XRAY SOVEREIGN OS (XOS V1.0)

tasks.xray.tech · /afk Fleet Plugin · Neon pgvector

XRAY Decision, Verification, Capacity & Agent Fleet User's Guide

How XRAY turns meeting debriefs, repository commits, and multi-agent sessions into self-verifying tasks, one-tap executive decisions, bounded daily Flight Plans, and a privacy-guarded company Seam.

4 Modes	Tier-0 & 90%+	1–3 Moves	100% Private
SOVEREIGN COCKPIT UI	AUTO-VERIFICATION ENGINE	BOUNDED DAILY FLIGHT PLAN	UNIVERSAL PRIVACY GATE

1. WHY XOS EXISTS: FROM PASSIVE BACKLOGS TO REALITY RECONCILIATION

EXECUTIVE OVERVIEW

The Old Friction

RETIRED

- **Zombie Tasks:** Features already shipped in code or ADRs still showed up as "New" in triage decks weeks later.
- **Context Fragmentation:** Follow-up meetings repeated proposals without folding earlier speaker quotes together.
- **Unbounded Backlogs:** 100+ item lists hid weekly capacity burn and caused context-switching fatigue.
- **Invisible Non-Git Wins:** High-impact work in Claude Cowork, spreadsheets, and Airtable went unrecorded.

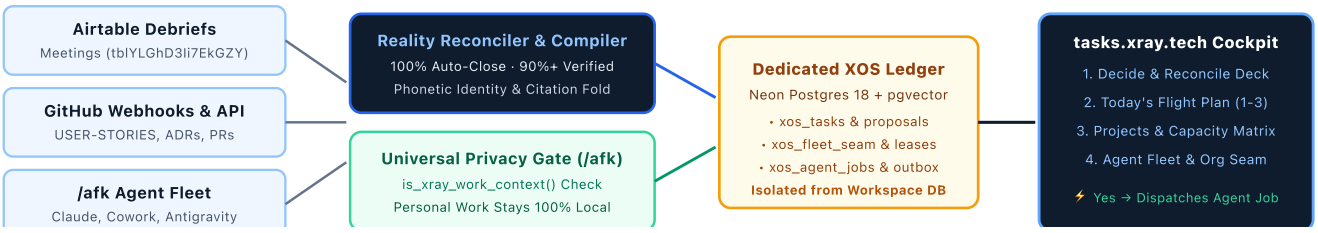
The XOS Target State

LIVE IN PRODUCTION

- **Autonomous Reality Reconciler:** Searches GitHub PRs, `USER-STORIES.md`, ADRs, and `/afk` wins before you triage.
- **Citation Folding:** Deduplicates recurring topics across Airtable Debriefs into one card with expandable speaker quotes.
- **"Yes → Agent Executes":** Approving a decision synthesizes an `ExecutionCapsule` and dispatches it to the agent fleet.
- **Company Seam via `/afk`:** Captures Wins, Decisions, Friction & Leases across repos and linked client folders alike.

2. END-TO-END ARCHITECTURE & DATA FLOW

SOVEREIGN TOPOLOGY



Web Cockpit

SSO READY

Visit `https://tasks.xray.tech` and sign in with your `@xray.tech` Google account, or open directly inside `workspace.xray.tech` via zero-prompt iframe SSO.

/afk Fleet Plugin

MCP + CLI

Installed via `XRAY-INC/afk`. Provides 7 XOS MCP tools and CLI commands (`afk link-project`, `afk seam`, `afk scratchpad`, `afk jobs`).

Commercial Isolation

ENFORCED

Backed by dedicated Neon database `xray-tasks-db`. Hard-guarded in `lib/db.js` to never touch commercial `THE-WORKSPACE-PRODUCT` data.

