

Tower Water-Vapor Mixing Ratio Value-Added Product Report

MS Levin
KL Gaustad

D Zhang

Revised June 2021



DISCLAIMER

This report was prepared as an account of work sponsored by the U.S. Government. Neither the United States nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Tower Water-Vapor Mixing Ratio Value-Added Product Report

MS Levin
D Zhang
KL Gaustad
All at Pacific Northwest National Laboratory

Revised June 2021

Work supported by the U.S. Department of Energy,
Office of Science, Office of Biological and Environmental Research

Acronyms and Abbreviations

ARM	Atmospheric Radiation Measurement
DQ	data quality
MET	surface meteorological instrumentation
SGP	Southern Great Plains
SMOS	surface meteorological observation system
THWAPS	temperature, humidity, wind, and pressure system
TWRMR	Tower Water-Vapor Mixing Ratio value-added product
VAP	value-added product

Contents

Acronyms and Abbreviations	iii
1.0 Introduction	1
2.0 Changes from Previous Release	1
3.0 Algorithm and Methodology	2
3.1 Calculating Atmospheric Pressure	2
3.2 Calculating Water Vapor Mixing Ratio	3
3.3 Calculating Vapor Pressure	3
4.0 Input Data	3
4.1 Input Data Sources	3
4.2 Input Data Variables.....	3
5.0 Output Data	5
5.1 Best-Estimate Variables	5
5.2 Source Variables	6
5.3 Quality Check Variables	7
5.4 Location Variables	10
5.5 Plots.....	10
6.0 Summary.....	12
7.0 References	12
Appendix A – Input Datastreams.....	A.1
Appendix B – Output Datastream.....	B.1

Figures

1 Water-vapor mixing ratio at the 2-meter, 25-meter, and 60-meter levels of the meteorological tower.....	11
2 Relative humidity at the 2-meter, 25-meter, and 60-meter levels of the meteorological tower.	11
3 Air temperature at the 2-meter, 25-meter, and 60-meter levels of the meteorological tower.	12

Tables

1 Input datastreams and their date ranges.....	3
2 Retrieval rules for input data variables.....	4
3 Method for deriving best-estimate variables.	6
4 Mapping of best-estimate variable to source variable.	7
5 Description of quality checks applied to output variables.....	7
6 Retrieval rules for location variables.....	10

1.0 Introduction

The purpose of the Tower Water-Vapor Mixing Ratio (TWRMR) value-added product (VAP) is to calculate water-vapor mixing ratios at the 25-meter and 60-meter levels of the meteorological tower and also report best-estimate temperature, relative humidity, and pressure measurements at the 2-meter, 25-meter, and 60-meter levels at the Atmospheric Radiation Measurement (ARM) user facility Southern Great Plains (SGP) Central Facility. Because there are no barometric pressure sensors at the 25-meter and 60-meter levels on the tower, the hypsometric equation is used along with surface pressure values from the surface meteorological instrumentation (MET), the surface meteorological observation system (SMOS), or the temperature, humidity, wind, and pressure system (THWAPS) to derive barometric pressures at those altitudes (Ritsche 2008, 2011a, 2011b). After this is done, water-vapor mixing ratio can be calculated directly.

The TWRMR VAP has been updated several times in order to support the addition of new instruments, removal of old instruments, or updates to existing instruments collecting data onsite. This update covers all three scenarios, as several hardware improvements have been made to datastreams used as input to TWRMR, new instruments have come online, and some older instruments have been taken offline. Details about the changes relating to this VAP can be found in Section 2.

This technical report describes the TWRMR process in terms of its inputs, outputs, and processing applied.

2.0 Changes from Previous Release

This VAP has undergone several changes since its last update in April 2013, as follows:

- Added support for the `sgptowermetC1.b1` input datastream, which came online in April 2019 and includes meteorological data for the 25-meter and 60-meter southeast (SE) and west (W) elevators. Meteorological data from the SE elevators are preferred over the W elevators.
- Renamed variables according to the following rules:
 - Variable names ending in “MET” were renamed to end with “preferred”
 - Variable names ending in “SMOS” or “THWAPS” were renamed to end with “alternate” and a source attribute was added to record input source.
- Added the following data variables:
 - `pres_02m_preferred`
 - `pres_02m_alternate`
- Added ancillary source variables to variables whose input source is time dependent. These variables are:
 - `source_pres_02m`
 - `source_temp_02m`, `source_temp_25m`, `source_temp_60m`
 - `source_rh_02m`, `source_rh_25m`, `source_rh_60m`

- source_vap_pres_02m, source_vap_pres_25m, source_vap_pres_60m
- Added source attributes to variables whose source is determined at runtime. These variables are:
 - pres_02m_preferred, pres_02m_alternate
 - temp_02m_preferred, temp_02m_alternate
 - temp_25m_SE, temp_25m_W
 - temp_60m_SE, temp_60m_W
 - rh_02m_preferred, rh_02m_alternate
 - rh_25m_SE, rh_25m_W
 - rh_60m_SE, rh_60m_W
 - vap_pres_02m_preferred, vap_pres_02m_alternate
 - vap_pres_25m_SE, vap_pres_25m_W
 - vap_pres_60m_SE, vap_pres_60m_W
- Applied stricter limit checks on variables and added quality assessment to variables at each measurement level (2-meters, 25-meters, 60-meters) based on the quality assessment of the temperature best-estimate variable at the measurement level.
- Moved quality check descriptions from global attributes to the qc-variable level.
- Changed units of pressure variables from “mb” to “hPa”.
- Added standard_name attributes where applicable.
- Made other minor adjustments to adhere to ARM Data Standards v1.3.

3.0 Algorithm and Methodology

This VAP uses several algorithms to calculate atmospheric pressure, water-vapor mixing ratios, and, in some cases, vapor pressure. These algorithms are described in Sections 3.1, 3.2, and 3.3 respectively. However, the primary methodology used for selecting output data is determined by datastream availability and data quality metrics. These methods are described in Sections 4 and 5.

3.1 Calculating Atmospheric Pressure

A variation of the hypsometric equation is used to calculate atmospheric pressure at the 25-meter and 60-meter levels of the meteorological tower. The equation used to calculate pressure (P) given average temperature (T_{avg}) in Kelvin, the altitude (h) and reference altitude (h_{ref}) in meters above ground level, and the reference pressure (P_{ref}) in hectopascals is given below.

$$P = \exp\left(\frac{-g}{R_{air} \times T_{avg}}(h - h_{ref})\right) P_{ref}$$

Note that g is the acceleration due to gravity (9.8 m/s^2) and R_{air} is the gas constant of dry air (287.06 J/[kg.K]).

3.2 Calculating Water-Vapor Mixing Ratio

The water-vapor mixing ratio is calculated at all levels of the meteorological tower. The water-vapor mixing ratio (R) is a measure of the mass concentration of water vapor in the surrounding atmosphere. As such, it can be calculated from the water-vapor pressure (P_{vap}) and the atmospheric pressure (P) using the equation below:

$$R = 0.622 \left(\frac{P_{vap}}{P - P_{vap}} \right)$$

Note that 0.622 is the molecular weight ratio of water vapor to dry air.

3.3 Calculating Vapor Pressure

Vapor pressure is usually retrieved directly from an input datastream that records the vapor pressure directly from instrument readings. However, in some cases the input vapor pressure may be missing at the 2-meter level of the meteorological tower. In these cases, vapor pressure is derived from the dewpoint temperature measurement.

4.0 Input Data

4.1 Input Data Sources

The input data sources have changed over time. Listed below are the input datastreams and their begin and end dates. More information on each input datastream can be found on the ARM website.

Table 1. Input datastreams and their date ranges.

<u>Datastream</u>	<u>Range</u>
sgpmetE13.b1	1993-07-21 – present
sgp1twr10xC1.b1	1997-09-12 – 2019-04-24
sgp1twr25mC1.b1	1998-11-03 – 2019-04-24
sgp1twr60mC1.b1	1998-11-03 – 2019-04-24
sgpthwapsC1.b1	1999-09-21 – 2016-01-26
sgpmawsC1.b1	2014-09-08 – present
sgptowermetC1.b1	2019-04-25 – present

4.2 Input Data Variables

Data variables are retrieved from input data sources in a hierarchical manner such that the data from sgpmetE13.b1 and data from instruments on the southeast face of the meteorological tower are used as the primary source of data where possible. The suffixes “preferred” and “alternate” are used in the output

data variables at the 2-meter output level to indicate the priority in which retrieved variables should be used in a best-estimate calculation. At the 25-meter and 60-meter levels of the meteorological tower, “SE” and “W” suffixes indicate the preferred and alternate sources, respectively.

Table 2. Retrieval rules for input data variables.

Output Variable	Source 1	Source 2
temp_02m_preferred	sgpmetE13.b1 <i>temp_mean</i>	
temp_02m_alternate	sgpthwapsC1.b1 <i>temp</i>	sgpmawsC1.b1 <i>atmospheric_temperature</i>
pres_02m_preferred	sgpmetE13.b1 <i>atmos_pressure</i>	
pres_02m_alternate	sgpthwapsC1.b1 <i>pres</i>	sgpmawsC1.b1 <i>atmospheric_pressure</i>
rh_02m_preferred	sgpmetE13.b1 <i>rh_mean</i>	
rh_02m_alternate	sgpthwapsC1.b1 <i>rh</i>	sgpmawsC1.b1 <i>atmospheric_relative_humidity</i>
vap_pres_02m_preferred	sgpmetE13.b1 <i>vapor_pressure_mean</i>	
vap_pres_02m_alternate	sgpthwapsC1.b1 <i>vap_pres</i>	sgpmawsC1.b1 <i>atmospheric_dew_point*</i>
mixing_ratio_02m_preferred	Calculated using <i>vap_pres_02m_preferred</i> and <i>pres_02m_preferred</i>	
mixing_ratio_02m_alternate	Calculated using <i>vap_pres_02m_alternate</i> and <i>pres_02m_alternate</i>	
temp_25m_SE	sgpltwr10xC1.b1 <i>temp_25m</i>	sgptowermetC1.b1 <i>temperature_SE_25m_avg</i>
temp_25m_W	sgpltwr25mC1.b1 <i>temp</i>	sgptowermetC1.b1 <i>temperature_W_25m_avg</i>
pres_25m_SE	Calculated using the hypsometric equation with the following variables: <i>temp_25m_SE, temp_02m, pres_02m</i>	
pres_25m_W	Calculated using the hypsometric equation with the following variables: <i>temp_25m_W, temp_02m, pres_02m</i>	
rh_25m_SE	sgpltwr10xC1.b1 <i>rh_25m</i>	sgptowermetC1.b1 <i>relative_humidity_SE_25m_avg</i>
rh_25m_W	sgpltwr25mC1.b1 <i>rh</i>	sgptowermetC1.b1 <i>relative_humidity_W_25m_avg</i>
vap_pres_25m_SE	sgpltwr10xC1.b1 <i>vap_pres_25m</i>	sgptowermetC1.b1 <i>vapor_pressure_SE_25m_avg</i>
vap_pres_25m_W	sgpltwr25mC1.b1 <i>vap_pres</i>	sgptowermetC1.b1 <i>vapor_pressure_W_25m_avg</i>
mixing_ratio_25m_SE	Calculated using <i>vap_pres_25m_SE</i> and <i>pres_25m_SE</i>	
mixing_ratio_25m_W	Calculated using <i>vap_pres_25m_W</i> and <i>pres_25m_W</i>	
temp_60m_SE	sgpltwr10xC1.b1 <i>temp_60m</i>	sgptowermetC1.b1 <i>temperature_SE_60m_avg</i>
temp_60m_W	sgpltwr60mC1.b1 <i>temp</i>	sgptowermetC1.b1 <i>temperature_W_60m_avg</i>
pres_60m_SE	Calculated using the hypsometric equation with the following variables: <i>temp_60m_SE, temp_02m, pres_02m</i>	

Output Variable	Source 1	Source 2
pres_60m_W	Calculated using the hypsometric equation with the following variables: <i>temp_60m_W, temp_02m, pres_02m</i>	
rh_60m_SE	<i>sgp1twr10xC1.b1</i> <i>rh_60m</i>	<i>sgptowermetC1.b1</i> <i>relative_humidity_SE_60m_avg</i>
rh_60m_W	<i>sgp1twr60mC1.b1</i> <i>rh</i>	<i>sgptowermetC1.b1</i> <i>relative_humidity_W_60m_avg</i>
vap_pres_60m_SE	<i>sgp1twr10xC1.b1</i> <i>vap_pres_60m</i>	<i>sgptowermetC1.b1</i> <i>vapor_pressure_SE_60m_avg</i>
vap_pres_60m_W	<i>sgp1twr60mC1.b1</i> <i>vap_pres</i>	<i>sgptowermetC1.b1</i> <i>vapor_pressure_W_60m_avg</i>
mixing_ratio_60m_SE	Calculated using <i>vap_pres_60m_SE</i> and <i>pres_60m_SE</i>	
mixing_ratio_60m_W	Calculated using <i>vap_pres_60m_W</i> and <i>pres_60m_W</i>	

* If *vap_pres_02m_alternate* cannot be retrieved from Source 1, it is calculated from Source 2 using methods described in Section 3.3.

5.0 Output Data

The TWRMR VAP has just one output datastream – *sgp1twrmrC1.c1*. Each file in this output datastream spans a 24-hour period in 1-minute increments, typically starting at 12:00am and ending with 11:59pm. Each data file is composed of global attributes and data variables. The global attributes are primarily used for data provenance and include information such as the input datastreams used, the version of ARM Data Standards with which the data set complies, the command used to run the VAP, and more. The data variables section is composed of variables described in Sections 5.1, 5.2, 5.3, and 5.4. Section 5.5 provides several plots produced to illustrate the output data set. An example output file can be found in Appendix B.

5.1 Best-Estimate Variables

The output variables described in Section 4.2 are used to calculate best-estimate variables in the output *sgp1twrmrC1.c1* datastream. The following table shows the best-estimate variable names and how they are derived from the table in Section 4.2.

Table 3. Method for deriving best-estimate variables.

Output Variable	Source 1	Source 2	Method
temp_02m	temp_02m_preferred	temp_02m_alterate	Use Source 1 if available, else Source 2.
pres_02m	pres_02m_preferred	pres_02m_alterate	Use same Source # as temp_02m.
rh_02m	rh_02m_preferred	rh_02m_alterate	Use same Source # as temp_02m.
vap_pres_02m	vap_pres_02m_preferred	vap_pres_02m_alterate	Use same Source # as temp_02m.
mixing_ratio_02m	mixing_ratio_02m_preferred	mixing_ratio_02m_alterate	Use same Source # as temp_02m.
temp_25m	temp_25m_SE	temp_25m_W	Use Source 1 if available, else Source 2.
pres_25m	pres_25m_SE	pres_25m_W	Use same Source # as temp_25m.
rh_25m	rh_25m_SE	rh_25m_W	Use same Source # as temp_25m.
vap_pres_25m	vap_pres_25m_SE	vap_pres_25m_W	Use same Source # as temp_25m.
mixing_ratio_25m	mixing_ratio_25m_SE	mixing_ratio_25m_W	Use same Source # as temp_25m.
temp_60m	temp_60m_SE	temp_60m_W	Use Source 1 if available, else Source 2.
pres_60m	pres_60m_SE	pres_60m_W	Use same Source # as temp_60m.
rh_60m	rh_60m_SE	rh_60m_W	Use same Source # as temp_60m.
vap_pres_60m	vap_pres_60m_SE	vap_pres_60m_W	Use same Source # as temp_60m.
mixing_ratio_60m	mixing_ratio_60m_SE	mixing_ratio_60m_W	Use same Source # as temp_60m.

5.2 Source Variables

Because each best-estimate variable is derived from its “preferred” and “alternate” or “SE” and “W” companion variables, each best-estimate variable can be composed of data from one or two input datastreams. An accompanying “source” variable is therefore defined for each best-estimate variable to track the time-dependent input source datastreams of each data point. The following best-estimate variables and their accompanying source variables are listed in the table below. Note that water-vapor mixing ratio variables and atmospheric pressure variables above the 2-meter level do not have accompanying source variables because they are always calculated and are not derived directly from an input datastream.

Table 4. Mapping of best-estimate variable to source variable.

Best-Estimate Variable	Accompanying Source Variable
temp_02m	source_temp_02m
pres_02m	source_pres_02m
rh_02m	source_rh_02m
vap_pres_02m	source_vap_pres_02m
temp_25m	source_temp_25m
rh_25m	source_rh_25m
vap_pres_25m	source_vap_pres_25m
temp_60m	source_temp_60m
rh_60m	source_rh_60m
vap_pres_60m	source_vap_pres_60m

An example of a source variable for the temp_02m variable is shown below.

```
int source_temp_02m(time);¶
→ source_temp_02m:long_name="Source for variable: Temperature at 2-m";¶
→ source_temp_02m:units="1";¶
→ source_temp_02m:description="This variable contains integer values which should be interpreted as listed.";¶
→ source_temp_02m:flag_method="integer";¶
→ source_temp_02m:flag_0_description="no source available";¶
→ source_temp_02m:flag_1_description="sgpmetE13.b1:temp_mean";¶
→ source_temp_02m:flag_2_description="sgpmawsC1.b1:atmospheric_temperature";¶
```

5.3 Quality Check Variables

Various quality checks are applied to the data set to ensure that quality problems are detected and reported early on, and to prevent data points of poor quality from being used in downstream analyses.

The table below shows the quality check variables used to record data quality in the TWRMR VAP and provides a summary of the checks applied in each case. For more information about the specific checks applied, see the attributes of individual quality check variables in any output file produced by this VAP.

Table 5. Description of quality checks applied to output variables.

Data Variable	Summary of Quality Checks Applied
<i>Ancillary QC Variable</i>	
temp_02m <i>qc_temp_02m</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
temp_02m_preferred <i>qc_temp_02m_preferred</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
temp_02m_alternate <i>qc_temp_02m_alternate</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
pres_02m <i>qc_pres_02m</i>	Bad or indeterminate transformation to 1-min grid, temp_02m quality is bad

Data Variable	Summary of Quality Checks Applied
<i>Ancillary QC Variable</i>	
pres_02m_preferred <i>qc_pres_02m_preferred</i>	Bad or indeterminate transformation to 1-min grid, temp_02m_preferred quality is bad
pres_02m_alternate <i>qc_pres_02m_alternate</i>	Bad or indeterminate transformation to 1-min grid, temp_02m_alternate quality is bad
rh_02m <i>qc_rh_02m</i>	Bad or indeterminate transformation to 1-min grid, temp_02m quality is bad
rh_02m_preferred <i>qc_rh_02m_preferred</i>	Bad or indeterminate transformation to 1-min grid, temp_02m_preferred quality is bad
rh_02m_alternate <i>qc_rh_02m_alternate</i>	Bad or indeterminate transformation to 1-min grid, temp_02m_alternate quality is bad
vap_pres_02m <i>qc_vap_pres_02m</i>	Bad or indeterminate transformation to 1-min grid, temp_02m quality is bad, value outside of valid range
vap_pres_02m_preferred <i>qc_vap_pres_02m_preferred</i>	Bad or indeterminate transformation to 1-min grid, temp_02m quality is bad
vap_pres_02m_alternate <i>qc_vap_pres_02m_alternate</i>	Bad or indeterminate transformation to 1-min grid, temp_02m quality is bad, value outside of valid range
mixing_ratio_02m <i>qc_mixing_ratio_02m</i>	pres_02m or vap_pres_02m data quality is bad or indeterminate
mixing_ratio_02m_preferred <i>qc_mixing_ratio_02m_preferred</i>	pres_02m_preferred or vap_pres_02m_preferred data quality is bad or indeterminate
mixing_ratio_02m_alternate <i>qc_mixing_ratio_02m_alternate</i>	pres_02m_alternate or vap_pres_02m_alternate data quality is bad or indeterminate
temp_25m <i>qc_temp_25m</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
temp_25m_SE <i>qc_temp_25m_SE</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
temp_25m_W <i>qc_temp_25m_W</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
pres_25m <i>qc_pres_25m</i>	Bad or indeterminate transformation to 1-min grid, temp_25m quality is bad, value outside of valid range
pres_25m_SE <i>qc_pres_25m_SE</i>	Bad or indeterminate transformation to 1-min grid, temp_25m_SE quality is bad, value outside of valid range
pres_25m_W <i>qc_pres_25m_W</i>	Bad or indeterminate transformation to 1-min grid, temp_25m_W quality is bad, value outside of valid range
rh_25m <i>qc_rh_25m</i>	Bad or indeterminate transformation to 1-min grid, temp_25m quality is bad
rh_25m_SE <i>qc_rh_25m_SE</i>	Bad or indeterminate transformation to 1-min grid, temp_25m_SE quality is bad
rh_25m_W <i>qc_rh_25m_W</i>	Bad or indeterminate transformation to 1-min grid, temp_25m_W quality is bad
vap_pres_25m <i>qc_vap_pres_25m</i>	Bad or indeterminate transformation to 1-min grid, temp_25m quality is bad
vap_pres_25m_SE <i>qc_vap_pres_25m_SE</i>	Bad or indeterminate transformation to 1-min grid, temp_25m_SE quality is bad
vap_pres_25m_W <i>qc_vap_pres_25m_W</i>	Bad or indeterminate transformation to 1-min grid, temp_25m_W quality is bad

