

Contest Log Analyzer - User Guide

Version: 0.40.1-Beta Date: 2025-08-24

--- Revision History ---

[0.40.1-Beta] - 2025-08-24

Added

- **Added the --debug-mults flag to the Command-Line Options list.**
- **Added "ARRL Field Day" to the list of supported contests.**
- **Added the 'chart_point_contribution' report to the list of available reports.**

Changed

- **Updated the description for SweepstakesSections.dat to include ARRL Field Day.**

[0.40.0-Beta] - 2025-08-19

Changed

- **Updated the "Available Reports" list to be complete.**

[0.37.0-Beta] - 2025-08-18

Changed

- **Aligned version with other documentation files.**
- **Corrected the list of required data files in Section 2.**
- **Updated the Command-Line Options list in Section 3 to include**
the --debug-data flag.

[0.36.8-Beta] - 2025-08-15

Changed

- **Updated lists of required data files, CLI options, and supported**
contexts to be complete and accurate.

[0.35.25-Beta] - 2025-08-15

Changed

- **Updated the "Available Reports" list and the --report argument**
description to be consistent with the current
codebase.

[0.30.31-Beta] - 2025-08-11

Changed

- **Updated the "Available Reports" section to be**

complete and accurate

based on the current project state.

[0.30.30-Beta] - 2025-08-05

- Updated environment variable and --report argument documentation.

[0.30.0-Beta] - 2025-08-05

- Initial release of Version 0.30.0-Beta.

- Standardized all project files to a common baseline version.

1. Introduction: What is the Contest Log Analyzer?

[cite_start]The Contest Log Analyzer is a powerful command-line tool designed for amateur radio contesters who want to perform deep, data-driven analysis of their operating performance. [cite: 824] [cite_start]It goes beyond the simple score summary provided by most logging software, allowing you to: [cite: 825]

- [cite_start]Process raw Cabrillo log files into a clean, standardized format. [cite: 826]
 - [cite_start]Automatically classify every QSO as "Run," "Search & Pounce," or "Unknown" to analyze your operating strategy. [cite: 827]
 - [cite_start]Generate detailed reports and charts that compare your log against one or more others. [cite: 828]
 - [cite_start]Analyze performance on a band-by-band basis to identify strengths and weaknesses. [cite: 829]
 - [cite_start]Calculate contest-specific QSO points for supported contests. [cite: 830] [cite_start]The ultimate goal of this program is to help you understand your contest operation in minute detail, identify missed opportunities, and improve your strategy for the next event. [cite: 830]
-

2. What You Need to Get Started

Before running the analyzer, you will need a few files:

- [cite_start]**Your Cabrillo Log File(s):** These are the standard log files generated by your contest logging software (e.g., `kd4d.log`, `n0ni.log`). [cite: 831] [cite_start]You can analyze a single log or compare

multiple logs at once. [cite: 832]

- [cite_start]**Data Files:** The program requires specific data files to be placed in a central `data/` directory. [cite: 833]
 - o [cite_start]`cty.dat`: Required for all contests. [cite: 834]
 - o [cite_start]`arrl_10_mults.dat`: Required for the ARRL 10 Meter contest. [cite: 835]
 - o [cite_start]`ARRLDXmults.dat`: Required for the ARRL DX contest. [cite: 836]
 - o [cite_start]`NAQPMults.dat`: Required for NAQP and CQ 160-Meter contests. [cite: 837]
 - o `SweepstakesSections.dat`: Required for ARRL Sweepstakes and ARRL Field Day.
 - [cite_start]**An Environment Variable:** You must tell the program where to find your data files by setting an environment variable named `CONTEST_LOGS_REPORTS`. [cite: 839] [cite_start]This variable should point to the root directory that contains your `Logs`, `data`, and `reports` subdirectories. [cite: 840]
-

3. How to Run the Analyzer

[cite_start]The program is run from your command prompt or terminal using `main_cli.py`. [cite: 841]

Basic Syntax

```
python main_cli.py --report <ReportID|all|chart|text|plot|animation> <LogFile1> [<LogFile2>..
```

