Architecture Roadmap

Version: 2.1.0 **Date:** 2025-12-18 **Status:** Phase 3 (Web Dashboard) - Active Sprint: "The Solo Audit"

Phase 1: Core Analytics Engine (Completed)

- ☒ **Log Ingest:** Cabrillo parser with contest-specific definitions (`.json`).
- ☒ **Heuristics:** Run/S&P/Unknown classification (`run_s_p.py`).
- ☒ **Aggregation:** Data Abstraction Layer (DAL) for TimeSeries, Matrix, and Multipliers.
- ☒ **Scoring:** Modular scoring engines for CQ WW, WAE (QTCs), NAQP, etc.

Phase 2: Visualization Engine (Completed)

- ☒ **Engine Migration:** Replaced Matplotlib with Plotly for interactive charts.
- ☒ **Styling:** Centralized `PlotlyStyleManager`.
- ☒ **Output:** Dual-stack generation (Static PNG + Interactive HTML).

Phase 3: The Web Dashboard (Active)

Goal: A stateless, containerized web interface ("The Strategy Board") for log analysis.

3.1. Infrastructure (Completed v0.102)

- ☒ Django Project Setup (Stateless).
- ☒ Docker/Docker-Compose configuration.
- ☒ Shared Template Layer (ADR-007).

3.2. Core Dashboard Features (Completed v0.115)

- ☒ **Session Management:** UUID-based workspaces (`/media/sessions/`).
- ☒ **The Scoreboard:** High-level scalars (Score, QSOs, Mults).
- ☒ **Interactive Animation:** Time-lapse playback of contest progression.
- ☒ **Sub-Dashboards:** Dedicated views for QSO Analysis and Multipliers.

3.3. Public Data Integration (Completed v0.126.0)

- ☒ **Log Fetcher:** Scraper for CQ WW Public Logs.
- ☒ **Typeahead Search:** AJAX API for competitor lookup.

3.4. "The Solo Audit" (Current Sprint - v0.126.x)

Goal: Adapt the dashboard for single-log uploads ("Self-Analysis") vs. multi-log uploads ("Competition").

- ☐ **Solo Mode Logic:** Detect `log_count == 1` in Views.
- ☐ **Correlation Analysis:** New Scatter Plot (Run % vs Rate/Mults).
- ☐ **UI Hardening:**
  - Suppress "Pairwise Strategy" tab in Solo Mode.
  - Switch "Missed Multipliers" (Red) to "Multiplier Matrix" (Blue).
- ☐ **Correlation Tab:** Integrate new report into `qso_dashboard`.

Phase 4: Future Scalability (Planned)

4.1. Modular Dashboard Architecture (Next Sprint)

Goal: Support non-CQ WW contests (WAE, Field Day) without monolithic templates.

- ☐ **Widget Pattern:** Refactor `dashboard.html` into `partials/widgets/`.
- ☐ **Dynamic Layouts:** Drive dashboard composition via `ContestDefinition`.

4.2. Advanced Metrics

- ☐ **Gap Analysis:** Report on off-times > 10 min.
- ☐ **Propagation Replay:** Map-based visualization of openings (WRTC style).