Data Variable	Summary of Quality Checks Applied
<i>Ancillary QC Variable</i>	
mixing_ratio_25m <i>qc_mixing_ratio_25m</i>	pres_25m or vap_pres_25m data quality is bad or indeterminate
mixing_ratio_25m_SE <i>qc_mixing_ratio_25m_SE</i>	pres_25m_SE or vap_pres_25m_SE data quality is bad or indeterminate
mixing_ratio_25m_W <i>qc_mixing_ratio_25m_W</i>	pres_25m_W or vap_pres_25m_W data quality is bad or indeterminate
temp_60m <i>qc_temp_60m</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
temp_60m_SE <i>qc_temp_60m_SE</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
temp_60m_W <i>qc_temp_60m_W</i>	Bad or indeterminate transformation to 1-min grid, value outside of valid range
pres_60m <i>qc_pres_60m</i>	Bad or indeterminate transformation to 1-min grid, temp_60m quality is bad, value outside of valid range
pres_60m_SE <i>qc_pres_60m_SE</i>	Bad or indeterminate transformation to 1-min grid, temp_60m_SE quality is bad, value outside of valid range
pres_60m_W <i>qc_pres_60m_W</i>	Bad or indeterminate transformation to 1-min grid, temp_60m_W quality is bad, value outside of valid range
rh_60m <i>qc_rh_60m</i>	Bad or indeterminate transformation to 1-min grid, temp_60m quality is bad
rh_60m_SE <i>qc_rh_60m_SE</i>	Bad or indeterminate transformation to 1-min grid, temp_60m_SE quality is bad
rh_60m_W <i>qc_rh_60m_W</i>	Bad or indeterminate transformation to 1-min grid, temp_60m_W quality is bad
vap_pres_60m <i>qc_vap_pres_60m</i>	Bad or indeterminate transformation to 1-min grid, temp_60m quality is bad
vap_pres_60m_SE <i>qc_vap_pres_60m_SE</i>	Bad or indeterminate transformation to 1-min grid, temp_60m_SE quality is bad
vap_pres_60m_W <i>qc_vap_pres_60m_W</i>	Bad or indeterminate transformation to 1-min grid, temp_60m_W quality is bad
mixing_ratio_60m <i>qc_mixing_ratio_60m</i>	pres_60m or vap_pres_60m data quality is bad or indeterminate
mixing_ratio_60m_SE <i>qc_mixing_ratio_60m_SE</i>	pres_60m_SE or vap_pres_60m_SE data quality is bad or indeterminate
mixing_ratio_60m_W <i>qc_mixing_ratio_60m_W</i>	pres_60m_W or vap_pres_60m_W data quality is bad or indeterminate

An example of a quality check variable for the temp_02m variable is shown below:

```

int qc_temp_02m(time);¶
qc_temp_02m:long_name="Quality check results on variable: Temperature at 2 m";¶
qc_temp_02m:units="1";¶
qc_temp_02m:description="This variable contains bit-packed integer values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests.";¶
qc_temp_02m:standard_name="quality_flag";¶
qc_temp_02m:flag_method="bit";¶
qc_temp_02m:bit_1_description="Transformation could not finish (all values bad or outside range, etc.), value set to missing_value.";¶
qc_temp_02m:bit_1_assessment="Bad";¶
qc_temp_02m:bit_2_description="Transformation resulted in an indeterminate outcome.";¶
qc_temp_02m:bit_2_assessment="Indeterminate";¶
qc_temp_02m:bit_3_description="Value is less than the valid_min, value set to missing_value.";¶
qc_temp_02m:bit_3_assessment="Bad";¶
qc_temp_02m:bit_4_description="Value is greater than the valid_max, value set to missing_value.";¶
qc_temp_02m:bit_4_assessment="Bad";¶

```

5.4 Location Variables

The “lat”, “lon”, and “alt” variables are included as the final data variables in this data set and are present to capture the latitude, longitude, and altitude of the location where the data was collected. Note that these variables are retrieved from the same datastream as that used to retrieve data for the “preferred” variables at the 2-meter level of the meteorological tower. That is, the retrieval rules for the “lat”, “lon”, and “alt” variables is as follows:

Table 6. Retrieval rules for location variables.

Output Variable	Source 1	Source 2	Source 3
lat	sgpmetE13.b1 <i>lat</i>	sgpchwapsC1.b1 <i>lat</i>	sgpmawsC1.b1 <i>lat</i>
lon	sgpmetE13.b1 <i>lon</i>	sgpchwapsC1.b1 <i>lon</i>	sgpmawsC1.b1 <i>lon</i>
alt	sgpmetE13.b1 <i>alt</i>	sgpchwapsC1.b1 <i>alt</i>	sgpmawsC1.b1 <i>alt</i>

5.5 Plots

TWRMR does not produce output plots, but shown below are several visualizations of data produced by this VAP. These plots were produced by ARM’s [DQ Zoom plotter tool](#) and have been included for illustration purposes.

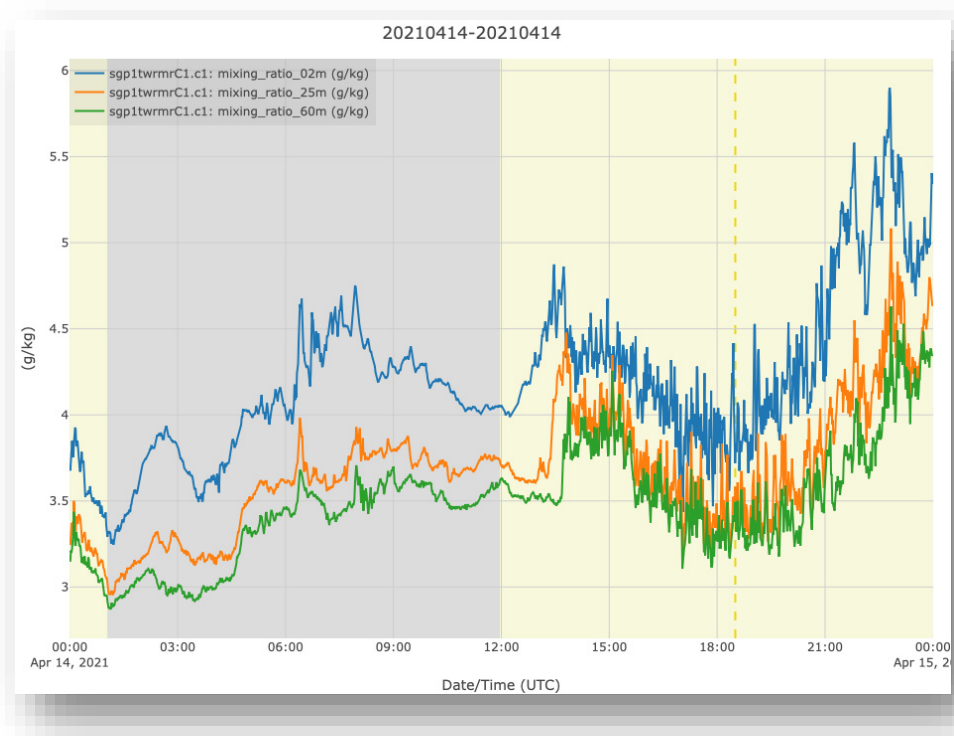


Figure 1. Water-vapor mixing ratio at the 2-meter, 25-meter, and 60-meter levels of the meteorological tower.

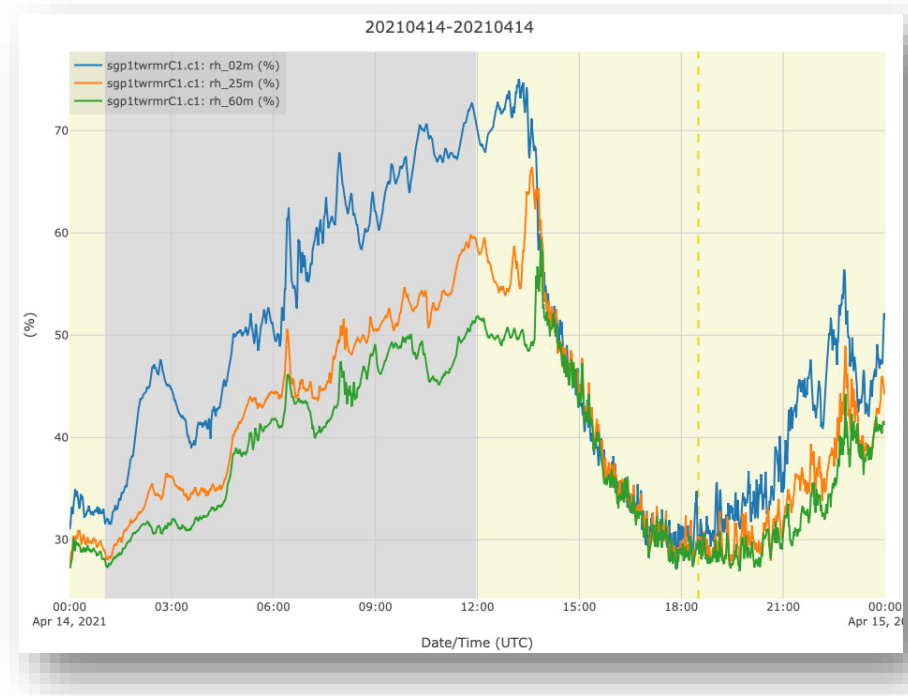


Figure 2. Relative humidity at the 2-meter, 25-meter, and 60-meter levels of the meteorological tower.

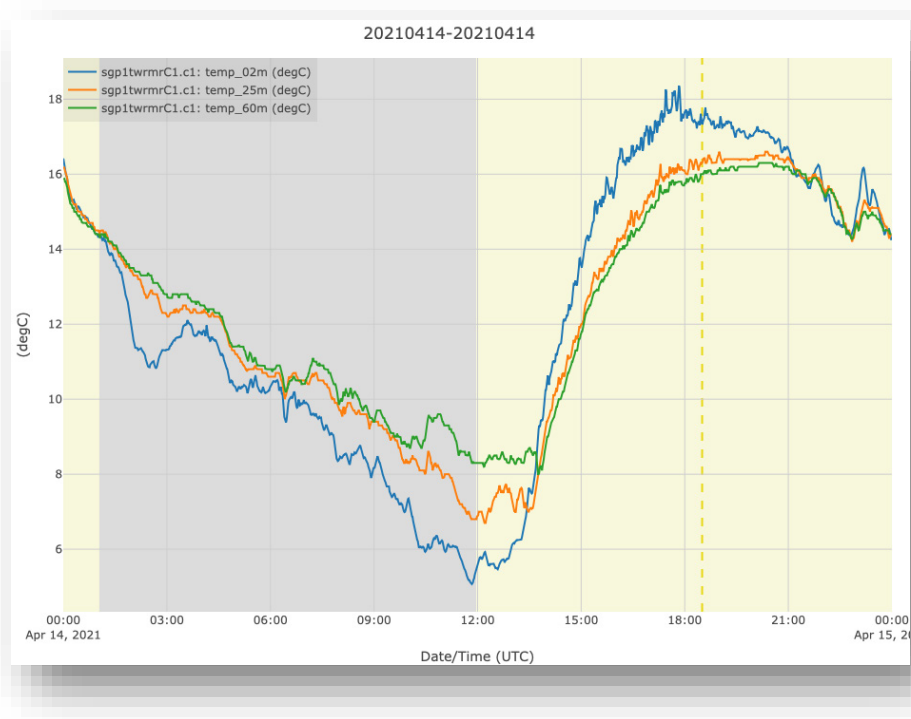


Figure 3. Air temperature at the 2-meter, 25-meter, and 60-meter levels of the meteorological tower.

6.0 Summary

The TWRMR VAP provides water-vapor mixing ratios at the 2-meter, 25-meter, and 60-meter levels using the best-estimate meteorological data at the ARM SGP site, which can be directly used to compare with model outputs. This report documents methods used to derive water-vapor mixing ratios at those levels and also major changes since its last update in April 2013.

7.0 References

- Ritsche, MT. 2008. Surface Meteorological Observation System (SMOS) Handbook. DOE/SC-ARM/TR-031. https://www.arm.gov/publications/tech_reports/handbooks/smos_handbook.pdf
- Ritsche, MT. 2011a. ARM Surface Meteorology Systems (MET) Handbook. DOE/SC-ARM/TR-086. https://www.arm.gov/publications/tech_reports/handbooks/met_handbook.pdf
- Ritsche, MT. 2011b. Temperature, Humidity, Wind, and Pressure System (THWAPS) Handbook. DOE/SC-ARM/TR-030. https://www.arm.gov/publications/tech_reports/handbooks/thwaps_handbook.pdf

Appendix A

Input Datastreams

sgpmetE13.b1:

```
netcdf sgpmetE13.b1.20210414.000000 {
dimensions:
    time = UNLIMITED ; // (1440 currently)
variables:
    int base_time ;
        base_time:string = "2021-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2021-04-14 00:00:00 0:00" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2021-04-14 00:00:00 0:00" ;
        time:standard_name = "time" ;
    float atmos_pressure(time) ;
        atmos_pressure:long_name = "Atmospheric pressure" ;
        atmos_pressure:units = "kPa" ;
        atmos_pressure:valid_min = 80.f ;
        atmos_pressure:valid_max = 110.f ;
        atmos_pressure:valid_delta = 1.f ;
        atmos_pressure:missing_value = -9999.f ;
    int qc_atmos_pressure(time) ;
        qc_atmos_pressure:long_name = "Quality check results on field: Atmospheric pressure" ;
        qc_atmos_pressure:units = "unitless" ;
        qc_atmos_pressure:description = "See global attributes for individual bit descriptions." ;
    float temp_mean(time) ;
        temp_mean:long_name = "Temperature mean" ;
        temp_mean:units = "degC" ;
        temp_mean:valid_min = -40.f ;
        temp_mean:valid_max = 50.f ;
        temp_mean:valid_delta = 20.f ;
```

```

    temp_mean:missing_value = -9999.f;
int qc_temp_mean(time);
    qc_temp_mean:long_name = "Quality check results on field: Temperature mean";
    qc_temp_mean:units = "unitless";
    qc_temp_mean:description = "See global attributes for individual bit descriptions.";
float temp_std(time);
    temp_std:long_name = "Temperature standard deviation";
    temp_std:units = "degC";
float rh_mean(time);
    rh_mean:long_name = "Relative humidity mean";
    rh_mean:units = "%";
    rh_mean:valid_min = -2.f;
    rh_mean:valid_max = 104.f;
    rh_mean:valid_delta = 30.f;
    rh_mean:missing_value = -9999.f;
int qc_rh_mean(time);
    qc_rh_mean:long_name = "Quality check results on field: Relative humidity mean";
    qc_rh_mean:units = "unitless";
    qc_rh_mean:description = "See global attributes for individual bit descriptions.";
float rh_std(time);
    rh_std:long_name = "Relative humidity standard deviation";
    rh_std:units = "%";
float vapor_pressure_mean(time);
    vapor_pressure_mean:long_name = "Vapor pressure mean, calculated";
    vapor_pressure_mean:units = "kPa";
    vapor_pressure_mean:valid_min = 0.f;
    vapor_pressure_mean:valid_max = 10.f;
    vapor_pressure_mean:valid_delta = 1.f;
    vapor_pressure_mean:missing_value = -9999.f;
int qc_vapor_pressure_mean(time);
    qc_vapor_pressure_mean:long_name = "Quality check results on field: Vapor pressure
mean, calculated";
    qc_vapor_pressure_mean:units = "unitless";
    qc_vapor_pressure_mean:description = "See global attributes for individual bit
descriptions.";
float vapor_pressure_std(time);
    vapor_pressure_std:long_name = "Vapor pressure standard deviation";
    vapor_pressure_std:units = "kPa";
float wspd_arith_mean(time);
    wspd_arith_mean:long_name = "Wind speed arithmetic mean";
    wspd_arith_mean:units = "m/s";
    wspd_arith_mean:valid_min = 0.f;
    wspd_arith_mean:valid_max = 60.f;
    wspd_arith_mean:valid_delta = 20.f;
    wspd_arith_mean:missing_value = -9999.f;
int qc_wspd_arith_mean(time);

```

```

        qc_wspd_arith_mean:long_name = "Quality check results on field: Wind speed
arithmetic mean" ;
        qc_wspd_arith_mean:units = "unitless" ;
        qc_wspd_arith_mean:description = "See global attributes for individual bit descriptions."
;
float wspd_vec_mean(time) ;
    wspd_vec_mean:long_name = "Wind speed vector mean" ;
    wspd_vec_mean:units = "m/s" ;
    wspd_vec_mean:valid_min = 0.f ;
    wspd_vec_mean:valid_max = 60.f ;
    wspd_vec_mean:valid_delta = 20.f ;
    wspd_vec_mean:missing_value = -9999.f ;
int qc_wspd_vec_mean(time) ;
    qc_wspd_vec_mean:long_name = "Quality check results on field: Wind speed vector
mean" ;
    qc_wspd_vec_mean:units = "unitless" ;
    qc_wspd_vec_mean:description = "See global attributes for individual bit descriptions." ;
float wdir_vec_mean(time) ;
    wdir_vec_mean:long_name = "Wind direction vector mean" ;
    wdir_vec_mean:units = "degree" ;
    wdir_vec_mean:valid_min = 0.f ;
    wdir_vec_mean:valid_max = 360.f ;
    wdir_vec_mean:missing_value = -9999.f ;
int qc_wdir_vec_mean(time) ;
    qc_wdir_vec_mean:long_name = "Quality check results on field: Wind direction vector
mean" ;
    qc_wdir_vec_mean:units = "unitless" ;
    qc_wdir_vec_mean:description = "See global attributes for individual bit descriptions." ;
float wdir_vec_std(time) ;
    wdir_vec_std:long_name = "Wind direction vector mean standard deviation" ;
    wdir_vec_std:units = "degree" ;
    wdir_vec_std:missing_value = -9999.f ;
float tbrg_precip_total(time) ;
    tbrg_precip_total:long_name = "TBRG precipitation total" ;
    tbrg_precip_total:units = "mm" ;
    tbrg_precip_total:valid_min = 0.f ;
    tbrg_precip_total:valid_max = 10.f ;
    tbrg_precip_total:missing_value = -9999.f ;
int qc_tbrg_precip_total(time) ;
    qc_tbrg_precip_total:long_name = "Quality check results on field: TBRG precipitation
total" ;
    qc_tbrg_precip_total:units = "unitless" ;
    qc_tbrg_precip_total:description = "See global attributes for individual bit descriptions." ;
float tbrg_precip_total_corr(time) ;
    tbrg_precip_total_corr:long_name = "TBRG precipitation total, corrected" ;
    tbrg_precip_total_corr:units = "mm" ;

```

```

    tbrg_precip_total_corr:valid_min = 0.f ;
    tbrg_precip_total_corr:valid_max = 10.f ;
    tbrg_precip_total_corr:missing_value = -9999.f ;
int qc_tbrg_precip_total_corr(time) ;
    qc_tbrg_precip_total_corr:long_name = "Quality check results on field: TBRG
precipitation total, corrected" ;
    qc_tbrg_precip_total_corr:units = "unitless" ;
    qc_tbrg_precip_total_corr:description = "See global attributes for individual bit
descriptions." ;
float org_precip_rate_mean(time) ;
    org_precip_rate_mean:long_name = "ORG precipitation rate mean" ;
    org_precip_rate_mean:units = "mm/hr" ;
    org_precip_rate_mean:valid_min = 0.f ;
    org_precip_rate_mean:valid_max = 500.f ;
    org_precip_rate_mean:missing_value = -9999.f ;
int qc_org_precip_rate_mean(time) ;
    qc_org_precip_rate_mean:long_name = "Quality check results on field: ORG
precipitation rate mean" ;
    qc_org_precip_rate_mean:units = "unitless" ;
    qc_org_precip_rate_mean:description = "See global attributes for individual bit
descriptions." ;
int pwd_err_code(time) ;
    pwd_err_code:long_name = "PWD alarm" ;
    pwd_err_code:units = "unitless" ;
    pwd_err_code:missing_value = -9999 ;
int pwd_mean_vis_1min(time) ;
    pwd_mean_vis_1min:long_name = "PWD 1 minute mean visibility" ;
    pwd_mean_vis_1min:units = "m" ;
    pwd_mean_vis_1min:valid_min = 0 ;
    pwd_mean_vis_1min:valid_max = 20000 ;
    pwd_mean_vis_1min:missing_value = -9999 ;
int qc_pwd_mean_vis_1min(time) ;
    qc_pwd_mean_vis_1min:long_name = "Quality check results on field: PWD 1 minute
mean visibility" ;
    qc_pwd_mean_vis_1min:units = "unitless" ;
    qc_pwd_mean_vis_1min:description = "See global attributes for individual bit
descriptions." ;
int pwd_mean_vis_10min(time) ;
    pwd_mean_vis_10min:long_name = "PWD 10 minute mean visibility" ;
    pwd_mean_vis_10min:units = "m" ;
    pwd_mean_vis_10min:valid_min = 0 ;
    pwd_mean_vis_10min:valid_max = 20000 ;
    pwd_mean_vis_10min:missing_value = -9999 ;
int qc_pwd_mean_vis_10min(time) ;
    qc_pwd_mean_vis_10min:long_name = "Quality check results on field: PWD 10 minute
mean visibility" ;

