsequent flattening of DAML volumes partly to that change.

The regulations that make it a system

The criminal law tells a firm what it must not do. The Money Laundering, Terrorist Financing and Transfer of Funds (Information on the Payer) Regulations 2017, now maintained under sections 49 and 54(2) of, and Schedule 2 to, the Sanctions and Anti-Money Laundering Act 2018, tell it what it must build.

At the time of writing, the most recent substantial change is the Money Laundering and Terrorist Financing (Amendment) Regulations 2026, S.I. 2026/621, made on 9 June 2026, most of it in force 21 days later on 30 June 2026. Two parts commence later: regulation 20, inserting a new regulation 34A on enhanced due diligence for cryptoasset exchange providers and custodian wallet providers in correspondent relationships, on 1 February 2027; and regulation 37, substituting Schedule 6B on changes in control of registered cryptoasset businesses, for most purposes on 25 October 2027. The instrument implements the Government's response to HM Treasury's 2024 consultation *Improving the effectiveness of the Money Laundering Regulations*, published on 17 July 2025, and its direction is broadly deregulatory.

Its most visible change is arithmetical: every euro-denominated threshold on the face of the Regulations was converted to sterling, and not always at par.

Provision	Subject	Before	After
Reg. 27(1)(b)	Occasional transaction, currency exchange or money transmission	€1,000	£800
Reg. 27(2)	Occasional transaction, general threshold	€15,000	£12,000
Reg. 27(3)	Occasional transaction, high value dealers and others	€10,000	£10,000
Reg. 27(5), (7)	Casino and related transactions	€2,000	£2,000
Reg. 14	High value dealers, casinos, auction platforms, art market participants	€10,000	£10,000
Reg. 38(1), (2)	Electronic money, simplified due diligence and redemption limits	€150, €50	£150, £50
Regs. 64C(4), 64G(1)(b)	Transfers of funds, information requirements	€1,000	£800

Three other changes matter more than they look. The phrase "complex or unusually large" in regulations 19 and 33 became "unusually complex or unusually large in each case given the nature of the transaction", a deliberate narrowing: a transaction is no longer high risk merely because it is structurally complicated. A new regulation 30ZA lets a credit institution open an account for a customer of a failed bank, and allow transactions from it, before completing due diligence, provided the customer is identified under regulation 28(2)(a) and the relationship begins within 30 days of the bank insolvency order. And a new block at regulation 29(10) to (18) governs pooled client accounts, requiring the firm to understand the account's purpose and proposed use, be satisfied this is consistent with what it knows of the customer, assess and mitigate the resulting risk, and demonstrate all of it to its supervisor. It also binds the customer, who must on request identify the persons whose money is held and their beneficial owners, keep written records of every payment in and out for five years, and provide information to law enforcement, subject to legal professional privilege.

Customer due diligence and beneficial ownership

Regulation 27 sets the triggers: establishing a business relationship, an occasional transaction above the applicable threshold, suspicion of money laundering or terrorist financing, and doubts about documents or information previously obtained. Regulation 28 sets the content: identify the customer and verify their identity from a reliable source independent of the customer; identify any beneficial owner and take reasonable measures to verify their identity so the firm is satisfied it knows who they are; where the customer is a legal person, understand its ownership and control structure; and assess the purpose and intended nature of the relationship. Regulation 28(11) requires ongoing monitoring, including scrutiny of transactions for consistency with knowledge of the customer and keeping information up to date.

Two misreadings deserve naming. Verification is not document collection: a passport photocopy verifies nothing unless the firm has satisfied itself the document is genuine and belongs to the person presenting it. And know your customer is not a one-off event at onboarding; regulation 28(11)'s ongoing obligation is what enforcement actions most often find missing, because the burden of periodic review across a large book is enormous and the incentive to defer it constant. Record-keeping runs under regulation 40: five years from the date the firm knows or has reasonable grounds to believe the relationship ended, after which regulation 40(5) requires deletion of the personal data unless another legal basis applies.

On ownership, regulation 5 covers bodies corporate and regulation 6 partnerships and trusts. The 25 per cent threshold in regulation 5(1)(b) is the number practitioners quote, but the limbs either side of it matter as much: an individual exercising ultimate control over management, and an individual who otherwise controls the body, are beneficial owners regardless of shareholding. Where none can be identified after all reasonable measures, the senior person responsible for managing the body may be treated as such, a fallback that should be rare and is not.

The public registers are the second leg. The United Kingdom's register of people with significant control, at Companies House, uses a threshold aligned to 25 per cent, and regulation 30A requires relevant persons to report discrepancies between what a customer tells them and what the register says. Register quality was for years the weak point, and the Economic Crime and Corporate Transparency Act 2023 is the response: at the time of writing, identity verification became a legal requirement for directors and people with significant control from 18 November 2025, with new people with significant control required to verify within 14 days of their appointment being notified. This does not solve the ownership problem, but it changes the cost of lying about it from nothing to something.

Enhanced due diligence

Regulation 33 sets out when enhanced customer due diligence and enhanced ongoing monitoring are mandatory, and the 2026 amendment made one materially important change. Previously regulation 33(1)(b) bit on relationships and transactions involving a "high-risk third country" as separately defined. From 30 June 2026 it bites on a "FATF call for action country", defined by reference to the Financial Action Task Force's list of High-Risk Jurisdictions subject to a Call for Action as it has effect from time to time. At the FATF plenary of 19 June 2026 that list comprised the Democratic People's Republic of Korea, Iran and Myanmar. The separate list of Jurisdictions under Increased Monitoring, informally the grey list, no longer triggers mandatory enhanced due diligence in the United Kingdom, though FATF mutual evaluations remain a geographical risk factor under regulation 33(6)(c) and a firm must still apply enhanced measures wherever it identifies high risk. HM Treasury's Money Laundering Advisory Notice of June 2026 sets out the position.

Politically exposed persons sit at regulation 35. The definition at regulation 35(12)(a) is an individual entrusted with prominent public functions other than as a middle-ranking or more junior official, with a non-exhaustive list at regulation 35(14) running from heads of state and government, ministers and legislators through senior judiciary, central bank boards, ambassadors and high-ranking armed forces officers to the boards of state-owned enterprises and international organisations. Family members include spouses and civil partners, children and their spouses or civil partners, and parents. Known close associates covers joint beneficial ownership or other close business relations, and sole beneficial ownership of an entity known to have been set up for the PEP's benefit, with regulation 35(15) limiting the enquiry to information in the firm's possession or credible publicly available information.

Where a PEP relationship is proposed or continued, regulation 35(5) requires senior management approval, adequate measures to establish source of wealth and source of funds, and enhanced ongoing

monitoring. The two sources are not synonyms, and conflating them is a common finding: source of wealth explains how the person's total assets were accumulated, source of funds where the money in this transaction came from. Regulation 35(9) requires the enhanced measures to continue for at least 12 months after the person ceases to be entrusted with the function, or longer if the firm considers it appropriate; regulation 35(11) provides that they fall away for family members and close associates when the PEP ceases to hold the function, whether or not that period has run.

Domestic PEPs are treated distinctly. Section 77 of the Financial Services and Markets Act 2023 required the Treasury to amend Part 3 of the Money Laundering Regulations so that where the customer is a domestic PEP, or a family member or known close associate of one, the starting point for the regulation 35(3) assessment is that they present a lower level of risk than a non-domestic PEP, and where no enhanced risk factors are present the extent of enhanced measures is less. That is now regulation 35(3A), with enhanced risk factors defined at regulation 35(12)(f) as risk factors other than the domestic PEP status itself. Section 78 required the FCA to review its PEP guidance and publish conclusions within twelve months. Its multi-firm review of 18 July 2024 found firms using definitions of relatives and close associates wider than the Regulations allow, ineffective arrangements for reassessing status after a person leaves office, risk ratings without clear rationale, and poor communication with affected customers. Finalised guidance FG25/3 followed on 7 July 2025, revised on 15 July 2025 to clarify that non-executive board members of United Kingdom civil service departments should not be treated as PEPs.

Simplified due diligence sits at regulation 37 and is permitted only where the firm has determined the relationship or transaction presents a low degree of risk, having taken account of the risk factors in Schedule 3. The 2026 amendment extended regulation 37(2)(a) to cover the new pooled account provisions at regulation 29(11) to (13), which was the point of that reform: making clear when simplified measures may be applied to accounts holding many clients' money.

Sanctions screening

Sanctions compliance is legally separate from anti-money-laundering compliance, and the vocabulary should stay separate too. In the United Kingdom, designations are made under regulations issued pursuant to the Sanctions and Anti-Money Laundering Act 2018. At the time of writing, and this is a recent change that will catch out any process built before it, the UK Sanctions List maintained by the Foreign, Commonwealth and Development Office is the single authoritative list of designations: the OFSI Consolidated List of Asset Freeze Targets closed on 28 January 2026 and is no longer updated. Any screening system still pointed at the old file is reading a frozen snapshot.

The Office of Financial Sanctions Implementation enforces financial sanctions. Under section 146 of the Policing and Crime Act 2017 as amended, and as set out in OFSI's *Financial sanctions enforcement and monetary penalties guidance* last updated on 9 February 2026, the maximum civil monetary penalty is the greater of £1,000,000 or 50 per cent of the estimated value of the funds or economic resources involved, or £1,000,000 where no value can be estimated. OFSI has stated an intention to seek to double this to the greater of £2,000,000 or 100 per cent of the breach value; at the time of writing that had been announced but the statutory cap was unchanged. Relevant firms must also notify HM Treasury where they know or have reasonable cause to suspect that a person is designated or has breached a prohibition, and where they hold frozen assets.

The United States position reaches further. The Office of Foreign Assets Control maintains the Specially Designated Nationals and Blocked Persons list and applies its 50 Percent Rule, under which an entity owned in the aggregate, directly or indirectly, 50 per cent or more by one or more blocked

persons is itself blocked whether or not it appears on any list. That is why ownership analysis and sanctions screening cannot be separate workflows: an entity can be blocked without ever being named.

The screening itself is a name-matching problem with unusual properties. Names arriving from customer records are not canonical: the same person may appear with given and family names reversed, with or without patronymics, with diacritics stripped, transliterated from Arabic, Cyrillic, Chinese or Korean by any of several competing conventions, abbreviated, or misspelled. Matching engines therefore combine exact matching with phonetic algorithms, edit-distance measures and token-based comparison, each returning a score, with a configurable threshold above which an alert is raised. Lower the threshold and recall rises while precision falls; raise it and the reverse. No setting gives both, and the threshold is a risk decision a supervisor will ask to see justified.

What reduces alert volumes without reducing detection is secondary identifiers and good data. Date of birth, place of birth, nationality, passport or national identity number and full address, held in structured fields rather than free text, let an engine discount a name match that is impossible on other grounds; list entries carry these to varying degrees, and the poorest-quality entries generate the most noise. Commonly cited industry estimates put the share of alerts closed with no report at between 85 and 95 per cent, but those figures come from vendors and industry bodies rather than regulators and should be treated as indicative. The cost also falls unevenly: common names in large populations, and naming conventions producing many identical full names, concentrate screening burdens on particular ethnic and national groups without anyone having designed that outcome. A firm that does not measure alert rates by customer segment will not know it is happening.

Transaction monitoring and suspicious activity reporting

Regulation 28(11) requires ongoing monitoring; how it is done is left to the firm. In practice there are two families of approach and mature firms run both.

Rules-based monitoring encodes typologies as deterministic tests: cash deposits aggregating above a value in a rolling window, rapid movement in and out leaving a near-zero balance, transactions structured just below a reporting threshold, activity inconsistent with the stated business, payments to jurisdictions the firm has rated high risk, or sudden dormancy reversal. Rules are explainable, testable and auditable, which regulators like, and trivially avoidable by anyone who learns the thresholds, which is their weakness. Model-based monitoring scores behaviour against learned baselines, per customer or per peer group, and surfaces the anomalous. It catches patterns nobody wrote a rule for, at the cost of two problems: the firm must explain to a supervisor why a model produced a given output, and a model trained on historically labelled data learns the biases in that labelling. Above-the-line and below-the-line testing and documented calibration are standard supervisory expectations rather than sophistication.

When suspicion arises, the firm's nominated officer, appointed under regulation 21(3) and known in FCA-regulated firms as the money laundering reporting officer, decides whether to report. In the Senior Managers and Certification Regime this is senior management function SMF17, a named individual with personal accountability. The report goes to the UK Financial Intelligence Unit at the National Crime Agency through the SAR Portal. From the SARs Annual Report 2025, covering April 2024 to March 2025:

Measure	2023-24	2024-25
SARs received	872,048	866,616
DAML requests received	57,081	57,666

Measure	2023-24	2024-25
Funds denied through DAML requests	£240.1m	£382.6m
Account freezing orders, forfeitures and restraints obtained through DAMLs	1,785	2,048
Terrorism Act SARs received	1,132	1,101
Defence Against Terrorist Financing requests	406	397
Reported SAR confidentiality breaches	8	7

Two further figures bear on how the system behaves. The average turnaround for a DAML decision was 2.8 days against the statutory seven working day period; and 151 refused cases required one or more moratorium extensions under section 336A, involving 18 law enforcement agencies and denying £120.6 million, of which £103 million came from a single DAML. The headline totals are heavily influenced by a few large cases, which is why the UKFIU publishes a three-year rolling average alongside them: £298.4 million for the period ending 2024-25.

The distribution by sector is equally telling. Of SARs submitted through the SAR Portal in 2024-25, banking accounted for 85.54 per cent and financial services a further 10.34 per cent, with virtual assets at 1.67 per cent, gambling 0.88 per cent, accountancy 0.71 per cent, legal 0.40 per cent and property 0.10 per cent. The UKFIU is careful to say it is for each sector and its supervisor to judge whether that volume is proportionate to risk. It is nonetheless clear that the overwhelming majority of the United Kingdom's financial intelligence comes from banks, which means anything a bank cannot see is largely invisible to the system.