3. PERSISTENT TOP TELEMETRY & CAPACITY BAR

ALWAYS VISIBLE

⏱️ 18.5h / 32h Committed | 🎯 2/3 Flight Moves | ✨ 4 Verified Ready | 🤖 3 Active Streams

🔍 RECONCILE SIGNALS

The top bar anchors every view in `tasks.xray.tech`. It displays your real-time **Weekly Committed Hours vs. 32h Target**, your active **Pacing Mode** (`⚡ Protection` caps Flight Plan to 2 atomic moves, `🟢 Sustainable` locks 3 moves, `🔥 High Output` allows up to 4 moves), your ✨ **Verified Ready** count, and live 🤖 **Fleet Leases & Streams**. Clicking 🔍 **Reconcile Signals** triggers an immediate cross-system verification pass.

4. MODE 1 — THE 3-LANE DECIDE & RECONCILE DECK (#INBOX)

RAPID TRIAGE

Lane 1: ✨ Verified

≥ 90% MATCH

Already Done in Reality:

- **100% Match (Tier 0):** Commits/PRs with `Fixes recXXXX` or `afk_complete_task` auto-close silently.
- **≥ 90% Match:** Tasks matching shipped stories (e.g. `US-AD-68` / `ADR-038`) appear here with an **Evidence Bundle**.
- Click **Clear All Verified** to close the entire lane in one tap.

Lane 2: ⚡ Proposals

4-OPTION DECIDE

Citation-Folded Decisions:

- Recurring topics across Airtable Debriefs merge into one card with expandable **Speaker Quotes**.
- **Accept:** Applies change or creates task.
- ⚡ **Yes — Agent Executes:** Creates task + dispatches `ExecutionCapsule` to the agent fleet queue.
- **Edit First / Reject:** Refine or dismiss with reason.

Lane 3: 🎯 Deal-a-Move

LOCK TO TODAY

Bounded Daily Selection:

- Surfaces your highest-priority unblocked tasks sorted by urgency and project commitment.
- Click 🎯 **Lock to Flight Plan** on 1–3 cards to commit your focus for today without scrolling a 100-item backlog.

5. MODES 2, 3 & 4 — FLIGHT PLAN, CAPACITY MATRIX & AGENT FLEET

EXECUTION & VISIBILITY

Mode 2: Today's Flight Plan Workbench

1–3 LOCKED MOVES

- Displays only your locked moves for today (bounded by Pacing Mode: 2 in Protection, 3 in Sustainable, 4 in High Output).
- Each move includes a pre-compiled **Execution Capsule** (target repo, files to touch, test command, and copyable prompt).
- Checking off a move updates both your Flight Plan counter and the underlying Airtable task status.

Mode 3: Projects & Capacity Matrix (#projects)

ALLOCATION VS. BURN

- Replaces flat task lists with an executive capacity matrix across all active XRAY Projects.
- Shows **Weekly Committed Hours**, **Burned Hours**, **Utilization %**, and linked **Outcomes & Milestones** count.
- Highlights over-allocated projects (>100% utilization) before deadlines slip.

Mode 4: Agent Fleet & Company Seam (#fleet)

REAL-TIME ORG STREAM

1. Live Org Seam Feed

Filterable company stream of `🏆 Win`, `🧭 Decision`, `⚠️ Friction`, and `🤝 Handoff` events logged by teammates and agents.

2. Active Scope Leases

Displays live file/glob reservations (e.g. `src/auth/*` on `THE-WORKSPACE-PRODUCT`) with TTL countdowns to prevent merge collisions.

3. Shared Scratchpad

Cross-session notes keyed by topic and project—ideal for non-Git workflows, spreadsheet audits, and multi-agent handoffs.

4. Dispatched Agent Jobs

Tracks queued, claimed, and completed `ExecutionCapsule` jobs triggered via ⚡ **Yes — Agent Executes**.