```

```

qc_pwd_mean_vis_10min:units = "unitless" ;
qc_pwd_mean_vis_10min:description = "See global attributes for individual bit
descriptions." ;
int pwd_pw_code_inst(time) ;
pwd_pw_code_inst:long_name = "PWD instantaneous present weather code" ;
pwd_pw_code_inst:units = "unitless" ;
pwd_pw_code_inst:valid_min = 0 ;
pwd_pw_code_inst:valid_max = 99 ;
pwd_pw_code_inst:missing_value = -9999 ;
int qc_pwd_pw_code_inst(time) ;
qc_pwd_pw_code_inst:long_name = "Quality check results on field: PWD instantaneous
present weather code" ;
qc_pwd_pw_code_inst:units = "unitless" ;
qc_pwd_pw_code_inst:description = "See global attributes for individual bit
descriptions." ;
int pwd_pw_code_15min(time) ;
pwd_pw_code_15min:long_name = "PWD 15 minute present weather code" ;
pwd_pw_code_15min:units = "unitless" ;
pwd_pw_code_15min:valid_min = 0 ;
pwd_pw_code_15min:valid_max = 99 ;
pwd_pw_code_15min:missing_value = -9999 ;
int qc_pwd_pw_code_15min(time) ;
qc_pwd_pw_code_15min:long_name = "Quality check results on field: PWD 15 minute
present weather code" ;
qc_pwd_pw_code_15min:units = "unitless" ;
qc_pwd_pw_code_15min:description = "See global attributes for individual bit
descriptions." ;
int pwd_pw_code_1hr(time) ;
pwd_pw_code_1hr:long_name = "PWD 1 hour present weather code" ;
pwd_pw_code_1hr:units = "unitless" ;
pwd_pw_code_1hr:valid_min = 0 ;
pwd_pw_code_1hr:valid_max = 99 ;
pwd_pw_code_1hr:missing_value = -9999 ;
int qc_pwd_pw_code_1hr(time) ;
qc_pwd_pw_code_1hr:long_name = "Quality check results on field: PWD 1 hour present
weather code" ;
qc_pwd_pw_code_1hr:units = "unitless" ;
qc_pwd_pw_code_1hr:description = "See global attributes for individual bit
descriptions." ;
float pwd_precip_rate_mean_1min(time) ;
pwd_precip_rate_mean_1min:long_name = "PWD 1 minute mean precipitation rate" ;
pwd_precip_rate_mean_1min:units = "mm/hr" ;
pwd_precip_rate_mean_1min:valid_min = 0.f ;
pwd_precip_rate_mean_1min:valid_max = 999.99f ;
pwd_precip_rate_mean_1min:valid_delta = 100.f ;
pwd_precip_rate_mean_1min:missing_value = -9999.f ;

```

```

int qc_pwd_precip_rate_mean_1min(time) ;
    qc_pwd_precip_rate_mean_1min:long_name = "Quality check results on field: PWD 1
minute mean precipitation rate" ;
    qc_pwd_precip_rate_mean_1min:units = "unitless" ;
    qc_pwd_precip_rate_mean_1min:description = "See global attributes for individual bit
descriptions." ;
float pwd_cumul_rain(time) ;
    pwd_cumul_rain:long_name = "PWD cumulative liquid precipitation" ;
    pwd_cumul_rain:units = "mm" ;
    pwd_cumul_rain:valid_min = 0.f ;
    pwd_cumul_rain:valid_max = 99.99f ;
    pwd_cumul_rain:valid_delta = 50.f ;
    pwd_cumul_rain:missing_value = -9999.f ;
int qc_pwd_cumul_rain(time) ;
    qc_pwd_cumul_rain:long_name = "Quality check results on field: PWD cumulative
liquid precipitation" ;
    qc_pwd_cumul_rain:units = "unitless" ;
    qc_pwd_cumul_rain:description = "See global attributes for individual bit descriptions." ;
float pwd_cumul_snow(time) ;
    pwd_cumul_snow:long_name = "PWD cumulative snow" ;
    pwd_cumul_snow:units = "mm" ;
    pwd_cumul_snow:valid_min = 0.f ;
    pwd_cumul_snow:valid_max = 999.f ;
    pwd_cumul_snow:valid_delta = 100.f ;
    pwd_cumul_snow:missing_value = -9999.f ;
int qc_pwd_cumul_snow(time) ;
    qc_pwd_cumul_snow:long_name = "Quality check results on field: PWD cumulative
snow" ;
    qc_pwd_cumul_snow:units = "unitless" ;
    qc_pwd_cumul_snow:description = "See global attributes for individual bit descriptions."
;

float logger_volt(time) ;
    logger_volt:long_name = "Logger voltage" ;
    logger_volt:units = "V" ;
    logger_volt:missing_value = -9999.f ;
    logger_volt:valid_min = 10.f ;
    logger_volt:valid_max = 15.f ;
    logger_volt:valid_delta = 5.f ;
int qc_logger_volt(time) ;
    qc_logger_volt:long_name = "Quality check results on field: Logger voltage" ;
    qc_logger_volt:units = "unitless" ;
    qc_logger_volt:description = "See global attributes for individual bit descriptions." ;
float logger_temp(time) ;
    logger_temp:long_name = "Logger temperature" ;
    logger_temp:units = "degC" ;
    logger_temp:missing_value = -9999.f ;

```

```

    logger_temp:valid_min = -25.f;
    logger_temp:valid_max = 50.f;
    logger_temp:valid_delta = 10.f;
int qc_logger_temp(time);
    qc_logger_temp:long_name = "Quality check results on field: Logger temperature" ;
    qc_logger_temp:units = "unitless" ;
    qc_logger_temp:description = "See global attributes for individual bit descriptions." ;
float lat ;
    lat:long_name = "North latitude" ;
    lat:units = "degree_N" ;
    lat:valid_min = -90.f;
    lat:valid_max = 90.f;
    lat:standard_name = "latitude" ;
float lon ;
    lon:long_name = "East longitude" ;
    lon:units = "degree_E" ;
    lon:valid_min = -180.f;
    lon:valid_max = 180.f;
    lon:standard_name = "longitude" ;
float alt ;
    alt:long_name = "Altitude above mean sea level" ;
    alt:units = "m" ;
    alt:standard_name = "altitude" ;

// global attributes:
:command_line = "met_ingest -s sgp -f E13" ;
:process_version = "ingest-met-4.45-0.e17" ;
:dod_version = "met-b1-7.3" ;
:input_source = "/data/collection/sgp/sgpmetE13.00/MET_Table1.20210414000000.dat"
;

:site_id = "sgp" ;
:platform_id = "met" ;
:facility_id = "E13" ;
:data_level = "b1" ;
:location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
:datastream = "sgpmetE13.b1" ;
:serial_number = "116" ;
:sampling_interval = "variable, see instrument handbook" ;
:averaging_interval = "60 seconds" ;
:averaging_interval_comment = "The time assigned to each data point indicates the end
of the averaging interval." ;
:tbrg = "Tipping Bucket Rain Gauge" ;
:pwd = "Present Weather Detector" ;
:wind_speed_offset = "0.000000" ;
:wind_speed_slope = "0.098000" ;

```



```

        :tbrg_precip_corr_info = "0.001000 * tbrg_precip_total^2 + 0.999000 *
tbrg_precip_total" ;
        :qc_bit_comment = "The QC field values are a bit packed representation of true/false
values for the tests that may have been performed. A QC value of zero means that none of the tests
performed on the value failed." ;
        :qc_bit_1_description = "Value is equal to missing_value." ;
        :qc_bit_1_assessment = "Bad" ;
        :qc_bit_2_description = "Value is less than the valid_min." ;
        :qc_bit_2_assessment = "Bad" ;
        :qc_bit_3_description = "Value is greater than the valid_max." ;
        :qc_bit_3_assessment = "Bad" ;
        :qc_bit_4_description = "Difference between current and previous values exceeds
valid_delta." ;
        :qc_bit_4_assessment = "Indeterminate" ;
        :history = "created by user dsmgr on machine zinc at 2021-04-14 02:49:00, using ingest-
met-4.45-0.e17" ;
    }

```

sgp1twr10xC1.b1:

```

netcdf sgp1twr10xC1.b1.20180414.000000 {
dimensions:
    time = UNLIMITED ; // (1440 currently)
variables:
    int base_time ;
        base_time:string = "2018-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2018-04-14 00:00:00 0:00" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2018-04-14 00:00:00 0:00" ;
    int qc_time(time) ;
        qc_time:long_name = "Quality check results on field: Time offset from midnight" ;
        qc_time:units = "unitless" ;
        qc_time:description = "This field contains bit packed values which should be interpreted
as listed. No bits set (zero) represents good data." ;
        qc_time:bit_1_description = "Delta time between current and previous samples is zero." ;
        qc_time:bit_1_assessment = "Indeterminate" ;
        qc_time:bit_2_description = "Delta time between current and previous samples is less
than the delta_t_lower_limit field attribute." ;
        qc_time:bit_2_assessment = "Indeterminate" ;

```

```

    qc_time:bit_3_description = "Delta time between current and previous samples is greater
than the delta_t_upper_limit field attribute." ;
    qc_time:bit_3_assessment = "Indeterminate" ;
    qc_time:delta_t_lower_limit = 57. ;
    qc_time:delta_t_upper_limit = 63. ;
    qc_time:prior_sample_flag = 1 ;
    qc_time:comment = "If the '\prior_sample_flag\' is set the first sample time from a new
raw file will be compared against the time just previous to it in the stored data. If it is not set the qc_time
value for the first sample will be set to 0." ;
    float temp_60m(time) ;
        temp_60m:long_name = "Air temperature at 60m" ;
        temp_60m:units = "C" ;
        temp_60m:valid_min = -50.f ;
        temp_60m:valid_max = 50.f ;
        temp_60m:missing_value = -9999.f ;
    int qc_temp_60m(time) ;
        qc_temp_60m:long_name = "Quality check results on field: Air temperature at 60m" ;
        qc_temp_60m:units = "unitless" ;
        qc_temp_60m:description = "See global attributes for individual bit descriptions." ;
    float temp_25m(time) ;
        temp_25m:long_name = "Air temperature at 25m" ;
        temp_25m:units = "C" ;
        temp_25m:valid_min = -50.f ;
        temp_25m:valid_max = 50.f ;
        temp_25m:missing_value = -9999.f ;
    int qc_temp_25m(time) ;
        qc_temp_25m:long_name = "Quality check results on field: Air temperature at 25m" ;
        qc_temp_25m:units = "unitless" ;
        qc_temp_25m:description = "See global attributes for individual bit descriptions." ;
    float rh_60m(time) ;
        rh_60m:long_name = "Relative humidity at 60m" ;
        rh_60m:units = "%" ;
        rh_60m:valid_min = 0.f ;
        rh_60m:valid_max = 102.f ;
        rh_60m:missing_value = -9999.f ;
    int qc_rh_60m(time) ;
        qc_rh_60m:long_name = "Quality check results on field: Relative humidity at 60m" ;
        qc_rh_60m:units = "unitless" ;
        qc_rh_60m:description = "See global attributes for individual bit descriptions." ;
    float rh_25m(time) ;
        rh_25m:long_name = "Relative humidity at 25m" ;
        rh_25m:units = "%" ;
        rh_25m:valid_min = 0.f ;
        rh_25m:valid_max = 102.f ;
        rh_25m:missing_value = -9999.f ;
    int qc_rh_25m(time) ;

```

```

qc_rh_25m:long_name = "Quality check results on field: Relative humidity at 25m" ;
qc_rh_25m:units = "unitless" ;
qc_rh_25m:description = "See global attributes for individual bit descriptions." ;
float vap_pres_60m(time) ;
vap_pres_60m:long_name = "Vapor pressure at 60m" ;
vap_pres_60m:units = "kPa" ;
vap_pres_60m:valid_min = 0.f ;
vap_pres_60m:valid_max = 10.f ;
vap_pres_60m:missing_value = -9999.f ;
int qc_vap_pres_60m(time) ;
qc_vap_pres_60m:long_name = "Quality check results on field: Vapor pressure at 60m" ;
qc_vap_pres_60m:units = "unitless" ;
qc_vap_pres_60m:description = "See global attributes for individual bit descriptions." ;
float vap_pres_25m(time) ;
vap_pres_25m:long_name = "Vapor pressure at 25m" ;
vap_pres_25m:units = "kPa" ;
vap_pres_25m:valid_min = 0.f ;
vap_pres_25m:valid_max = 10.f ;
vap_pres_25m:missing_value = -9999.f ;
int qc_vap_pres_25m(time) ;
qc_vap_pres_25m:long_name = "Quality check results on field: Vapor pressure at 25m" ;
qc_vap_pres_25m:units = "unitless" ;
qc_vap_pres_25m:description = "See global attributes for individual bit descriptions." ;
float vbat(time) ;
vbat:long_name = "Battery voltage" ;
vbat:units = "V" ;
vbat:valid_min = 10.4f ;
vbat:valid_max = 15.f ;
vbat:missing_value = -9999.f ;
vbat:comment = "This is the voltage of the CR10X battery for QA/QC purposes,\n",
                "prior to 19:30 GMT on 10/30/2000 this was a CR21X logger" ;
int qc_vbat(time) ;
qc_vbat:long_name = "Quality check results on field: Battery voltage" ;
qc_vbat:units = "unitless" ;
qc_vbat:description = "See global attributes for individual bit descriptions." ;
float lat ;
lat:long_name = "North latitude" ;
lat:units = "degree_N" ;
lat:valid_min = -90.f ;
lat:valid_max = 90.f ;
float lon ;
lon:long_name = "East longitude" ;
lon:units = "degree_E" ;
lon:valid_min = -180.f ;
lon:valid_max = 180.f ;
float alt ;

```

```
alt:long_name = "Altitude above mean sea level" ;
alt:units = "m" ;
```

```
// global attributes:
```

```
:command_line = "twr_ingest -n twr10x -s sgp -f C1" ;
:process_version = "ingest-twr-8.4-0.el6" ;
:dod_version = "1twr10x-b1-1.1" ;
:data_level = "b1" ;
:input_source = "/data/collection/sgp/sgptwr10xC1.00/1523660700.icm" ;
:site_id = "sgp" ;
:facility_id = "C1: Lamont, Oklahoma" ;
:datastream = "sgp1twr10xC1.b1" ;
:sampling_interval = "1 second" ;
:averaging_interval = "1 minute" ;
:serial_number = "N/A" ;
:comment = "N/A" ;
:resolution_description = "The resolution field attributes refer to the number of
```

significant digits relative to the decimal point that should be used in calculations. Using fewer digits might result in greater uncertainty. Using a larger number of digits should have no effect and thus is unnecessary. However, analyses based on differences in values with a larger number of significant digits than indicated could lead to erroneous results or misleading scientific conclusions.\n",

```
"\n",
```

```
"resolution for lat = 0.001\n",
```

```
"resolution for lon = 0.001\n",
```

```
"resolution for alt = 1" ;
```

```
:sensor_height = "Sensors heights (above base \alt\):\n",
"60m for temp_60m, rh_60m, and vap_pres_60m;\n",
"25m for temp_25m, rh_25m, and vap_pres_25m." ;
```

```
:qc_method = "Standard Mentor QC" ;
```

```
:qc_standards_version = "1.0" ;
```

:qc_comment = "The QC field values are a bit packed representation of true/false values for the tests that may have been performed. A QC value of zero means that none of the tests performed on the value failed.\n",

```
"\n",
```

"The QC field values make use of the internal binary format to store the results of the individual QC tests. This allows the representation of multiple QC states in a single value. If the test associated with a particular bit fails the bit is turned on. Turning on the bit equates to adding the integer value of the failed test to the current value of the field. The QC field's value can be interpreted by applying bit logic using bitwise operators, or by examining the QC value's integer representation. A QC field's integer representation is the sum of the individual integer values of the failed tests. The bit and integer equivalents for the first 5 bits are listed below:\n",

```
"\n",
```

```
"bit_1 = 00000001 = 0x01 = 2^0 = 1\n",
```

```
"bit_2 = 00000010 = 0x02 = 2^1 = 2\n",
```

```
"bit_3 = 00000100 = 0x04 = 2^2 = 4\n",
```

```
"bit_4 = 00001000 = 0x08 = 2^3 = 8\n",
```

```

        "bit_5 = 00010000 = 0x10 = 2^4 = 16" ;
:qc_bit_1_description = "Value is equal to missing_value." ;
:qc_bit_1_assessment = "Bad" ;
:qc_bit_2_description = "Value is less than the valid_min." ;
:qc_bit_2_assessment = "Bad" ;
:qc_bit_3_description = "Value is greater than the valid_max." ;
:qc_bit_3_assessment = "Bad" ;
:qc_bit_4_description = "Difference between current and previous values exceeds
valid_delta." ;
:qc_bit_4_assessment = "Indeterminate" ;
:history = "created by user dsmgr on machine ruby at 2018-04-14 00:12:00, using ingest-
twr-8.4-0.el6" ;
}

```

sgp1twr25mC1.b1:

```

netcdf sgp1twr25mC1.b1.20180414.000000 {
dimensions:
    time = UNLIMITED ; // (1440 currently)
variables:
    int base_time ;
        base_time:string = "2018-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2018-04-14 00:00:00 0:00" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2018-04-14 00:00:00 0:00" ;
    int qc_time(time) ;
        qc_time:long_name = "Quality check results on field: Time offset from midnight" ;
        qc_time:units = "unitless" ;
        qc_time:description = "This field contains bit packed values which should be interpreted
as listed. No bits set (zero) represents good data." ;
        qc_time:bit_1_description = "Delta time between current and previous samples is zero." ;
        qc_time:bit_1_assessment = "Indeterminate" ;
        qc_time:bit_2_description = "Delta time between current and previous samples is less
than the delta_t_lower_limit field attribute." ;
        qc_time:bit_2_assessment = "Indeterminate" ;
        qc_time:bit_3_description = "Delta time between current and previous samples is greater
than the delta_t_upper_limit field attribute." ;
        qc_time:bit_3_assessment = "Indeterminate" ;
        qc_time:delta_t_lower_limit = 57. ;

```

qc_time:delta_t_upper_limit = 63. ;
 qc_time:prior_sample_flag = 1 ;
 qc_time:comment = "If the '\prior_sample_flag\' is set the first sample time from a new raw file will be compared against the time just previous to it in the stored data. If it is not set the qc_time value for the first sample will be set to 0." ;

```
float temp(time) ;
  temp:long_name = "Air temperature" ;
  temp:units = "C" ;
  temp:valid_min = -50.f ;
  temp:valid_max = 50.f ;
  temp:resolution = 0.01f ;
  temp:missing_value = -9999.f ;
int qc_temp(time) ;
  qc_temp:long_name = "Quality check results on field: Air temperature" ;
  qc_temp:units = "unitless" ;
  qc_temp:description = "See global attributes for individual bit descriptions." ;
float rh(time) ;
  rh:long_name = "Relative humidity" ;
  rh:units = "%" ;
  rh:valid_min = 0.f ;
  rh:valid_max = 102.f ;
  rh:resolution = 0.1f ;
  rh:missing_value = -9999.f ;
int qc_rh(time) ;
  qc_rh:long_name = "Quality check results on field: Relative humidity" ;
  qc_rh:units = "unitless" ;
  qc_rh:description = "See global attributes for individual bit descriptions." ;
float vap_pres(time) ;
  vap_pres:long_name = "Vapor pressure" ;
  vap_pres:units = "kPa" ;
  vap_pres:valid_min = 0.f ;
  vap_pres:valid_max = 10.f ;
  vap_pres:resolution = 0.001f ;
  vap_pres:missing_value = -9999.f ;
int qc_vap_pres(time) ;
  qc_vap_pres:long_name = "Quality check results on field: Vapor pressure" ;
  qc_vap_pres:units = "unitless" ;
  qc_vap_pres:description = "See global attributes for individual bit descriptions." ;
float aspirator(time) ;
  aspirator:long_name = "Aspirator flow status (% of time with proper flow)" ;
  aspirator:units = "%" ;
  aspirator:valid_min = 90.f ;
  aspirator:valid_max = 100.f ;
  aspirator:resolution = 0.1f ;
  aspirator:missing_value = -9999.f ;
int qc_aspirator(time) ;
```

```

    qc_aspirator:long_name = "Quality check results on field: Aspirator flow status (% of
time with proper flow)" ;
    qc_aspirator:units = "unitless" ;
    qc_aspirator:description = "See global attributes for individual bit descriptions." ;
float vbat(time) ;
    vbat:long_name = "Battery voltage" ;
    vbat:units = "V" ;
    vbat:valid_min = 10.4f ;
    vbat:valid_max = 15.f ;
    vbat:resolution = 0.01f ;
    vbat:missing_value = -9999.f ;
    vbat:comment = "This is the voltage of the CR10X battery for QA/QC purposes, prior to
19:30 GMT on 10/30/2000 this was a CR21X logger" ;
int qc_vbat(time) ;
    qc_vbat:long_name = "Quality check results on field: Battery voltage" ;
    qc_vbat:units = "unitless" ;
    qc_vbat:description = "See global attributes for individual bit descriptions." ;
float lat ;
    lat:long_name = "North latitude" ;
    lat:units = "degree_N" ;
    lat:valid_min = -90.f ;
    lat:valid_max = 90.f ;
float lon ;
    lon:long_name = "East longitude" ;
    lon:units = "degree_E" ;
    lon:valid_min = -180.f ;
    lon:valid_max = 180.f ;
float alt ;
    alt:long_name = "Altitude above mean sea level" ;
    alt:units = "m" ;

// global attributes:
:command_line = "twr_ingest -n twr25m -s sgp -f C1" ;
:process_version = "ingest-twr-8.4-0.e16" ;
:dod_version = "1twr25m-b1-1.1" ;
:data_level = "b1" ;
:input_source = "/data/collection/sgp/sgptwr25mC1.00/1523660880.icm" ;
:site_id = "sgp" ;
:facility_id = "C1: Lamont, Oklahoma" ;
:datastream = "sgp1twr25mC1.b1" ;
:sampling_interval = "1 second" ;
:averaging_interval = "1 minute" ;
:serial_number = "N/A" ;
:comment = "The time assigned to each data point indicates the end of any period of
averaging of the geophysical data." ;