The United States regime under the Bank Secrecy Act is more mechanical. Under 31 CFR 1020.320 a bank must file a suspicious activity report no later than 30 calendar days after initial detection of facts that may constitute a basis for filing, with an additional 30 where no suspect has been identified, and may not disclose the report or anything revealing its existence. Alongside it sits the currency transaction report for cash above $10,000, a threshold report rather than a suspicion report, with a corresponding offence of structuring to evade it.

The risk-based approach

The risk-based approach is FATF Recommendation 1, and it is not permission to do less. It is an instruction to allocate effort in proportion to assessed risk, which requires a firm to have assessed risk in writing. In the United Kingdom that is regulation 18, requiring a written risk assessment that takes account of information from supervisors and considers risk factors relating to customers, countries, products, services, transactions and delivery channels, and regulation 19, requiring policies, controls and procedures proportionate to that assessment and approved by senior management.

The national layer sits above it. HM Treasury and the Home Office published the *National Risk Assessment of Money Laundering and Terrorist Financing 2025* on 17 July 2025, the fourth comprehensive United Kingdom assessment. It rates retail banking, money service businesses, and payment and electronic money institutions as high risk for both money laundering and terrorist financing, and cryptoasset service providers as high for money laundering and medium for terrorist financing. It assesses that trade-based schemes move over £10 billion, and describes a realistic possibility that up to £10 billion could be laundered through the United Kingdom property market annually. A firm's regulation 18 assessment is expected to engage with those findings, not restate them.

Internationally the emphasis has shifted. In February 2025 the FATF revised Recommendation 1 and its Interpretive Note to strengthen the requirement that controls be proportionate and to encourage countries to promote financial inclusion, and later published *Guidance on Financial Inclusion and Anti-Money Laundering and Terrorist Financing Measures*. The FATF is explicit, including in the standing text accompanying its increased-monitoring list, that its standards do not envisage de-risking or cutting off entire classes of customer, and that flows for humanitarian assistance, legitimate non-profit activity and remittances should be neither disrupted nor discouraged. That followed its February 2021 project on the unintended consequences of the standards, which led in 2023 to amendments to Recommendation 8 curbing disproportionate measures against non-profit organisations.

Correspondent banking and de-risking

A correspondent relationship is one bank providing services, typically an account and payment execution, to another bank in another jurisdiction. It is the mechanism by which a bank in a small economy reaches the dollar, the euro or sterling at all.

The due diligence is heavier than for an ordinary customer because the correspondent takes on not just the respondent but, indirectly, the respondent's entire customer base. Regulation 34 requires a firm entering a correspondent relationship with a respondent from outside the United Kingdom to gather sufficient information to understand its business fully, determine its reputation and the quality of its supervision from credible public sources, assess its financial crime controls, obtain senior management approval, and document the respective responsibilities. Relationships with shell banks are prohibited outright, as is continuing one with an institution known to allow its accounts to be used by a shell bank. From 1 February 2027 the new regulation 34A extends materially the same requirements to cryptoasset exchange providers and custodian wallet providers dealing with third-country providers of similar services. The standard instrument for gathering this information is the Wolfsberg Group's Correspondent Banking Due Diligence Questionnaire, at the time of writing at version 1.4, with the Group's FAQs recommending a refresh cycle of 12 to 18 months.

The pattern in the data is a long, uneven contraction. The Committee on Payments and Market Infrastructures at the Bank for International Settlements publishes a quantitative review of correspondent banking data; its August 2020 commentary reported that the number of active correspondents worldwide had declined by about 22 per cent between 2011 and 2019 and by about 3 per cent in 2019 alone, with country-level declines ranging from 23 per cent in advanced economies to 41 per cent in small island developing states and dependent territories, and regional declines over the same period from 13 per cent in Northern America to 34 per cent in Latin America. Meanwhile the value and volume of payments running through those networks kept growing, by roughly 5 and 4 per cent respectively in the final year measured. Fewer relationships carrying more traffic is concentration, not shrinkage, and concentration means a single correspondent's exit can disconnect a jurisdiction.

The consequences for smaller markets are structural. Where a country loses its last correspondent for a currency, payments must route through an additional intermediary, raising cost, lengthening settlement and reducing transparency. The National Risk Assessment 2025 records a rise in nesting, where correspondent banks hold accounts for other correspondent banks so the chain becomes longer and more opaque, and attributes this partly to de-risking. That is how de-risking makes the underlying problem worse: the flows do not stop, they move somewhere with less visibility. The same report notes that de-risking of non-profit organisations, combined with the absence of formal banking where they operate, pushes them towards cash and less well-regulated channels. The Financial Stability Board launched a four-point action plan in November 2015 to address the decline, and correspondent

banking remains part of the G20 cross-border payments roadmap, but none of it changes the arithmetic facing a bank weighing bounded revenue against unbounded liability.

Supervision, enforcement and what will date this chapter

The United Kingdom's supervisory architecture is fragmented. The FCA supervises banks, payment institutions, electronic money institutions and registered cryptoasset businesses. His Majesty's Revenue and Customs supervises money service businesses, trust or company service providers, high value dealers, art market participants, and estate agency and accountancy businesses not otherwise supervised. Professional body supervisors cover the legal and accountancy sectors, overseen by the Office for Professional Body Anti-Money Laundering Supervision within the FCA. The Gambling Commission supervises casinos.

Enforcement inside the FCA's perimeter is generally brought under the Principles for Businesses rather than the Regulations, most often Principle 3 on management and control systems. The Nationwide Building Society case of December 2025 is representative of both mechanics and amounts: £44,078,500, reduced by 30 per cent from £62,969,297 for early resolution, for inadequate anti-financial-crime systems and controls between October 2016 and July 2021, specifically ineffective arrangements for keeping due diligence and risk assessments current across personal current account customers and for monitoring their transactions. The pattern across such cases is instructive: the failings are almost never an absence of policy. They are failures to apply it at scale, to keep customer information current, to tune monitoring to the actual business, and to close the loop between an alert and a decision.

The European Union is moving the opposite way. Regulation (EU) 2024/1624, the single rulebook, applies directly in all Member States from 10 July 2027, with Directive (EU) 2024/1640 to be transposed by the same date and Regulation (EU) 2024/1620 establishing the Authority for Anti-Money Laundering and Countering the Financing of Terrorism, based in Frankfurt and operational since 1 July 2025, expected to begin directly supervising a small group of the highest-risk cross-border institutions from 2028.

Four things here will date, and a reader in 2029 should verify all of them. The United Kingdom's Regulations are mid-reform: the 2026 amendment has commencement dates still ahead of it, on 1 February 2027 and 25 October 2027, and HM Treasury committed in July 2025 to further guidance on the meaning of establishing a business relationship, when source of funds checks are required, and when enhanced due diligence is mandatory. The lists move by design: FATF updates its at each plenary, roughly three times a year, and the United Kingdom's mandatory trigger now tracks the call for action list as it has effect from time to time, while the UK Sanctions List became the single authoritative designation list on 28 January 2026. OFSI's announced doubling of its civil penalty maximum had not taken effect as this was written, and the FCA enforcement figure was current at December 2025. And the European regime has not yet started, so the comparison drawn here between a deregulating United Kingdom and a harmonising European Union is a hypothesis that will by then have been tested.

52.98 Common wrong ideas

Wrong: Laundering is about hiding money. **Right:** It is about explaining it; the car wash owner does not conceal the criminal cash but manufactures a believable answer to where it came from, and pays tax on it because the tax bill is the receipt that proves the story.

Wrong: The anti-money-laundering regime is a net for catching launderers. **Right:** Firms are measured on process rather than outcomes and can comply impeccably while the money still gets away;

the published numbers describe a very large privately funded intelligence-gathering exercise from which comparatively few interventions are extracted.

Wrong: A wrong name match is an error to be engineered away. **Right:** It is the intended output of a system deliberately tuned to over-report, because missing a genuine sanctions target is a strict-liability breach while stopping an innocent customer for ten minutes is an inconvenience.

Wrong: A low-risk customer can be given lighter sanctions screening. **Right:** The risk-based approach governs due diligence only; an asset freeze is an absolute prohibition, and for breaches on or after 15 June 2022 any requirement that the person knew, suspected or believed anything is disregarded.

Wrong: Knowing who the customer is settles the identity question. **Right:** Identity is the easy half; the "more than 25 per cent" bright line is designed around by four holders at 25 per cent each, by chains of companies across jurisdictions, and by nominees, trusts and companies limited by guarantee.

Wrong: Collecting a passport copy verifies identity. **Right:** Verification requires the firm to satisfy itself from a source independent of the customer that the document is genuine and belongs to the person presenting it.

Wrong: Know your customer is something you complete at onboarding. **Right:** Regulation 28(11) makes scrutiny of transactions and keeping information up to date a continuing obligation, and it is the limb enforcement actions most often find missing.

Wrong: Source of wealth and source of funds are two phrases for the same check. **Right:** Source of wealth explains how the person's total assets were accumulated and source of funds where the money in this transaction came from, and conflating them is a common finding.

Wrong: The bank is being obstructive when it refuses to explain a delay. **Right:** Tipping off is a criminal offence in the United Kingdom, so the staff facing the customer are legally forbidden to explain, which is at the bottom of almost every furious story of this kind.

Wrong: De-risking shows a firm has failed to apply the risk-based approach. **Right:** It is a rational answer to bounded revenue weighed against an unbounded and asymmetric downside, and its effect is that flows move somewhere with less visibility rather than stopping.

52.99 Chapter summary in 20 lines

1. A banknote carries nothing to say where it came from, so what catches a criminal is the story attached to the money rather than the money itself.
2. Money laundering is the manufacture of believable answers to the question "where did that come from", and anti-money-laundering is the business of asking it properly.
3. A cash business launders by writing down more takings than it had, and its owner will happily pay tax on money never earned, because the tax bill is the receipt.
4. The three moves are getting the cash in, moving it around until the trail is hard to follow, and spending it on something ordinary.
5. Every rule you have been annoyed by exists to interfere with one of those three moves, which is why opening a business account takes eleven days.
6. Each onboarding question does a job: who you are, what the business does, who else benefits, and where the opening money came from.
7. Identity is the easiest of those and the least important; establishing who ultimately benefits is a judgement rather than a fact.

8. Sanctions screening is a legally separate world from money laundering, with absolute prohibition and strict liability where the other has suspicion and reasonableness.
9. Screening deliberately looks for near matches, so a bank taking on four thousand customers a week may see roughly 120 possible matches of which perhaps two are real.
10. That is a property of names rather than a defect in software, and its cost falls unevenly on people whose naming conventions produce many identical full names.
11. Once the account is open, monitoring compares behaviour against what the customer said and against what similar businesses do, and the great majority of alerts are ordinary life.
12. When a firm reports, it must not tell the customer, because tipping off is itself an offence, which explains the unexplained delay.
13. The architecture rests on Part 7 of the Proceeds of Crime Act 2002, whose principal offences sit at sections 327 to 329 and whose day-to-day grip comes from the failure to disclose and tipping-off provisions.
14. The Defence Against Money Laundering process gives the National Crime Agency seven working days to refuse, after which moratorium extensions can lawfully immobilise funds for over half a year on suspicion alone.
15. The Money Laundering Regulations 2017, as amended in 2026, set out what a firm must build: written risk assessment, due diligence, enhanced measures, ongoing monitoring, record-keeping and reporting.
16. The 2026 amendment converted euro thresholds to sterling, narrowed "complex or unusually large", added rules for pooled client accounts and moved the mandatory enhanced due diligence trigger onto the FATF call for action list.
17. Transaction monitoring runs on rules, which are explainable and avoidable once the thresholds are known, and on models, which find what nobody wrote a rule for and must still be explained to a supervisor.
18. Banking supplies about 85 per cent of the United Kingdom's suspicious activity reports, so anything a bank cannot see is largely invisible to the system.
19. Enforcement finds failures to apply policy at scale rather than absent policy, and against penalties measured in tens of millions banks exit whole customer categories.
20. De-risking does not stop the flows; it lengthens the chain through nesting and pushes activity towards cash and less regulated channels, which is a policy problem rather than a compliance one.

Sources, all consulted in August 2026: the Proceeds of Crime Act 2002, sections 327 to 333A, 335, 336A, 339A and 342; the Terrorism Act 2000; the Money Laundering, Terrorist Financing and Transfer of Funds (Information on the Payer) Regulations 2017 as amended by S.I. 2026/621; the Sanctions and Anti-Money Laundering Act 2018; sections 77 and 78 of the Financial Services and Markets Act 2023; section 146 of the Policing and Crime Act 2017 as amended; the Economic Crime and Corporate Transparency Act 2023 and Companies House identity verification guidance; 31 CFR 1020.320; the UKFIU SARs Annual Report 2025; the National Risk Assessment of Money Laundering and Terrorist Financing 2025; HM Treasury's Money Laundering Advisory Notice of June 2026; the FCA, including FG25/3, the 2024 multi-firm review on politically exposed persons and the Nationwide announcement of December 2025; OFSI's financial sanctions enforcement and monetary penalties guidance; the FCDO UK Sanctions List; the Financial Action Task Force, including the June 2026 plenary lists, the February 2025 revision to Recommendation 1 and the financial inclusion guidance; BIS CPMI correspondent banking commentaries; the Financial Stability Board; the Wolfsberg Group CBDDQ v1.4; Regulations (EU) 2024/1620 and 2024/1624 and Directive (EU) 2024/1640; the United Nations Office on Drugs and Crime; and the Bank Policy Institute, Getting to Effectiveness, 2018.

