

# **CLA Reports Style Guide**

**Version: 1.3.0 Date: 2025-12-20**

---

--- Revision History ---

[1.3.0] - 2025-12-20

Changed

- Updated Section 2.1 to include a "Redundancy Exception" for single-mode datasets.

[1.2.0] - 2025-12-20

Changed

- Renamed application to "Contest Log Analytics".

- Implemented "Three-Line Title" standard (Report, Context, Scope).

- Added "Metadata Header/Footer" standards for Text and Visual reports.

[1.1.0] - 2025-12-06

Added

- Added "The Hierarchical Aggregation Standard" to prevent misleading sums.

- Added "Explicit Scope Labeling" to clarify data subsets in comparisons.

- Added "General UX Principles" (Smart Suppression).

[1.0.0] - 2025-12-06

Changed

- Major update to support Plotly interactive charts and dashboard layouts.

- Codified the "Block Layout" and "Drill-Down" patterns for text reports.

- Split Visualization Standards into Legacy (Matplotlib) and Modern (Plotly).

- Defined distinction between File Titles and Section Headers.

## 1. Introduction

This document is the single source of truth for the visual and formatting standards of all report outputs generated by the **Contest Log Analytics (CLA)**. Its purpose is to ensure that all reports, regardless of type, have a consistent, clean, and professional appearance. All new reports must adhere to these guidelines.

### 1.1. General UX Principles

- **Signal-to-Noise Ratio:** Maximize signal, minimize noise. Reports should prioritize "Actionable Intelligence" over raw data dumps.
  - **Smart Suppression:** If a "Detail Block" provides no new information compared to its parent "Summary Block" (e.g., a Single-Mode band where Band Total == Mode Total), the Detail Block **should be suppressed** to reduce clutter.
- 

## 2. Report Headers & Titles

### 2.1. File Titles (Top of Document)

All generated reports must use the standard **Three-Line Title** format.

- **Line 1 (Report Name):** The official name (e.g., "Cumulative Difference Plots").
- **Line 2 (Context):** The event context (e.g., "2025 NAQP CW AUG - K1ABC vs W1XYZ").
- **Line 3 (Scope):** The data scope. **Mandatory.** (e.g., "All Bands", "20 Meters", "20 Meters [CW]").
  - *Exception:* The Mode label should be omitted if the dataset contains only a single mode (e.g., in a CW-only contest).

### 2.2. Branding & Metadata (Headers/Footers)

Every report must include the application signature and CTY provenance.

**Text Reports (Upper Right Header)** Aligned to the right margin of the content block:

Contest Log Analytics by KD4D  
CTY File: YYYY-MM-DD CTY-XXXX

**Visual Reports (Bottom Center Footer)** Centered below the plot area:

Contest Log Analytics by KD4D  
CTY File: YYYY-MM-DD CTY-XXXX

### 2.3. Section Headers (Internal Dividers)

For reports that contain multiple distinct blocks or tables, strict visual separation is required.

- **Separator:** A section must begin with a full-width visual separator (e.g., 80 dashes --- in text reports).
  - **Header Text:** The section title must be centered relative to the content block it precedes.
- 

## 3. Text Reports (.txt)

This section defines the standards for all plain-text reports.

### 3.1. Library Standards

The creation of text-based tables is governed by the following two-tool approach:

- **For Complex Reports (`prettytable`):** Recommended for fixed-width layouts, precise column alignment, or "stitching" multiple tables.
- **For Simple Reports (`tabulate`):** Suitable for single, standalone tables where complex alignment is not a concern.

### 3.2. The Drill-Down Pattern

To ensure scannability, text reports must follow the "Drill-Down" structure:

1. **Executive Summary:** The first block must be a high-level summary (e.g., All Bands, Total Score).
2. **Sequential Details:** Subsequent blocks should break the data down by specific segments (e.g., Band-by-Band Details), ordered logically (e.g., 160M to 10M).