Command-Line Options

- [cite_start]`--report <ReportID|all|chart|text|plot|animation>`: (Required) Specifies which report to generate. [cite: 842] [cite_start]Use a specific `ReportID` (e.g., `score_report`), `all` to generate every available report, or a category keyword like `chart` to run all chart reports. [cite: 843]
- [cite_start]`<LogFile1> . . .`: (Required) One or more paths to the Cabrillo log files you want to analyze. [cite: 844]
- `--verbose`: (Optional) Enables detailed `INFO`-level status messages for debugging.
- [cite_start]`--include-dupes`: (Optional) By default, duplicate QSOs are ignored. [cite: 845] [cite_start]Use this flag to include them in all calculations. [cite: 846]
- [cite_start]`--mult-name <name>`: (Optional) For reports that analyze multipliers (like `missed_multipliers`), this specifies which multiplier to use (e.g., 'Countries', 'Zones'). [cite: 847]
- [cite_start]`--metric <qsos|points>`: (Optional) For the `cumulative_difference_plots` report, this specifies whether to compare QSO counts or Point totals. [cite: 848] Defaults to 'qsos'.
- [cite_start]`--debug-data`: (Optional) When used with a visual report (chart, plot, animation), this saves the report's source data to a `.txt` file in a `Debug/` subdirectory. [cite: 849]
- `--debug-mults`: (Optional) Save intermediate multiplier lists from text reports for debugging.

Examples

- **Generate all available reports for two logs:**

```
python main_cli.py --report all 2025/cq-160-cw/kd4d.log 2025/cq-160-cw/n0ni.log
```

- **Generate only the text reports for two logs:**

```
python main_cli.py --report text 2025/cq-160-cw/kd4d.log 2025/cq-160-cw/n0ni.log
```

- **Generate a specific report (Score Summary) for a single log:**

```
python main_cli.py --report score_report 2025/cq-160-cw/kd4d.log
```

- **Generate a Missed Multipliers report for CQ WW Zones:**

```
python main_cli.py --report missed_multipliers --mult-name Zones Logs/2024/cq-ww-cw/k3lr.log :
```

4. Supported Contests

[cite_start]The analyzer uses the `CONTEST:` field in your Cabrillo file header to automatically apply the correct rules. [cite: 850] The following contests are currently supported:

- ARRL 10 Meter
- ARRL DX (CW & SSB)
- ARRL Field Day
- ARRL Sweepstakes (CW & SSB)
- CQ 160-Meter (CW & SSB)
- CQ WPX (CW & SSB)
- CQ World Wide DX (CW & SSB)
- North American QSO Party (NAQP) (CW & SSB)

5. Available Reports

[cite_start]Use the `Report ID` with the `--report` command-line option. [cite: 851]

Animation Reports (`animations/`)

- `hourly_animation`: Hourly Rate Animation

Chart Reports (`charts/`)

- `chart_point_contribution`: Point Contribution Breakdown (Comparative)
- `chart_point_contribution_single`: Point Contribution Breakdown (Single Log)
- `qso_breakdown_chart`: QSO Breakdown by Run/S&P

Plot Reports (`plots/`)

- `band_activity_heatmap`: Band Activity Heatmap
- `comparative_band_activity`: Comparative Band Activity
- `comparative_band_activity_heatmap`: Comparative Band Activity Heatmap
- `cumulative_difference_plots`: Cumulative Difference Plot
- `point_rate_plots`: Cumulative Point Rate Plot
- `qso_rate_plots`: Cumulative QSO Rate Plot

Text Reports (`text/`)

- `comparative_continent_summary`: Comparative Continent Summary
-

- comparative_score_report: Comparative Score Report
- continent_breakdown: Continent Breakdown by Run/S&P
- continent_summary: Continent Summary
- missed_multipliers: Missed Multipliers
- multiplier_summary: Multiplier Summary
- multipliers_by_hour: Multipliers by Hour
- qso_comparison: QSO Comparison Summary
- rate_sheet: Rate Sheet (per hour)
- rate_sheet_comparison: Rate Sheet Comparison
- score_report: Score Report
- summary: QSO Summary by Run/S&P