53

CHAPTER

Regulation

53.0 What this chapter gives you

1. You will be able to explain why a firm holding £900,000 of other people's money on an ordinary Tuesday is required to have only 125,000 euros of its own capital, and why that is not the mismatch it looks like.
2. You will be able to work out which licence a firm needs — authorised payment institution, small payment institution, authorised or small e-money institution, or registered account information service provider — from its volumes and the services it intends to provide.
3. You will be able to explain what safeguarding actually is, why it dominates everything else in this sector, and what changed on 7 May 2026 when CASS 15 came into force.
4. You will be able to tell somebody why safeguarded money is not insured money, and why the average shortfall across twelve failed payments firms was 65 per cent with distributions taking 2.3 years.
5. You will be able to distinguish the £85,000 that appears in scam reimbursement from the £85,000 people wrongly imagine sits behind safeguarded balances.
6. You will be able to explain why a merchant's card costs keep rising even though interchange is capped at 0.2% on consumer debit and 0.3% on consumer credit.
7. You will be able to name the four public bodies and one department that share the United Kingdom's payments perimeter, and say which statute each draws its powers from.
8. You will be able to say why "PSD2 applies in the UK" is false, and what to say instead when quoting a rule.
9. You will be able to describe what an important business service and an impact tolerance are, and why a service is not made unimportant by being well protected.
10. You will be able to separate what is law from what is announced policy — the abolition of the PSR, the PSD3 package, the repeal of assimilated law — and date any statement you make about this field.

The plain version

Imagine a left-luggage office at a railway station. You arrive with a suitcase, hand it over, and get a paper ticket with a number on it. When you come back you hand over the ticket and you expect the same suitcase.

Now ask what could go wrong, and notice that there are two completely different kinds of wrong.

The first kind: the office runs out of money. Rent goes up, custom falls away, and one morning the shutters stay down. That is a business failing. It is sad, it is common, and mostly the world copes.

The second kind: you come back and your suitcase is gone, because the owner sold it. Or because the owner kept all the suitcases in one heap, lost track of whose was whose, and used a few to prop open a door. That is a different sort of failure entirely, and no amount of the owner's own money in the till would have prevented it, because the problem was never the till. The problem was the room out the back.

Almost everything in payments regulation attempts to make the second kind of failure impossible. When a firm holds your money on its way somewhere else, your money is the suitcase and the firm's own money is the till. Regulators care about the till, because a firm that runs out of cash stops working. They care far more about the room out the back, because that is where your money actually is.

So let us follow a firm through it.

Rina runs a small business called Northgate Pay. Her customers are letting agents. When a tenant pays rent, the money comes to Northgate Pay first, sits for a day or two while the agent's software matches it to a property, and then goes out to the landlord. Northgate Pay never owns that money; it holds it in transit. On an average day about £900,000 belonging to other people is sitting in Rina's accounts, and across a month roughly £27 million passes through.

Four things now happen to Rina, and they are the four things this chapter is about.

She needs permission. Moving other people's money as a business is not something you may simply do. You have to be licensed by the Financial Conduct Authority, the FCA, and the licence comes in different sizes. Because Rina's monthly volume is well over the small-firm threshold, she needs the full version: authorisation as what the law calls an authorised payment institution. That is not a form you fill in over a weekend. It is a business plan, a three-year financial forecast, background checks on everyone who runs or owns the firm, and written descriptions of how she will handle complaints, spot money laundering, cope when her systems break, and keep customer money separate. The regulator has three months to decide, but only once the application is genuinely complete, and most first drafts are not.

She needs starting money of her own. As things stood at the time of writing in August 2026, the law required a firm doing what Rina does to hold initial capital of 125,000 euros. Note the currency: British law here is still written in euros, a leftover from where these rules came from. And note the size: 125,000 euros against £900,000 of customer money sitting there on a normal Tuesday. The capital is not there to cover the customer money and could not possibly cover it. It is there so the firm has a cushion of its own, enough to absorb an ordinary bad year without immediately becoming somebody else's problem.

She must keep the customer money somewhere else entirely. This is the room out the back, and in the trade it is called safeguarding. Rina must hold the rent money in a separate account, at a different bank, clearly marked as containing other people's money, with a letter from that bank acknowledging in writing that it has no right to grab the money to settle Rina's overdraft. Anything she is still holding at the end of the working day after the day it arrived has to be in there. Not most of it. All of it. And under rules that came into force on 7 May 2026, she must count it every business day, twice over: once against her own records, once against what the bank says is actually there. Differences are fixed the same day. Once a month she sends the regulator a return. Once a year an outside auditor checks the arrangement, unless she is small enough to be exempt, and the exemption line was drawn at £100,000 of customer money over a fifty-three week look-back.

And she must decide, in advance, how long she is allowed to be broken. The regulator does not ask Rina to promise her systems will never fail. It asks her to name the handful of things she does that

would genuinely hurt somebody if they stopped — for her, getting rent to landlords — and then to write down the longest outage a customer could survive. Four hours. Twelve. A day. Then she must map everything that service depends on, including other people's computers she does not control, and test whether she could really stay inside her own number when things go badly wrong. The regulator's interest is not in the promise. It is in whether she has done the arithmetic honestly.

Now the fifth thing, which is not about Rina at all, but about the price of card payments.

When you buy a £50 pair of shoes with a debit card, the shop does not get £50. A small slice is taken out and passed back to the bank that issued your card. That slice is called interchange. Left alone it would be set by the card networks, who are in the odd position of setting a fee paid by one of their customers to another, with no particular pressure to keep it low.

So the law caps it. As the rules stood at the time of writing in August 2026, interchange on a consumer debit card transaction within the UK was capped at 0.2% of the value, and on a consumer credit card at 0.3%. On your £50 shoes, that is ten pence on debit and fifteen pence on credit. It sounds trivial. Across the country it is billions of pounds a year, which is why it is one of the most bitterly litigated numbers in British commerce.

And finally, the thing that makes this chapter the one most likely to date: the map of who is in charge is itself being redrawn. There were, at the time of writing, two payments regulators in the UK. The FCA licenses and supervises firms. The Payment Systems Regulator, the PSR, regulates the shared plumbing, and it was the PSR that set the interchange rules and the rules on refunding scam victims. In March 2025 the government announced it would abolish the PSR and move its work into the FCA, and the consultation response confirming that decision was published on 21 April 2026. As at the time of writing in August 2026 the PSR still existed and still had its powers, because the merger needs an Act of Parliament that had not yet been passed.

That is the plain version. Permission, capital, the room out the back, an honest number for how long you may be broken, a cap on the card fee, and two regulators becoming one.

Where the plain version stops being true

Safeguarded is not insured, and the £85,000 you have heard of is not this £85,000. The analogy suggests that if the office closes you walk round the back and collect your suitcase. It is not that clean. Money held by a payment institution or an e-money institution is not a bank deposit and is not covered by the Financial Services Compensation Scheme. There is no government-backed £85,000 guarantee behind it. What safeguarding creates is a pool of assets meant to be paid out to customers ahead of the failed firm's ordinary creditors, and whether that pool is full is a question of fact. Historically it very often was not. The FCA stated in its September 2024 consultation CP24/20 that of the twelve payments firms that became insolvent between the first quarter of 2018 and the second quarter of 2023, the average shortfall between funds owed to customers and funds actually safeguarded was 65%, and for e-money institutions alone 80%. Where money was eventually distributed it took on average 2.3 years to reach customers. Sixty-five pence in the pound, two and a bit years late, is a materially different outcome from collecting your suitcase. Separately, the £85,000 that appears in British payments conversation usually refers to something else entirely: the maximum reimbursement per claim under the mandatory refund rules for authorised push payment scams, confirmed by the PSR in policy statement PS24/7 and applying since 7 October 2024. Same number, different mechanism, different money, no connection.

There is no single "payments regulator", and several of the most binding rulebooks are not law at all. The plain version names two regulators and implies they cover the field. They do not. The Bank of England supervises the payment systems that matter most to financial stability under Part 5 of the Banking Act 2009; the Prudential Regulation Authority supervises banks' safety and soundness; anti-money-laundering supervision runs on its own track under the Money Laundering Regulations 2017; data protection sits with the Information Commissioner's Office. And running alongside all of it are rulebooks that bind harder than most legislation while being nobody's law: the Visa and Mastercard operating rules, private contracts revised at least annually and enforced with fines through the acquiring chain; Pay.UK's scheme rules for Bacs, Faster Payments and Direct Debit; the PCI Data Security Standard, published by an industry council with no statutory authority whatsoever, which nonetheless determines whether a merchant may accept cards at all. A firm can be perfectly compliant with the Payment Services Regulations and be shut down in a fortnight by a card scheme. Practitioners spend more of their week on the private rulebooks than on the statutes.

PSD2 is not, and never was, the law in the United Kingdom, and the British copies of the European rules are actively drifting. People say "PSD2 applies in the UK" as shorthand and it is false in a way that matters. The Second Payment Services Directive was a directive: an instruction to member states to legislate. The United Kingdom's legislation was the Payment Services Regulations 2017 and a set of FCA rules, and after departure from the European Union what remains is assimilated law that Parliament and the FCA can change unilaterally, and have. The Financial Services and Markets Act 2023 provides for the wholesale repeal of assimilated financial services law, to be replaced by FCA Handbook rules, on a timetable still running. Divergence already exists: on 19 March 2026 FCA rule changes took effect removing the mandatory £100 single-transaction contactless limit and allowing firms with strong fraud controls to set their own, a change with no European equivalent. Meanwhile the European Union is replacing PSD2 outright. Anyone quoting an article number should say which body of law they are quoting and on what date.

The interchange cap does not cap what a merchant pays, and it does not apply to most of the transactions merchants complain about. Interchange is one component of the merchant service charge. The other two are the scheme fees, charged directly by the card networks and not capped at all, and the acquirer's own margin. Capping interchange at 0.2% has never stopped the total cost of card acceptance rising. Further, the caps apply to consumer debit and credit cards on transactions within the relevant territory. They do not apply to commercial cards, nor where the issuer is outside the territory, and three-party schemes such as American Express fall outside them except in defined circumstances. This is not a loophole; it is the deliberate scope of the rule. It is also why the most expensive card in a merchant's mix is usually a foreign-issued commercial credit card, on which no cap bites at all.

The technical version

The institutional map, as at August 2026

Four public bodies and one department share the United Kingdom's payments perimeter, and they derive their powers from four different statutes.

The **Financial Conduct Authority** authorises, registers and supervises payment service providers under the Payment Services Regulations 2017 (SI 2017/752) and electronic money issuers under the Electronic Money Regulations 2011 (SI 2011/99). For banks and building societies it acts as conduct regulator under the Financial Services and Markets Act 2000. It also publishes the finalised guidance *Payment Services and Electronic Money — Our Approach*, not binding but the document practitioners

actually read; version 6 appeared in November 2024 and an amended version was published in draft alongside PS25/12 for May 2026.

The **Payment Systems Regulator** is the economic regulator of designated payment systems under Part 5 of the Financial Services (Banking Reform) Act 2013. It is also the competent authority for the interchange fee caps under the Payment Card Interchange Fee Regulations 2015 (SI 2015/1911), and holds functions under Part 8 of the PSRs 2017 on access to payment systems and under the Payment Accounts Regulations 2015 on the Current Account Switch Service.

The **Bank of England** supervises recognised payment systems and specified service providers under Part 5 of the Banking Act 2009, and operates CHAPS and the real-time gross settlement infrastructure. The **Prudential Regulation Authority** supervises deposit-takers and designated investment firms. **HM Treasury** sets the framework: its November 2024 National Payments Vision and the associated Payments Forward Plan set the direction, and the Payments Vision Delivery Committee coordinates.

On 11 March 2025, as part of its Regulation Action Plan, the government announced its intention to abolish the PSR and consolidate its functions primarily within the FCA. It consulted in September 2025 in *A Streamlined Approach to Payment Systems Regulation* and published the consultation response on 21 April 2026. The response confirms the intention to transfer the PSR's functions entirely to the FCA, integrating them into the FSMA framework so far as practicable; to retain a designation regime for bringing payment systems in and out of scope; to carry over objectives equivalent in substance to the PSR's competition, innovation and service-user objectives; to transfer its functions under the PSRs 2017, the Payment Card Interchange Fee Regulations 2015 and the Payment Accounts Regulations 2015, together with the authorised push payment reimbursement requirements; to create a single access regime by removing the Part 8 PSRs route in favour of the FSBRA one; and to make specific directions appealable to the High Court rather than the Competition Appeal Tribunal. Consolidation requires primary legislation, and the response says only that the government will legislate "as soon as parliamentary time allows". As at the time of writing in August 2026 that legislation had not been introduced and the PSR remained a separate body.

Authorisation categories under the PSRs 2017 and EMRs 2011

Five categories matter. The distinctions are about scale, about whether the firm issues e-money, and about which of the payment services in Schedule 1 to the PSRs 2017 it intends to provide.

Category	May do	Scale limit	Initial capital	Ongoing own funds
Authorised payment institution	Any Schedule 1 payment service, including payment initiation and account information	None	EUR 125,000 general; EUR 50,000 payment initiation only; EUR 20,000 money remittance only; none for account information only	Method A, B or C as the FCA directs
Small payment institution	Schedule 1 services other than payment initiation and account information	Monthly average of payment transactions over the preceding 12 months not exceeding EUR 3 million	None	None

Category	May do	Scale limit	Initial capital	Ongoing own funds
Authorised e-money institution	Issue e-money and provide payment services	None	EUR 350,000	2% of average outstanding e-money; Method A, B or C for unrelated payment services
Small e-money institution	Issue e-money and payment services other than payment initiation and account information	Average outstanding e-money not exceeding EUR 5 million; unrelated payment transactions not exceeding EUR 3 million monthly average	Set by Schedule 2 EMRs	2% of average outstanding e-money where required
Registered account information service provider	Account information services only	None	None	None

All figures in that table are as they stood at the time of writing in August 2026, and all are stated in euros in the United Kingdom's own legislation, the amounts having been carried across from the European originals without conversion.

