

ChorusControl — Enterprise Agent Control Plane Specification

Vendor: Insight IT Solutions LLC (insightits.com)

Version: 1.0.0 Enterprise Release

Date: July 2026

1. Executive Summary & System Architecture

ChorusControl is the unified enterprise management, observability, and administration control plane for the **ChorusGraph** and **Prism Pack** ecosystem (PrismGuard, PrismShine, PrismCortex, PrismRAG, PrismLib, CHORUS Fabric).

The platform solves four critical operational gaps in multi-agent enterprise deployment:

1. **Opaque Multi-Agent Failures:** Provides real-time execution tracing and zero-token offline replay debugging.
2. **Silent RAG Decay & Category Bleed:** Provides visual taxonomy mapping, vector density heatmaps, and keyword projections.
3. **Unverifiable Agent Memory:** Offers bitemporal fact inspection, manual sleep-cycle triggers, and conflict resolution.
4. **Probabilistic Security Risks:** Delivers deterministic WAF gate logging and zero-false-positive prompt injection controls.

Two-Sided Deployment Model

- **Side 1 (insightits.com Control Portal):** Customer billing (Stripe integration), support ticketing system, and Ed25519-signed offline JWT license key issuing.
- **Side 2 (Self-Hosted ChorusControl Container):** Runs inside the customer's VPC/On-Prem environment alongside ChorusGraph workers. Performs 100% offline license validation, serves the administrative web UI, and connects to the fleet via PrismAPI / CHORUS Fabric.

2. Architectural Solutions to Core Technical Gaps

Gap 1 Solved: Cross-Fleet State Invalidation Sync

When an administrator edits a document chunk in PrismRAG or resolves a memory conflict in PrismCortex via ChorusControl, ChorusControl broadcasts a high-frequency `INVALIDATE_CACHE` signal across all fleet worker nodes over PrismAPI / CHORUS Fabric.

Worker nodes receive this tensor frame and execute instant local eviction (`cache.invalidate_tags()` / `cache.invalidate_where()`) without requiring application restarts.

Gap 2 Solved: Mutex Locking & Asynchronous Background Operations

Heavy maintenance operations—such as PrismCortex / PrismResonance background sleep cycles (Passes 1–4: temporal decay, synaptic alignment, centroid recompute, group merge) or large taxonomy re-indexes—are executed via a non-blocking asynchronous queue (asyncio.Lock + thread executor). This prevents database lock contention and ensures incoming worker digest() and recall() requests experience zero latency spikes.

Gap 3 Solved: Cryptographic Administrative Audit Trail

Every administrative write operation (RAG taxonomy updates, memory conflict resolutions, or security gate policy changes) generates an immutable, cryptographically signed audit envelope (using Ed25519 signatures). Audit records are appended locally to choruscontrol_audit.jsonl and streamed to persistent storage (e.g., PostgreSQL / pgvector).

3. UI & Operational Tab Specification

Tab Name	Key Features & Capabilities
1. System Overview (/overview)	5-layer health matrix (L0–L5), real-time token tax savings counter (PrismCache & PrismLang token reduction), PrismDriver sub-ms read latency gauge, and dogfooding status indicators.
2. Execution Trace (/trace)	Live wire trace (PrismGuard → ChorusGraph → PrismShine), step-by-step Route Ledger inspector with resolution gates, and zero-token offline replay simulation button.
3. RAG & Taxonomy (/taxonomy)	64-d keyword/entity vector projection search, PrismRAG category tree visualizer, chunk health & decay map, and soft-path re-index trigger.
4. Cortex Memory (/memory)	Bitemporal knowledge graph inspector (active vs. superseded facts), interactive background sleep cycle trigger button (PrismResonance Passes 1–4), and conflict resolution console.
5. Security WAF (/guard)	PrismGuard prompt injection logs with exact resolution_gate outputs, shadow mode comparison (web_chat vs. light ONNX), and tenant lexicon / domain overlay editor.
6. Admin & License (/admin)	Ed25519 offline license status and key uploader, tenant isolation matrix, cryptographic audit trail console with SOC2/HIPAA export, user RBAC management, and embedded link

Tab Name	Key Features & Capabilities
	to insightits.com support ticketing.

4. Licensing & Authentication Protocol

The self-hosted container reads CHORUSCONTROL_LICENSE_KEY from environment variables on boot. It validates the cryptographic signature using the embedded public key from chorusmesh/license.py entirely offline, enforcing node limits, expiration dates, and tier capabilities with zero external phone-home calls.