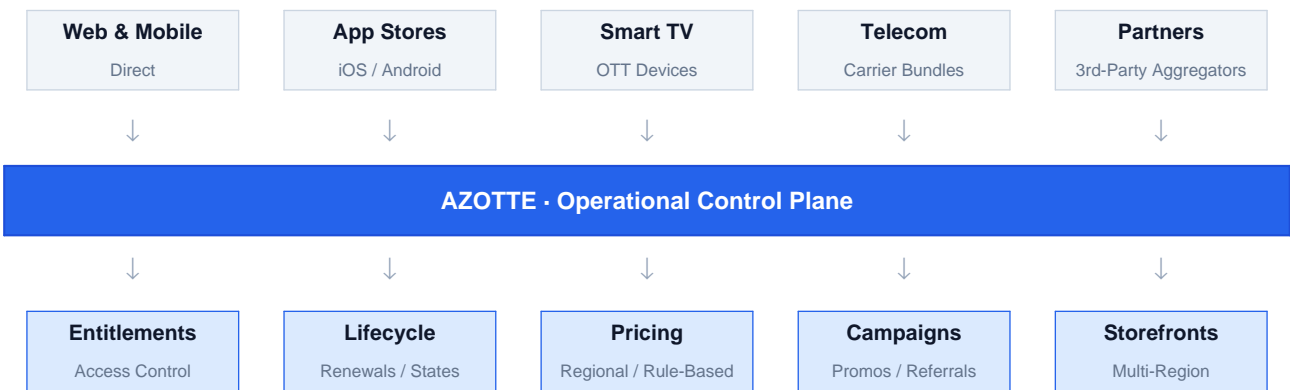


# Operational Control Plane for OTT & Video Streaming.

As OTT businesses scale across regions, app stores, device platforms, and channel partners, subscription operations fragment. Lifecycle logic diverges. Entitlements drift. Operational overhead compounds.

Azotte is the orchestration layer that centralizes subscription truth across every channel, region, and partner relationship.



## Operational Fragmentation at Scale

- Entitlement Drift** When app stores, web, and telecom channels each manage access independently, entitlement state becomes inconsistent. Support cost rises. User trust erodes.
- Lifecycle Divergence** Renewal logic, trial behavior, and upgrade paths that differ per channel create operational debt that compounds with every new integration.
- Pricing Logic Duplication** Regional pricing, promotional rules, and bundle configurations maintained separately per channel introduce error risk and block fast iteration.
- Disconnected Promotions** Promotion codes and referral programs that operate per-channel cannot deliver consistent subscriber experiences or unified campaign performance data.

**These are not billing problems. They are operational architecture problems.**

# Orchestration, Not Billing

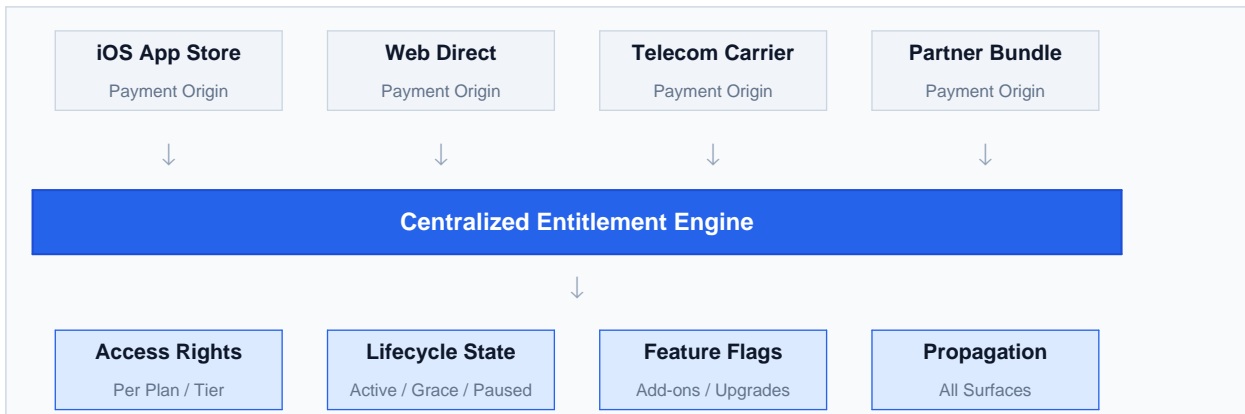
Azotte separates the operational layer from the payment layer. Payment origin does not define lifecycle logic. Channel of acquisition does not determine entitlement behavior. The orchestration layer holds the truth.

## ENTITLEMENT-FIRST ARCHITECTURE

### Subscriptions deliver entitlements. Azotte manages both.

In most platforms, the subscription record is the source of truth. In Azotte, entitlements are. Access rights are derived from a centralized entitlement model, not inferred from payment records across disconnected channels.

This separation means entitlement state remains consistent whether a subscriber arrived via the iOS App Store, a telecom bundle, a web checkout, or a partner aggregator. Upgrades, pauses, grace periods, and cancellations propagate through one policy engine.



