

# CashLine ERP Complete Architecture Diagram Pack

Application, integration, deployment, DevOps, data architecture, and ERD

## Pack Scope

- Application Architecture Diagram
- Integration Architecture Diagram
- Deployment Architecture Diagram
- DevOps Architecture Diagram
- Data Architecture Diagram
- Detailed ERD

## Repository-Backed Baseline

- Angular shell and role-based user channels
- ASP.NET Core API host
- Layered backend with Contracts, Application, Domain, Infrastructure, and SharedKernel
- Worker-based async orchestration
- EF-backed operational relational store
- Managed document storage and release artifacts

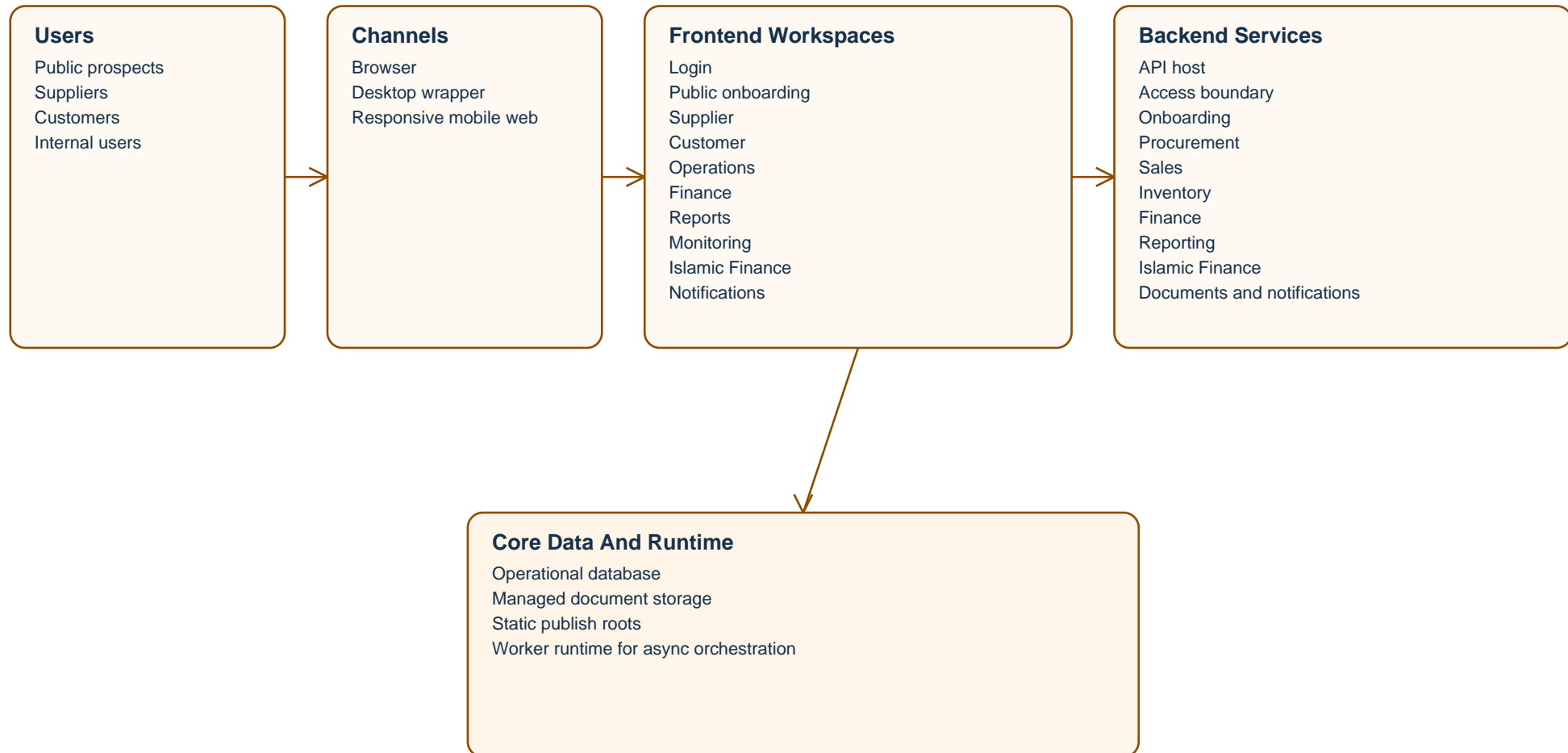
## Engineering Note

Editable Figma-friendly sources are delivered as Mermaid files under docs/architecture.

This PDF is the presentation-ready review pack; the Mermaid sources remain the redraw and refinement baseline for Figma and technical documentation updates.