Three points a practitioner will insist on. First, a payment institution may not issue e-money; an e-money institution may do both, which is why firms issuing prepaid cards, wallets or stored balances need the EMI licence and the higher capital. Second, initial capital is a floor applying from day one, whereas own funds is a continuing calculation. Method A is 10% of the previous financial year's fixed overheads. Method B is a tiered percentage of payment volume: 4% of the first EUR 5 million of monthly volume annualised, 2.5% of the next EUR 5 million, 1% of the next EUR 90 million, 0.5% of the next EUR 150 million and 0.25% of the remainder, times a scaling factor of 0.5 for money remittance and 1 otherwise. Method C applies tiered percentages to a relevant indicator built from interest income and expense, commissions and other operating income, with a floor of 80% of the three-year average. The FCA directs which applies. Third, providing payment services in the UK without authorisation or registration is a criminal offence under regulation 138 of the PSRs 2017, not a technical breach.

On process: under regulation 9 of the PSRs 2017 and regulation 9 of the EMRs, the FCA must determine a complete application within three months of receiving it, and an incomplete one within twelve months. The clock runs from completeness, not submission, which is why published median timelines bear little relation to the statutory figure; the Approach Document confirms the point at paragraph 3.196. Agents are separately registered under regulation 34 and the principal remains fully responsible for their acts and omissions, while distributors of e-money are not registered in the same way, a difference that has caused repeated supervisory difficulty.

Safeguarding, and why it dominates everything else

The prudential logic here is unusual. Capital requirements in banking exist to absorb losses so the institution continues as a going concern. In payments the capital numbers are deliberately small relative to the customer balances held, because these firms do not take credit risk on customer money as a

bank does. The regulatory weight therefore falls almost entirely on the gone-concern question: when this firm fails, do its customers get their money back, in full, quickly?

Under regulation 23 of the PSRs 2017, an authorised payment institution must safeguard "relevant funds": broadly, sums received from or for the benefit of a payment service user for the execution of a payment transaction, and sums received from another provider for that purpose. Regulations 20 to 22 of the EMRs impose the equivalent obligation on e-money institutions for funds received in exchange for e-money issued.

There are two permitted methods and a firm may use each for different funds. The **segregation method** requires relevant funds to be kept separate from the firm's own money and, where still held at the end of the business day following the day of receipt, placed in a separate account with an authorised credit institution or the Bank of England, or invested in secure liquid assets. The **insurance or guarantee method** requires cover by a policy with an authorised insurer, or a comparable guarantee from an authorised insurer or credit institution, with proceeds payable into a separate account on insolvency. In practice the overwhelming majority segregate, because insurers write very little of this cover and price it punitively.

The gap between what the regulations say and what happens in insolvency was exposed by *Re Ipagoo LLP* [2022] EWCA Civ 302, decided on 9 March 2022. Ipagoo was an authorised e-money institution that went into administration having failed to safeguard sufficient relevant funds. The FCA argued that the EMRs impose a statutory trust over relevant funds from the moment of receipt. The Court of Appeal held that they do not. It did hold that the regulations create a statutory asset pool for the benefit of e-money holders, and that where relevant funds should have been safeguarded but were not, the pool falls to be topped up from the general estate. Customers therefore rank ahead of ordinary creditors in respect of the pool, but without the proprietary certainty a trust would give, and insolvency costs can bite in ways they would not under a clean trust.

That case, together with the insolvency data, drove consultation CP24/20, published on 25 September 2024 and closed on 17 December 2024, which proposed two stages: an interim set of rules to raise compliance with the existing statutory requirements, and an end-state regime replacing the statutory safeguarding provisions with a sourcebook modelled on the Client Assets Sourcebook, including an express statutory trust.

The outcome was policy statement **PS25/12, published on 7 August 2025**, with the interim rules renamed the **Supplementary Regime** and the end-state rules the **Post-Repeal Regime**. The Supplementary Regime, made by the Payments and Electronic Money (Safeguarding) Instrument 2025 and inserted into the Handbook as **CASS 15**, **came into force on 7 May 2026**. Its principal requirements, as stated in PS25/12 and accurate at the time of writing in August 2026, are internal and external reconciliations of safeguarded funds on each business day, not required on weekends, public holidays and days when relevant foreign markets are closed, with discrepancies resolved promptly; a new monthly safeguarding return, revised after a pilot exercise; an annual safeguarding audit by a qualified auditor for authorised payment institutions and authorised e-money institutions, subject to an exemption where the firm has not been required to safeguard more than £100,000 of relevant funds at any time over a period of at least fifty-three weeks; a contingency plan at least three months before the expiry of any safeguarding insurance policy or comparable guarantee, that lead period retained despite industry objection; retention of the existing range of permitted secure liquid assets, the FCA explicitly maintaining a low risk tolerance towards broadening it; and resolution pack requirements aligned to CASS practice. The FCA also removed the proposed limited assurance engagement for firms claiming

to hold no relevant funds, replacing it with guidance that a statutory auditor should notify the FCA if a firm has failed to safeguard.

Two significant CP24/20 proposals did **not** proceed at this stage: the requirement to receive relevant funds directly into a designated safeguarding account, and the corresponding treatment of funds received through agents and distributors. Both were held back pending further consideration alongside the statutory trust proposal. The FCA has said it will not proceed with the Post-Repeal Regime without further consultation, and will review implementation once a full audit period has been completed.

The regime applies to authorised payment institutions other than those providing only payment initiation or account information services, authorised e-money institutions, small e-money institutions, and credit unions issuing e-money in the UK. Small payment institutions may opt in.

Underneath all of this sits the **Payment and Electronic Money Institution Insolvency Regulations 2021** (**SI 2021/716**), a bespoke special administration regime for these firms with the return of relevant funds as a statutory objective and a court-approved distribution plan mechanism. HM Treasury launched an independent review of it in 2024 and the report has since been published.

Interchange regulation

The caps entered European law through Regulation (EU) 2015/751 on interchange fees for card-based payment transactions, applying from 9 December 2015. That regulation was assimilated into United Kingdom law and amended so that its operative provisions now refer to UK transactions.

As the assimilated regulation stood at the time of writing in August 2026, Article 3 provides that payment service providers must not offer or request a per-transaction interchange fee of more than 0.2% of transaction value for any UK debit card transaction, with a Treasury power to lower that percentage, impose a fixed maximum, or permit an alternative not exceeding the sterling equivalent of EUR 0.05 per transaction, provided the scheme's aggregate annual interchange stays within 0.2% of annual transaction value. Article 4 sets the equivalent cap for any UK credit card transaction at 0.3%, again with a Treasury power to lower but not raise it. Legislation.gov.uk records the regulation as subject to revocation under Schedule 1 to the Financial Services and Markets Act 2023, a change not yet applied at the time of writing.

The regulation is not only about price. Articles 6 to 11 carry the business rules that shape the acquiring market: licensing, prohibiting territorial restrictions; separation of payment card scheme and processing entities; co-badging and the payer's right to choose the brand or application at the point of sale; unblending, which requires acquirers to offer merchant service charges itemised by category and brand of card rather than a single blended rate; the Honour All Cards rule, which prevents schemes obliging merchants to accept every card of a brand irrespective of category; and steering, which permits merchants to steer customers towards cheaper instruments. Unblending made the true cost of card acceptance visible to merchants, and is the reason interchange-plus pricing exists at all. Separately, regulation 166 of the PSRs 2017 prohibits surcharging on most consumer card payments in the UK.

The post-departure history is the interesting part. After the end of the transition period, Mastercard and Visa increased interchange on card-not-present transactions between the UK and the European Economic Area, for consumer debit and credit cards, from 0.2% and 0.3% to 1.15% and 1.5% respectively. The PSR opened market review MR22/2. Its final report, MR22/2.7, found that Mastercard and Visa, which between them account for 99% of UK debit and credit card payments, were not subject to effective competitive constraints; that the increases were costing UK businesses an additional £150 million to £200 million annually; and that neither scheme could demonstrate any specific assessment

underpinning the decision. The PSR consulted on a two-stage remedy: an interim cap followed by a longer-term cap.

Visa, Mastercard and Revolut challenged the PSR's power to impose such a cap by judicial review. **The High Court dismissed all three challenges on 15 January 2026, in [2026] EWHC 64, upholding the PSR's statutory power to impose price caps on cross-border interchange fees.** The PSR then decided, in MR22/2.9, not to proceed with an interim cap, preferring to move directly to a longer-term cap once it had developed a robust methodology, on which it consulted in MR22/2.8. As at the time of writing in August 2026, no cap on UK-EEA cross-border interchange had been imposed. HM Treasury's April 2026 consultation response states that the ability to impose price controls will carry over to the FCA on consolidation.

Running in parallel, and quite separately, is the private litigation. The Competition Appeal Tribunal's judgment of 27 June 2025 in the umbrella interchange proceedings, brought by over two thousand merchant claimants, held that the schemes' default interchange fee arrangements restricted competition in the acquiring market contrary to Article 101(1) of the Treaty on the Functioning of the European Union in respect of inter-regional and commercial card transactions. Permission to appeal was sought in August 2025 and the appellate proceedings were live at the time of writing. Interchange is therefore regulated on two tracks at once — a regulatory cap track and a competition law damages track — and they can reach different answers about the same fee.

Operational resilience

The FCA's operational resilience regime was made by policy statement **PS21/3, published in March 2021**, with the rules in SYSC 15A taking effect on **31 March 2022** and a transitional period ending on **31 March 2025**. Its scope expressly includes entities authorised and registered under the PSRs 2017 and the EMRs, alongside banks, building societies, insurers, enhanced-scope Senior Managers and Certification Regime firms and recognised investment exchanges.

The regime rests on four obligations. A firm must identify its **important business services**, defined as services provided to clients which, if disrupted, could cause intolerable levels of harm to one or more clients or pose a risk to the soundness, stability or resilience of the UK financial system. It must set an **impact tolerance** for each, being the maximum tolerable level of disruption, measured as a length of time and any other relevant metric. It must **map** the people, processes, technology, facilities and information supporting each service, **test** its ability to remain within tolerance under severe but plausible scenarios, and produce a board-approved **self-assessment**.

By 31 March 2025 firms were required to have completed mapping and testing to the sophistication necessary to demonstrate that they can remain within impact tolerances. The FCA's supervisory observations emphasise several things firms routinely get wrong: that a service should not be excluded from the list on the strength of a single factor; that identification must be done without reference to the firm's response and recovery capabilities, so a service is not made unimportant by being well protected; that interdependencies and sub-contracting chains must be considered; and, critically, that the firm remains responsible for staying within its impact tolerance even where a third party delivers the service. A firm may rely on a third party's own testing, but must satisfy itself that the methodology and scenarios are adequate.

Two further layers sit on top.

Incident and third-party reporting. Regulation 98 of the PSRs 2017 requires payment service providers to maintain a framework for managing operational and security risks with an annual assessment, and regulation 99 requires notification of major operational or security incidents without

undue delay. Following consultation CP24/28, published 13 December 2024 and closed 13 March 2025, the FCA published **policy statement PS26/2 on 18 March 2026**, with finalised guidance FG26/3 and FG26/4. PS26/2 creates a single regime shared across the FCA, the PRA and the Bank of England, and **its rules come into force on 18 March 2027**. It defines an operational incident, sets reporting thresholds, and introduces a standardised single submission regardless of which regulator the report is for, with a two-tier model of standard and enhanced reporting. The FCA threshold is that the firm reasonably believes the incident poses a risk of intolerable harm to consumers from which they cannot easily recover, or a risk to the safety and soundness of the firm or other market participants, or a risk to market stability, integrity or confidence in the UK financial system. Reportability is not limited to incidents affecting an important business service, and payment service providers will discharge their regulation 99 obligations through this regime. The third-party rules apply to a narrower population expressly including authorised e-money institutions and authorised payment institutions, requiring notification of new or significantly changed material third party arrangements and an annually submitted register.

Critical third parties. The Financial Services and Markets Act 2023 gave HM Treasury power to designate a third party providing services to regulated firms as a critical third party, and gave the regulators rule-making and enforcement powers over designated entities. The final rules, published jointly as FCA PS24/16 and PRA PS16/24 on 12 November 2024, **took effect on 1 January 2025**, but apply to a provider only from the date its designation order comes into force. **On 10 July 2026 HM Treasury designated the first critical third parties, with effect from 13 July 2026: Microsoft Ireland Operations Limited, Google Cloud EMEA Limited, Amazon Web Services EMEA SARL and Oracle Corporation UK Limited.** Designation does not reduce the responsibility of the firms relying on those providers; it adds a supervisory relationship with the provider itself.

The European Union position, and where the two regimes now differ

The Second Payment Services Directive, Directive (EU) 2015/2366, applied from 13 January 2018, with the regulatory technical standards on strong customer authentication and common and secure communication, Delegated Regulation (EU) 2018/389, applying from 14 September 2019. The Second Electronic Money Directive, 2009/110/EC, sits alongside it. The European Banking Authority issues guidelines and technical standards; national competent authorities supervise.

On 28 June 2023 the European Commission published two proposals: **PSD3**, COM(2023)366, a directive covering licensing, prudential requirements and supervision, and the **Payment Services Regulation**, COM(2023)367, a directly applicable regulation covering conduct, transparency and rights and obligations. Splitting the regime this way is the point: conduct rules become uniform without national transposition, while authorisation remains a directive.

The legislative position, as at the time of writing in August 2026: Parliament adopted its first-reading position on 23 April 2024; Parliament and Council reached provisional political agreement on 27 November 2025; and the text agreed at early second-reading negotiations was approved in the ECON committee on 5 May 2026. The European Parliament's legislative train summary, updated 20 June 2026, records that the deal still needs formal adoption by Parliament and Council before it can enter into force, and the Parliament's own Legislative Observatory still shows the file as awaiting the Council's first-reading position, with an indicative plenary sitting date of 14 December 2026. Nothing has therefore reached the Official Journal, and the practitioner expectation of publication in the first half of 2026 has already been overtaken. The application clock itself is not speculation: the final compromise texts published in the Council's public document register, 8221/26 for the Regulation and 8222/26 for the Directive, both dated 17 April 2026, provide that the Regulation applies twenty-one months after

its entry into force, that member states must adopt, publish and apply their transposing measures within the same twenty-one months, and that the payee verification provisions in Articles 50 and 57 of the Regulation apply only from twenty-seven months. Entry into force is twenty days after Official Journal publication, so the whole calendar can be counted precisely from the day that publication happens, and not a day before.

