Manitoba Métis Federation Weather Station Data Cookbook - Python Edition

Prepared for the **Manitoba Métis Federation Weather Keeper Program** in collaboration with the **Centre for Earth Observation Science (CEOS)** , University of Manitoba.

Document Control

Version History

Version	Author(s)	Type	Date Modified	Comments
2.0	Campbell, Y.	Working Copy	2026-02-19	Python-based workflow. Alternative to R scripts.

Document Location

A digital copy of this document is stored in the Weather Keeper Program repository.

License

1. Overview

This cookbook describes the Python-based workflow used to clean, standardize, and quality-check weather station data collected from DataGarrison loggers deployed at Manitoba Métis Federation sites in Northern Manitoba. This workflow is another option to the previous R-based scripts and provides a reproducible approach to processing these files.

The workflow is implemented in a single processing engine (`processing.py`) and can be executed either through a Streamlit interface or through a pure Python script.

2. Raw Weather Station Files

Raw DataGarrison files are tab-delimited text files that contain:

- Metadata rows preceding the header
- Unstandardized variable names that include sensor identifiers
- Timestamp values in the format MM/DD/YY HH:MM:SS
- Wind speed values recorded in kilometres per hour (most recent logger)
- Measurements logged at fifteen-minute intervals

Raw files are stored in the Weather Keeper Program repository under station-specific directories –> Data –> raw.

3. Python Cleaning Workflow

Script and app files can be found here.

All cleaning logic is centralized in:

```
processing/processing.py
```

The workflow consists of the following stages.

3.1 Reading Raw Files

- Detects the header row by scanning for the first line containing the word “temperature”.
- Removes metadata rows if selected.
- Loads the file into a pandas DataFrame.

3.2 Structural & Variable Standardization

- Removes unnamed or empty columns.
- Renames raw variable names to standardized names using a `COLUMN_MAP`.
- Inserts result value qualifier (RVQ) columns immediately after each measurement variable to account for potentially bad values.

3.3 Wind Unit Conversion

- Raw wind values can be converted to m/s or km/hr, depending on the provided `raw_units` and `convert_choice`

3.4 Quality Control

- Converts timestamps to datetime.
- Adds year, month, and day helper columns.
- Applies range checks for each variable.
- Applies special rules for winter precipitation, wind speed, and wind direction.

3.5 Finalization

- Sorts rows chronologically.
- Removes duplicates.
- Reorders columns into a consistent structure.
- Formats timestamps as ISO strings.

4. Quality Control Rules

Quality control rules are applied in the `apply_qc_rules` function.

5.1 Timestamp Handling

- Raw timestamps are parsed using the format MM/DD/YY HH:MM:SS.
- Final timestamps are formatted as ISO strings (YYYY-MM-DDTHH:MM:SS).

5.2 Range Checks

Based on sensor ranges.

Variable	Lower Bound	Upper Bound	RVQ Code
air_pressure (mbar)	< 660	> 1070	BDL or ADL
photosynthetically_active_radiation (uE)	< 0	> 2500	BDL or ADL
air_temperature (deg_C)	< -40	> 75	BDL or ADL
relative_humidity (%)	< 0	> 100	BDL or ADL
precip (mm)	< 0	> 127	BDL or ADL
wind_speed (m/s)	< 0	> 100	BDL or ADL
wind_speed_of_gust (m/s)	< 0	> 100	BDL or ADL

5.3 Special Rules

Condition	Rule Applied	RVQ
wind_speed equals wind_speed_of_gust and both exceed 11 m/s	Both flagged	prob_bad
wind_speed exceeds 30 m/s	Flagged	prob_bad
wind_gust exceeds 35 m/s	Flagged	prob_bad
wind_from_direction between 355 and 360	Flagged	prob_bad
Month is December, January, or February	All precipitation flagged	prob_bad

4. Variable Transformation Table

The processing engine generates a dictionary table that documents how each variable is transformed. The table includes:

- The original variable name from the raw file
- The standardized cleaned name
- The units applied after cleaning

Table 4.1. Variable Transformations

Original Name	Cleaned Name	Units
Date_Time	date_and_time	UTC
Pressure_20812849_mbar	air_pressure	mbar
PAR_21181960_uE	photosynthetically_active_radiation	uE
Temperature_21238286_deg_C	air_temperature	deg_C
RH_21238286_%	relative_humidity	%
Rain_21201869_mm	precip	mm
Wind Speed_21292310_km/h	wind_speed	km/h or m/s
Gust Speed_21292310_km/h	wind_speed_of_gust	km/h or m/s
Wind Direction_21292310_deg	wind_from_direction	deg
Backup_Batts_21296930_V	battery_output	V

6. Output Files

6.1 Cleaned Files

Each raw file produces one cleaned CSV containing:

- Standardized variable names
- RVQ columns
- Optional wind unit conversion
- ISO-formatted timestamps

6.2 Compiled File

Multiple cleaned files may be combined into a single compiled dataset.

6.3 Dictionary Table

The dictionary table documents variable transformations and units. It is generated automatically in the Streamlit workflow and may be exported if needed.

7. Running the Workflow

7.1 Streamlit Application

The Streamlit interface provides a guided workflow:

1. Upload raw files

2. Preview raw data
3. Select wind unit conversion
4. Clean files
5. Preview cleaned data
6. Preview dictionary table
7. Compile files
8. Download outputs

The streamlit application can be found here: https://dg-cleaner.umcanwin.ca

7.2 Pure Python Script

The workflow may also be executed using:

```
python clean_datagarrison_file.py
```

Configuration options are stored in:

```
settings/config.py
```

See the README in data-garrison-script-workflow.