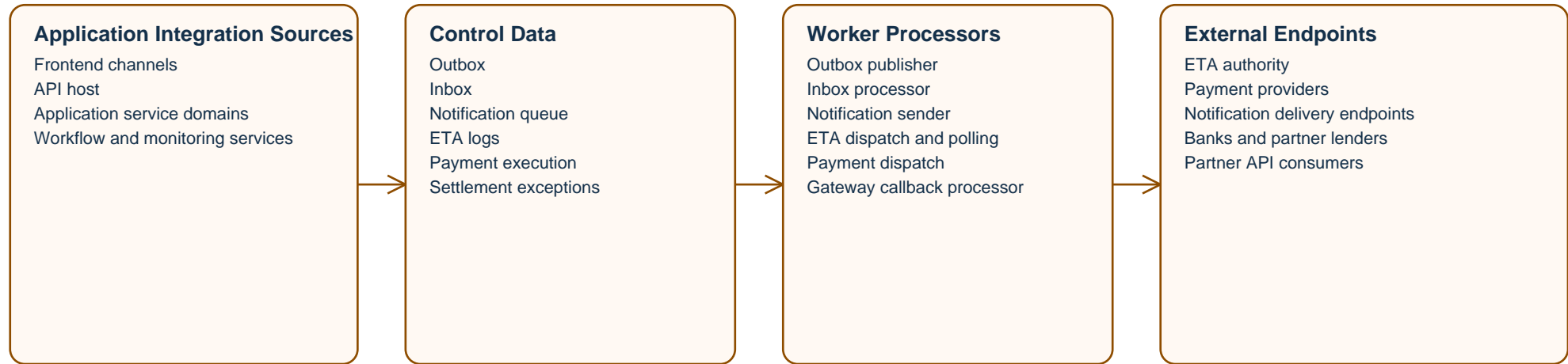
# Application Architecture Diagram

Full application landscape across channels, modules, services, and storage



# Integration Architecture Diagram

Internal and external integrations, APIs, messaging, and exchange points

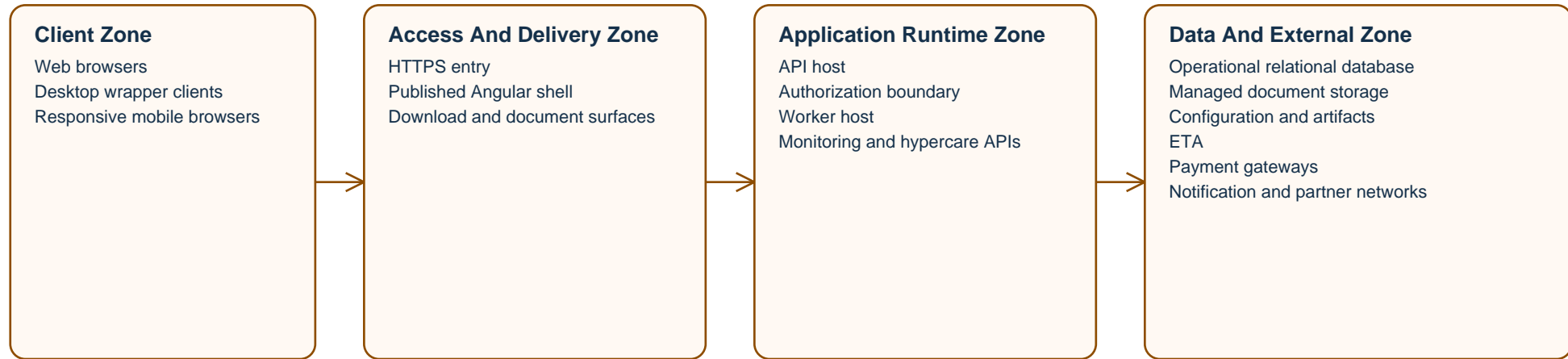


## Communication Pattern

Synchronous API calls handle direct business actions. Asynchronous worker-driven flows handle outbox publication, inbox processing, ETA events, notification delivery, payment execution dispatch, and callback-driven settlement mutation.

# Deployment Architecture Diagram

Runtime topology, environments, hosting layers, network edges, and storage dependencies