The substantive changes that matter most, as reported in the agreed texts: e-money institutions cease to exist as a separate category and become a sub-category of payment institutions, with e-money issuance treated as a payment service in the annex to PSD3 and the Second Electronic Money Directive repealed; payment institutions gain an option, at national discretion, to safeguard customer funds at a central bank; providers must offer a payee verification service and inform the payer of discrepancies; liability extends to failures to implement appropriate fraud prevention mechanisms, and to impersonation fraud where the customer has reported the matter to the police and the provider; online platforms that fail to remove fraudulent financial content after notification become liable to providers that have reimbursed defrauded customers; advertisers of financial services must demonstrate to very large online platforms and search engines that they are authorised or exempt; and users gain a right of access to human customer support.

Separately, and already in force, the **Instant Payments Regulation (EU) 2024/886**, adopted on 13 March 2024, amends the SEPA Regulation to require euro instant credit transfers, charge parity with ordinary transfers, a free verification of payee service, and at-least-daily sanctions screening of customers rather than transaction-by-transaction screening. Its deadlines, as published by the European Central Bank, are staggered:

Requirement	Applies to	Deadline
Receiving instant payments, and equality of charges	Euro area member states	9 January 2025
Sending instant payments, and verification of payee	Euro area member states	9 October 2025
Receiving instant payments, and equality of charges	Non-euro area member states	9 January 2027
Receiving instant payments	EMIs and PIs, euro and non-euro area	9 April 2027
Sending instant payments	EMIs and PIs in euro area member states	9 April 2027
Sending instant payments, and verification of payee	Non-euro area member states, and their EMIs and PIs	9 July 2027
Sending outside business hours from national currency accounts	Non-euro area member states	9 June 2028

The same regulation, through amendments to PSD2 and the Settlement Finality Directive, makes payment institutions and e-money institutions eligible to participate directly in designated payment systems. Since October 2025 non-bank payment service providers meeting the TARGET Guideline requirements have been able to access TARGET services including T2 and TIPS. That is a structural change of the first order: it removes the requirement for a non-bank to hold its settlement position at a commercial bank that is also its competitor.

On operational resilience, the European regime is the **Digital Operational Resilience Act**, Regulation (EU) 2022/2554, which has applied since **17 January 2025**. It covers ICT risk management, incident classification and reporting, resilience testing including threat-led penetration testing for significant

entities, ICT third-party risk management with prescribed contractual terms, and direct oversight of critical ICT third-party providers by the European Supervisory Authorities. Firms operating on both sides will find the UK and EU regimes conceptually similar and mechanically incompatible: impact tolerances and important business services on one side, ICT risk management and register-of-information reporting on the other, with different thresholds and different clocks.

The United Kingdom's own moving parts

Three UK changes bear directly on how a payments firm operates, and each has an explicit date.

The **Payment Services (Amendment) Regulations 2024 came into force on 30 October 2024**, permitting a payment service provider to delay execution of an outbound sterling payment within the UK where it has reasonable grounds to suspect fraud or dishonesty and needs to make further enquiries. The delay must be no longer than necessary and in any event no longer than the end of the fourth business day from receipt of the payment order. The provider must tell the customer of the delay and its reasons unless disclosure would be unlawful, and is liable for any charges or interest the customer incurs.

The **authorised push payment reimbursement requirement has applied since 7 October 2024**, imposed by the PSR through Specific Direction 20 for Faster Payments and mirrored by the Bank of England for CHAPS. The **maximum reimbursement is £85,000 per claim**, confirmed in PSR policy statement PS24/7 in October 2024, which reduced a previously confirmed maximum of £415,000 before go-live. The PSR estimated that £85,000 fully covers 99.8% of Faster Payments authorised push payment scams by volume and 90% by value.

The **contactless rule changes announced by the FCA in December 2025 took effect on 19 March 2026**, removing the mandatory £100 single-transaction limit and the cumulative limit and giving firms with strong fraud controls the flexibility to set their own. Nothing obliges a firm to change anything, and terminal and scheme-rule changes are also needed before higher-value contactless card transactions can be processed. The pattern will recur: a fixed numerical rule replaced by a risk-based standard, with the supervisory question shifting from compliance with a threshold to the demonstrable quality of a firm's own fraud monitoring.

What this means for a firm actually operating

The standing obligations of a mid-sized authorised payment institution, as at August 2026, run wider than most people building a payments product expect: own funds on the applicable method; daily safeguarding reconciliation and the monthly return; the annual safeguarding audit; the regulation 98 risk framework and its annual assessment; the SYSC 15A self-assessment, mapping and testing; professional indemnity insurance for payment initiation or account information services; the Money Laundering Regulations 2017; the Consumer Duty in PRIN 2A where it applies; complaints under DISP; strong customer authentication; the reporting cycle under SUP 16.13; registration of agents before they act; and, from 18 March 2027, notification of material third party arrangements and the register.

The single most common cause of serious regulatory difficulty in this sector is not any of those in isolation. It is the gap between what a firm's safeguarding documentation says and what its bank accounts actually contain, discovered late. Everything the FCA has done since CP24/20 shortens the interval at which a discrepancy becomes visible: from an annual audit, to a monthly return, to a daily reconciliation. A firm that treats the daily reconciliation as a compliance chore has misunderstood it. It is the early warning system for the only failure mode that genuinely destroys customers.

What is most likely to be wrong in this chapter first

Three things, and it is better to name them than to pretend otherwise.

The abolition of the Payment Systems Regulator is government policy with a published consultation response but no Act of Parliament. When the legislation passes, every reference here to the PSR as a separate body becomes historical, and the statutory basis for interchange regulation and for the reimbursement rules will move.

The European Union's PSD3 and Payment Services Regulation package had reached committee approval of an agreed text but, on the author's checking in August 2026, had not been verified as adopted and published in the Official Journal. Once it applies, e-money institutions cease to be a separate category in the European Union while remaining one in the United Kingdom: the largest structural divergence between the two regimes since departure.

And the whole of the United Kingdom's assimilated payment services law is subject to repeal under the Financial Services and Markets Act 2023, with firm-facing requirements expected to move into FCA rules. When that happens, the regulation numbers used throughout this chapter — regulation 23 for safeguarding, regulation 98 for security risk, regulation 138 for the criminal offence, Schedule 3 for capital — will cease to exist even where the substance survives in renumbered form. Every figure and deadline here should be checked against the source before it is relied on, and the sources are named below precisely so that it can be.

53.98 Common wrong ideas

Wrong: Money held by a payment institution or e-money institution is protected up to £85,000 like a bank deposit. **Right:** It is not a deposit and the Financial Services Compensation Scheme does not cover it; safeguarding creates a pool meant to be paid to customers ahead of ordinary creditors, and whether that pool is full is a question of fact.

Wrong: The £85,000 figure in British payments is the safeguarding guarantee. **Right:** It is the maximum reimbursement per claim under the mandatory authorised push payment scam rules confirmed in PSR policy statement PS24/7, applying since 7 October 2024 — the same number attached to a different mechanism, with no connection between them.

Wrong: Safeguarded funds are held on trust from the moment they are received. **Right:** *Re Ipagoo LLP* held that the Electronic Money Regulations impose no statutory trust; they create a statutory asset pool topped up from the general estate, so customers rank ahead of ordinary creditors but without the proprietary certainty a trust would give.

Wrong: There is a payments regulator, and complying with its rules is what keeps you trading. **Right:** The FCA, the PSR, the Bank of England, the PRA and HM Treasury all hold pieces of the perimeter, and the private rulebooks — the Visa and Mastercard operating rules, Pay.UK's scheme rules, the PCI Data Security Standard — bind harder than much legislation; a firm perfectly compliant with the Payment Services Regulations can be shut down in a fortnight by a card scheme.

Wrong: PSD2 applies in the United Kingdom. **Right:** PSD2 was a directive instructing member states to legislate; the United Kingdom's legislation was the Payment Services Regulations 2017 plus FCA rules, now assimilated law that Parliament and the FCA can and do change unilaterally, as the removal of the mandatory £100 contactless limit on 19 March 2026 demonstrates.

Wrong: Capping interchange caps what a merchant pays to accept cards. **Right:** Interchange is one component of the merchant service charge alongside uncapped scheme fees and the acquirer's own margin, and capping it has never stopped the total cost of acceptance rising.

Wrong: The interchange caps apply to all cards. **Right:** They apply to consumer debit and credit cards on transactions within the relevant territory; commercial cards, foreign-issued cards and three-party schemes such as American Express fall outside them by design, which is why the most expensive card in a merchant's mix is usually one on which no cap bites at all.

Wrong: The FCA has three months to decide an authorisation application, so budget three months. **Right:** The clock runs from completeness rather than submission, an incomplete application attracts a twelve-month deadline, and most first drafts are not complete.

Wrong: Outsourcing an important business service moves the resilience obligation to the supplier. **Right:** The firm remains responsible for staying within its own impact tolerance even where a third party delivers the service, and the designation of a critical third party adds supervision of the provider without reducing anybody else's responsibility.

Wrong: The daily safeguarding reconciliation is a compliance chore. **Right:** It is the early warning system for the only failure mode that genuinely destroys customers, which is why the FCA has shortened the interval at which a discrepancy becomes visible from an annual audit to a monthly return to a daily count.

53.99 Chapter summary in 20 lines

1. Payments regulation is built around one failure in particular: a firm losing the customer money it was only holding in transit, which no amount of the firm's own capital would have prevented.
2. Moving other people's money as a business requires a licence from the Financial Conduct Authority, and the licence comes in sizes.
3. Five categories matter, distinguished by scale, by whether the firm issues e-money, and by which of the Schedule 1 payment services it provides.
4. Initial capital ranges from nothing for an account information service provider to EUR 350,000 for an authorised e-money institution, and the amounts are still stated in euros in British legislation because they were carried across from the European originals without conversion.
5. Initial capital is a floor from day one, whereas ongoing own funds is a continuing calculation under Method A, B or C as the FCA directs.
6. Providing payment services without authorisation or registration is a criminal offence under regulation 138 of the Payment Services Regulations 2017, not a technical breach.
7. The statutory three-month determination period runs from the point an application is complete, which is why published timelines bear so little relation to it.
8. Safeguarding carries far more regulatory weight than capital, because the question these rules ask is a gone-concern question: when this firm fails, do its customers get their money back, in full, quickly?
9. The honest answer historically was no — an average shortfall of 65 per cent across twelve insolvencies, 80 per cent for e-money institutions alone, and 2.3 years to distribution.
10. *Re Ipagoo LLP* established that the regulations create a statutory asset pool rather than a trust, which is why customers rank ahead of ordinary creditors without proprietary certainty.
11. Consultation CP24/20 and policy statement PS25/12 followed, and the Supplementary Regime in CASS 15 came into force on 7 May 2026 with daily internal and external reconciliations, a monthly return and an annual audit subject to a £100,000 exemption.

12. Two proposals were held back for further work: receiving relevant funds directly into a designated safeguarding account, and the express statutory trust of the Post-Repeal Regime.
13. Operational resilience under SYSC 15A asks a firm to identify its important business services, set an impact tolerance in time for each, map the dependencies, test against severe but plausible scenarios and self-assess at board level.
14. A service is not made unimportant by being well protected, and the firm stays responsible for its tolerance even where a third party delivers the service.
15. A single incident and third-party reporting regime under PS26/2 comes into force on 18 March 2027, and HM Treasury designated the first critical third parties with effect from 13 July 2026.
16. Interchange on UK consumer debit is capped at 0.2% of value and on consumer credit at 0.3% under the assimilated Regulation (EU) 2015/751, with a Treasury power to lower but not to raise.
17. That cap reaches only one component of one part of the market, which is why total card acceptance costs have kept rising and why the priciest cards escape it entirely.
18. Interchange is contested on two tracks at once — the PSR's cross-border market review, whose capping power the High Court upheld on 15 January 2026, and the competition damages litigation in the Competition Appeal Tribunal — and the two can reach different answers about the same fee.
19. The institutional map is being redrawn: the government confirmed on 21 April 2026 that the PSR's functions will transfer to the FCA, but consolidation needs an Act of Parliament that had not been introduced by August 2026.
20. The whole body of assimilated payment services law is subject to repeal under the Financial Services and Markets Act 2023, so the regulation numbers used here may survive only in renumbered form, and every figure and date should be checked against the named source before it is relied on.

Sources: the Payment Services Regulations 2017, the Electronic Money Regulations 2011, the Payment and Electronic Money Institution Insolvency Regulations 2021, the Payment Card Interchange Fee Regulations 2015 and assimilated Regulation (EU) 2015/751, all via legislation.gov.uk; FCA CP24/20, PS25/12, CP24/28, PS26/2, PS24/16, PS21/3, the December 2025 contactless statement and the Approach Document Payment Services and Electronic Money — Our Approach; PRA and Bank of England PS16/24; PSR MR22/2.7, MR22/2.8, MR22/2.9 and PS24/7; HM Treasury's National Payments Vision, its April 2026 consultation response A Streamlined Approach to Payment Systems Regulation and its July 2026 critical third party designations; Re Ipagoo LLP [2022] EWCA Civ 302 and the High Court judgment of 15 January 2026; the European Parliament legislative train file on the payment services regulation; and the European Central Bank's Instant Payments Regulation implementation table.

