

ArchitectureRoadmap.md

Version: 2.1.0 **Date:** 2025-12-13 **Status:** Phase 3 Complete / Phase 4 (Analytics Refactor) Active

1. The Strategic Vision

Goal: Create a single analysis engine that powers both a Command-Line Interface (CLI) and a Web Interface (Django). **Core Constraint:** "Write Once, Render Everywhere." Logic must not be duplicated. **Deployment Model:** Stateless & Portable.

- No User Accounts. No User Database. Atomic Sessions.
- Containerized (Docker) to ensure "Laptop == Server".

2. The "Golden Path" Workflow

This defines the specific user experience and data flow for the Web Interface.

1. **The Three-Slot Model:** The analyzer presents three generic input slots (Log A, Log B, Log C).
2. **Identity Agnostic:** There is no concept of "My Log" vs. "Competitor."
3. **Source Agnostic:** Each slot can be filled by either:
 - **Direct Upload:** User uploads a local .log file.
 - **Public Fetch:** User selects a contest/year/callsign (Phase 4.2).
4. **Session Persistence:**
 - Files exist in `media/sessions/<session_key>/` for the duration of the analysis session.
 - Lazy Cleanup ensures old sessions are purged automatically.

3. Architectural Decision Records (ADRs)

ADR-001: Data Abstraction Layer (DAL)

- **Decision:** All business logic exists in `data_aggregators/`. Returns Pure Python Primitives.

ADR-002: Unified Visualization (Client-Side Rendering)

- **Decision:** Replace Matplotlib with Plotly.
- **Animation:** Replace server-side MP4 generation with Plotly HTML Animations.

ADR-007: Shared Presentation Layer

- **Decision:** The Web App (Django) must point to the existing `contest_tools/templates`.

- **Reason:** Ensures CLI HTML reports and Web Views are bit-for-bit identical.

ADR-009: Session-Scope Persistence

- **Decision:** Replace "Ephemeral I/O" (tempfile) with Session-Scope Storage (`media/sessions/`).
 - **Reason:** Required to support drill-down navigation and static asset serving (images/HTML) in the browser.
-

4. Master Transition Timeline

Phase 1: Data Decoupling (COMPLETE)

- **Status:** Complete. DAL is operational.

Phase 2: Visualization Standardization (COMPLETE)

- **Status:** Complete. Static charts migrated to Plotly.

Phase 2.5: Animation Modernization (COMPLETE)

- **Status:** Complete. `plot_interactive_animation.py` implemented.

Phase 3: The Web Pathfinder (COMPLETE)

- **Status:** Complete.
- **Tasks:**
 - Containerization:** Create `Dockerfile` and `docker-compose.yml`.
 - Django Bootstrap:** Initialize the project skeleton.
 - UI Implementation:** Three-Slot Upload Form & Basic Dashboard.
 - Validation:** "Pathfinder Run" successful with CQ WW logs.

Phase 4: The Strategy Board & Analytics Refactor (ACTIVE)

- **Status:** In Progress.
- **Goal:** Implement the "Top-Down" UI strategy ("The Arena", "The War Room").
- **Tasks:**
 - Step 1: The Strategy Board:** Upgrade Dashboard table with "Run %" metrics.
 - Step 2: The Drill-Down UI:** Implement the "Triptych" (Animation/Plots/Mults) using Session Persistence.
 - Step 3: Sub-Page Views:** Create generic viewer views for the drill-down reports.
 - Step 4: Public Log Fetcher:** (Deferred to Phase 4.1).