6. HOW EVERY ROLE USES /AFK (GIT & NON-GIT WORKFLOWS)

ORG-WIDE ADOPTION

Not everyone works inside a GitHub repository. Consultants, ops leads, and analysts frequently work in local client folders, spreadsheets, Airtable bases, or Claude Desktop/Cowork chats. XOS is baked directly into the `/afk` plugin so **every teammate and agent** shares the same Seam, Scratchpad, and Task Verification loop.

📊 Chats, Spreadsheets & Ops

NO GIT NEEDED

Example (Kate / Ops / Enablement):

- Working in a local client folder or analyzing Airtable exports in Claude.
- Ask Claude: *"Link this folder to XRAY project Executive Enablement for client Acme Corp."*
- Claude calls `afk_link_project` (writes `.xray-project.json`).
- When analysis finishes, Claude calls `afk_log_seam(tag="win", ...)` or `afk_complete_task` — instantly visible on `tasks.xray.tech#fleet`!

🐝 Multi-Agent Swarms

SHARED STATE

Example (Doc / Rachael / Research):

- Running parallel research agents or switching between Claude Desktop and Claude Code.
- Agent A writes findings to the shared scratchpad: `afk_scratchpad_write(topic="Q4 Pricing", body="...")`.
- Agent B (or another teammate) reads it immediately via `afk_scratchpad_read(topic="Q4 Pricing")`.

💻 Software Engineering

ZERO CONFIG

Example (Mark / Tom / Engineering):

- Any repo cloned from `XRayWorkflow/*` or `XRAY-INC/*` is recognized automatically—no setup file required.
- Claiming a file scope via `afk claim "src/lib/*"` automatically broadcasts a cross-developer lease to `tasks.xray.tech`.
- GitHub pushes & PRs stream directly to the Reality Reconciler via webhooks.

7. COPY-PASTE PLAYBOOKS FOR COMMON WORKFLOWS

STEP-BY-STEP

Playbook A: Link Any Non-Git Folder & Log a Win

CLI OR MCP

Run once inside any client directory (or ask your AI agent to call `afk_link_project`):

```
# 1. Link directory to XRAY Project / Client / Task
afk link-project \
  --project "Executive Enablement" \
  --client "Acme Corp" \
  --task-id "recAcmeDeck123"

# 2. Log a milestone win (auto-reconciles linked task!)
afk seam win "Completed Acme Q3 enablement run sheet" \
  --task-id "recAcmeDeck123"
```

Playbook B: Claim & Execute a Dispatched Decision

YES → AGENT EXECUTES

After clicking ⚡ **Yes — Agent Executes** on a proposal in `tasks.xray.tech`:

```
# 1. Claim the next queued ExecutionCapsule job
afk jobs claim --agent claude-code

# 2. Agent executes the capsule prompt & verificationCmd
# 3. Mark job complete (auto-closes the Airtable task!)
afk jobs complete job_8f92a1c0 \
  --summary "Shipped holding-pen redirect & passing tests" \
  --commit-sha "203b493"
```

Playbook C: Cross-Developer Scope Reservations (Preventing Agent Collisions)

LAYER 2 CONCURRENCY

When two teammates (or two subagents) work on the same repository from their own laptops, `afk claim` normalizes the GitHub repository slug (e.g. `XRAY-INC/THE-WORKSPACE-PRODUCT`) and checks `tasks.xray.tech` for wildcard glob overlaps (`src/*` vs. `src/auth/login.ts`). If another developer's agent holds an overlapping lease, `tasks.xray.tech` returns `409 Conflict` with the holder's name and remaining TTL.

```
# Reserve glob before multi-file edits (syncs to XOS Fleet)
afk claim "src/provisioning/*" --agent claude --ttl 1800

# Release all leases when done
afk release --all --agent claude
```

8. THE UNIVERSAL PRIVACY RULE (IS_XRAY_WORK_CONTEXT) — ZERO SPECIAL CASES

PRIVACY FIREWALL

One Identical Rule on Every Machine: There is zero machine-specific code in `/afk`. Whether you are on Mark's MacBook, Kate's laptop, or a cloud runner, `/afk` evaluates the exact same 3-signal check before sending a single byte to `tasks.xray.tech`.