54

CHAPTER

What Breaks and What It Costs

54.0 What this chapter gives you

1. You will be able to explain why a forty-minute terminal outage produces three weeks of consequences, and why the two numbers are not related to each other.
2. You will be able to do the arithmetic that turns 12,400 uncertain items into roughly 413 hours of manual research, and give senior management a recovery estimate they will not like but cannot argue with.
3. You will be able to explain why a partial batch is more dangerous than a total failure, and why resubmitting one converts a partial failure into a duplicate.
4. You will be able to name the five phases of an incident and say which of them firms are habitually bad at.
5. You will be able to list the clocks that start at once in a United Kingdom incident — four hours from first detection, seventy-two hours for a personal data breach, the end of the next business day for an unauthorised transaction, five business days for a scam claim — and say what triggers each.
6. You will be able to explain why the regulators fined TSB for the governance around the decision to go live rather than for the fault itself.
7. You will be able to say which rails can be pulled back and which cannot, and explain that settlement finality is a deliberately created legal state rather than a property of a message.
8. You will be able to explain why batch file submission has no idempotency key, and why duplicate protection is therefore a control the submitter must build rather than a service the scheme provides.
9. You will be able to distinguish operational resilience, the critical third parties regime and incident reporting, and explain why the three are one policy.
10. You will be able to specify a control that alerts on the absence of an expected file, which is the cheapest fix for the most expensive silent failure in the industry.

Every chapter of this book so far has described something working. Cards authorising, files clearing, settlement landing, mandates honoured, disputes adjudicated. That is the right way round, because you cannot understand a failure in a system you do not understand working. But it leaves an impression that needs correcting before we finish, which is that payments is a set of mechanisms that occasionally break.

It is not. Payments is a set of mechanisms that break continually at a low rate, and the entire apparatus around them — the reconciliation, the exception queues, the indemnities, the redress rules, the regulators, the two o'clock in the morning phone trees — exists because that is the normal condition rather than the exceptional one. A payment system is not a machine that works and sometimes fails. It is a machine that fails at a known rate, wrapped in enough correction to make the failures survivable.

The published record is unambiguous about the rate. In March 2025 the House of Commons Treasury Committee published correspondence from nine of the largest banks and building societies operating in the United Kingdom, and the aggregate was at least 803 hours of unplanned technology and systems outage — more than thirty-three days — across at least 158 separate incidents between January 2023 and February 2025. Nine well-capitalised, heavily supervised institutions, in a mature market, with mature controls. That number is not a scandal. It is the baseline.

This chapter is about what happens inside those hours and, much more importantly, in the weeks after them. The thing practitioners learn early and outsiders almost never see is that the outage is the small part. What outlasts it is the reconciliation, and the reconciliation is where the money actually is.

The plain version

Start with a water main.

A pipe bursts under a road at seven in the morning. The water company gets a crew there by nine and the leak is stopped by half past ten. Three and a half hours, and the problem in the technical sense is over. Nobody thinks the story ends there. The road is dug up for a month. The trench has to be backfilled, the tarmac replaced, the two shops whose cellars flooded dealt with, the claims from people whose cars were damaged assessed one at a time, and somebody has to work out whether the pipe that burst was the one that was supposed to have been replaced in 2019.

Payments incidents are exactly like this, and people consistently misjudge them for exactly the same reason. The visible event — the app is down, the card is declined, the wages have not arrived — is the leak. Fixing the leak is a few hours of engineering. The road is the ledger, and the ledger takes weeks.

Here is a real-shaped example with real arithmetic.

Riverside Court is a care home. It employs eighty-four people and pays them monthly, and the total that leaves its bank account on payday is £176,400, an average of £2,100 each. The way it pays them is old and reliable: on the Wednesday, the payroll clerk sends a list to the bank containing eighty-four lines, each line saying who to pay, how much, and which account. Three working days later the money lands.

One month, the confirmation screen does not load. The clerk waits, refreshes, sees nothing, and sensibly assumes the list did not go. So she sends it again. Both lists were correctly signed, both were validly formatted, and the bank has no way of knowing that the second one is not a genuine second run of wages. On the Friday, £352,800 leaves the account. Every single member of staff is paid twice.

Now watch how long each part takes.

The cause takes about four seconds to explain and about four minutes to fix: the software gains a check that warns you when you are about to send a list identical to one sent in the last twenty-four hours. That is the whole technical remedy. It could be written on a Monday morning.

Detection takes three days, and this is the first surprise. Nobody notices on Friday. The staff certainly do not ring up to complain about being paid twice. The bank does not stop it, because nothing about the second list was invalid. The alarm is not an alarm at all: on Monday the finance manager opens the bank statement for an unrelated reason, sees a balance £176,400 lower than she expected, and has to work backwards from a number to a cause. In payments, this is the ordinary way things are found. Not a red light. A figure that is wrong.

Then comes the part that takes eleven weeks.

The money has gone into eighty-four accounts belonging to eighty-four adults, and it is now legally theirs to argue about. Riverside writes to all of them on the Tuesday. Sixty send it back within a fortnight: £126,000 recovered. Fifteen more come back over the following six weeks, some awkwardly, because they had already spent part of it — one had cleared a credit card, one had paid a deposit on a holiday. That is another £31,500, some of it in instalments. Six agree to repay over four months. Three never pay at all: one has left the country, one disputes being overpaid, one simply stops answering. £6,300 is written off.

Meanwhile the books do not balance, in a specific and irritating way. The payroll system recorded one payroll. The bank recorded two payments to each person. So for eleven weeks there is an unexplained hole that starts at £176,400 and shrinks unevenly as money dribbles back, and every returned payment has to be matched to the right person by hand, because people send money back with references like "wages" and "sorry". The cash flow forecast was wrong for two months. The overdraft was used, costing £940 in interest and fees. The auditor asks about it the following March, so somebody writes the whole thing up again seven months later.

Add it up. Four minutes of code. Roughly sixty-five hours of finance and payroll time at Riverside. £6,300 written off, £940 of borrowing costs, one very uncomfortable staff meeting, and a company that will be nervous about payday for a year.

The same shape appears when nothing goes out at all. A café's card terminal stops taking payments at half past twelve on a Saturday and is fixed in forty minutes, and the café loses perhaps £600 of lunchtime trade that never comes back. But some of the failed taps were not clean failures. A handful of customers were charged, saw a decline on the terminal, and paid again with a different card. Sorting out fourteen double charges — finding them, matching them to receipts, refunding them, answering the two customers who go to their banks instead and start a formal dispute — takes the owner three weeks of intermittent effort and one furious online review.

Forty minutes of outage, three weeks of consequences. If you remember one sentence from this chapter, make it that one: the outage is measured in hours, the recovery is measured in weeks, and the two numbers are not related.

One last thing the plain version needs to establish, because it is why all of this is so laborious. When money moves and settles, it is genuinely gone. It is not sitting somewhere temporary waiting to be recalled. Somebody else now owns it, and getting it back requires either their agreement or a court. That is not a design flaw. It is the entire point of a payment: to end the argument about who owes what. A payment you could quietly reverse would not be a payment, it would be a loan with extra steps. The price of finality — the thing that lets the shopkeeper hand over the goods — is that a mistake is expensive to undo.

Where the plain version stops being true

The clock that matters does not start when you understand the problem. The Riverside story has detection on the Monday and everything flowing from there. In a regulated firm, the regulatory clock is deliberately not tied to your understanding. At the time of writing in August 2026, a UK payment service provider that becomes aware of a major operational or security incident must notify the Financial Conduct Authority without undue delay under regulation 99 of the Payment Services Regulations 2017, with the timing currently specified through the European Banking Authority's Guidelines on incident reporting under PSD2, EBA/GL/2017/10 as issued on 27 July 2017, applied by SUP 15.14 of the FCA Handbook. That guidance requires an initial report within four hours of first detection. Not from root cause, not from confirmed impact, not from the incident bridge agreeing what to call it.

Firms must therefore be able to file a report saying, in effect, something is wrong, we do not yet know what, here is what we do know. The instinct to wait until you can give a clean answer is the instinct that produces a breach.

Duplicate is easier to explain than partial, and partial is far more dangerous. The Riverside example is a duplicate, which is tidy: you know exactly what happened, twice, to exactly whom. Real incidents are more often partial, and partial is worse in a way that is not intuitive. If a batch of 40,000 payments fails completely, you know the state of the world: nothing happened, resend. If 31,600 succeeded, 2,900 failed cleanly, and 5,500 returned no answer at all, you do not know the state of the world and cannot find out from your own records. You have to reconcile against somebody else's. Until you have, you cannot safely resubmit, because resubmitting the ambiguous 5,500 might duplicate them and not resubmitting them might leave 5,500 people unpaid. A clean total failure is a good day. Ambiguity is the expensive one.

The cost is almost never the outage. Riverside's direct loss was small. The pattern in published cases is that the operational loss is a rounding error against the redress, the remediation, the enforcement and the supervisory tail. TSB's platform migration in April 2018 gives the clearest published ratio: the Financial Conduct Authority and Prudential Regulation Authority stated when fining the bank a combined £48,650,000 on 20 December 2022 that all of TSB's branches and a significant proportion of its then 5.2 million customers were affected by the initial issues, that the problems were not fully resolved for eight months, and that TSB had by then paid £32.7 million in redress. Against that, the engineering hours to restore service were trivial. The asymmetry runs the other way too: the Bank of England's Real-Time Gross Settlement system was down for roughly nine hours on 20 October 2014, delaying 142,759 CHAPS payments worth £289.3 billion, and the Bank's published response of 25 March 2015 records that it settled nine compensation claims totalling £4,056.89. A national settlement outage produced about four thousand pounds of claims; a retail migration produced tens of millions. Where the harm lands, not how big the system is, determines the bill.

"The technical fix is the short part" has a real exception, and it is the migration case. When the failure is a design decision rather than a fault — a new platform that does not behave as the old one did, a data model that lost information in transit, a capacity assumption wrong by an order of magnitude — there is no short fix, because there is nothing to restore to. TSB's disruption ran for eight months. Deloitte's review found the root cause of the 2014 RTGS outage to be defects introduced in functionality changes made in April 2013 and May 2014, meaning the fault had been latent for around eighteen months before it expressed itself. Incidents caused by a change you deliberately made are a different species from incidents caused by a component that failed, and they behave differently in every dimension: longer, more expensive, more likely to attract enforcement, and much more likely to be attributed to governance rather than technology.

The technical version

On dating

Every figure, deadline, threshold and rule in this section is stated as accurate at the time of writing, in August 2026. Operational resilience is one of the fastest-moving areas of financial regulation in the United Kingdom and the European Union, and three specific things were in motion as this was written: the FCA's and PRA's new operational incident and third-party reporting regime had been finalised but did not come into force until 18 March 2027; HM Treasury had made its first designations under the critical third parties regime only weeks earlier; and the EU's Digital Operational Resilience Act had been applicable for less than two years, with its supervisory practice still forming. Verify every number below against the primary source before relying on it.

The anatomy of an incident

An incident has five phases, and firms tend to be good at the second and third and bad at the first, fourth and fifth.

Detection. The commonest detection mechanism in payments is not an error alert. It is a volume anomaly: throughput that has departed from its expected shape for the time of day, the day of the week and the point in the month. This matters because the failure modes that hurt most produce no errors at all. A file that is never picked up throws nothing. A downstream system that silently truncates a field returns success. A duplicate submission is, by construction, two valid events. Error-rate monitoring finds the loud failures; throughput and reconciliation monitoring find the quiet ones, and the quiet ones cost more. Mature operations therefore alert on the absence of expected traffic — no Bacs input report by 23:00, no settlement file by 06:00, credit volumes 40 per cent below the same weekday last week — rather than only on the presence of errors.

Classification. Classification is a regulatory act, not an engineering one, because it starts clocks. The problem is that it demands an impact estimate at the moment when impact is least knowable. The discipline that works is to classify on plausible worst case and revise downwards later, because every reporting regime tolerates an overstated report far better than a late one.

Containment. A genuine tension runs through every serious incident: stopping the bleeding usually destroys evidence. Failing over destroys the state of the failed node. Restarting a stuck process discards the queue you needed to count. Reprocessing a file overwrites the log that told you what the first attempt did. Good runbooks resolve this in advance by specifying what must be captured before each containment action, so the decision is not being made by a tired engineer at four in the morning who will later be asked, by a skilled person under section 166 of the Financial Services and Markets Act 2000, why there is no record of what happened.

Recovery. Restoring service is not the same as restoring correctness, and conflating them is the classic mistake. Service is restored when payments flow again. Correctness is restored when every payment that should have happened has happened exactly once, and every payment that should not have happened has been reversed or paid for. The gap between those two moments is where the reconciliation lives: typically two to eight weeks for a material incident.

Aftermath. Redress, reporting, root cause analysis, remediation, and the supervisory conversation, which continues long after the engineering is closed.

Command structure is where competent firms differ visibly from incompetent ones. There is one incident commander, who does not fix anything. There is a separate communications lead, because the person diagnosing must not also be the person drafting the customer statement. There is a scribe

keeping a timestamped log, because the log is the evidence and cannot be reconstructed afterwards. And there is a named accountable senior manager: in the United Kingdom, operational resilience typically sits with the Chief Operations function, SMF24, under the Senior Managers and Certification Regime, which means the failure has a person's name attached to it in a way that predates the incident.

The clocks

The single hardest operational fact about a modern incident is that several unrelated deadlines start at once, they are triggered by different events, and they are owed to different people.

Obligation	Trigger	Deadline	Basis, as at August 2026
UK major payments incident, initial	First detection	4 hours	PSRs 2017 reg. 99, SUP 15.14, EBA/GL/2017/10
UK personal data breach	Awareness	72 hours	UK GDPR Art. 33
EU major ICT incident, initial	Classification as major	4 hours, and no later than 24 hours from becoming aware	DORA Art. 19(4)(a)
EU major ICT incident, intermediate	Initial notification	72 hours	DORA Art. 19(4)(b)
EU major ICT incident, final	Intermediate report	1 month	DORA Art. 19(4)(c)
Refund of unauthorised transaction	Notification by payer	End of the next business day	PSRs 2017 reg. 76
APP scam reimbursement	Claim	5 business days, subject to stop the clock	PSR reimbursement requirement, from 7 Oct 2024
UK incident report, from 18 Mar 2027	Threshold determination	Ordinarily 24 hours, but 4 hours from first detection for PSPs	FCA PS26/2, FG26/3; PRA PS7/26, SS1/26

Two of those deserve elaboration.