```

:resolution_description = "The resolution field attributes refer to the number of significant digits relative to the decimal point that should be used in calculations. Using fewer digits might result in greater uncertainty. Using a larger number of digits should have no effect and thus is unnecessary. However, analyses based on differences in values with a larger number of significant digits than indicated could lead to erroneous results or misleading scientific conclusions.\n",

"\n",

"resolution for lat = 0.001\n",

"resolution for lon = 0.001\n",

"resolution for alt = 1" ;

:sensor_height = "Sensors heights (above base \alt\): 25m for temp, rh, vap_pres and aspirator." ;

:qc_method = "Standard Mentor QC" ;

:qc_standards_version = "1.0" ;

:qc_comment = "The QC field values are a bit packed representation of true/false values for the tests that may have been performed. A QC value of zero means that none of the tests performed on the value failed.\n",

"\n",

"The QC field values make use of the internal binary format to store the results of the individual QC tests. This allows the representation of multiple QC states in a single value. If the test associated with a particular bit fails the bit is turned on. Turning on the bit equates to adding the integer value of the failed test to the current value of the field. The QC field's value can be interpreted by applying bit logic using bitwise operators, or by examining the QC value's integer representation. A QC field's integer representation is the sum of the individual integer values of the failed tests. The bit and integer equivalents for the first 5 bits are listed below:\n",

"\n",

"bit_1 = 00000001 = 0x01 = 2^0 = 1\n",

"bit_2 = 00000010 = 0x02 = 2^1 = 2\n",

"bit_3 = 00000100 = 0x04 = 2^2 = 4\n",

"bit_4 = 00001000 = 0x08 = 2^3 = 8\n",

"bit_5 = 00010000 = 0x10 = 2^4 = 16" ;

:qc_bit_1_description = "Value is equal to missing_value." ;

:qc_bit_1_assessment = "Bad" ;

:qc_bit_2_description = "Value is less than the valid_min." ;

:qc_bit_2_assessment = "Bad" ;

:qc_bit_3_description = "Value is greater than the valid_max." ;

:qc_bit_3_assessment = "Bad" ;

:qc_bit_4_description = "Difference between current and previous values exceeds valid_delta." ;

:qc_bit_4_assessment = "Indeterminate" ;

:history = "created by user dsmgr on machine ruby at 2018-04-14 00:12:00, using ingest-twr-8.4-0.el6" ;

}

sgp1twr60mC1.b1:


```

netcdf sgp1twr60mC1.b1.20180414.000000 {
dimensions:
    time = UNLIMITED ; // (1440 currently)
variables:
    int base_time ;
        base_time:string = "2018-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2018-04-14 00:00:00 0:00" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2018-04-14 00:00:00 0:00" ;
    int qc_time(time) ;
        qc_time:long_name = "Quality check results on field: Time offset from midnight" ;
        qc_time:units = "unitless" ;
        qc_time:description = "This field contains bit packed values which should be interpreted
as listed. No bits set (zero) represents good data." ;
        qc_time:bit_1_description = "Delta time between current and previous samples is zero." ;
        qc_time:bit_1_assessment = "Indeterminate" ;
        qc_time:bit_2_description = "Delta time between current and previous samples is less
than the delta_t_lower_limit field attribute." ;
        qc_time:bit_2_assessment = "Indeterminate" ;
        qc_time:bit_3_description = "Delta time between current and previous samples is greater
than the delta_t_upper_limit field attribute." ;
        qc_time:bit_3_assessment = "Indeterminate" ;
        qc_time:delta_t_lower_limit = 57. ;
        qc_time:delta_t_upper_limit = 63. ;
        qc_time:prior_sample_flag = 1 ;
        qc_time:comment = "If the '\prior_sample_flag\' is set the first sample time from a new
raw file will be compared against the time just previous to it in the stored data. If it is not set the qc_time
value for the first sample will be set to 0." ;
    float temp(time) ;
        temp:long_name = "Air temperature" ;
        temp:units = "C" ;
        temp:valid_min = -50.f ;
        temp:valid_max = 50.f ;
        temp:resolution = 0.01f ;
        temp:missing_value = -9999.f ;
    int qc_temp(time) ;
        qc_temp:long_name = "Quality check results on field: Air temperature" ;
        qc_temp:units = "unitless" ;
        qc_temp:description = "See global attributes for individual bit descriptions." ;
    float rh(time) ;
        rh:long_name = "Relative humidity" ;

```

```

    rh:units = "%" ;
    rh:valid_min = 0.f ;
    rh:valid_max = 102.f ;
    rh:resolution = 0.1f ;
    rh:missing_value = -9999.f ;
int qc_rh(time) ;
    qc_rh:long_name = "Quality check results on field: Relative humidity" ;
    qc_rh:units = "unitless" ;
    qc_rh:description = "See global attributes for individual bit descriptions." ;
float vap_pres(time) ;
    vap_pres:long_name = "Vapor pressure" ;
    vap_pres:units = "kPa" ;
    vap_pres:valid_min = 0.f ;
    vap_pres:valid_max = 10.f ;
    vap_pres:resolution = 0.001f ;
    vap_pres:missing_value = -9999.f ;
int qc_vap_pres(time) ;
    qc_vap_pres:long_name = "Quality check results on field: Vapor pressure" ;
    qc_vap_pres:units = "unitless" ;
    qc_vap_pres:description = "See global attributes for individual bit descriptions." ;
float aspirator(time) ;
    aspirator:long_name = "Aspirator flow status (% of time with proper flow)" ;
    aspirator:units = "%" ;
    aspirator:valid_min = 90.f ;
    aspirator:valid_max = 100.f ;
    aspirator:resolution = 0.1f ;
    aspirator:missing_value = -9999.f ;
int qc_aspirator(time) ;
    qc_aspirator:long_name = "Quality check results on field: Aspirator flow status (% of
time with proper flow)" ;
    qc_aspirator:units = "unitless" ;
    qc_aspirator:description = "See global attributes for individual bit descriptions." ;
float vbat(time) ;
    vbat:long_name = "Battery voltage" ;
    vbat:units = "V" ;
    vbat:valid_min = 10.4f ;
    vbat:valid_max = 15.f ;
    vbat:resolution = 0.01f ;
    vbat:missing_value = -9999.f ;
    vbat:comment = "This is the voltage of the CR10X battery for QA/QC purposes, prior to
19:30 GMT on 10/30/2000 this was a CR21X logger" ;
int qc_vbat(time) ;
    qc_vbat:long_name = "Quality check results on field: Battery voltage" ;
    qc_vbat:units = "unitless" ;
    qc_vbat:description = "See global attributes for individual bit descriptions." ;
float lat ;

```

```

lat:long_name = "North latitude" ;
lat:units = "degree_N" ;
lat:valid_min = -90.f ;
lat:valid_max = 90.f ;
float lon ;
lon:long_name = "East longitude" ;
lon:units = "degree_E" ;
lon:valid_min = -180.f ;
lon:valid_max = 180.f ;
float alt ;
alt:long_name = "Altitude above mean sea level" ;
alt:units = "m" ;

// global attributes:
:command_line = "twr_ingest -n twr60m -s sgp -f C1" ;
:process_version = "ingest-twr-8.4-0.el6" ;
:dod_version = "1twr60m-b1-1.1" ;
:data_level = "b1" ;
:input_source = "/data/collection/sgp/sgptwr60mC1.00/1523660940.icm" ;
:site_id = "sgp" ;
:facility_id = "C1: Lamont, Oklahoma" ;
:datastream = "sgp1twr60mC1.b1" ;
:sampling_interval = "1 second" ;
:averaging_interval = "1 minute" ;
:serial_number = "N/A" ;
:comment = "The time assigned to each data point indicates the end of any period of\n",
           "averaging of the geophysical data." ;
:resolution_description = "The resolution field attributes refer to the number of
significant digits relative to the decimal point that should be used in calculations. Using fewer digits
might result in greater uncertainty. Using a larger number of digits should have no effect and thus is
unnecessary. However, analyses based on differences in values with a larger number of significant digits
than indicated could lead to erroneous results or misleading scientific conclusions.\n",
           "\n",
           "resolution for lat = 0.001\n",
           "resolution for lon = 0.001\n",
           "resolution for alt = 1" ;
:sensor_height = "Sensors heights (above base '\alt\'): 60m for temp, rh, vap_pres and
aspirator." ;
:qc_method = "Standard Mentor QC" ;
:qc_standards_version = "1.0" ;
:qc_comment = "The QC field values are a bit packed representation of true/false values
for the tests that may have been performed. A QC value of zero means that none of the tests performed on
the value failed.\n",
           "\n",
           "The QC field values make use of the internal binary format to store the results of
the individual QC tests. This allows the representation of multiple QC states in a single value. If the test

```

associated with a particular bit fails the bit is turned on. Turning on the bit equates to adding the integer value of the failed test to the current value of the field. The QC field's value can be interpreted by applying bit logic using bitwise operators, or by examining the QC value's integer representation. A QC field's integer representation is the sum of the individual integer values of the failed tests. The bit and integer equivalents for the first 5 bits are listed below:\n",

```

        "\n",
        "bit_1 = 00000001 = 0x01 = 2^0 = 1\n",
        "bit_2 = 00000010 = 0x02 = 2^1 = 2\n",
        "bit_3 = 00000100 = 0x04 = 2^2 = 4\n",
        "bit_4 = 00001000 = 0x08 = 2^3 = 8\n",
        "bit_5 = 00010000 = 0x10 = 2^4 = 16" ;
:qc_bit_1_description = "Value is equal to missing_value." ;
:qc_bit_1_assessment = "Bad" ;
:qc_bit_2_description = "Value is less than the valid_min." ;
:qc_bit_2_assessment = "Bad" ;
:qc_bit_3_description = "Value is greater than the valid_max." ;
:qc_bit_3_assessment = "Bad" ;
:qc_bit_4_description = "Difference between current and previous values exceeds
valid_delta." ;
:qc_bit_4_assessment = "Indeterminate" ;
:history = "created by user dsmgr on machine ruby at 2018-04-14 00:12:00, using ingest-
twr-8.4-0.el6" ;
}

```

sgpthwapsC1.b1:

```

netcdf sgpthwapsC1.b1.20140414.000000 {
dimensions:
    time = UNLIMITED ; // (288 currently)
variables:
    int base_time ;
        base_time:string = "2014-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
        base_time:ancillary_variables = "time_offset" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2014-04-14 00:00:00 0:00" ;
        time_offset:ancillary_variables = "base_time" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2014-04-14 00:00:00 0:00" ;
        time:standard_name = "time" ;
    float pres(time) ;

```

```

pres:long_name = "Atmospheric pressure" ;
pres:units = "hPa" ;
pres:standard_name = "surface_air_pressure" ;
pres:valid_min = 800.f ;
pres:valid_max = 1100.f ;
pres:resolution = 0.1f ;
pres:missing_value = -9999.f ;
int qc_pres(time) ;
qc_pres:long_name = "Quality check results on field: Atmospheric pressure" ;
qc_pres:units = "unitless" ;
qc_pres:description = "See global attributes for individual bit descriptions." ;
float temp(time) ;
temp:long_name = "Temperature" ;
temp:units = "degC" ;
temp:standard_name = "air_temperature" ;
temp:valid_min = -40.f ;
temp:valid_max = 50.f ;
temp:resolution = 0.1f ;
temp:missing_value = -9999.f ;
int qc_temp(time) ;
qc_temp:long_name = "Quality check results on field: Temperature" ;
qc_temp:units = "unitless" ;
qc_temp:description = "See global attributes for individual bit descriptions." ;
float rh(time) ;
rh:long_name = "Relative humidity" ;
rh:units = "%" ;
rh:standard_name = "relative_humidity" ;
rh:valid_min = -2.f ;
rh:valid_max = 104.f ;
rh:resolution = 1.f ;
rh:missing_value = -9999.f ;
int qc_rh(time) ;
qc_rh:long_name = "Quality check results on field: Relative humidity" ;
qc_rh:units = "unitless" ;
qc_rh:description = "See global attributes for individual bit descriptions." ;
float vap_pres(time) ;
vap_pres:long_name = "Vapor pressure" ;
vap_pres:units = "kPa" ;
vap_pres:standard_name = "water_vapor_partial_pressure_in_air" ;
vap_pres:valid_min = 0.f ;
vap_pres:valid_max = 10.f ;
vap_pres:resolution = 0.001f ;
vap_pres:missing_value = -9999.f ;
int qc_vap_pres(time) ;
qc_vap_pres:long_name = "Quality check results on field: Vapor pressure" ;
qc_vap_pres:units = "unitless" ;

```

```

    qc_vap_pres:description = "See global attributes for individual bit descriptions." ;
float wspd(time) ;
    wspd:long_name = "Mean wind speed" ;
    wspd:units = "m/s" ;
    wspd:standard_name = "wind_speed" ;
    wspd:valid_min = 0.f ;
    wspd:valid_max = 45.f ;
    wspd:resolution = 0.1f ;
    wspd:missing_value = -9999.f ;
int qc_wspd(time) ;
    qc_wspd:long_name = "Quality check results on field: Mean wind speed" ;
    qc_wspd:units = "unitless" ;
    qc_wspd:description = "See global attributes for individual bit descriptions." ;
float wdir(time) ;
    wdir:long_name = "Unit vector wind direction" ;
    wdir:units = "degree" ;
    wdir:standard_name = "wind_from_direction" ;
    wdir:valid_min = 0.f ;
    wdir:valid_max = 360.f ;
    wdir:resolution = 1.f ;
    wdir:missing_value = -9999.f ;
int qc_wdir(time) ;
    qc_wdir:long_name = "Quality check results on field: Unit vector wind direction" ;
    qc_wdir:units = "unitless" ;
    qc_wdir:description = "See global attributes for individual bit descriptions." ;
float sd_wdir(time) ;
    sd_wdir:long_name = "Standard deviation of wind direction" ;
    sd_wdir:units = "degree" ;
    sd_wdir:valid_min = 0.f ;
    sd_wdir:valid_max = 90.f ;
    sd_wdir:resolution = 1.f ;
    sd_wdir:missing_value = -9999.f ;
int qc_sd_wdir(time) ;
    qc_sd_wdir:long_name = "Quality check results on field: Standard deviation of wind
direction" ;
    qc_sd_wdir:units = "unitless" ;
    qc_sd_wdir:description = "See global attributes for individual bit descriptions." ;
float sd_pres(time) ;
    sd_pres:long_name = "Standard deviation of atmospheric pressure" ;
    sd_pres:units = "hPa" ;
    sd_pres:valid_min = 0.f ;
    sd_pres:resolution = 0.1f ;
    sd_pres:missing_value = -9999.f ;
int qc_sd_pres(time) ;
    qc_sd_pres:long_name = "Quality check results on field: Standard deviation of
atmospheric pressure" ;

```

```

    qc_sd_pres:units = "unitless" ;
    qc_sd_pres:description = "See global attributes for individual bit descriptions." ;
float sd_temp(time) ;
    sd_temp:long_name = "Standard deviation of temperature" ;
    sd_temp:units = "degC" ;
    sd_temp:valid_min = 0.f ;
    sd_temp:valid_max = 2.f ;
    sd_temp:resolution = 0.1f ;
    sd_temp:missing_value = -9999.f ;
int qc_sd_temp(time) ;
    qc_sd_temp:long_name = "Quality check results on field: Standard deviation of
temperature" ;
    qc_sd_temp:units = "unitless" ;
    qc_sd_temp:description = "See global attributes for individual bit descriptions." ;
float sd_rh(time) ;
    sd_rh:long_name = "Standard deviation of relative humidity" ;
    sd_rh:units = "%" ;
    sd_rh:valid_min = 0.f ;
    sd_rh:valid_max = 20.f ;
    sd_rh:resolution = 1.f ;
    sd_rh:missing_value = -9999.f ;
int qc_sd_rh(time) ;
    qc_sd_rh:long_name = "Quality check results on field: Standard deviation of relative
humidity" ;
    qc_sd_rh:units = "unitless" ;
    qc_sd_rh:description = "See global attributes for individual bit descriptions." ;
float sd_vap_pres(time) ;
    sd_vap_pres:long_name = "Standard deviation of vapor pressure" ;
    sd_vap_pres:units = "kPa" ;
    sd_vap_pres:valid_min = 0.f ;
    sd_vap_pres:resolution = 0.001f ;
    sd_vap_pres:missing_value = -9999.f ;
int qc_sd_vap_pres(time) ;
    qc_sd_vap_pres:long_name = "Quality check results on field: Standard deviation of
vapor pressure" ;
    qc_sd_vap_pres:units = "unitless" ;
    qc_sd_vap_pres:description = "See global attributes for individual bit descriptions." ;
float vbat(time) ;
    vbat:long_name = "Battery voltage" ;
    vbat:units = "V" ;
    vbat:valid_min = 9.6f ;
    vbat:valid_max = 16.f ;
    vbat:resolution = 0.01f ;
    vbat:missing_value = -9999.f ;
int qc_vbat(time) ;
    qc_vbat:long_name = "Quality check results on field: Battery voltage" ;

```

```

qc_vbat:units = "unitless" ;
qc_vbat:description = "See global attributes for individual bit descriptions." ;
float lat ;
lat:long_name = "North latitude" ;
lat:units = "degree_N" ;
lat:standard_name = "latitude" ;
lat:valid_min = -90.f ;
lat:valid_max = 90.f ;
float lon ;
lon:long_name = "East longitude" ;
lon:units = "degree_E" ;
lon:standard_name = "longitude" ;
lon:valid_min = -180.f ;
lon:valid_max = 180.f ;
float alt ;
alt:long_name = "Altitude above mean sea level" ;
alt:units = "m" ;
alt:standard_name = "altitude" ;

// global attributes:
:command_line = "thwaps_ingest -s sgp -f C1" ;
:process_version = "ingest-thwaps-8.2-0.e16" ;
:dod_version = "thwaps-b1-2.1" ;
:input_source = "/data/collection/sgp/sgpthwapsC1.00/1397430600.icm" ;
:site_id = "sgp" ;
:platform_id = "thwaps" ;
:facility_id = "C1" ;
:data_level = "b1" ;
:location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
:datastream = "sgpthwapsC1.b1" ;
:resolution_description = "The resolution field attributes refer to the number of
significant digits relative to the decimal point that should be used in calculations. Using fewer digits
might result in greater uncertainty. Using a larger number of digits should have no effect and thus is
unnecessary. However, analyses based on differences in values with a larger number of significant digits
than indicated could lead to erroneous results or misleading scientific conclusions.\n",
"\n",
"resolution for lat = 0.001\n",
"resolution for lon = 0.001\n",
"resolution for alt = 1" ;
:sampling_interval = "3 seconds" ;
:averaging_interval = "5 minutes" ;
:averaging_interval_comment = "The time assigned to each data point indicates the end
of the averaging interval." ;
:comment = "The wind speed is the 'mean horizontal wind speed' which is
(1/N)*SUM(Si).\n",
"\n",

```


"The wind direction is the \unit vector wind direction\ which is defined as $\text{ARCTAN}(U_x/U_y)$, where $U_x = \text{SUM}(\text{Sin}(\text{THETA}_i))/N$ and $U_y = \text{SUM}(\text{Cos}(\text{THETA}_i))/N$.\n",
 "\n",

"The standard deviation of wind direction is computed using the Yamartino algorithm in which $\text{StdDevTHETA} = \text{ARCSIN}(\text{epsilon}) * [1 + 0.1547 * \text{epsilon} ** 3]$, where $\text{epsilon} = \text{SQRT}[1 - (U_x ** 2 + U_y ** 2)]$ " ;

:qc_bit_comment = "The QC field values are a bit packed representation of true/false values for the tests that may have been performed. A QC value of zero means that none of the tests performed on the value failed." ;

:qc_bit_1_description = "Value is equal to missing_value." ;

:qc_bit_1_assessment = "Bad" ;

:qc_bit_2_description = "Value is less than the valid_min." ;

:qc_bit_2_assessment = "Bad" ;

:qc_bit_3_description = "Value is greater than the valid_max." ;

:qc_bit_3_assessment = "Bad" ;

:qc_bit_4_description = "Difference between current and previous values exceeds valid_delta." ;

:qc_bit_4_assessment = "Indeterminate" ;

:history = "created by user dsmgr on machine tin at 2014-04-14 01:11:01, using ingest-thwaps-8.2-0.el6" ;