✅ Syncs to tasks.xray.tech (XRAY Work Context)

SYNCED

- **Signal 1 (Explicit Tag):** Command/tool call passes an explicit XRAY `project`, `client`, or `task_id` (`rec...`).
- **Signal 2 (Linked Folder):** Current directory (or a parent) contains `.xray-project.json` created via `afk link-project`.
- **Signal 3 (XRAY Git Remote):** `git remote.origin.url` belongs to `github.com/(XRayWorkflow|XRAY-INC)/*`.

🔒 Stays 100% Local (Personal / Unlinked Context)

LOCAL ONLY

- **Unlinked Directories & Personal Repos:** If none of the 3 signals above match, `is_xray_work_context()` returns `False`.
- **Zero Outbound Network Calls:** Seam events, scratchpad notes, and file leases stay strictly in your local `~/.afk/` ledger (and personal `BLACKBOARD_URL` if configured).

9. COMPLETE /AFK CLI & MCP TOOL REFERENCE

CHEAT SHEET

MCP TOOL (CLAUDE / AGY)	CLI COMMAND EQUIVALENT	PURPOSE & BEHAVIOR
`afk_link_project`	`afk link-project --project P --client C`	Writes `.xray-project.json` so non-Git folders & chats sync to XOS.
`afk_log_seam`	`afk seam <win|decision|friction> "msg"`	Logs high-signal milestone to local ledger + XOS Fleet Seam (if XRAY work).
`afk_complete_task`	`afk seam win "msg" --task-id recXXXX`	Proves task completion from an agent session; triggers Tier-0 auto-close.
`afk_scratchpad_write`	`afk scratchpad write --topic T --body B`	Publishes shared cross-session / cross-agent context note.
`afk_scratchpad_read`	`afk scratchpad read [--project P]`	Reads shared scratchpad notes for the active project or topic.
`afk_claim_job`	`afk jobs claim [--agent claude]`	Claims a queued `ExecutionCapsule` dispatched via "Yes → Agent Executes".
`afk_complete_job`	`afk jobs complete <id> --summary "Done"`	Completes claimed job, logs an Org Seam Win, and closes the linked task.

10. KEY XOS REST ENDPOINTS

TASKS.XRAY.TECH

ENDPOINT	DESCRIPTION
`GET /api/v1/cockpit/state`	4-Mode telemetry & capacity
`POST /api/v1/reconciler/scan`	Scan tasks against repo signals
`POST /api/v1/reconciler/clear-verified`	1-Tap batch clear (≥90% match)
`POST /api/v1/flight-plan/lock`	Lock move to daily Flight Plan
`POST /api/v1/fleet/seam`	Ingest /afk Wins & Decisions
`POST /api/v1/fleet/leases`	Claim cross-dev glob lease
`POST /api/webhooks/github`	HMAC-verified GitHub webhook

11. THE 5-MINUTE DAILY XOS RITUAL

BEST PRACTICE

Every Morning at tasks.xray.tech

5 MINS

- **Step 1 — Clear Verified (30s):** Check Lane 1 (`✨ Verified`). Review evidence badges from overnight PRs/agent runs and tap **Clear All Verified**.
- **Step 2 — Decide Proposals (2m):** Sweep Lane 2 (`⚡ Proposals`). Expand speaker quotes if needed; tap **Accept**, ⚡ **Yes — Agent Executes**, or **Reject**.
- **Step 3 — Lock 3 Flight Moves (1.5m):** Set your Pacing Mode in the top bar and lock up to 3 moves from `🎯 Deal-a-Move` into Today's Flight Plan.
- **Step 4 — Execute with /afk (1m):** Copy the Execution Capsule or run `afk jobs claim`; let `/afk` auto-log wins and close tasks as you ship.