DORA — Regulation (EU) 2022/2554, applicable since 17 January 2025 — sets its initial four-hour clock from classification rather than detection, but caps it by requiring notification no later than twenty-four hours from becoming aware of the incident. Classification itself must be done without undue delay, against the criteria in the Commission Delegated Regulation on classification of major ICT-related incidents. The design intent is to stop firms deferring the clock by deferring the decision.

The United Kingdom is replacing its arrangements. On 18 March 2026 the FCA published PS26/2, with finalised guidance FG26/3 on operational incident reporting and FG26/4 on material third-party reporting, alongside the PRA's PS7/26 and supervisory statement SS1/26. The rules come into force on 18 March 2027. They create a single definition of an operational incident, a single template and a single submission route through FCA Connect for firms that today report separately to two or three authorities. An operational incident is a single event or a series of linked events which disrupt the firm's operations such that they disrupt delivery of a service to an end user external to the firm, or affect the availability, integrity, authenticity or confidentiality of that end user's data. Reporting is split into standard and enhanced tiers, with payment service providers in the enhanced tier and, crucially, retaining the accelerated four-hour deadline from first detection while other enhanced firms work to twenty-four hours from threshold determination. The EBA guidelines will be disapplied for UK PSPs,

who will discharge regulation 99 solely through the new regime. Final reports are due within thirty working days of resolution, or no later than sixty with reasons for the delay.

Failure modes, and why some are worse than others

Duplicate submission and duplicate settlement. The commonest serious payments incident, and the one with the worst recovery profile, because a duplicate is by definition a valid instruction. The largest published UK example is Santander's Christmas Day 2021 incident, in which approximately £130 million was paid out in around 75,000 duplicate transactions originating from about 2,000 business accounts. Recovering a duplicate credit means recovering funds from third parties who have no contractual relationship with the party that made the mistake, which is why the money often does not come back.

Stuck file. The instruction was created but never delivered, or delivered and never picked up. Zero errors, zero alarms, zero payments. Only a positive control catches this: a reconciliation that expects a specific file, of roughly a specific size, by a specific time, and alerts on its absence.

Partial batch. Some items processed, some did not, and the boundary is unknown. This is the failure mode that most often produces the wrong recovery action, because the temptation to resubmit the whole file is enormous and doing so converts a partial failure into a duplicate.

Silent corruption. Field-level truncation and encoding faults are especially nasty in the file-based world because the legacy formats are fixed-width and unforgiving. In the Bacs Standard 18 format, the fields carrying the counterparty's name and the payment reference are eighteen characters. Anything that overruns is cut, and a cut reference is a payment that arrives but cannot be applied. The sender sees success. The receiver sees an unidentifiable credit and a reconciliation break.

Idempotency, and why batch payments lack it. Modern payment APIs are built around an idempotency key: a client-generated identifier that lets the server recognise a retry and return the original result rather than performing the action twice. That solves the Riverside problem completely. Batch file submission has no equivalent primitive. A Bacs submission is authenticated and non-repudiable, and is therefore trusted; it is not deduplicated, because two genuine payroll runs to the same eighty-four people for the same amounts on consecutive days is a legitimate business event no scheme can distinguish from an error. Duplicate protection in the file world is a control submitters must build themselves.

Reversal asymmetry

Whether a payment can be pulled back is not a question about technology. It is a question about which rail it went through, and the answers differ sharply.

Rail	Reversal mechanism	Practical character, as at August 2026
Bacs Direct Credit	No unilateral reversal; recall by request to the receiving PSP, requiring the beneficiary's consent	Weak. Success depends on goodwill and on the money still being there
Bacs Direct Debit	Direct Debit Indemnity Claim under the Direct Debit Guarantee	Strong for the payer. The paying PSP refunds first and recovers from the originator afterwards, with a challenge process available to the service user

Rail	Reversal mechanism	Practical character, as at August 2026
Faster Payments	Recall request; no guaranteed return	Weak, and fast money is fast in both directions
CHAPS	Irrevocable between direct participants once settled	None, absent the beneficiary's agreement
Card	Chargeback under scheme rules, within scheme-defined windows	Strong for the cardholder, structured, and adversarial

Underneath all of this sits settlement finality. The Financial Markets and Insolvency (Settlement Finality) Regulations 1999, S.I. 1999/2979, allow payment systems to be designated so that transfer orders effected through them are protected from being unwound by insolvency law. CHAPS was designated in 2000 and is a recognised interbank payment system under Part 5 of the Banking Act 2009. Finality is not a technical property of a message; it is a legal state, deliberately created, that exists precisely so that a settled payment cannot be reopened by later events.

The consequences of that design are best shown by the largest published misdirected-payment case of recent years. On 11 August 2020, Citibank, acting as administrative agent on a Revlon term loan, sent lenders the full outstanding principal when it had intended to send only an interest payment. Some returned the money. Others did not. The sum in dispute — principal and accrued interest owed to the lenders who declined to return it — was $558,558,375.74. The United States District Court for the Southern District of New York held in February 2021 that the discharge-for-value rule gave those lenders a defence against restitution, because they were owed exactly what they received and had no notice of the mistake. The Second Circuit vacated that ruling on 8 September 2022 in *In re Citibank August 11, 2020 Wire Transfers*, No. 21-487, holding among other things that the debt was not then due and that the recipients had constructive notice. The point for a practitioner is not the eventual outcome. It is that recovering an erroneous payment of over half a billion dollars from sophisticated counterparties took two years and two courts, and at the halfway mark the money was gone. Payment systems are built to make payments stick. They succeed.

The reconciliation aftermath

This is the part of an incident that no press release describes and every practitioner remembers.

Reconciliation after an incident is a three-way problem, not a two-way one. There is the scheme's or acquirer's view, expressed in settlement and clearing files; the bank's view, expressed as movements on the settlement or client account; and the firm's own ledger, its record of what it believed it did. In normal operation these three agree to within a small and stable break rate. After an incident they do not, and the work is to reduce the disagreement to zero, item by item, with each item explained.

Breaks fall into recognisable categories, and the category determines the remedy:

Break type	Meaning	Typical remedy
In the ledger, not at the bank	We think we paid; no money moved	Reissue after confirming non-payment
At the bank, not in the ledger	Money moved; we have no record of instructing it	Investigate as potential duplicate or unauthorised
Matched, wrong amount	Both agree it happened, disagree on value	Adjust, then find the truncation or rounding fault

Break type	Meaning	Typical remedy
Matched, wrong date	Value-dating disagreement	Usually interest and reporting impact only
Unidentified receipt	Money arrived, cannot be applied	Suspense, then manual research, then escalating attempts to contact
Duplicate	Two records, one intended event	Recall attempt, then recovery, then write-off

Unmatched items go to a suspense account, which is not a solution but a holding pen with a clock on it. Two disciplines matter: ageing, because a two-day break is operations and a ninety-day break is a provision, and named ownership, because unowned breaks are immortal.

The reason this outlasts the outage is arithmetic. Suppose an incident leaves 12,400 items in an uncertain state. Automated matching on amount, date, counterparty and reference will typically resolve 85 to 95 per cent of them, so call it 1,240 items left. Manual research on a payments break — pulling the original instruction, the scheme record and the bank movement, and forming a view — averages fifteen to twenty-five minutes when straightforward and runs to hours when it involves contacting a beneficiary. At twenty minutes each, 1,240 items is roughly 413 hours: eleven weeks for one full-time person, or three weeks for a team of four who also have their normal work to do. Then there is the tail, the items requiring a third party's cooperation, which arrive back over months, and the small residue that never resolves and becomes a write-off with an audit trail.

That is why a forty-minute outage produces a six-week recovery, and why the honest answer to "when will this be finished?" on the day of an incident is a number that senior management will not like.

Customer impact, redress and the legal floor

Redress in the United Kingdom is not discretionary generosity. Several distinct regimes apply at once, and they attach to different failures.

For unauthorised transactions, regulation 76 of the Payment Services Regulations 2017 requires the payer's payment service provider to refund the amount as soon as practicable and in any event no later than the end of the business day following the day on which it becomes aware of, or is notified of, the transaction, and to restore the account to the state it would have been in. There are exceptions where the provider has reasonable grounds to suspect fraud and reports them appropriately. Regulations 91 to 95 deal with non-execution, defective execution and late execution, allocating liability between the payer's and payee's providers and requiring the payment to be traced.

For authorised push payment scams, the Payment Systems Regulator's reimbursement requirement has applied to eligible Faster Payments claims since 7 October 2024, with the Bank of England setting the equivalent for CHAPS. Reimbursement is due within five business days, subject to a stop-the-clock provision where the sending provider needs more information, and the maximum mandatory level is £85,000 per claim, a figure the PSR settled on having previously proposed £415,000. Firms may reimburse more, and losses above it can be taken to the Financial Ombudsman Service.

For everything else — the distress and inconvenience of an outage, the consequential losses, the missed mortgage completion, the declined card at the till — there is complaint handling under DISP in the FCA Handbook, the Financial Ombudsman Service, and since July 2023 the Consumer Duty, which requires firms to act to deliver good outcomes for retail customers and specifically covers consumer

support. At the time of writing, the Ombudsman's award limits rose on 1 April 2026 to £455,000 for complaints about acts or omissions on or after 1 April 2019, and £205,000 for earlier ones.

The scale of voluntary distress payments is now public record because the Treasury Committee asked. Barclays confirmed that during its outage from 31 January to 2 February 2025, 56 per cent of online payments failed as a result of severe degradation of mainframe processing performance, and that it expected to pay between £5 million and £7.5 million for inconvenience or distress, taking its total potential payout over the two-year period to as much as £12.5 million. The second-highest figure among the nine firms was £350,000, from Bank of Ireland. One very large number, then a cliff: these payments are driven by the severity and timing of individual events, not by aggregate downtime. The Barclays outage landed on payday.

Four published incidents, and what each one teaches

Bank of England RTGS, 20 October 2014. Roughly nine hours of outage in the system that settles CHAPS. Operating hours were extended to 20:00 and all submitted payments, 142,759 of them worth £289.3 billion, settled the same day. Deloitte's independent review, published with the Bank's response on 25 March 2015, found the root cause to be defects introduced as part of functionality changes made in April 2013 and May 2014. The Bank aims for 99.95 per cent availability of RTGS settlement services to CHAPS, and had achieved 100 per cent from 2008 to 2013. Nine compensation claims were settled, totalling £4,056.89. The lessons the Bank accepted were almost entirely about governance, testing and crisis management rather than software: reconstituting the RTGS board under a Deputy Governor, deferring non-routine changes, separating test and pre-production environments, and reducing the barriers to invoking the MIRS contingency solution so that it could switch and fix in parallel rather than treating failover as a last resort. That last point is the most transferable. A contingency you are afraid to invoke is not a contingency.

Visa Europe, 1 June 2018. Charlotte Hogg, then Visa's European chief executive, wrote to the Treasury Committee explaining that the disruption ran from 14:35 on 1 June to 00:45 the following day, that 51.2 million transactions were initiated across Europe in that window and 5.2 million failed, and that 2.4 million UK transactions failed to process. The cause was a very rare partial failure of a component within a switch at the primary of two data centres, either of which was designed to handle 100 per cent of European volume alone. The partial nature of the failure is the whole lesson: because the primary had not cleanly failed, the backup switch did not activate, and the malfunctioning primary kept trying to synchronise messages with the secondary site, creating a backlog that degraded the secondary's own processing. It took nearly five hours to fully deactivate the failing system. Redundancy protects against a component that stops. It does not automatically protect against a component that continues, badly.

TSB, April 2018. A platform migration in which, as the regulators put it, the data itself migrated successfully but the platform immediately experienced technical failures. On 20 December 2022 the FCA fined TSB £29,750,000 and the PRA £18,900,000, a combined £48,650,000 after a 30 per cent early-resolution discount, for operational risk management and governance failures including management of outsourcing risk. Note what was penalised. Not the fault. The governance around the decision to go live.

CrowdStrike, 19 July 2024. At 04:09 UTC a faulty content configuration update to the Falcon sensor caused an out-of-bounds memory read in the Windows sensor client, crashing machines into a boot loop. The update was reverted at 05:27 UTC, seventy-eight minutes later, and machines that booted after the reversion were unaffected. Microsoft estimated roughly 8.5 million Windows devices were affected, less than one per cent worldwide. Recovery took days, because many machines needed in-

dividual physical intervention and could not be fixed remotely by the very software that had broken them. This is the chapter's central ratio at global scale: the vendor's fix took seventy-eight minutes and the world took a week. It is also why the concentration of financial firms on a handful of shared suppliers became a regulatory priority rather than a procurement one.

The regulatory frame around resilience

Three regimes now sit on top of the failure modes described above, and they are worth distinguishing because firms routinely conflate them.

Operational resilience is about outcomes, not systems. Under the FCA's PS21/3 and the corresponding PRA rules, firms must identify their important business services, set an impact tolerance for each — the maximum tolerable level of disruption, expressed as a duration alongside other relevant metrics, beyond which further disruption would cause intolerable harm to consumers or risk to market integrity — and map and test to the point where they can remain within those tolerances in severe but plausible scenarios. Services had to be identified and tolerances set by 31 March 2022, and the transition period for being able to stay within them ended on 31 March 2025. The conceptual shift is from availability targets, which are about systems, to impact tolerances, which are about customers and which assume disruption will happen rather than trying to prevent it.