}

sgpmawsC1.b1:

netcdf sgpmaawsC1.b1.20210414.000007 {

dimensions:

time = UNLIMITED ; // (1432 currently)

bound = 2 ;

variables:

int base_time ;

base_time:string = "2021-04-14 00:00:00 0:00" ;

base_time:long_name = "Base time in Epoch" ;

base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;

base_time:ancillary_variables = "time_offset" ;

double time_offset(time) ;

time_offset:long_name = "Time offset from base_time" ;

time_offset:units = "seconds since 2021-04-14 00:00:00 0:00" ;

time_offset:ancillary_variables = "base_time" ;

double time(time) ;

time:long_name = "Time offset from midnight" ;

time:units = "seconds since 2021-04-14 00:00:00 0:00" ;

time:bounds = "time_bounds" ;

time:calendar = "gregorian" ;

time:standard_name = "time" ;

```

double time_bounds(time, bound) ;
    time_bounds:long_name = "Time cell bounds" ;
    time_bounds:bound_offsets = -60., 0. ;
float atmospheric_pressure(time) ;
    atmospheric_pressure:long_name = "Atmospheric pressure" ;
    atmospheric_pressure:units = "hPa" ;
    atmospheric_pressure:valid_min = 800.f ;
    atmospheric_pressure:valid_max = 1100.f ;
    atmospheric_pressure:missing_value = -9999.f ;
    atmospheric_pressure:cell_methods = "time: mean" ;
    atmospheric_pressure:standard_name = "air_pressure" ;
    atmospheric_pressure:ancillary_variables = "qc_atmospheric_pressure" ;
int qc_atmospheric_pressure(time) ;
    qc_atmospheric_pressure:long_name = "Quality check results on field: Atmospheric
pressure" ;
    qc_atmospheric_pressure:units = "unitless" ;
    qc_atmospheric_pressure:description = "This field contains bit packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_atmospheric_pressure:flag_method = "bit" ;
    qc_atmospheric_pressure:bit_1_description = "Value is equal to missing_value." ;
    qc_atmospheric_pressure:bit_1_assessment = "Bad" ;
    qc_atmospheric_pressure:bit_2_description = "Value is less than the valid_min." ;
    qc_atmospheric_pressure:bit_2_assessment = "Bad" ;
    qc_atmospheric_pressure:bit_3_description = "Value is greater than the valid_max." ;
    qc_atmospheric_pressure:bit_3_assessment = "Bad" ;
float atmospheric_temperature(time) ;
    atmospheric_temperature:long_name = "Dry bulb temperature" ;
    atmospheric_temperature:units = "degC" ;
    atmospheric_temperature:ancillary_variables = "qc_atmospheric_temperature" ;
    atmospheric_temperature:valid_min = -40.f ;
    atmospheric_temperature:valid_max = 50.f ;
    atmospheric_temperature:missing_value = -9999.f ;
    atmospheric_temperature:cell_methods = "time: mean" ;
    atmospheric_temperature:standard_name = "air_temperature" ;
int qc_atmospheric_temperature(time) ;
    qc_atmospheric_temperature:long_name = "Quality check results on field: Dry bulb
temperature" ;
    qc_atmospheric_temperature:units = "unitless" ;
    qc_atmospheric_temperature:description = "This field contains bit packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_atmospheric_temperature:flag_method = "bit" ;
    qc_atmospheric_temperature:bit_1_description = "Value is equal to missing_value." ;
    qc_atmospheric_temperature:bit_1_assessment = "Bad" ;
    qc_atmospheric_temperature:bit_2_description = "Value is less than the valid_min." ;

```

```

qc_atmospheric_temperature:bit_2_assessment = "Bad" ;
qc_atmospheric_temperature:bit_3_description = "Value is greater than the valid_max." ;
qc_atmospheric_temperature:bit_3_assessment = "Bad" ;
float atmospheric_relative_humidity(time) ;
atmospheric_relative_humidity:long_name = "Atmospheric relative humidity" ;
atmospheric_relative_humidity:units = "%" ;
atmospheric_relative_humidity:valid_min = -2.f ;
atmospheric_relative_humidity:valid_max = 104.f ;
atmospheric_relative_humidity:missing_value = -9999.f ;
atmospheric_relative_humidity:cell_methods = "time: mean" ;
atmospheric_relative_humidity:standard_name = "relative_humidity" ;
atmospheric_relative_humidity:ancillary_variables =
"qc_atmospheric_relative_humidity" ;
int qc_atmospheric_relative_humidity(time) ;
qc_atmospheric_relative_humidity:long_name = "Quality check results on field:
Atmospheric relative humidity" ;
qc_atmospheric_relative_humidity:units = "unitless" ;
qc_atmospheric_relative_humidity:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_atmospheric_relative_humidity:flag_method = "bit" ;
qc_atmospheric_relative_humidity:bit_1_description = "Value is equal to
missing_value." ;
qc_atmospheric_relative_humidity:bit_1_assessment = "Bad" ;
qc_atmospheric_relative_humidity:bit_2_description = "Value is less than the
valid_min." ;
qc_atmospheric_relative_humidity:bit_2_assessment = "Bad" ;
qc_atmospheric_relative_humidity:bit_3_description = "Value is greater than the
valid_max." ;
qc_atmospheric_relative_humidity:bit_3_assessment = "Bad" ;
float atmospheric_dew_point(time) ;
atmospheric_dew_point:long_name = "Atmospheric dew point temperature" ;
atmospheric_dew_point:units = "degC" ;
atmospheric_dew_point:valid_min = -75.f ;
atmospheric_dew_point:valid_max = 51.f ;
atmospheric_dew_point:missing_value = -9999.f ;
atmospheric_dew_point:cell_methods = "time: mean" ;
atmospheric_dew_point:standard_name = "dew_point_temperature" ;
atmospheric_dew_point:ancillary_variables = "qc_atmospheric_dew_point" ;
int qc_atmospheric_dew_point(time) ;
qc_atmospheric_dew_point:long_name = "Quality check results on field: Atmospheric
dew point temperature" ;
qc_atmospheric_dew_point:units = "unitless" ;
qc_atmospheric_dew_point:description = "This field contains bit packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;

```

```

qc_atmospheric_dew_point:flag_method = "bit" ;
qc_atmospheric_dew_point:bit_1_description = "Value is equal to missing_value." ;
qc_atmospheric_dew_point:bit_1_assessment = "Bad" ;
qc_atmospheric_dew_point:bit_2_description = "Value is less than the valid_min." ;
qc_atmospheric_dew_point:bit_2_assessment = "Bad" ;
qc_atmospheric_dew_point:bit_3_description = "Value is greater than the valid_max." ;
qc_atmospheric_dew_point:bit_3_assessment = "Bad" ;
float wind_speed(time) ;
wind_speed:long_name = "Wind speed" ;
wind_speed:units = "m/s" ;
wind_speed:valid_min = 0.f ;
wind_speed:valid_max = 60.f ;
wind_speed:missing_value = -9999.f ;
wind_speed:cell_methods = "time: mean (arithmetic)" ;
wind_speed:standard_name = "wind_speed" ;
wind_speed:ancillary_variables = "qc_wind_speed" ;
int qc_wind_speed(time) ;
qc_wind_speed:long_name = "Quality check results on field: Wind speed" ;
qc_wind_speed:units = "unitless" ;
qc_wind_speed:description = "This field contains bit packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_wind_speed:flag_method = "bit" ;
qc_wind_speed:bit_1_description = "Value is equal to missing_value." ;
qc_wind_speed:bit_1_assessment = "Bad" ;
qc_wind_speed:bit_2_description = "Value is less than the valid_min." ;
qc_wind_speed:bit_2_assessment = "Bad" ;
qc_wind_speed:bit_3_description = "Value is greater than the valid_max." ;
qc_wind_speed:bit_3_assessment = "Bad" ;
qc_wind_speed:bit_4_description = "Outlier detected: deviation from median of 30
minute window > 12.5 * MAD" ;
qc_wind_speed:bit_4_assessment = "Bad" ;
float wind_direction(time) ;
wind_direction:long_name = "Wind direction" ;
wind_direction:units = "degree" ;
wind_direction:valid_min = 0.f ;
wind_direction:valid_max = 360.f ;
wind_direction:missing_value = -9999.f ;
wind_direction:cell_methods = "time: mean (arithmetic)" ;
wind_direction:standard_name = "wind_from_direction" ;
wind_direction:ancillary_variables = "qc_wind_direction" ;
int qc_wind_direction(time) ;
qc_wind_direction:long_name = "Quality check results on field: Wind direction" ;
qc_wind_direction:units = "unitless" ;

```

qc_wind_direction:description = "This field contains bit packed integer values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;

qc_wind_direction:flag_method = "bit" ;

qc_wind_direction:bit_1_description = "Value is equal to missing_value." ;

qc_wind_direction:bit_1_assessment = "Bad" ;

qc_wind_direction:bit_2_description = "Value is less than the valid_min." ;

qc_wind_direction:bit_2_assessment = "Bad" ;

qc_wind_direction:bit_3_description = "Value is greater than the valid_max." ;

qc_wind_direction:bit_3_assessment = "Bad" ;

float lat ;

lat:long_name = "North latitude" ;

lat:units = "degree_N" ;

lat:valid_min = -90.f ;

lat:valid_max = 90.f ;

lat:standard_name = "latitude" ;

float lon ;

lon:long_name = "East longitude" ;

lon:units = "degree_E" ;

lon:valid_min = -180.f ;

lon:valid_max = 180.f ;

lon:standard_name = "longitude" ;

float alt ;

alt:long_name = "Altitude above mean sea level" ;

alt:units = "m" ;

alt:standard_name = "altitude" ;

// global attributes:

:command_line = "maws_ingest -s sgp -f C1" ;

:Conventions = "ARM-1.0" ;

:process_version = "ingest-maws-1.0-2.e17" ;

:dod_version = "maws-b1-1.0" ;

:input_source = "/data/collection/sgp/sgpmawsC1.00/LOG1_20210413.txt" ;

:site_id = "sgp" ;

:platform_id = "maws" ;

:facility_id = "C1" ;

:data_level = "b1" ;

:location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;

:datastream = "sgpmawsC1.b1" ;

:doi = "10.5439/1162061" ;

:history = "created by user dsmgr on machine zinc at 2021-04-14 00:30:01, using ingest-maws-1.0-2.e17" ;

}

sgptowermetC1.b1:

```

netcdf sgptowermetC1.b1.20210414.000000 {
dimensions:
    time = UNLIMITED ; // (1440 currently)
    bound = 2 ;
variables:
    int base_time ;
        base_time:string = "2021-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
        base_time:ancillary_variables = "time_offset" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2021-04-14 00:00:00 0:00" ;
        time_offset:ancillary_variables = "base_time" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2021-04-14 00:00:00 0:00" ;
        time:calendar = "gregorian" ;
        time:standard_name = "time" ;
        time:bounds = "time_bounds" ;
    double time_bounds(time, bound) ;
        time_bounds:long_name = "Time cell bounds" ;
        time_bounds:bound_offsets = 0., 60. ;
    float temperature_SE_25m_avg(time) ;
        temperature_SE_25m_avg:long_name = "Air temperature at southeast 25m tower" ;
        temperature_SE_25m_avg:units = "degC" ;
        temperature_SE_25m_avg:missing_value = -9999.f ;
        temperature_SE_25m_avg:valid_min = -7999.f ;
        temperature_SE_25m_avg:valid_max = 7999.f ;
        temperature_SE_25m_avg:ancillary_variables = "qc_temperature_SE_25m_avg" ;
        temperature_SE_25m_avg:standard_name = "air_temperature" ;
    int qc_temperature_SE_25m_avg(time) ;
        qc_temperature_SE_25m_avg:long_name = "Quality check results on field: Air
temperature at southeast 25m tower" ;
        qc_temperature_SE_25m_avg:units = "1" ;
        qc_temperature_SE_25m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
        qc_temperature_SE_25m_avg:fail_min = -50.f ;
        qc_temperature_SE_25m_avg:fail_max = 50.f ;
        qc_temperature_SE_25m_avg:flag_method = "bit" ;
        qc_temperature_SE_25m_avg:bit_1_description = "Value is equal to missing_value." ;
        qc_temperature_SE_25m_avg:bit_1_assessment = "Bad" ;
        qc_temperature_SE_25m_avg:bit_2_description = "Value is less than valid_min." ;
        qc_temperature_SE_25m_avg:bit_2_assessment = "Bad" ;

```

```

qc_temperature_SE_25m_avg:bit_3_description = "Value is greater than valid_max." ;
qc_temperature_SE_25m_avg:bit_3_assessment = "Bad" ;
qc_temperature_SE_25m_avg:bit_4_description = "Value is less than fail_min." ;
qc_temperature_SE_25m_avg:bit_4_assessment = "Bad" ;
qc_temperature_SE_25m_avg:bit_5_description = "Value is greater than fail_max." ;
qc_temperature_SE_25m_avg:bit_5_assessment = "Bad" ;
float temperature_SE_25m_std(time) ;
temperature_SE_25m_std:long_name = "Standard deviation of air temperature at
southeast 25m tower" ;
temperature_SE_25m_std:units = "degC" ;
temperature_SE_25m_std:missing_value = -9999.f ;
temperature_SE_25m_std:valid_min = -7999.f ;
temperature_SE_25m_std:valid_max = 7999.f ;
float relative_humidity_SE_25m_avg(time) ;
relative_humidity_SE_25m_avg:long_name = "Relative humidity at southeast 25m
tower" ;
relative_humidity_SE_25m_avg:units = "%" ;
relative_humidity_SE_25m_avg:missing_value = -9999.f ;
relative_humidity_SE_25m_avg:valid_min = -7999.f ;
relative_humidity_SE_25m_avg:valid_max = 7999.f ;
relative_humidity_SE_25m_avg:ancillary_variables =
"qc_relative_humidity_SE_25m_avg" ;
relative_humidity_SE_25m_avg:standard_name = "relative_humidity" ;
int qc_relative_humidity_SE_25m_avg(time) ;
qc_relative_humidity_SE_25m_avg:long_name = "Quality check results on field:
Relative humidity at southeast 25m tower" ;
qc_relative_humidity_SE_25m_avg:units = "1" ;
qc_relative_humidity_SE_25m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_relative_humidity_SE_25m_avg:fail_min = 0.f ;
qc_relative_humidity_SE_25m_avg:fail_max = 100.f ;
qc_relative_humidity_SE_25m_avg:flag_method = "bit" ;
qc_relative_humidity_SE_25m_avg:bit_1_description = "Value is equal to
missing_value." ;
qc_relative_humidity_SE_25m_avg:bit_1_assessment = "Bad" ;
qc_relative_humidity_SE_25m_avg:bit_2_description = "Value is less than valid_min." ;
qc_relative_humidity_SE_25m_avg:bit_2_assessment = "Bad" ;
qc_relative_humidity_SE_25m_avg:bit_3_description = "Value is greater than
valid_max." ;
qc_relative_humidity_SE_25m_avg:bit_3_assessment = "Bad" ;
qc_relative_humidity_SE_25m_avg:bit_4_description = "Value is less than fail_min." ;
qc_relative_humidity_SE_25m_avg:bit_4_assessment = "Bad" ;
qc_relative_humidity_SE_25m_avg:bit_5_description = "Value is greater than fail_max."
;
qc_relative_humidity_SE_25m_avg:bit_5_assessment = "Bad" ;

```

```

float relative_humidity_SE_25m_std(time) ;
    relative_humidity_SE_25m_std:long_name = "Standard deviation of relative humidity at
southeast 25m tower" ;
    relative_humidity_SE_25m_std:units = "%" ;
    relative_humidity_SE_25m_std:missing_value = -9999.f ;
    relative_humidity_SE_25m_std:valid_min = -7999.f ;
    relative_humidity_SE_25m_std:valid_max = 7999.f ;
float vapor_pressure_SE_25m_avg(time) ;
    vapor_pressure_SE_25m_avg:long_name = "Vapor pressure at southeast 25m tower" ;
    vapor_pressure_SE_25m_avg:units = "kPa" ;
    vapor_pressure_SE_25m_avg:missing_value = -9999.f ;
    vapor_pressure_SE_25m_avg:valid_min = -7999.f ;
    vapor_pressure_SE_25m_avg:valid_max = 7999.f ;
    vapor_pressure_SE_25m_avg:ancillary_variables = "qc_vapor_pressure_SE_25m_avg" ;
    vapor_pressure_SE_25m_avg:standard_name = "water_vapor_partial_pressure_in_air" ;
int qc_vapor_pressure_SE_25m_avg(time) ;
    qc_vapor_pressure_SE_25m_avg:long_name = "Quality check results on field: Vapor
pressure at southeast 25m tower" ;
    qc_vapor_pressure_SE_25m_avg:units = "1" ;
    qc_vapor_pressure_SE_25m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_vapor_pressure_SE_25m_avg:fail_min = 0.f ;
    qc_vapor_pressure_SE_25m_avg:fail_max = 10.f ;
    qc_vapor_pressure_SE_25m_avg:flag_method = "bit" ;
    qc_vapor_pressure_SE_25m_avg:bit_1_description = "Value is equal to missing_value."
;
    qc_vapor_pressure_SE_25m_avg:bit_1_assessment = "Bad" ;
    qc_vapor_pressure_SE_25m_avg:bit_2_description = "Value is less than valid_min." ;
    qc_vapor_pressure_SE_25m_avg:bit_2_assessment = "Bad" ;
    qc_vapor_pressure_SE_25m_avg:bit_3_description = "Value is greater than valid_max."
;
    qc_vapor_pressure_SE_25m_avg:bit_3_assessment = "Bad" ;
    qc_vapor_pressure_SE_25m_avg:bit_4_description = "Value is less than fail_min." ;
    qc_vapor_pressure_SE_25m_avg:bit_4_assessment = "Bad" ;
    qc_vapor_pressure_SE_25m_avg:bit_5_description = "Value is greater than fail_max." ;
    qc_vapor_pressure_SE_25m_avg:bit_5_assessment = "Bad" ;
float vapor_pressure_SE_25m_std(time) ;
    vapor_pressure_SE_25m_std:long_name = "Standard deviation of vapor pressure at
southeast 25m tower" ;
    vapor_pressure_SE_25m_std:units = "kPa" ;
    vapor_pressure_SE_25m_std:missing_value = -9999.f ;
    vapor_pressure_SE_25m_std:valid_min = -7999.f ;
    vapor_pressure_SE_25m_std:valid_max = 7999.f ;
float temperature_SE_60m_avg(time) ;
    temperature_SE_60m_avg:long_name = "Air temperature at southeast 60m tower" ;

```



```

temperature_SE_60m_avg:units = "degC" ;
temperature_SE_60m_avg:missing_value = -9999.f ;
temperature_SE_60m_avg:valid_min = -7999.f ;
temperature_SE_60m_avg:valid_max = 7999.f ;
temperature_SE_60m_avg:ancillary_variables = "qc_temperature_SE_60m_avg" ;
temperature_SE_60m_avg:standard_name = "air_temperature" ;
int qc_temperature_SE_60m_avg(time) ;
    qc_temperature_SE_60m_avg:long_name = "Quality check results on field: Air
temperature at southeast 60m tower" ;
    qc_temperature_SE_60m_avg:units = "1" ;
    qc_temperature_SE_60m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_temperature_SE_60m_avg:fail_min = -50.f ;
    qc_temperature_SE_60m_avg:fail_max = 50.f ;
    qc_temperature_SE_60m_avg:flag_method = "bit" ;
    qc_temperature_SE_60m_avg:bit_1_description = "Value is equal to missing_value." ;
    qc_temperature_SE_60m_avg:bit_1_assessment = "Bad" ;
    qc_temperature_SE_60m_avg:bit_2_description = "Value is less than valid_min." ;
    qc_temperature_SE_60m_avg:bit_2_assessment = "Bad" ;
    qc_temperature_SE_60m_avg:bit_3_description = "Value is greater than valid_max." ;
    qc_temperature_SE_60m_avg:bit_3_assessment = "Bad" ;
    qc_temperature_SE_60m_avg:bit_4_description = "Value is less than fail_min." ;
    qc_temperature_SE_60m_avg:bit_4_assessment = "Bad" ;
    qc_temperature_SE_60m_avg:bit_5_description = "Value is greater than fail_max." ;
    qc_temperature_SE_60m_avg:bit_5_assessment = "Bad" ;
float temperature_SE_60m_std(time) ;
    temperature_SE_60m_std:long_name = "Standard deviation of air temperature at
southeast 60m tower" ;
    temperature_SE_60m_std:units = "degC" ;
    temperature_SE_60m_std:missing_value = -9999.f ;
    temperature_SE_60m_std:valid_min = -7999.f ;
    temperature_SE_60m_std:valid_max = 7999.f ;
float relative_humidity_SE_60m_avg(time) ;
    relative_humidity_SE_60m_avg:long_name = "Relative humidity at southeast 60m
tower" ;
    relative_humidity_SE_60m_avg:units = "%" ;
    relative_humidity_SE_60m_avg:missing_value = -9999.f ;
    relative_humidity_SE_60m_avg:valid_min = -7999.f ;
    relative_humidity_SE_60m_avg:valid_max = 7999.f ;
    relative_humidity_SE_60m_avg:ancillary_variables =
"qc_relative_humidity_SE_60m_avg" ;
    relative_humidity_SE_60m_avg:standard_name = "relative_humidity" ;
int qc_relative_humidity_SE_60m_avg(time) ;
    qc_relative_humidity_SE_60m_avg:long_name = "Quality check results on field:
Relative humidity at southeast 60m tower" ;

