# Build multi-acquirer payment orchestration

A payment-resilience and optimisation architecture using multiple acquirers with independent routing, tokenisation, fraud and reconciliation.

LAYER 1 / REQUIRED

## Primary and secondary acquirers

Create genuinely independent processing paths.

- ☐ Market overlap
- ☐ Token portability
- ☐ Commercial and risk diversity

Adyen — Enterprise acquiring
Checkout.com — Enterprise acquiring
Worldpay — Global acquiring
Nuvei — Global payments
Global Payments — Acquiring and processing
Worldline — Merchant acquiring and regional payment platforms
Elavon — Merchant acquiring and integrated payment gateways

LAYER 2 / REQUIRED

## Orchestration and vault

Route, tokenise and fail over across acquirers.

- ☐ Connector depth
- ☐ Network tokens
- ☐ Routing controls and observability

Primer — Workflows and orchestration
Gr4vy — Cloud orchestration
Spreedly — Vault and routing
IXOPAY — Enterprise orchestration
CellPoint Digital — Payment optimisation
Basis Theory — Processor-independent token vault and payment-data routing
VGS — Payment-data vaulting and tokenisation

LAYER 3 / REQUIRED

## Fraud and authentication

Apply consistent risk policy across processors.

- ☐ Data availability before routing
- ☐ 3-D Secure orchestration
- ☐ Feedback loops

Featurespace — Behavioural fraud
Feedzai — Payment fraud
SEON — Digital fraud

LAYER 4 / REQUIRED

## Reconciliation and analytics

Reconcile heterogeneous processor and settlement data.

- ☐ Canonical data model
- ☐ Fee and settlement reconciliation
- ☐ Route-performance attribution

IXOPAY — Orchestration and reconciliation
Primer — Workflow data
Form3 — Payment processing infrastructure
Payrails — Payment operations and multi-provider reconciliation
Juspay — Payment reconciliation and operational tooling

## Questions to ask providers

- ☐ Which legal entity contracts for each service and in which jurisdictions?
- ☐ Which responsibilities remain with the product operator rather than the provider?
- ☐ What are the implementation, approval, testing and migration timelines?
- ☐ How are incidents, exits, data portability and business continuity handled?
- ☐ What failure modes are actually independent across the selected providers?