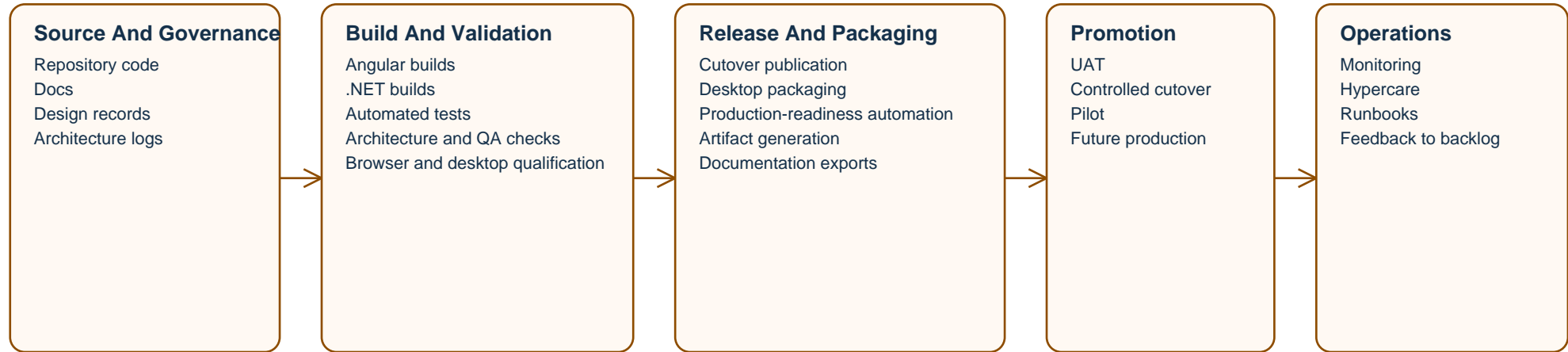


## Environment Topology

The current repo-backed posture supports UAT and controlled cutover today, with future production represented as the next promotion target once ETA and payment target-environment closure is complete.

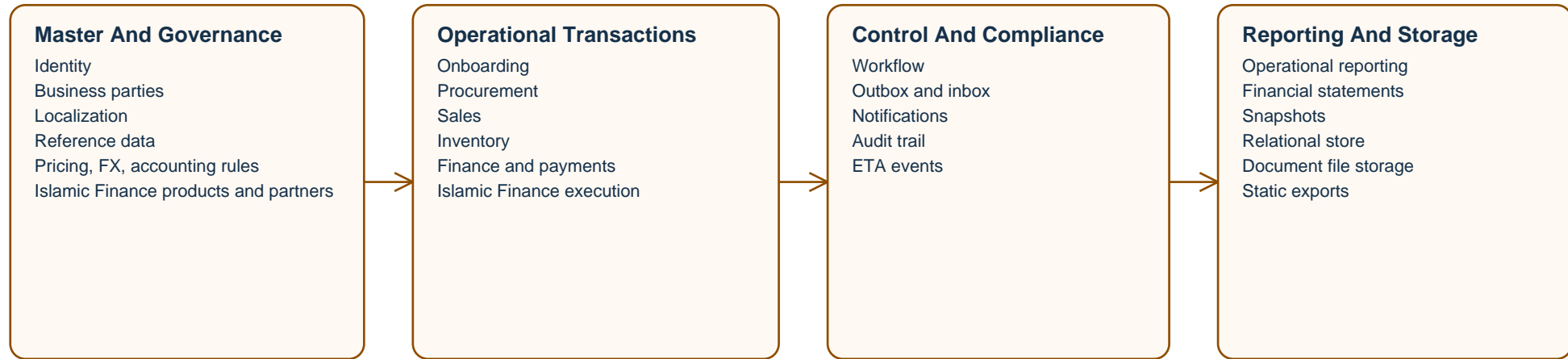
# DevOps Architecture Diagram

Source flow, validation, release, promotion, and operational support



# Data Architecture Diagram

Data domains, ownership, storage layers, and reporting movement

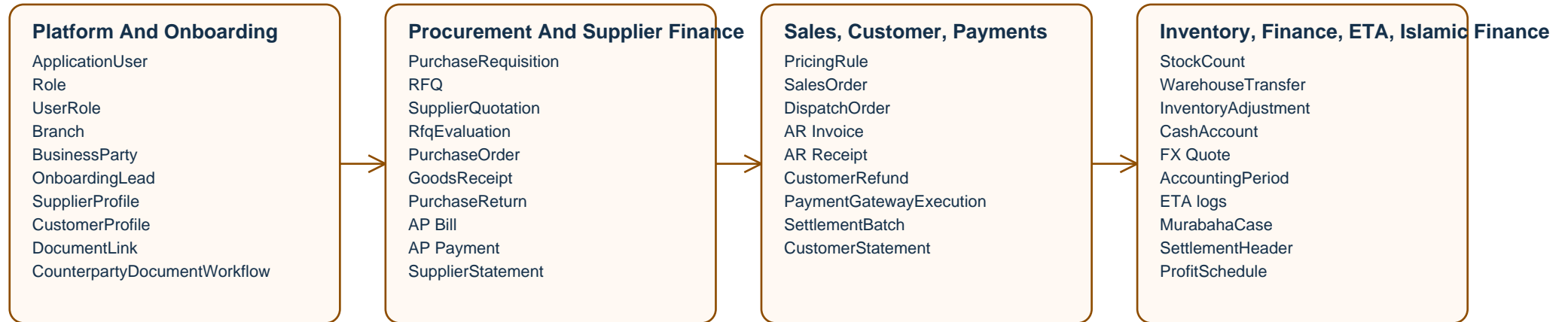


## Ownership Model

Master and governance data define platform controls. Transaction domains execute business events. Workflow, messaging, and audit data preserve platform control. Reporting consumes operational and control-state evidence without introducing a separate reporting database in the current baseline.

# Detailed ERD

Core business entities, master and transactional relationships, and major domains



## Detailed Source

The full ERD source with PK and FK-oriented relationships is provided in docs/architecture/cashline-full-erd-2026-04-12.mmd for redraw in Figma or deeper engineering refinement.