Contest Log Analyzer - User Guide

Version: 0.36.8-Beta Date: 2025-08-15

--- Revision History ---

[0.36.8-Beta] - 2025-08-15

Changed

- Updated lists of required data files, CLI options, and supported

contests 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?

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. It goes beyond the simple score summary provided by most logging software, allowing you to:

- Process raw Cabrillo log files into a clean, standardized format.
- Automatically classify every QSO as "Run," "Search & Pounce," or "Unknown" to analyze your operating strategy.
- Generate detailed reports and charts that compare your log against one or more others.
- Analyze performance on a band-by-band basis to identify strengths and weaknesses.
- Calculate contest-specific QSO points for supported contests. 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.

---

2. What You Need to Get Started

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

- **Your Cabrillo Log File(s)**: These are the standard log files generated by your contest logging software (e.g., `kd4d.log`, `n0ni.log`). You can analyze a single log or compare multiple logs at once.
- **Data Files**: The program requires specific data files to be placed in a central `data/` directory.
    - `cty.dat`: Required for all contests.
    - `arrl_10_mults.dat`: Required for the ARRL 10 Meter contest.
    - `ARRLDXmults.dat`: Required for the ARRL DX contest.
    - `CQ160mults.dat`: Required for the CQ WW 160-Meter contest.
    - `NAQPmults.dat`: Required for North American QSO Party contests.
    - `SweepstakesSections.dat`: Required for ARRL Sweepstakes.
- **An Environment Variable**: You must tell the program where to find your data files by setting an environment variable named `CONTEST_LOGS_REPORTS`. This variable should point to the root directory

that contains your `Logs`, `data`, and `reports` subdirectories.

---

3. How to Run the Analyzer

The program is run from your command prompt or terminal using `main_cli.py`.

Basic Syntax

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

Command-Line Options

- `--report <ReportID|all|chart|text|plot|animation>`: (Required) Specifies which report to generate. 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.
- `<LogFile1> ...`: (Required) One or more paths to the Cabrillo log files you want to analyze.
- `--verbose`: (Optional) Enables detailed `INFO`-level status messages for debugging.
- `--include-dupes`: (Optional) By default, duplicate QSOs are ignored. Use this flag to include them in all calculations.
- `--mult-name <name>`: (Optional) For reports that analyze multipliers (like `missed_multipliers`), this specifies which multiplier to use (e.g., 'Countries', 'Zones').
- `--metric <qsos|points>`: (Optional) For the `cumulative_difference_plots` report, this specifies whether to compare QSO counts or Point totals. Defaults to 'qsos'.

Examples

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

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

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

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

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

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

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

CODE_BLOCK python main_cli.py --report missed_multipliers --mult-name Zones Logs/2024/cq-ww-cw/k3lr.log Logs/2024/cq-ww-cw/kc1xx.log CODE_BLOCK

4. Supported Contests

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

- ARRL 10 Meter
- ARRL DX (CW & SSB)
- 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

Use the `Report ID` with the `--report` command-line option.

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/`)

- `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