## LIFECYCLE NORMALIZATION

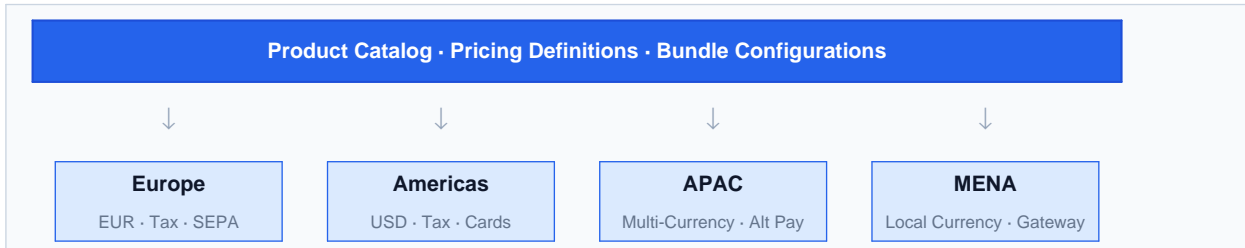
### Channel-agnostic lifecycle execution.

Each channel — app store, web, telecom, partner — has its own billing mechanics. Azotte abstracts these differences behind a normalized lifecycle model. Trial expiry, renewal retry, grace period, cancellation, and reactivation behave consistently regardless of payment origin.

<p><b>Renewal Orchestration</b></p> <p>Retry logic, grace periods, and dunning workflows operate from a single policy definition applied across all channels.</p>	<p><b>Lifecycle State Management</b></p> <p>Pause, hold, and reactivation states are managed centrally. No per-channel implementation divergence.</p>
<p><b>Upgrade &amp; Downgrade Paths</b></p> <p>Plan transitions follow a consistent policy. Proration, entitlement changes, and billing adjustments are coordinated automatically.</p>	<p><b>Family &amp; Group Orchestration</b></p> <p>Seat-based plans, family sharing, and group entitlements are managed as first-class lifecycle objects.</p>

## One Catalog. Regional Operational Independence.

Azotte operates a multi-region storefront model where a single product catalog supports independent regional configurations. Each storefront manages its own pricing, currency, tax logic, payment gateway routing, and promotional strategy without duplicating catalog definitions.



## Centralized Marketing Logic. Consistent Execution.

Promotional campaigns, discount rules, referral programs, and lifecycle-triggered offers are defined once in Azotte and executed consistently across all channels. A promotion code issued for a web campaign applies with identical rules on a smart TV app, a telecom bundle portal, or a partner storefront.

<p><b>■ Universal Promotion Engine</b></p> <p>One promotion definition applies across web, app stores, telecom portals, and partner surfaces. Consistent eligibility, redemption limits, and reporting.</p>	<p><b>■ Two-Sided Referral Orchestration</b></p> <p>Referral reward logic for both sender and recipient is managed centrally. Configurable per plan, region, and channel without engineering intervention.</p>
<p><b>■ Rule-Based Lifecycle Campaigns</b></p> <p>Lifecycle events trigger automated campaign actions. Trial expiry, renewal failure, or downgrade can initiate targeted retention offers based on subscriber context.</p>	<p><b>■ Elastic Scaling</b></p> <p>The orchestration layer handles subscriber volume spikes during live events, premieres, or viral campaigns without lifecycle divergence or entitlement inconsistency.</p>

# Why Enterprises Adopt Azotte

Subscription businesses that outgrow fragmented billing stacks do not need a better billing system. They need an orchestration layer. The outcomes below are architectural consequences of operating from a single control plane.

## 01 Eliminated Entitlement Drift

Consistent access state across every channel and device. No reconciliation between app store, web, and partner records.

## 02 Reduced Lifecycle Engineering Overhead

Renewal, retry, grace, and cancellation logic is defined once. New channel integrations inherit existing behavior without re-implementation.

## 03 Faster Regional Expansion

A new regional storefront inherits the product catalog and lifecycle model. Regional operational config — pricing, tax, gateway, language — is layered on top without duplicating core logic.

## 04 Unified Promotional Governance

Promotion codes, referral programs, and campaign rules operate from one definition. Marketing teams govern campaigns without engineering tickets per channel.

## 05 Telecom & Partner Orchestration

Carrier billing systems and third-party aggregators connect through normalized integration contracts. Subscribers acquired via partners enter the same lifecycle model.

## 06 Operational Consistency at Spike Scale

Cloud-native architecture absorbs subscriber volume spikes during live events or campaigns. The orchestration model does not diverge under load.

**Design once.**

**Operate everywhere.**

Azotte is not a billing system upgrade. It is the subscription orchestration infrastructure required when operational complexity exceeds what fragmented stacks can govern.

We work with OTT operators to assess subscription operational complexity and define the right orchestration architecture.

■ [azotte.com](https://azotte.com)

■ [Info@azotte.com](mailto:Info@azotte.com)

[Schedule a Technical Review](#)