Critical third parties. Section 312L of the Financial Services and Markets Act 2000, as amended by the Financial Services and Markets Act 2023, allows HM Treasury to designate a third-party service provider as a critical third party where, in its opinion, failure or disruption of its services to firms could threaten the stability of, or confidence in, the UK financial system. The joint rules of the Bank of England, PRA and FCA, published on 12 November 2024 as PRA PS16/24 and FCA PS24/16, took effect on 1 January 2025 but bite on a provider only when a designation order comes into force. HM Treasury announced its first designations on 10 July 2026, effective 13 July 2026: Microsoft Ireland Operations Limited, Google Cloud EMEA Limited, Amazon Web Services EMEA SARL and Oracle Corporation UK Limited. Two features are commonly misunderstood. Oversight extends only to the systemic services provided to the financial sector, not to the provider's wider business. And designation transfers nothing: firms remain fully accountable for the risks in their own third-party arrangements, and a designated provider is not thereby safer than an undesignated one.

Incident and third-party reporting is the data layer that makes the other two supervisable. PS26/2 and PS7/26 also require in-scope firms to notify material third-party arrangements before commitments are finalised and to submit an annual register of them, explicitly so that the regulators can identify concentration risk and inform future critical third party designations. The three regimes are one policy.

What good operational practice actually looks like

Strip away the frameworks and the practices that distinguish firms which recover well are unglamorous and specific.

Design for reconciliation before designing for throughput. Every payment instruction should carry an identifier that survives every hop and appears in every record on both sides, so that three-way matching is a lookup rather than an investigation. Retrofitting this after an incident is close to impossible, which is why it has to be a build-time decision.

Make duplicate prevention a control, not a hope. Hash every outbound file and compare it against submissions from the preceding days. Maintain a submission register that operations staff, not just engineers, can read. Require a second authoriser for any submission matching a recent one in value and item count. Set value limits with the sponsoring bank so that an anomalous total cannot be sub-

mitted at all. Riverside's four minutes of code is the entire fix, and almost nobody writes it until after the first time.

Alert on absence, not only on errors. The expected file that never arrives is the most expensive silent failure in the industry.

Rehearse the failover you are afraid of. The Bank of England's own accepted recommendation after 2014 was to reduce the barriers to invoking its contingency solution so that it could switch and fix in parallel. A failover that has never been executed in anger, or that requires a decision nobody wants to make, is a document rather than a capability. This is what the impact-tolerance regime is really testing.

Pre-agree the recovery paperwork: recall templates, beneficiary letters, the customer notice, the redress matrix saying what is paid for what category of harm. Drafting these during an incident guarantees they will be slow, inconsistent and later criticised.

Separate the restore-service team from the restore-correctness team on day one. They have different skills, timescales and definitions of done, and if they are the same people the reconciliation starts two weeks late because everyone is exhausted.

Write a blameless post-incident review with a real timeline, publish the actions with owners and dates, and then check them. The commonest finding in enforcement notices is not that a firm failed to identify a weakness. It is that the firm identified it, wrote it down, and did not fix it in time. Keep the incident log as if a regulator will read it, because one may: contemporaneous, timestamped, factual, including the decisions not taken and why.

Returning to the tap

This book opened in a shop in Leeds, with a card touched against a terminal, and with the observation that almost nobody's description of what happens next is correct. Money did not move. No object travelled. A record was edited, and everyone accepted the edit because everyone accepted the record-keeper.

Fifty-three chapters later, you know what the edit involves. That the terminal runs an application which must satisfy a kernel specification, that the token in the phone is not the card number, that the authorisation travels through an acquirer to a scheme to an issuer and back inside a second or two, and that it is a promise rather than a payment. That the movement happens hours later in a settlement file, and finality later still across accounts at the Bank of England. You know who is liable if the card was stolen, who is liable if the shopkeeper never delivers, what the interchange fee is and why it is capped, why the merchant's money arrives on Wednesday, and which regulator would care if it did not.

You also now know the thing the tap is designed to conceal. Behind that gesture sit several dozen organisations, any of which can fail; a legal framework whose entire purpose is to make the result stick even when somebody has made a mistake; a reconciliation apparatus that quietly resolves millions of small disagreements every day; and people whose job, when the disagreements stop being small, is to spend six weeks matching numbers to numbers until the two sides of a ledger agree again.

In 2024, according to UK Finance's UK Payment Markets report, nearly 49 billion payment transactions were made in the United Kingdom, of which 18.9 billion were contactless. Pay.UK's figures for 2025 show approximately 5.03 billion Direct Debits and 1.83 billion Bacs Direct Credits, including around 378 million payroll payments and 267 million state pension payments, running through a system whose cycle is still three working days and whose input files are still eighteen characters wide in

the fields that matter. The overwhelming majority of those payments worked. The ones that did not were caught, in most cases, by somebody looking at a number that was wrong.

That is what money moving actually is. Not a substance travelling, but a very large number of records being written, checked against each other, and corrected. Ada's notebook, with more lawyers.

The tap takes half a second. Everything in this book is what makes the half-second safe to trust, and everything in this chapter is what happens on the days when it is not. Both halves are the system. A reader who understands only the first half understands payments the way somebody who has only watched aircraft take off understands aviation.

Now you have seen the landings too.

54.98 Common wrong ideas

Wrong: The regulatory clock starts when you understand what has gone wrong. **Right:** The initial report is due within four hours of first detection, not from root cause or confirmed impact, so a firm must be able to file a report that says something is wrong, we do not yet know what, here is what we do know.

Wrong: A complete failure is the worst outcome. **Right:** A clean total failure tells you the state of the world and can be resent; ambiguity is the expensive one, because items that returned no answer at all can be neither safely resubmitted nor safely abandoned.

Wrong: The cost of an incident is the outage. **Right:** The operational loss is usually a rounding error against redress, remediation, enforcement and the supervisory tail — TSB's combined fine was £48,650,000 against £32.7 million already paid in redress, while nine hours of national settlement outage in 2014 produced nine claims totalling £4,056.89.

Wrong: The bigger the system that failed, the bigger the bill. **Right:** Where the harm lands determines the bill, which is why a retail migration cost tens of millions and an RTGS outage cost four thousand pounds.

Wrong: The technical fix is always the short part. **Right:** Where the failure is a design decision rather than a fault there is nothing to restore to, which is why TSB's disruption ran for eight months and why the 2014 RTGS defects had been latent for around eighteen months.

Wrong: The incident is over when service is restored. **Right:** Service is restored when payments flow again; correctness is restored when every payment that should have happened has happened exactly once, and the gap between those two moments is typically two to eight weeks.

Wrong: Redundancy protects against the failure of a component. **Right:** It protects against a component that stops, not one that continues badly — Visa's primary switch in June 2018 failed partially, so the backup never activated and the malfunctioning primary degraded the secondary site as well.

Wrong: Error-rate monitoring will find the incident. **Right:** The failure modes that cost most produce no errors at all, so mature operations alert on the absence of expected traffic rather than only on the presence of errors.

Wrong: A settled payment sent by mistake can be pulled back. **Right:** Finality exists precisely to stop that, which is why recovering Citibank's $558,558,375.74 of mis-sent Revlon principal took two years and two courts, with the money gone at the halfway mark.

Wrong: Designating a cloud provider as a critical third party makes the firms relying on it safer. **Right:** Oversight extends only to the systemic services provided to the financial sector, designation transfers nothing, and firms remain fully accountable for the risks in their own third-party arrangements.

54.99 Chapter summary in 20 lines

1. A payment system is not a machine that works and sometimes fails, but a machine that fails at a known rate wrapped in enough correction to make the failures survivable.
2. Nine of the largest United Kingdom banks reported at least 803 hours of unplanned outage across at least 158 incidents between January 2023 and February 2025, which is the baseline rather than a scandal.
3. The visible event is the leak and the ledger is the road, and the road takes weeks after the leak is stopped.
4. Riverside Court's duplicated payroll of £176,400 needed four minutes of code to prevent, took three days to detect and eleven weeks to reconcile, and ended in £6,300 written off plus £940 of borrowing costs.
5. Detection in payments is ordinarily a wrong number noticed for an unrelated reason, not a red light.
6. Once money has moved and settled it is genuinely gone, and getting it back requires the recipient's agreement or a court, because ending the argument about who owes what is the whole purpose of a payment.
7. Several unrelated regulatory clocks start at once, triggered by different events and owed to different people, and the United Kingdom four-hour clock runs from first detection.
8. Classification is a regulatory act rather than an engineering one, because it starts those clocks at the moment impact is least knowable.
9. Containment usually destroys evidence, so good runbooks specify in advance what must be captured before each containment action.
10. Restoring service is not restoring correctness, and conflating them is the classic mistake.
11. Partial failures are worse than total ones because the state of the world cannot be established from the firm's own records, and the temptation to resubmit the whole file converts a partial failure into a duplicate.
12. Batch file rails have no idempotency key, because two identical payroll runs on consecutive days are a legitimate business event no scheme can distinguish from an error, so duplicate prevention is a control submitters must build themselves.
13. Whether a payment can be reversed depends on the rail: Direct Debit and cards give strong structured remedies, Bacs credits and Faster Payments give only a request, and CHAPS gives none once settled.
14. Post-incident reconciliation is a three-way problem between the scheme's view, the bank's view and the firm's own ledger, and every break must be categorised, aged and given a named owner.
15. The arithmetic is unforgiving: 12,400 uncertain items, ninety per cent matched automatically, twenty minutes each on the remainder, is about 413 hours before the tail and the write-offs even begin.
16. Redress is not discretionary generosity but a stack of regimes — regulation 76 for unauthorised transactions, regulations 91 to 95 for defective execution, the £85,000 scam reimbursement requirement, DISP, the Ombudsman and the Consumer Duty.
17. Barclays expected to pay between £5 million and £7.5 million in distress payments for one outage that landed on payday, while the next highest figure among nine firms was £350,000, because severity and timing drive these payments rather than aggregate downtime.

18. The four published incidents teach four distinct lessons: a contingency you are afraid to invoke is not a contingency, redundancy does not protect against a component that continues badly, regulators penalise the governance of the decision rather than the fault, and a seventy-eight-minute vendor fix can take the world a week.
19. Operational resilience, the critical third parties regime and the new incident and third-party reporting rules coming into force on 18 March 2027 are three parts of one policy, the last being the data layer that makes the other two supervisable.
20. Money moving is not a substance travelling but a very large number of records being written, checked against each other and corrected, and this chapter is what happens on the days when the correcting becomes visible.

Sources, all consulted in August 2026: the Payment Services Regulations 2017, regulations 76, 91 to 95 and 99, and FCA Handbook SUP 15.14; EBA Guidelines EBA/GL/2017/10; FCA PS26/2 with FG26/3 and FG26/4, and PRA PS7/26 with SS1/26, published 18 March 2026 and in force 18 March 2027; FCA PS21/3 and the FCA's operational resilience insights pages; PRA PS16/24 and FCA PS24/16 on critical third parties, with HM Treasury's designation announcement of 10 July 2026; section 312L of the Financial Services and Markets Act 2000 as amended by FSMA 2023; Regulation (EU) 2022/2554 (DORA), Article 19; the Financial Markets and Insolvency (Settlement Finality) Regulations 1999, S.I. 1999/2979; the Bank of England's response to the independent review of the RTGS outage of 20 October 2014, published 25 March 2015; the Bank of England and FCA announcement of the TSB fine, 20 December 2022; Visa's letter to the Treasury Committee of 15 June 2018; the Treasury Committee's publication of bank IT failure correspondence, 6 March 2025; Payment Systems Regulator PS24/7 and its APP reimbursement pages; Financial Ombudsman Service award limits for 1 April 2026; Pay.UK's Bacs glossary and Bacs annual processing statistics 2025; UK Finance UK Payment Markets 2025; Microsoft's statement of 20 July 2024 and CISA's alert of 19 July 2024; and In re Citibank August 11, 2020 Wire Transfers, No. 21-487 (2d Cir., 8 September 2022).

ABOUT THE AUTHOR

Shikhar Singh

Founder & Chairman

KedByte Technologies Private Limited

Shikhar Singh is the Founder & Chairman of KedByte Technologies Private Limited.

He wrote this book from the position of someone building payment software rather than someone commentating on it, and the difference shows in what it chooses to explain. The chapters that go deepest are the ones where he has watched capable engineers lose days: the gap between authorisation and settlement, the several identifiers all called "the transaction number", and reconciliation, which is unglamorous, unavoidable, and where more young financial technology companies fail than for any more interesting reason.

His conviction is that payments has no genuinely hard idea in it, only a very long chain of simple ones, most of which are documented for people who already understand them. The barrier is not difficulty. It is that the industry writes for itself. That is a solvable problem, and this book is an attempt at solving it.

Under his direction, KedByte Technologies Private Limited treats technical education as infrastructure rather than as a product feature: knowledge that should be transferable, verifiable, and free of the vocabulary barriers that keep capable people out of the field.

ABOUT THE PUBLISHER

KedByte Technologies Private Limited

KedByte Technologies Private Limited publishes technical work under the KedByte Press imprint.

The editorial standard applied to this volume is simple to state and difficult to meet. Every explanation must work twice: once for a reader who has never opened a terminal, and once for an engineer who needs the exact figure. Every claim that can be checked must be checked. Every figure that will go stale must carry a date. Where the field genuinely disagrees, both positions are given rather than one being quietly chosen.

Where this book uses real evidence, it uses it unaltered. The network trace in Part F, the failed pushes in Part G and the error messages quoted throughout are reproduced exactly as they were observed. Diagnosis is taught against ambiguous real data, because that is the only kind there is.

KEDBYTE

TECHNOLOGIES PRIVATE LIMITED

COLOPHON

This first edition of *How Money Moves* runs to five volumes and fifty-four chapters, and was set entirely from plain text.

The body text is set in TeX Gyre Pagella, a digital revival of Hermann Zapf's Palatino, chosen for long-form reading. Headings, tables and the KedByte identity are set in Poppins. Code, command output and diagrams are set in DejaVu Sans Mono, and every diagram in this book is drawn in plain ASCII so that it survives being copied anywhere.

The book was composed with pandoc and typeset with XeLaTeX on A4 stock. The single accent colour used on rules, chapter numbers and section headings is KedByte navy.

Shikhar Singh

Founder & Chairman, KedByte Technologies Private Limited

First Edition • KedByte Press