```

```

qc_relative_humidity_SE_60m_avg:units = "1" ;
qc_relative_humidity_SE_60m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_relative_humidity_SE_60m_avg:fail_min = 0.f ;
qc_relative_humidity_SE_60m_avg:fail_max = 100.f ;
qc_relative_humidity_SE_60m_avg:flag_method = "bit" ;
qc_relative_humidity_SE_60m_avg:bit_1_description = "Value is equal to
missing_value." ;
qc_relative_humidity_SE_60m_avg:bit_1_assessment = "Bad" ;
qc_relative_humidity_SE_60m_avg:bit_2_description = "Value is less than valid_min." ;
qc_relative_humidity_SE_60m_avg:bit_2_assessment = "Bad" ;
qc_relative_humidity_SE_60m_avg:bit_3_description = "Value is greater than
valid_max." ;
qc_relative_humidity_SE_60m_avg:bit_3_assessment = "Bad" ;
qc_relative_humidity_SE_60m_avg:bit_4_description = "Value is less than fail_min." ;
qc_relative_humidity_SE_60m_avg:bit_4_assessment = "Bad" ;
qc_relative_humidity_SE_60m_avg:bit_5_description = "Value is greater than fail_max."
;
qc_relative_humidity_SE_60m_avg:bit_5_assessment = "Bad" ;
float relative_humidity_SE_60m_std(time) ;
relative_humidity_SE_60m_std:long_name = "Standard deviation of relative humidity at
southeast 60m tower" ;
relative_humidity_SE_60m_std:units = "%" ;
relative_humidity_SE_60m_std:missing_value = -9999.f ;
relative_humidity_SE_60m_std:valid_min = -7999.f ;
relative_humidity_SE_60m_std:valid_max = 7999.f ;
float vapor_pressure_SE_60m_avg(time) ;
vapor_pressure_SE_60m_avg:long_name = "Vapor pressure at southeast 60m tower" ;
vapor_pressure_SE_60m_avg:units = "kPa" ;
vapor_pressure_SE_60m_avg:missing_value = -9999.f ;
vapor_pressure_SE_60m_avg:valid_min = -7999.f ;
vapor_pressure_SE_60m_avg:valid_max = 7999.f ;
vapor_pressure_SE_60m_avg:ancillary_variables = "qc_vapor_pressureSE_60m_avg" ;
vapor_pressure_SE_60m_avg:standard_name = "water_vapor_partial_pressure_in_air" ;
int qc_vapor_pressure_SE_60m_avg(time) ;
qc_vapor_pressure_SE_60m_avg:long_name = "Quality check results on field: Vapor
pressure at southeast 60m tower" ;
qc_vapor_pressure_SE_60m_avg:units = "1" ;
qc_vapor_pressure_SE_60m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vapor_pressure_SE_60m_avg:fail_min = 0.f ;
qc_vapor_pressure_SE_60m_avg:fail_max = 10.f ;
qc_vapor_pressure_SE_60m_avg:flag_method = "bit" ;

```

```

qc_vapor_pressure_SE_60m_avg:bit_1_description = "Value is equal to missing_value."
;
qc_vapor_pressure_SE_60m_avg:bit_1_assessment = "Bad" ;
qc_vapor_pressure_SE_60m_avg:bit_2_description = "Value is less than valid_min." ;
qc_vapor_pressure_SE_60m_avg:bit_2_assessment = "Bad" ;
qc_vapor_pressure_SE_60m_avg:bit_3_description = "Value is greater than valid_max."
;
qc_vapor_pressure_SE_60m_avg:bit_3_assessment = "Bad" ;
qc_vapor_pressure_SE_60m_avg:bit_4_description = "Value is less than fail_min." ;
qc_vapor_pressure_SE_60m_avg:bit_4_assessment = "Bad" ;
qc_vapor_pressure_SE_60m_avg:bit_5_description = "Value is greater than fail_max." ;
qc_vapor_pressure_SE_60m_avg:bit_5_assessment = "Bad" ;
float vapor_pressure_SE_60m_std(time) ;
vapor_pressure_SE_60m_std:long_name = "Standard deviation of vapor pressure at
southeast 60m tower" ;
vapor_pressure_SE_60m_std:units = "kPa" ;
vapor_pressure_SE_60m_std:missing_value = -9999.f ;
vapor_pressure_SE_60m_std:valid_min = -7999.f ;
vapor_pressure_SE_60m_std:valid_max = 7999.f ;
float temperature_W_25m_avg(time) ;
temperature_W_25m_avg:long_name = "Air temperature at west 25m tower" ;
temperature_W_25m_avg:units = "degC" ;
temperature_W_25m_avg:missing_value = -9999.f ;
temperature_W_25m_avg:valid_min = -7999.f ;
temperature_W_25m_avg:valid_max = 7999.f ;
temperature_W_25m_avg:ancillary_variables = "qc_temperature_W_25m_avg" ;
temperature_W_25m_avg:standard_name = "air_temperature" ;
int qc_temperature_W_25m_avg(time) ;
qc_temperature_W_25m_avg:long_name = "Quality check results on field: Air
temperature at west 25m tower" ;
qc_temperature_W_25m_avg:units = "1" ;
qc_temperature_W_25m_avg:description = "This field contains bit packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_temperature_W_25m_avg:fail_min = -50.f ;
qc_temperature_W_25m_avg:fail_max = 50.f ;
qc_temperature_W_25m_avg:flag_method = "bit" ;
qc_temperature_W_25m_avg:bit_1_description = "Value is equal to missing_value." ;
qc_temperature_W_25m_avg:bit_1_assessment = "Bad" ;
qc_temperature_W_25m_avg:bit_2_description = "Value is less than valid_min." ;
qc_temperature_W_25m_avg:bit_2_assessment = "Bad" ;
qc_temperature_W_25m_avg:bit_3_description = "Value is greater than valid_max." ;
qc_temperature_W_25m_avg:bit_3_assessment = "Bad" ;
qc_temperature_W_25m_avg:bit_4_description = "Value is less than fail_min." ;
qc_temperature_W_25m_avg:bit_4_assessment = "Bad" ;
qc_temperature_W_25m_avg:bit_5_description = "Value is greater than fail_max." ;

```

```

    qc_temperature_W_25m_avg:bit_5_assessment = "Bad" ;
float temperature_W_25m_std(time) ;
    temperature_W_25m_std:long_name = "Standard deviation of air temperature at west
25m tower" ;
    temperature_W_25m_std:units = "degC" ;
    temperature_W_25m_std:missing_value = -9999.f ;
    temperature_W_25m_std:valid_min = -7999.f ;
    temperature_W_25m_std:valid_max = 7999.f ;
float relative_humidity_W_25m_avg(time) ;
    relative_humidity_W_25m_avg:long_name = "Relative humidity at west 25m tower" ;
    relative_humidity_W_25m_avg:units = "%" ;
    relative_humidity_W_25m_avg:missing_value = -9999.f ;
    relative_humidity_W_25m_avg:valid_min = -7999.f ;
    relative_humidity_W_25m_avg:valid_max = 7999.f ;
    relative_humidity_W_25m_avg:ancillary_variables =
"qc_relative_humidity_W_25m_avg" ;
    relative_humidity_W_25m_avg:standard_name = "relative_humidity" ;
int qc_relative_humidity_W_25m_avg(time) ;
    qc_relative_humidity_W_25m_avg:long_name = "Quality check results on field:
Relative humidity at west 25m tower" ;
    qc_relative_humidity_W_25m_avg:units = "1" ;
    qc_relative_humidity_W_25m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_relative_humidity_W_25m_avg:fail_min = 0.f ;
    qc_relative_humidity_W_25m_avg:fail_max = 100.f ;
    qc_relative_humidity_W_25m_avg:flag_method = "bit" ;
    qc_relative_humidity_W_25m_avg:bit_1_description = "Value is equal to
missing_value." ;
    qc_relative_humidity_W_25m_avg:bit_1_assessment = "Bad" ;
    qc_relative_humidity_W_25m_avg:bit_2_description = "Value is less than valid_min." ;
    qc_relative_humidity_W_25m_avg:bit_2_assessment = "Bad" ;
    qc_relative_humidity_W_25m_avg:bit_3_description = "Value is greater than
valid_max." ;
    qc_relative_humidity_W_25m_avg:bit_3_assessment = "Bad" ;
    qc_relative_humidity_W_25m_avg:bit_4_description = "Value is less than fail_min." ;
    qc_relative_humidity_W_25m_avg:bit_4_assessment = "Bad" ;
    qc_relative_humidity_W_25m_avg:bit_5_description = "Value is greater than fail_max."
;
    qc_relative_humidity_W_25m_avg:bit_5_assessment = "Bad" ;
float relative_humidity_W_25m_std(time) ;
    relative_humidity_W_25m_std:long_name = "Standard deviation of relative humidity at
west 25m tower" ;
    relative_humidity_W_25m_std:units = "%" ;
    relative_humidity_W_25m_std:missing_value = -9999.f ;
    relative_humidity_W_25m_std:valid_min = -7999.f ;

```

```

    relative_humidity_W_25m_std:valid_max = 7999.f;
float vapor_pressure_W_25m_avg(time);
    vapor_pressure_W_25m_avg:long_name = "Vapor pressure at west 25m tower";
    vapor_pressure_W_25m_avg:units = "kPa";
    vapor_pressure_W_25m_avg:missing_value = -9999.f;
    vapor_pressure_W_25m_avg:valid_min = -7999.f;
    vapor_pressure_W_25m_avg:valid_max = 7999.f;
    vapor_pressure_W_25m_avg:ancillary_variables = "qc_vapor_pressure_W_25m_avg";
    vapor_pressure_W_25m_avg:standard_name = "water_vapor_partial_pressure_in_air";
int qc_vapor_pressure_W_25m_avg(time);
    qc_vapor_pressure_W_25m_avg:long_name = "Quality check results on field: Vapor
pressure at west 25m tower";
    qc_vapor_pressure_W_25m_avg:units = "1";
    qc_vapor_pressure_W_25m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests.";
    qc_vapor_pressure_W_25m_avg:fail_min = 0.f;
    qc_vapor_pressure_W_25m_avg:fail_max = 10.f;
    qc_vapor_pressure_W_25m_avg:flag_method = "bit";
    qc_vapor_pressure_W_25m_avg:bit_1_description = "Value is equal to missing_value.";
    qc_vapor_pressure_W_25m_avg:bit_1_assessment = "Bad";
    qc_vapor_pressure_W_25m_avg:bit_2_description = "Value is less than valid_min.";
    qc_vapor_pressure_W_25m_avg:bit_2_assessment = "Bad";
    qc_vapor_pressure_W_25m_avg:bit_3_description = "Value is greater than valid_max.";
    qc_vapor_pressure_W_25m_avg:bit_3_assessment = "Bad";
    qc_vapor_pressure_W_25m_avg:bit_4_description = "Value is less than fail_min.";
    qc_vapor_pressure_W_25m_avg:bit_4_assessment = "Bad";
    qc_vapor_pressure_W_25m_avg:bit_5_description = "Value is greater than fail_max.";
    qc_vapor_pressure_W_25m_avg:bit_5_assessment = "Bad";
float vapor_pressure_W_25m_std(time);
    vapor_pressure_W_25m_std:long_name = "Standard deviation of vapor pressure at west
25m tower";
    vapor_pressure_W_25m_std:units = "kPa";
    vapor_pressure_W_25m_std:missing_value = -9999.f;
    vapor_pressure_W_25m_std:valid_min = -7999.f;
    vapor_pressure_W_25m_std:valid_max = 7999.f;
float temperature_W_60m_avg(time);
    temperature_W_60m_avg:long_name = "Air temperature at west 60m tower";
    temperature_W_60m_avg:units = "degC";
    temperature_W_60m_avg:missing_value = -9999.f;
    temperature_W_60m_avg:valid_min = -7999.f;
    temperature_W_60m_avg:valid_max = 7999.f;
    temperature_W_60m_avg:ancillary_variables = "qc_temperature_W_60m_avg";
    temperature_W_60m_avg:standard_name = "air_temperature";
int qc_temperature_W_60m_avg(time);

```

```

qc_temperature_W_60m_avg:long_name = "Quality check results on field: Air
temperature at west 60m tower" ;
qc_temperature_W_60m_avg:units = "1" ;
qc_temperature_W_60m_avg:description = "This field contains bit packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_temperature_W_60m_avg:fail_min = -50.f ;
qc_temperature_W_60m_avg:fail_max = 50.f ;
qc_temperature_W_60m_avg:flag_method = "bit" ;
qc_temperature_W_60m_avg:bit_1_description = "Value is equal to missing_value." ;
qc_temperature_W_60m_avg:bit_1_assessment = "Bad" ;
qc_temperature_W_60m_avg:bit_2_description = "Value is less than valid_min." ;
qc_temperature_W_60m_avg:bit_2_assessment = "Bad" ;
qc_temperature_W_60m_avg:bit_3_description = "Value is greater than valid_max." ;
qc_temperature_W_60m_avg:bit_3_assessment = "Bad" ;
qc_temperature_W_60m_avg:bit_4_description = "Value is less than fail_min." ;
qc_temperature_W_60m_avg:bit_4_assessment = "Bad" ;
qc_temperature_W_60m_avg:bit_5_description = "Value is greater than fail_max." ;
qc_temperature_W_60m_avg:bit_5_assessment = "Bad" ;
float temperature_W_60m_std(time) ;
temperature_W_60m_std:long_name = "Standard deviation of air temperature at west
60m tower" ;
temperature_W_60m_std:units = "degC" ;
temperature_W_60m_std:missing_value = -9999.f ;
temperature_W_60m_std:valid_min = -7999.f ;
temperature_W_60m_std:valid_max = 7999.f ;
float relative_humidity_W_60m_avg(time) ;
relative_humidity_W_60m_avg:long_name = "Relative humidity at west 60m tower" ;
relative_humidity_W_60m_avg:units = "%" ;
relative_humidity_W_60m_avg:missing_value = -9999.f ;
relative_humidity_W_60m_avg:valid_min = -7999.f ;
relative_humidity_W_60m_avg:valid_max = 7999.f ;
relative_humidity_W_60m_avg:ancillary_variables =
"qc_relative_humidity_W_60m_avg" ;
relative_humidity_W_60m_avg:standard_name = "relative_humidity" ;
int qc_relative_humidity_W_60m_avg(time) ;
qc_relative_humidity_W_60m_avg:long_name = "Quality check results on field:
Relative humidity at west 60m tower" ;
qc_relative_humidity_W_60m_avg:units = "1" ;
qc_relative_humidity_W_60m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_relative_humidity_W_60m_avg:fail_min = 0.f ;
qc_relative_humidity_W_60m_avg:fail_max = 100.f ;
qc_relative_humidity_W_60m_avg:flag_method = "bit" ;

```

```

qc_relative_humidity_W_60m_avg:bit_1_description = "Value is equal to
missing_value." ;
qc_relative_humidity_W_60m_avg:bit_1_assessment = "Bad" ;
qc_relative_humidity_W_60m_avg:bit_2_description = "Value is less than valid_min." ;
qc_relative_humidity_W_60m_avg:bit_2_assessment = "Bad" ;
qc_relative_humidity_W_60m_avg:bit_3_description = "Value is greater than
valid_max." ;
qc_relative_humidity_W_60m_avg:bit_3_assessment = "Bad" ;
qc_relative_humidity_W_60m_avg:bit_4_description = "Value is less than fail_min." ;
qc_relative_humidity_W_60m_avg:bit_4_assessment = "Bad" ;
qc_relative_humidity_W_60m_avg:bit_5_description = "Value is greater than fail_max."
;
qc_relative_humidity_W_60m_avg:bit_5_assessment = "Bad" ;
float relative_humidity_W_60m_std(time) ;
relative_humidity_W_60m_std:long_name = "Standard deviation of relative humidity at
west 60m tower" ;
relative_humidity_W_60m_std:units = "%" ;
relative_humidity_W_60m_std:missing_value = -9999.f ;
relative_humidity_W_60m_std:valid_min = -7999.f ;
relative_humidity_W_60m_std:valid_max = 7999.f ;
float vapor_pressure_W_60m_avg(time) ;
vapor_pressure_W_60m_avg:long_name = "Vapor pressure at west 60m tower" ;
vapor_pressure_W_60m_avg:units = "kPa" ;
vapor_pressure_W_60m_avg:missing_value = -9999.f ;
vapor_pressure_W_60m_avg:valid_min = -7999.f ;
vapor_pressure_W_60m_avg:valid_max = 7999.f ;
vapor_pressure_W_60m_avg:ancillary_variables = "qc_vapor_pressure_W_60m_avg" ;
vapor_pressure_W_60m_avg:standard_name = "water_vapor_partial_pressure_in_air" ;
int qc_vapor_pressure_W_60m_avg(time) ;
qc_vapor_pressure_W_60m_avg:long_name = "Quality check results on field: Vapor
pressure at west 60m tower" ;
qc_vapor_pressure_W_60m_avg:units = "1" ;
qc_vapor_pressure_W_60m_avg:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vapor_pressure_W_60m_avg:fail_min = 0.f ;
qc_vapor_pressure_W_60m_avg:fail_max = 10.f ;
qc_vapor_pressure_W_60m_avg:flag_method = "bit" ;
qc_vapor_pressure_W_60m_avg:bit_1_description = "Value is equal to missing_value." ;
qc_vapor_pressure_W_60m_avg:bit_1_assessment = "Bad" ;
qc_vapor_pressure_W_60m_avg:bit_2_description = "Value is less than valid_min." ;
qc_vapor_pressure_W_60m_avg:bit_2_assessment = "Bad" ;
qc_vapor_pressure_W_60m_avg:bit_3_description = "Value is greater than valid_max." ;
qc_vapor_pressure_W_60m_avg:bit_3_assessment = "Bad" ;
qc_vapor_pressure_W_60m_avg:bit_4_description = "Value is less than fail_min." ;
qc_vapor_pressure_W_60m_avg:bit_4_assessment = "Bad" ;

```

```

    qc_vapor_pressure_W_60m_avg:bit_5_description = "Value is greater than fail_max." ;
    qc_vapor_pressure_W_60m_avg:bit_5_assessment = "Bad" ;
float vapor_pressure_W_60m_std(time) ;
    vapor_pressure_W_60m_std:long_name = "Standard deviation of vapor pressure at west
60m tower" ;
    vapor_pressure_W_60m_std:units = "kPa" ;
    vapor_pressure_W_60m_std:missing_value = -9999.f ;
    vapor_pressure_W_60m_std:valid_min = -7999.f ;
    vapor_pressure_W_60m_std:valid_max = 7999.f ;
float logger_temperature(time) ;
    logger_temperature:long_name = "Logger temperature" ;
    logger_temperature:units = "degC" ;
    logger_temperature:missing_value = -9999.f ;
    logger_temperature:valid_min = -7999.f ;
    logger_temperature:valid_max = 7999.f ;
    logger_temperature:ancillary_variables = "qc_logger_temperature" ;
int qc_logger_temperature(time) ;
    qc_logger_temperature:long_name = "Quality check results on field: Logger
temperature" ;
    qc_logger_temperature:units = "1" ;
    qc_logger_temperature:description = "This field contains bit packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_logger_temperature:fail_min = 0.f ;
    qc_logger_temperature:fail_max = 50.f ;
    qc_logger_temperature:flag_method = "bit" ;
    qc_logger_temperature:bit_1_description = "Value is equal to missing_value." ;
    qc_logger_temperature:bit_1_assessment = "Bad" ;
    qc_logger_temperature:bit_2_description = "Value is less than valid_min." ;
    qc_logger_temperature:bit_2_assessment = "Bad" ;
    qc_logger_temperature:bit_3_description = "Value is greater than valid_max." ;
    qc_logger_temperature:bit_3_assessment = "Bad" ;
    qc_logger_temperature:bit_4_description = "Value is less than fail_min." ;
    qc_logger_temperature:bit_4_assessment = "Bad" ;
    qc_logger_temperature:bit_5_description = "Value is greater than fail_max." ;
    qc_logger_temperature:bit_5_assessment = "Bad" ;
float battery_voltage(time) ;
    battery_voltage:long_name = "Logger voltage" ;
    battery_voltage:units = "V" ;
    battery_voltage:missing_value = -9999.f ;
    battery_voltage:valid_min = -7999.f ;
    battery_voltage:valid_max = 7999.f ;
    battery_voltage:ancillary_variables = "qc_battery_voltage" ;
int qc_battery_voltage(time) ;
    qc_battery_voltage:long_name = "Quality check results on field: Logger voltage" ;
    qc_battery_voltage:units = "1" ;

```


qc_battery_voltage:description = "This field contains bit packed integer values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;

```
qc_battery_voltage:fail_min = 10.f ;
qc_battery_voltage:fail_max = 15.f ;
qc_battery_voltage:flag_method = "bit" ;
qc_battery_voltage:bit_1_description = "Value is equal to missing_value." ;
qc_battery_voltage:bit_1_assessment = "Bad" ;
qc_battery_voltage:bit_2_description = "Value is less than valid_min." ;
qc_battery_voltage:bit_2_assessment = "Bad" ;
qc_battery_voltage:bit_3_description = "Value is greater than valid_max." ;
qc_battery_voltage:bit_3_assessment = "Bad" ;
qc_battery_voltage:bit_4_description = "Value is less than fail_min." ;
qc_battery_voltage:bit_4_assessment = "Bad" ;
qc_battery_voltage:bit_5_description = "Value is greater than fail_max." ;
qc_battery_voltage:bit_5_assessment = "Bad" ;
```

float lat ;

```
lat:long_name = "North latitude" ;
lat:units = "degree_N" ;
lat:valid_min = -90.f ;
lat:valid_max = 90.f ;
lat:standard_name = "latitude" ;
```

float lon ;

```
lon:long_name = "East longitude" ;
lon:units = "degree_E" ;
lon:valid_min = -180.f ;
lon:valid_max = 180.f ;
lon:standard_name = "longitude" ;
```

float alt ;

```
alt:long_name = "Altitude above mean sea level" ;
alt:units = "m" ;
alt:standard_name = "altitude" ;
```

