

Post-Trade Autopilot

Automated settlement readiness, lifecycle coverage, and exception management for structured instruments - first version in 6-7 months.

Prepared for: Fayssal | Author: Kaouter Karboub | Date: January 09, 2026

Executive overview

Post-trade work for bonds/notes/convertibles and OTC derivatives (CDS/IRS) is still far more manual than it should be. Teams reconcile the same trade across systems, chase missing fields, manage lifecycle events, and spend a lot of time resolving settlement “breaks”. The result is avoidable cost, delays, operational risk, and weak auditability.

Post-Trade Autopilot can be delivered as software in the form of a web-based Ops Control Tower, a desktop application, and/or a mobile companion app - backed by an API layer. It ingests trade/confirmation data and lifecycle events, normalizes them into a single “golden record”, detects breaks early, and guides (or automates) resolution with a complete audit trail.

To keep the first version affordable and realistically deliverable by one person, the plan is to start with one corridor (one product slice) and cap integrations. With that scope discipline, a first version is realistic in ~6-7 months.

Why now

This is the kind of problem buyers will pay for because it hits daily operations and risk.

- **Breaks are predictable:** missing/incorrect SSI, mismatched economics, inconsistent schedules, incomplete trade attributes.
- **Lifecycle events are hard:** coupons, resets, credit events, corporate actions, conversions - all require correct rules and approvals.
- **Manual exception handling is expensive:** ops time goes to chasing and re-keying instead of analysis and control.
- **Auditability matters:** errors become bigger problems when you need evidence and traceability.

What we are building

A unified operational layer that turns post-trade chaos into a controlled pipeline:

Ingest → Normalize → Validate → Detect breaks → Resolve → Export settlement-ready outputs + audit trail

Core pieces in V1:

- **Golden record:** canonical trade + lifecycle record for the chosen corridor.
- **Validation + break detection:** rules to flag common break patterns with clear explanations (“why is this flagged?”).
- **Ops Control Tower (web/desktop/mobile):** pipeline view, breaks queue, drill-down, suggested resolution steps, and approvals for sensitive changes.
- **Audit trail:** every change traced to a rule, user action, or upstream source.
- **APIs + connectors:** ingestion endpoints and export endpoints for downstream systems.

First version scope (the affordable MVP)

To deliver within ~6-7 months, V1 is intentionally narrow and measurable.

- **Choose one corridor for V1:** (A) Derivatives-first (IRS + CDS) focused on confirmations/lifecycle/reconciliation, or (B) Structured securities-first (bonds/notes/convertibles) focused on term-sheet ingestion → schedules → settlement readiness.
- **Integration cap:** maximum 1-2 integrations in V1 (one upstream source + one downstream export/format).
- **Included:** ingestion, golden record, validation rules, breaks queue, suggested fixes + approvals, audit log, and a basic KPI dashboard.
- **Excluded (for V1):** full multi-asset coverage, many custodians/CSD/CCP integrations at once, complex multi-market corporate actions, and enterprise billing/multi-tenant features unless required for the pilot.

How it works (high level)

The goal is to make every trade/event follow a reliable, observable path from ingestion to resolution.

- Ingest trades/confirmations and lifecycle events (file drop, API, or agreed feed).
- Normalize into a canonical schema (consistent identifiers, dates, terms).
- Validate required fields, schedules, and basic economic consistency.
- Detect breaks using rules + simple heuristics (and log the reason).
- Resolve via guided workflows and approvals; record who did what and why.
- Export settlement-ready outputs, reconciliation reports, and audit evidence.

Technical principles (so it stays reliable)

- Idempotent ingestion + safe retries so reprocessing doesn't duplicate or corrupt state.
- Clear rule catalog so ops teams trust the system and can tune it over time.
- Observability (logs/metrics/alerts) so you can operate it like a real product.

Delivery plan (solo build, 6-7 months)

I'll run this as a weekly-demo build: small increments, visible progress, and early feedback.

- Weeks 1-3 - Discovery + spec lock: confirm the V1 corridor, define KPIs, map inputs/outputs, decide the 1-2 integrations, and produce wireframes + backlog with acceptance criteria.
- Weeks 4-16 - MVP build: canonical schema + ingestion pipeline, validation rules + break detection, Control Tower UI (pipeline + breaks queue + case view), and audit trail.
- Weeks 17-28 (or up to Week 30) - Integration + hardening: build the agreed integrations, add monitoring/error handling/retries, RBAC basics, pilot runbooks, and weekly KPI reporting.

Business potential and monetization

This can be sold as a product (or used as a wedge for client engagements) with recurring revenue.

- **Who buys:** buy-side and sell-side ops teams, custodians/fund administrators, and fintech vendors selling post-trade tooling.
- **Revenue model:** annual subscription (tiered by volume/users/entities) + implementation/integration fees + usage-based add-ons (per processed event or per case) + premium SLA/support.
- **Expansion path:** add instruments (from one corridor to multi-asset), add connectors, then add enterprise controls (SSO, reporting packs, multi-tenant support).

What I need from you next

- Pick the V1 corridor (derivatives-first vs structured securities-first).
- Identify the pilot client profile (or a potential client contact to validate with).
- Confirm the first 1-2 systems or formats we will integrate (source + export).
- Agree on 2-3 KPIs and acceptance criteria for the “first version” (so we can measure success quickly).