### 3.3. Block Layout Standards

Unlike graphical plots, text reports must **NOT** be globally centered on the page/terminal.

- **Left Anchor:** All tables and content blocks must be anchored to the left margin (Column 0).
- **Relative Centering:** Table titles and column headers must be centered relative to the *width of the specific table*, not the global document width.
- *Correct:* Title is centered over the 60-character table below it.
- *Incorrect:* Title is centered at character 40 of an 80-character terminal while the table sits at the left margin.

### 3.4. The Hierarchical Aggregation Standard

When presenting aggregated data (e.g., Band Totals) alongside constituent parts (e.g., Mode Counts), the report **must** use a Parent/Child layout to prevent misleading mental arithmetic (e.g., "10 + 10 = 20" when the union is actually 12).

- **Structure:**

- **Parent Row:** Labeled ALL or TOTAL. Displays the **Union** (deduplicated total).
- **Child Rows:** Indented. Display strict, specific counts for that subset (e.g., CW, PH).
- **Visual Cue:** Parent rows must be visually distinguished (e.g., via a separator line ----- or all-caps label) to signal they are summaries, not addends.

### 3.5. Explicit Scope Labeling

To prevent "Visual Lies" where a user mistakes a subset statistic for a global statistic, column headers must explicitly state the **Scope** of the data if it is a subset.

- **Mandate:** Do not use generic labels like "Run" or "Total" if they refer to a filtered subset (like "Unique Run").
  - **Pattern:** Use "Super-Headers" or compound labels.
    - *Bad:* | Run | S&P | (inside a Unique table)
    - *Good:* | UNIQUE (Run) | UNIQUE (S&P) | OR | Unique Run | Unique S&P |
- 

## 4. HTML Reports (.html)

This section defines the standards for all web-based reports.

### 4.1. Styling Framework

All styling for HTML reports must be implemented using **Tailwind CSS**. No custom CSS in `<style>` blocks should be used unless absolutely necessary.

### 4.2. Layout & Table Styling

HTML reports often act as dashboards. Therefore, strict horizontal centering is **not required** if a multi-column dashboard layout is used.

- **Border Hierarchy:** A two-level border system must be used:
  - **Dark Lines** (`border-gray-500`): Main outline and major divisions.
  - **Medium Lines** (`border-gray-400`): Internal grid lines.

### 4.3. Interactive Embeds

Future reports will rely on Javascript-based interactivity.

- **Containers:** Charts must be injected into named `<div>` containers (e.g., `<div id="chart_main"></div>`).
  - **Hydration:** Scripts should target these IDs for Plotly hydration.
- 

## 5. Visualization Standards

### 5.1. Legacy Static Plots (Matplotlib)

For static image generation (PNG/PDF), Matplotlib remains the engine.

- **Layout Engine:** Use `GridSpec` for any layout more complex than a single subplot. Automatic managers like `tight_layout` are discouraged for complex reports.
- **Implementation Pattern:** Create a "Master Grid" dividing the figure into zones (Title, Plot Area, Legend), then nest sub-grids within those zones.

### 5.2. Interactive Plots (Plotly) [NEW]

All new interactive visualization features must be built using **Plotly**.

#### 5.2.1. Engine Mandates

- **Graph Objects:** Use `plotly.graph_objects` (`go.Figure`) for maximum control. Avoid `plotly.express` for production reports unless the chart is trivial.
- **Subplots:** Use `plotly.subplots.make_subplots` for multi-panel layouts.

#### 5.2.2. Styling Consistency

- **Style Manager:** All trace styling (colors, fonts, margins) must be retrieved from the project's `PlotlyStyleManager` to ensure visual consistency across the application. Hard-coding hex values in report files is prohibited.

#### 5.2.3. Export Standards

- **Static:** Use `fig.write_image("file.png")` for static snapshots.
- **Web:** Use `fig.to_json()` (or `fig.write_json()`) to export the figure definition for embedding into HTML reports.