// global attributes:

```
:command_line = "towermet_ingest -s sgp -f C1" ;
:Conventions = "ARM-1.2" ;
:process_version = "ingest-towermet-1.1-0.e17" ;
:dod_version = "towermet-b1-1.0" ;
:input_source =
```

"/data/collection/sgp/sgptowermetC1.00/SGPTWR_1min.20210414000000.dat" ;

```
:site_id = "sgp" ;
:platform_id = "towermet" ;
:facility_id = "C1" ;
:data_level = "b1" ;
:location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
:datastream = "sgptowermetC1.b1" ;
```

```
:doi = "10.5439/1501941" ;  
:history = "created by user dsmgr on machine zinc at 2021-04-14 02:40:00, using ingest-  
towermet-1.1-0.el7" ;  
}
```

Appendix B

Output Datastream

sgp1twrmrC1.b1:

```

netcdf sgp1twrmrC1.c1.20210414.000000 {
dimensions:
    time = UNLIMITED ; // (1440 currently)
    bound = 2 ;
variables:
    int base_time ;
        base_time:string = "2021-04-14 00:00:00 0:00" ;
        base_time:long_name = "Base time in Epoch" ;
        base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
        base_time:ancillary_variables = "time_offset" ;
    double time_offset(time) ;
        time_offset:long_name = "Time offset from base_time" ;
        time_offset:units = "seconds since 2021-04-14 00:00:00 0:00" ;
        time_offset:ancillary_variables = "base_time" ;
    double time(time) ;
        time:long_name = "Time offset from midnight" ;
        time:units = "seconds since 2021-04-14 00:00:00 0:00" ;
        time:bounds = "time_bounds" ;
        time:standard_name = "time" ;
    double time_bounds(time, bound) ;
        time_bounds:long_name = "Time cell bounds" ;
        time_bounds:bound_offsets = -60., 0. ;
    float pres_02m(time) ;
        pres_02m:long_name = "Barometric pressure at 2 m" ;
        pres_02m:units = "hPa" ;
        pres_02m:missing_value = -9999.f ;
        pres_02m:standard_name = "air_pressure" ;
        pres_02m:ancillary_variables = "qc_pres_02m source_pres_02m" ;
        pres_02m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
    int qc_pres_02m(time) ;
        qc_pres_02m:long_name = "Quality check results on variable: Barometric pressure at 2
m" ;

```

```

qc_pres_02m:units = "1" ;
qc_pres_02m:description = "This variable contains bit-packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_02m:standard_name = "quality_flag" ;
qc_pres_02m:flag_method = "bit" ;
qc_pres_02m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
qc_pres_02m:bit_1_assessment = "Bad" ;
qc_pres_02m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_pres_02m:bit_2_assessment = "Indeterminate" ;
qc_pres_02m:bit_3_description = "temp_02m assessment is Bad, value set to
missing_value." ;
qc_pres_02m:bit_3_assessment = "Bad" ;
int source_pres_02m(time) ;
source_pres_02m:long_name = "Source for variable: Barometric pressure at 2 m" ;
source_pres_02m:units = "1" ;
source_pres_02m:description = "This variable contains integer values which should be
interpreted as listed." ;
source_pres_02m:flag_method = "integer" ;
source_pres_02m:flag_0_description = "no_source_available" ;
source_pres_02m:flag_1_description = "sgpmetE13.b1:atmos_pressure" ;
source_pres_02m:flag_2_description = "sgpmawsC1.b1:atmospheric_pressure" ;
float pres_02m_preferred(time) ;
pres_02m_preferred:long_name = "Pressure at surface from MET" ;
pres_02m_preferred:units = "hPa" ;
pres_02m_preferred:missing_value = -9999.f ;
pres_02m_preferred:standard_name = "air_pressure" ;
pres_02m_preferred:ancillary_variables = "qc_pres_02m_preferred" ;
pres_02m_preferred:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
pres_02m_preferred:source = "sgpmetE13.b1:atmos_pressure" ;
int qc_pres_02m_preferred(time) ;
qc_pres_02m_preferred:long_name = "Quality check results on variable: Pressure at
surface from MET" ;
qc_pres_02m_preferred:units = "1" ;
qc_pres_02m_preferred:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_02m_preferred:standard_name = "quality_flag" ;
qc_pres_02m_preferred:flag_method = "bit" ;
qc_pres_02m_preferred:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_pres_02m_preferred:bit_1_assessment = "Bad" ;
qc_pres_02m_preferred:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;

```

```

qc_pres_02m_preferred:bit_2_assessment = "Indeterminate" ;
qc_pres_02m_preferred:bit_3_description = "temp_02m_preferred assessment is Bad,
value set to missing_value." ;
qc_pres_02m_preferred:bit_3_assessment = "Bad" ;
float pres_02m_alternate(time) ;
pres_02m_alternate:long_name = "Pressure at surface from THWAPS/MAWS" ;
pres_02m_alternate:units = "hPa" ;
pres_02m_alternate:missing_value = -9999.f ;
pres_02m_alternate:standard_name = "air_pressure" ;
pres_02m_alternate:ancillary_variables = "qc_pres_02m_alternate" ;
pres_02m_alternate:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
pres_02m_alternate:source = "sgpmawsC1.b1:atmospheric_pressure" ;
int qc_pres_02m_alternate(time) ;
qc_pres_02m_alternate:long_name = "Quality check results on variable: Pressure at
surface from THWAPS/MAWS" ;
qc_pres_02m_alternate:units = "1" ;
qc_pres_02m_alternate:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_02m_alternate:standard_name = "quality_flag" ;
qc_pres_02m_alternate:flag_method = "bit" ;
qc_pres_02m_alternate:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_pres_02m_alternate:bit_1_assessment = "Bad" ;
qc_pres_02m_alternate:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_pres_02m_alternate:bit_2_assessment = "Indeterminate" ;
qc_pres_02m_alternate:bit_3_description = "temp_02m_alternate assessment is Bad,
value set to missing_value." ;
qc_pres_02m_alternate:bit_3_assessment = "Bad" ;
float pres_25m(time) ;
pres_25m:long_name = "Calculated barometric pressure at 25 m" ;
pres_25m:units = "hPa" ;
pres_25m:missing_value = -9999.f ;
pres_25m:valid_min = 800.f ;
pres_25m:valid_max = 1100.f ;
pres_25m:standard_name = "air_pressure" ;
pres_25m:ancillary_variables = "qc_pres_25m" ;
int qc_pres_25m(time) ;
qc_pres_25m:long_name = "Quality check results on variable: Calculated barometric
pressure at 25 m" ;
qc_pres_25m:units = "1" ;
qc_pres_25m:description = "This variable contains bit-packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_25m:standard_name = "quality_flag" ;

```

```

qc_pres_25m:flag_method = "bit" ;
qc_pres_25m:bit_1_description = "pres_02m or temp_02m assessment is Bad, value set
to missing_value." ;
qc_pres_25m:bit_1_assessment = "Bad" ;
qc_pres_25m:bit_2_description = "pres_02m or temp_02m assessment is Indeterminate."
;
qc_pres_25m:bit_2_assessment = "Indeterminate" ;
qc_pres_25m:bit_3_description = "temp_25m assessment is Bad, value set to
missing_value." ;
qc_pres_25m:bit_3_assessment = "Bad" ;
qc_pres_25m:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
qc_pres_25m:bit_4_assessment = "Bad" ;
qc_pres_25m:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
qc_pres_25m:bit_5_assessment = "Bad" ;
float pres_25m_SE(time) ;
pres_25m_SE:long_name = "Calculated pressure at 25m from SE elevator" ;
pres_25m_SE:units = "hPa" ;
pres_25m_SE:valid_min = 800.f ;
pres_25m_SE:valid_max = 1100.f ;
pres_25m_SE:missing_value = -9999.f ;
pres_25m_SE:standard_name = "air_pressure" ;
pres_25m_SE:ancillary_variables = "qc_pres_25m_SE" ;
int qc_pres_25m_SE(time) ;
qc_pres_25m_SE:long_name = "Quality check results on variable: Calculated pressure at
25m from SE elevator" ;
qc_pres_25m_SE:units = "1" ;
qc_pres_25m_SE:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_25m_SE:standard_name = "quality_flag" ;
qc_pres_25m_SE:flag_method = "bit" ;
qc_pres_25m_SE:bit_1_description = "pres_02m or temp_02m assessment is Bad, value
set to missing_value." ;
qc_pres_25m_SE:bit_1_assessment = "Bad" ;
qc_pres_25m_SE:bit_2_description = "pres_02m or temp_02m assessment is
Indeterminate." ;
qc_pres_25m_SE:bit_2_assessment = "Indeterminate" ;
qc_pres_25m_SE:bit_3_description = "temp_25m_SE assessment is Bad, value set to
missing_value." ;
qc_pres_25m_SE:bit_3_assessment = "Bad" ;
qc_pres_25m_SE:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
qc_pres_25m_SE:bit_4_assessment = "Bad" ;

```

```

    qc_pres_25m_SE:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
    qc_pres_25m_SE:bit_5_assessment = "Bad" ;
float pres_25m_W(time) ;
    pres_25m_W:long_name = "Calculated pressure at 25m from W elevator" ;
    pres_25m_W:units = "hPa" ;
    pres_25m_W:valid_min = 800.f ;
    pres_25m_W:valid_max = 1100.f ;
    pres_25m_W:missing_value = -9999.f ;
    pres_25m_W:standard_name = "air_pressure" ;
    pres_25m_W:ancillary_variables = "qc_pres_25m_W" ;
int qc_pres_25m_W(time) ;
    qc_pres_25m_W:long_name = "Quality check results on variable: Calculated pressure at
25m from W elevator" ;
    qc_pres_25m_W:units = "1" ;
    qc_pres_25m_W:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_pres_25m_W:standard_name = "quality_flag" ;
    qc_pres_25m_W:flag_method = "bit" ;
    qc_pres_25m_W:bit_1_description = "pres_02m or temp_02m assessment is Bad, value
set to missing_value." ;
    qc_pres_25m_W:bit_1_assessment = "Bad" ;
    qc_pres_25m_W:bit_2_description = "pres_02m or temp_02m assessment is
Indeterminate." ;
    qc_pres_25m_W:bit_2_assessment = "Indeterminate" ;
    qc_pres_25m_W:bit_3_description = "temp_25m_W assessment is Bad, value set to
missing_value." ;
    qc_pres_25m_W:bit_3_assessment = "Bad" ;
    qc_pres_25m_W:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
    qc_pres_25m_W:bit_4_assessment = "Bad" ;
    qc_pres_25m_W:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
    qc_pres_25m_W:bit_5_assessment = "Bad" ;
float pres_60m(time) ;
    pres_60m:long_name = "Calculated barometric pressure at 60 m" ;
    pres_60m:units = "hPa" ;
    pres_60m:valid_min = 800.f ;
    pres_60m:valid_max = 1100.f ;
    pres_60m:missing_value = -9999.f ;
    pres_60m:standard_name = "air_pressure" ;
    pres_60m:ancillary_variables = "qc_pres_60m" ;
int qc_pres_60m(time) ;
    qc_pres_60m:long_name = "Quality check results on variable: Calculated barometric
pressure at 60 m" ;

```

```

qc_pres_60m:units = "1" ;
qc_pres_60m:description = "This variable contains bit-packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_60m:standard_name = "quality_flag" ;
qc_pres_60m:flag_method = "bit" ;
qc_pres_60m:bit_1_description = "pres_02m or temp_02m assessment is Bad, value set
to missing_value." ;
qc_pres_60m:bit_1_assessment = "Bad" ;
qc_pres_60m:bit_2_description = "pres_02m or temp_02m assessment is Indeterminate."
;
qc_pres_60m:bit_2_assessment = "Indeterminate" ;
qc_pres_60m:bit_3_description = "temp_60m assessment is Bad, value set to
missing_value." ;
qc_pres_60m:bit_3_assessment = "Bad" ;
qc_pres_60m:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
qc_pres_60m:bit_4_assessment = "Bad" ;
qc_pres_60m:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
qc_pres_60m:bit_5_assessment = "Bad" ;
float pres_60m_SE(time) ;
pres_60m_SE:long_name = "Calculated pressure at 60m from SE elevator" ;
pres_60m_SE:units = "hPa" ;
pres_60m_SE:valid_min = 800.f ;
pres_60m_SE:valid_max = 1100.f ;
pres_60m_SE:missing_value = -9999.f ;
pres_60m_SE:standard_name = "air_pressure" ;
pres_60m_SE:ancillary_variables = "qc_pres_60m_SE" ;
int qc_pres_60m_SE(time) ;
qc_pres_60m_SE:long_name = "Quality check results on variable: Calculated pressure at
60m from SE elevator" ;
qc_pres_60m_SE:units = "1" ;
qc_pres_60m_SE:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_60m_SE:standard_name = "quality_flag" ;
qc_pres_60m_SE:flag_method = "bit" ;
qc_pres_60m_SE:bit_1_description = "pres_02m or temp_02m assessment is Bad, value
set to missing_value." ;
qc_pres_60m_SE:bit_1_assessment = "Bad" ;
qc_pres_60m_SE:bit_2_description = "pres_02m or temp_02m assessment is
Indeterminate." ;
qc_pres_60m_SE:bit_2_assessment = "Indeterminate" ;
qc_pres_60m_SE:bit_3_description = "temp_60m_SE assessment is Bad, value set to
missing_value." ;

```



```

qc_pres_60m_SE:bit_3_assessment = "Bad" ;
qc_pres_60m_SE:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
qc_pres_60m_SE:bit_4_assessment = "Bad" ;
qc_pres_60m_SE:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
qc_pres_60m_SE:bit_5_assessment = "Bad" ;
float pres_60m_W(time) ;
pres_60m_W:long_name = "Calculated pressure at 60m from W elevator" ;
pres_60m_W:units = "hPa" ;
pres_60m_W:valid_min = 800.f ;
pres_60m_W:valid_max = 1100.f ;
pres_60m_W:missing_value = -9999.f ;
pres_60m_W:standard_name = "air_pressure" ;
pres_60m_W:ancillary_variables = "qc_pres_60m_W" ;
int qc_pres_60m_W(time) ;
qc_pres_60m_W:long_name = "Quality check results on variable: Calculated pressure at
60m from W elevator" ;
qc_pres_60m_W:units = "1" ;
qc_pres_60m_W:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_pres_60m_W:standard_name = "quality_flag" ;
qc_pres_60m_W:flag_method = "bit" ;
qc_pres_60m_W:bit_1_description = "pres_02m or temp_02m assessment is Bad, value
set to missing_value." ;
qc_pres_60m_W:bit_1_assessment = "Bad" ;
qc_pres_60m_W:bit_2_description = "pres_02m or temp_02m assessment is
Indeterminate." ;
qc_pres_60m_W:bit_2_assessment = "Indeterminate" ;
qc_pres_60m_W:bit_3_description = "temp_60m_W assessment is Bad, value set to
missing_value." ;
qc_pres_60m_W:bit_3_assessment = "Bad" ;
qc_pres_60m_W:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
qc_pres_60m_W:bit_4_assessment = "Bad" ;
qc_pres_60m_W:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
qc_pres_60m_W:bit_5_assessment = "Bad" ;
float temp_02m(time) ;
temp_02m:long_name = "Temperature at 2 m" ;
temp_02m:units = "degC" ;
temp_02m:missing_value = -9999.f ;
temp_02m:valid_min = -50.f ;
temp_02m:valid_max = 60.f ;
temp_02m:standard_name = "air_temperature" ;

```

```

temp_02m:ancillary_variables = "qc_temp_02m source_temp_02m" ;
temp_02m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_temp_02m(time) ;
temp_02m:qc_temp_02m:long_name = "Quality check results on variable: Temperature at 2 m" ;
temp_02m:qc_temp_02m:units = "1" ;
temp_02m:qc_temp_02m:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
temp_02m:qc_temp_02m:standard_name = "quality_flag" ;
temp_02m:qc_temp_02m:flag_method = "bit" ;
temp_02m:qc_temp_02m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
temp_02m:qc_temp_02m:bit_1_assessment = "Bad" ;
temp_02m:qc_temp_02m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
temp_02m:qc_temp_02m:bit_2_assessment = "Indeterminate" ;
temp_02m:qc_temp_02m:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
temp_02m:qc_temp_02m:bit_3_assessment = "Bad" ;
temp_02m:qc_temp_02m:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
temp_02m:qc_temp_02m:bit_4_assessment = "Bad" ;
int source_temp_02m(time) ;
temp_02m:source_temp_02m:long_name = "Source for variable: Temperature at 2 m" ;
temp_02m:source_temp_02m:units = "1" ;
temp_02m:source_temp_02m:description = "This variable contains integer values which should be
interpreted as listed." ;
temp_02m:source_temp_02m:flag_method = "integer" ;
temp_02m:source_temp_02m:flag_0_description = "no_source_available" ;
temp_02m:source_temp_02m:flag_1_description = "sgpmetE13.b1:temp_mean" ;
temp_02m:source_temp_02m:flag_2_description = "sgpmawsC1.b1:atmospheric_temperature" ;
float temp_02m_preferred(time) ;
temp_02m_preferred:long_name = "Temperature at surface from MET" ;
temp_02m_preferred:units = "degC" ;
temp_02m_preferred:missing_value = -9999.f ;
temp_02m_preferred:valid_min = -50.f ;
temp_02m_preferred:valid_max = 60.f ;
temp_02m_preferred:standard_name = "air_temperature" ;
temp_02m_preferred:ancillary_variables = "qc_temp_02m_preferred" ;
temp_02m_preferred:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
temp_02m_preferred:source = "sgpmetE13.b1:temp_mean" ;
int qc_temp_02m_preferred(time) ;
temp_02m_preferred:qc_temp_02m_preferred:long_name = "Quality check results on variable: Temperature at
surface from MET" ;
temp_02m_preferred:qc_temp_02m_preferred:units = "1" ;

```

```

qc_temp_02m_preferred:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_temp_02m_preferred:standard_name = "quality_flag" ;
qc_temp_02m_preferred:flag_method = "bit" ;
qc_temp_02m_preferred:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_temp_02m_preferred:bit_1_assessment = "Bad" ;
qc_temp_02m_preferred:bit_2_description = "Transformation resulted in an
indeterminate outcome." ;
qc_temp_02m_preferred:bit_2_assessment = "Indeterminate" ;
qc_temp_02m_preferred:bit_3_description = "Value is less than the valid_min, value set
to missing_value." ;
qc_temp_02m_preferred:bit_3_assessment = "Bad" ;
qc_temp_02m_preferred:bit_4_description = "Value is greater than the valid_max, value
set to missing_value." ;
qc_temp_02m_preferred:bit_4_assessment = "Bad" ;
float temp_02m_alternate(time) ;
temp_02m_alternate:long_name = "Temperature at surface from THWAPS/MAWS" ;
temp_02m_alternate:units = "degC" ;
temp_02m_alternate:missing_value = -9999.f ;
temp_02m_alternate:valid_min = -50.f ;
temp_02m_alternate:valid_max = 60.f ;
temp_02m_alternate:standard_name = "air_temperature" ;
temp_02m_alternate:ancillary_variables = "qc_temp_02m_alternate" ;
temp_02m_alternate:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
temp_02m_alternate:source = "sgpmawsC1.b1:atmospheric_temperature" ;
int qc_temp_02m_alternate(time) ;
qc_temp_02m_alternate:long_name = "Quality check results on variable: Temperature at
surface from THWAPS/MAWS" ;
qc_temp_02m_alternate:units = "1" ;
qc_temp_02m_alternate:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_temp_02m_alternate:standard_name = "quality_flag" ;
qc_temp_02m_alternate:flag_method = "bit" ;
qc_temp_02m_alternate:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_temp_02m_alternate:bit_1_assessment = "Bad" ;
qc_temp_02m_alternate:bit_2_description = "Transformation resulted in an
indeterminate outcome." ;
qc_temp_02m_alternate:bit_2_assessment = "Indeterminate" ;
qc_temp_02m_alternate:bit_3_description = "Value is less than the valid_min, value set
to missing_value." ;
qc_temp_02m_alternate:bit_3_assessment = "Bad" ;

