Version: 1.0.0 **Target:** 0.124.0-Beta

Implementation Plan - Phase 3: The Opportunity Matrix

This plan executes Phase 3 of the Multiplier Dashboard roadmap. It transforms the dashboard from a static file list into an interactive analytical tool by implementing the **Opportunity Matrix**.

Proposed Changes

1. Modified File: `contest_tools/data_aggregators/multiplier_stats.py`

Purpose: Extend the aggregator to provide high-level matrix metrics (Common, Differential, Missed) without the overhead of detailed callsign strings.

- **New Method:** `get_dashboard_matrix_data(self)`.
- **Logic:**
  - Iterates through all multiplier rules defined for the contest.
  - Grouping: Aggregates data by **Band** (and **Mode** if applicable).
  - Engine Integration: Delegates set logic to `ComparativeEngine`.
  - Output: Returns a structured dictionary suitable for rendering the "Opportunity Matrix" grid (Integers for Common, Diff, Missed).

2. Modified File: `web_app/analyzer/views.py`

Purpose: Update the `multiplier_dashboard` view to hydrate live data instead of just scanning text files.

- **Logic Update:**
  - Re-initializes `LogManager` and loads the logs stored in the session directory.
  - Instantiates `MultiplierStatsAggregator`.
  - Calls `get_dashboard_matrix_data()` to fetch the matrix metrics.
  - Context Update: Passes the `matrix_data` and `logs` metadata to the template.

3. Modified File: `web_app/analyzer/templates/analyzer/multiplier_dashboard.html`

Purpose: Implement the "Three-Pane" layout (Panes 1 & 2 only for this phase).

- **Pane 1 (Scoreboards):** Adds a top row of mini-scoreboards for the logs being compared.
- **Pane 2 (Opportunity Matrix):** Adds a dynamic table rendering the "Common / Diff / Missed" integers.

– **Visuals:** Uses standard Bootstrap styling. Negative "Missed" numbers are highlighted in red.
- **Legacy Support:** Retains the existing "Deep Dive" file list at the bottom for continuity.

Affected Modules Checklist

- `contest_tools/data_aggregators/multiplier_stats.py`
- `web_app/analyzer/views.py`
- `web_app/analyzer/templates/analyzer/multiplier_dashboard.html`

Pre-Flight Check

- **Inputs:** Session ID.
- **Expected Outcome:** A dashboard page showing a "Scoreboard" and an "Opportunity Matrix" grid populated with correct integers, followed by the original report links.
- **Backward Compatibility:** Existing text reports are preserved and accessible via the "Deep Dive" section.
- **Architecture Compliance:** Aggregation logic remains in `contest_tools`; View handles I/O; Template handles presentation.
- **Constraint Check:** "Unknown" multipliers are filtered by the underlying engine (Phase 1).

--- BEGIN DIFF --- (Surgical diffs not provided for full file replacements in this context to ensure clean integration of the new class structures.) --- END DIFF ---