```

```

    qc_temp_02m_alternate:bit_4_description = "Value is greater than the valid_max, value
set to missing_value." ;
    qc_temp_02m_alternate:bit_4_assessment = "Bad" ;
float temp_25m(time) ;
    temp_25m:long_name = "Temperature at 25 m" ;
    temp_25m:units = "degC" ;
    temp_25m:missing_value = -9999.f ;
    temp_25m:valid_min = -50.f ;
    temp_25m:valid_max = 60.f ;
    temp_25m:standard_name = "air_temperature" ;
    temp_25m:ancillary_variables = "qc_temp_25m source_temp_25m" ;
    temp_25m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_temp_25m(time) ;
    qc_temp_25m:long_name = "Quality check results on variable: Temperature at 25 m" ;
    qc_temp_25m:units = "1" ;
    qc_temp_25m:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_temp_25m:standard_name = "quality_flag" ;
    qc_temp_25m:flag_method = "bit" ;
    qc_temp_25m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
    qc_temp_25m:bit_1_assessment = "Bad" ;
    qc_temp_25m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
    qc_temp_25m:bit_2_assessment = "Indeterminate" ;
    qc_temp_25m:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
    qc_temp_25m:bit_3_assessment = "Bad" ;
    qc_temp_25m:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
    qc_temp_25m:bit_4_assessment = "Bad" ;
int source_temp_25m(time) ;
    source_temp_25m:long_name = "Source for variable: Temperature at 25 m" ;
    source_temp_25m:units = "1" ;
    source_temp_25m:description = "This variable contains integer values which should be
interpreted as listed." ;
    source_temp_25m:flag_method = "integer" ;
    source_temp_25m:flag_0_description = "no_source_available" ;
    source_temp_25m:flag_1_description = "sgptowermetC1.b1:temperature_SE_25m_avg"
;
    source_temp_25m:flag_2_description = "sgptowermetC1.b1:temperature_W_25m_avg" ;
float temp_25m_SE(time) ;
    temp_25m_SE:long_name = "Temperature at 25 m from SE elevator" ;
    temp_25m_SE:units = "degC" ;
    temp_25m_SE:missing_value = -9999.f ;

```

```

temp_25m_SE:valid_min = -50.f ;
temp_25m_SE:valid_max = 60.f ;
temp_25m_SE:standard_name = "air_temperature" ;
temp_25m_SE:ancillary_variables = "qc_temp_25m_SE" ;
temp_25m_SE:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
temp_25m_SE:source = "sgptowermetC1.b1:temperature_SE_25m_avg" ;
int qc_temp_25m_SE(time) ;
    qc_temp_25m_SE:long_name = "Quality check results on variable: Temperature at 25 m
from SE elevator" ;
    qc_temp_25m_SE:units = "1" ;
    qc_temp_25m_SE:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_temp_25m_SE:standard_name = "quality_flag" ;
    qc_temp_25m_SE:flag_method = "bit" ;
    qc_temp_25m_SE:bit_1_description = "Transformation could not finish (all values bad
or outside range, etc.), value set to missing_value." ;
    qc_temp_25m_SE:bit_1_assessment = "Bad" ;
    qc_temp_25m_SE:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
    qc_temp_25m_SE:bit_2_assessment = "Indeterminate" ;
    qc_temp_25m_SE:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
    qc_temp_25m_SE:bit_3_assessment = "Bad" ;
    qc_temp_25m_SE:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
    qc_temp_25m_SE:bit_4_assessment = "Bad" ;
float temp_25m_W(time) ;
    temp_25m_W:long_name = "Temperature at 25 m from W elevator" ;
    temp_25m_W:units = "degC" ;
    temp_25m_W:missing_value = -9999.f ;
    temp_25m_W:valid_min = -50.f ;
    temp_25m_W:valid_max = 60.f ;
    temp_25m_W:standard_name = "air_temperature" ;
    temp_25m_W:ancillary_variables = "qc_temp_25m_W" ;
    temp_25m_W:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
    temp_25m_W:source = "sgptowermetC1.b1:temperature_W_25m_avg" ;
int qc_temp_25m_W(time) ;
    qc_temp_25m_W:long_name = "Quality check results on variable: Temperature at 25 m
from W elevator" ;
    qc_temp_25m_W:units = "1" ;
    qc_temp_25m_W:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_temp_25m_W:standard_name = "quality_flag" ;
    qc_temp_25m_W:flag_method = "bit" ;

```

```

    qc_temp_25m_W:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
    qc_temp_25m_W:bit_1_assessment = "Bad" ;
    qc_temp_25m_W:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
    qc_temp_25m_W:bit_2_assessment = "Indeterminate" ;
    qc_temp_25m_W:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
    qc_temp_25m_W:bit_3_assessment = "Bad" ;
    qc_temp_25m_W:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
    qc_temp_25m_W:bit_4_assessment = "Bad" ;
float temp_60m(time) ;
    temp_60m:long_name = "Temperature at 60 m" ;
    temp_60m:units = "degC" ;
    temp_60m:missing_value = -9999.f ;
    temp_60m:valid_min = -50.f ;
    temp_60m:valid_max = 60.f ;
    temp_60m:standard_name = "air_temperature" ;
    temp_60m:ancillary_variables = "qc_temp_60m source_temp_60m" ;
    temp_60m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_temp_60m(time) ;
    qc_temp_60m:long_name = "Quality check results on variable: Temperature at 60 m" ;
    qc_temp_60m:units = "1" ;
    qc_temp_60m:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_temp_60m:standard_name = "quality_flag" ;
    qc_temp_60m:flag_method = "bit" ;
    qc_temp_60m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
    qc_temp_60m:bit_1_assessment = "Bad" ;
    qc_temp_60m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
    qc_temp_60m:bit_2_assessment = "Indeterminate" ;
    qc_temp_60m:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
    qc_temp_60m:bit_3_assessment = "Bad" ;
    qc_temp_60m:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
    qc_temp_60m:bit_4_assessment = "Bad" ;
int source_temp_60m(time) ;
    source_temp_60m:long_name = "Source for variable: Temperature at 60 m" ;
    source_temp_60m:units = "1" ;
    source_temp_60m:description = "This variable contains integer values which should be
interpreted as listed." ;

```

```

source_temp_60m:flag_method = "integer" ;
source_temp_60m:flag_0_description = "no_source_available" ;
source_temp_60m:flag_1_description = "sgptowermetC1.b1:temperature_SE_60m_avg"
;
source_temp_60m:flag_2_description = "sgptowermetC1.b1:temperature_W_60m_avg" ;
float temp_60m_SE(time) ;
temp_60m_SE:long_name = "Temperature at 60 m from SE elevator" ;
temp_60m_SE:units = "degC" ;
temp_60m_SE:missing_value = -9999.f ;
temp_60m_SE:valid_min = -50.f ;
temp_60m_SE:valid_max = 60.f ;
temp_60m_SE:standard_name = "air_temperature" ;
temp_60m_SE:ancillary_variables = "qc_temp_60m_SE" ;
temp_60m_SE:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
temp_60m_SE:source = "sgptowermetC1.b1:temperature_SE_60m_avg" ;
int qc_temp_60m_SE(time) ;
qc_temp_60m_SE:long_name = "Quality check results on variable: Temperature at 60 m
from SE elevator" ;
qc_temp_60m_SE:units = "1" ;
qc_temp_60m_SE:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_temp_60m_SE:standard_name = "quality_flag" ;
qc_temp_60m_SE:flag_method = "bit" ;
qc_temp_60m_SE:bit_1_description = "Transformation could not finish (all values bad
or outside range, etc.), value set to missing_value." ;
qc_temp_60m_SE:bit_1_assessment = "Bad" ;
qc_temp_60m_SE:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_temp_60m_SE:bit_2_assessment = "Indeterminate" ;
qc_temp_60m_SE:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
qc_temp_60m_SE:bit_3_assessment = "Bad" ;
qc_temp_60m_SE:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
qc_temp_60m_SE:bit_4_assessment = "Bad" ;
float temp_60m_W(time) ;
temp_60m_W:long_name = "Temperature at 60 m from W elevator" ;
temp_60m_W:units = "degC" ;
temp_60m_W:missing_value = -9999.f ;
temp_60m_W:valid_min = -50.f ;
temp_60m_W:valid_max = 60.f ;
temp_60m_W:standard_name = "air_temperature" ;
temp_60m_W:ancillary_variables = "qc_temp_60m_W" ;
temp_60m_W:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
temp_60m_W:source = "sgptowermetC1.b1:temperature_W_60m_avg" ;

```

```

int qc_temp_60m_W(time) ;
    qc_temp_60m_W:long_name = "Quality check results on variable: Temperature at 60 m
from W elevator" ;
    qc_temp_60m_W:units = "1" ;
    qc_temp_60m_W:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_temp_60m_W:standard_name = "quality_flag" ;
    qc_temp_60m_W:flag_method = "bit" ;
    qc_temp_60m_W:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
    qc_temp_60m_W:bit_1_assessment = "Bad" ;
    qc_temp_60m_W:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
    qc_temp_60m_W:bit_2_assessment = "Indeterminate" ;
    qc_temp_60m_W:bit_3_description = "Value is less than the valid_min, value set to
missing_value." ;
    qc_temp_60m_W:bit_3_assessment = "Bad" ;
    qc_temp_60m_W:bit_4_description = "Value is greater than the valid_max, value set to
missing_value." ;
    qc_temp_60m_W:bit_4_assessment = "Bad" ;
float rh_02m(time) ;
    rh_02m:long_name = "Relative humidity at 2 m" ;
    rh_02m:units = "%" ;
    rh_02m:missing_value = -9999.f ;
    rh_02m:standard_name = "relative_humidity" ;
    rh_02m:ancillary_variables = "qc_rh_02m source_rh_02m" ;
    rh_02m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_rh_02m(time) ;
    qc_rh_02m:long_name = "Quality check results on variable: Relative humidity at 2 m" ;
    qc_rh_02m:units = "1" ;
    qc_rh_02m:description = "This variable contains bit-packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_rh_02m:standard_name = "quality_flag" ;
    qc_rh_02m:flag_method = "bit" ;
    qc_rh_02m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
    qc_rh_02m:bit_1_assessment = "Bad" ;
    qc_rh_02m:bit_2_description = "Transformation resulted in an indeterminate outcome." ;
    qc_rh_02m:bit_2_assessment = "Indeterminate" ;
    qc_rh_02m:bit_3_description = "temp_02m assessment is Bad, value set to
missing_value." ;
    qc_rh_02m:bit_3_assessment = "Bad" ;
int source_rh_02m(time) ;
    source_rh_02m:long_name = "Source for variable: Relative humidity at 2 m" ;

```



```

source_rh_02m:units = "1" ;
source_rh_02m:description = "This variable contains integer values which should be
interpreted as listed." ;
source_rh_02m:flag_method = "integer" ;
source_rh_02m:flag_0_description = "no_source_available" ;
source_rh_02m:flag_1_description = "sgpmetE13.b1:rh_mean" ;
source_rh_02m:flag_2_description = "sgpmawsC1.b1:atmospheric_relative_humidity" ;
float rh_02m_preferred(time) ;
rh_02m_preferred:long_name = "Relative humidity at surface from MET" ;
rh_02m_preferred:units = "%" ;
rh_02m_preferred:missing_value = -9999.f ;
rh_02m_preferred:standard_name = "relative_humidity" ;
rh_02m_preferred:ancillary_variables = "qc_rh_02m_preferred" ;
rh_02m_preferred:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
rh_02m_preferred:source = "sgpmetE13.b1:rh_mean" ;
int qc_rh_02m_preferred(time) ;
qc_rh_02m_preferred:long_name = "Quality check results on variable: Relative humidity
at surface from MET" ;
qc_rh_02m_preferred:units = "1" ;
qc_rh_02m_preferred:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_rh_02m_preferred:standard_name = "quality_flag" ;
qc_rh_02m_preferred:flag_method = "bit" ;
qc_rh_02m_preferred:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_rh_02m_preferred:bit_1_assessment = "Bad" ;
qc_rh_02m_preferred:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_rh_02m_preferred:bit_2_assessment = "Indeterminate" ;
qc_rh_02m_preferred:bit_3_description = "temp_02m_preferred assessment is Bad,
value set to missing_value." ;
qc_rh_02m_preferred:bit_3_assessment = "Bad" ;
float rh_02m_alternate(time) ;
rh_02m_alternate:long_name = "Relative humidity at surface from THWAPS/MAWS" ;
rh_02m_alternate:units = "%" ;
rh_02m_alternate:missing_value = -9999.f ;
rh_02m_alternate:standard_name = "relative_humidity" ;
rh_02m_alternate:ancillary_variables = "qc_rh_02m_alternate" ;
rh_02m_alternate:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
rh_02m_alternate:source = "sgpmawsC1.b1:atmospheric_relative_humidity" ;
int qc_rh_02m_alternate(time) ;
qc_rh_02m_alternate:long_name = "Quality check results on variable: Relative humidity
at surface from THWAPS/MAWS" ;
qc_rh_02m_alternate:units = "1" ;

```

```

qc_rh_02m_alternate:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_rh_02m_alternate:standard_name = "quality_flag" ;
qc_rh_02m_alternate:flag_method = "bit" ;
qc_rh_02m_alternate:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_rh_02m_alternate:bit_1_assessment = "Bad" ;
qc_rh_02m_alternate:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_rh_02m_alternate:bit_2_assessment = "Indeterminate" ;
qc_rh_02m_alternate:bit_3_description = "temp_02m_alternate assessment is Bad, value
set to missing_value." ;
qc_rh_02m_alternate:bit_3_assessment = "Bad" ;
float rh_25m(time) ;
rh_25m:long_name = "Relative humidity at 25 m" ;
rh_25m:units = "%" ;
rh_25m:missing_value = -9999.f ;
rh_25m:standard_name = "relative_humidity" ;
rh_25m:ancillary_variables = "qc_rh_25m source_rh_25m" ;
rh_25m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_rh_25m(time) ;
qc_rh_25m:long_name = "Quality check results on variable: Relative humidity at 25 m" ;
qc_rh_25m:units = "1" ;
qc_rh_25m:description = "This variable contains bit-packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_rh_25m:standard_name = "quality_flag" ;
qc_rh_25m:flag_method = "bit" ;
qc_rh_25m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
qc_rh_25m:bit_1_assessment = "Bad" ;
qc_rh_25m:bit_2_description = "Transformation resulted in an indeterminate outcome." ;
qc_rh_25m:bit_2_assessment = "Indeterminate" ;
qc_rh_25m:bit_3_description = "temp_25m assessment is Bad, value set to
missing_value." ;
qc_rh_25m:bit_3_assessment = "Bad" ;
int source_rh_25m(time) ;
source_rh_25m:long_name = "Source for variable: Relative humidity at 25 m" ;
source_rh_25m:units = "1" ;
source_rh_25m:description = "This variable contains integer values which should be
interpreted as listed." ;
source_rh_25m:flag_method = "integer" ;
source_rh_25m:flag_0_description = "no_source_available" ;
source_rh_25m:flag_1_description =
"sgptowermetC1.b1:relative_humidity_SE_25m_avg" ;

```

```

    source_rh_25m:flag_2_description =
"sgptowermetC1.b1:relative_humidity_W_25m_avg" ;
    float rh_25m_SE(time) ;
        rh_25m_SE:long_name = "Relative humidity at 25 m from SE elevator" ;
        rh_25m_SE:units = "%" ;
        rh_25m_SE:missing_value = -9999.f ;
        rh_25m_SE:standard_name = "relative_humidity" ;
        rh_25m_SE:ancillary_variables = "qc_rh_25m_SE" ;
        rh_25m_SE:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
        rh_25m_SE:source = "sgptowermetC1.b1:relative_humidity_SE_25m_avg" ;
    int qc_rh_25m_SE(time) ;
        qc_rh_25m_SE:long_name = "Quality check results on variable: Relative humidity at 25
m from SE elevator" ;
        qc_rh_25m_SE:units = "1" ;
        qc_rh_25m_SE:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
        qc_rh_25m_SE:standard_name = "quality_flag" ;
        qc_rh_25m_SE:flag_method = "bit" ;
        qc_rh_25m_SE:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
        qc_rh_25m_SE:bit_1_assessment = "Bad" ;
        qc_rh_25m_SE:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
        qc_rh_25m_SE:bit_2_assessment = "Indeterminate" ;
        qc_rh_25m_SE:bit_3_description = "temp_25m_SE assessment is Bad, value set to
missing_value." ;
        qc_rh_25m_SE:bit_3_assessment = "Bad" ;
    float rh_25m_W(time) ;
        rh_25m_W:long_name = "Relative humidity at 25 m from W elevator" ;
        rh_25m_W:units = "%" ;
        rh_25m_W:missing_value = -9999.f ;
        rh_25m_W:standard_name = "relative_humidity" ;
        rh_25m_W:ancillary_variables = "qc_rh_25m_W" ;
        rh_25m_W:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
        rh_25m_W:source = "sgptowermetC1.b1:relative_humidity_W_25m_avg" ;
    int qc_rh_25m_W(time) ;
        qc_rh_25m_W:long_name = "Quality check results on variable: Relative humidity at 25
m from W elevator" ;
        qc_rh_25m_W:units = "1" ;
        qc_rh_25m_W:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
        qc_rh_25m_W:standard_name = "quality_flag" ;
        qc_rh_25m_W:flag_method = "bit" ;

```

```

        qc_rh_25m_W:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
        qc_rh_25m_W:bit_1_assessment = "Bad" ;
        qc_rh_25m_W:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
        qc_rh_25m_W:bit_2_assessment = "Indeterminate" ;
        qc_rh_25m_W:bit_3_description = "temp_25m_W assessment is Bad, value set to
missing_value." ;
        qc_rh_25m_W:bit_3_assessment = "Bad" ;
float rh_60m(time) ;
    rh_60m:long_name = "Relative humidity at 60 m" ;
    rh_60m:units = "%" ;
    rh_60m:missing_value = -9999.f ;
    rh_60m:standard_name = "relative_humidity" ;
    rh_60m:ancillary_variables = "qc_rh_60m source_rh_60m" ;
    rh_60m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int source_rh_60m(time) ;
    source_rh_60m:long_name = "Source for variable: Relative humidity at 60 m" ;
    source_rh_60m:units = "1" ;
    source_rh_60m:description = "This variable contains integer values which should be
interpreted as listed." ;
    source_rh_60m:flag_method = "integer" ;
    source_rh_60m:flag_0_description = "no_source_available" ;
    source_rh_60m:flag_1_description =
"sgptowermetC1.b1:relative_humidity_SE_60m_avg" ;
    source_rh_60m:flag_2_description =
"sgptowermetC1.b1:relative_humidity_W_60m_avg" ;
int qc_rh_60m(time) ;
    qc_rh_60m:long_name = "Quality check results on variable: Relative humidity at 60 m" ;
    qc_rh_60m:units = "1" ;
    qc_rh_60m:description = "This variable contains bit-packed integer values, where each
bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description for
those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_rh_60m:standard_name = "quality_flag" ;
    qc_rh_60m:flag_method = "bit" ;
    qc_rh_60m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
    qc_rh_60m:bit_1_assessment = "Bad" ;
    qc_rh_60m:bit_2_description = "Transformation resulted in an indeterminate outcome." ;
    qc_rh_60m:bit_2_assessment = "Indeterminate" ;
    qc_rh_60m:bit_3_description = "temp_60m assessment is Bad, value set to
missing_value." ;
    qc_rh_60m:bit_3_assessment = "Bad" ;
float rh_60m_SE(time) ;
    rh_60m_SE:long_name = "Relative humidity at 60 m from SE elevator" ;
    rh_60m_SE:units = "%" ;

```

```

rh_60m_SE:missing_value = -9999.f;
rh_60m_SE:standard_name = "relative_humidity" ;
rh_60m_SE:ancillary_variables = "qc_rh_60m_SE" ;
rh_60m_SE:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
rh_60m_SE:source = "sgptowermetC1.b1:relative_humidity_SE_60m_avg" ;
int qc_rh_60m_SE(time) ;
  qc_rh_60m_SE:long_name = "Quality check results on variable: Relative humidity at 60
m from SE elevator" ;
  qc_rh_60m_SE:units = "1" ;
  qc_rh_60m_SE:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
  qc_rh_60m_SE:standard_name = "quality_flag" ;
  qc_rh_60m_SE:flag_method = "bit" ;
  qc_rh_60m_SE:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
  qc_rh_60m_SE:bit_1_assessment = "Bad" ;
  qc_rh_60m_SE:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
  qc_rh_60m_SE:bit_2_assessment = "Indeterminate" ;
  qc_rh_60m_SE:bit_3_description = "temp_60m_SE assessment is Bad, value set to
missing_value." ;
  qc_rh_60m_SE:bit_3_assessment = "Bad" ;
float rh_60m_W(time) ;
  rh_60m_W:long_name = "Relative humidity at 60 m from W elevator" ;
  rh_60m_W:units = "%" ;
  rh_60m_W:missing_value = -9999.f;
  rh_60m_W:standard_name = "relative_humidity" ;
  rh_60m_W:ancillary_variables = "qc_rh_60m_W" ;
  rh_60m_W:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
  rh_60m_W:source = "sgptowermetC1.b1:relative_humidity_W_60m_avg" ;
int qc_rh_60m_W(time) ;
  qc_rh_60m_W:long_name = "Quality check results on variable: Relative humidity at 60
m from W elevator" ;
  qc_rh_60m_W:units = "1" ;
  qc_rh_60m_W:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
  qc_rh_60m_W:standard_name = "quality_flag" ;
  qc_rh_60m_W:flag_method = "bit" ;
  qc_rh_60m_W:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
  qc_rh_60m_W:bit_1_assessment = "Bad" ;
  qc_rh_60m_W:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
  qc_rh_60m_W:bit_2_assessment = "Indeterminate" ;

```

```

qc_rh_60m_W:bit_3_description = "temp_60m_W assessment is Bad, value set to
missing_value." ;
qc_rh_60m_W:bit_3_assessment = "Bad" ;
float vap_pres_02m(time) ;
vap_pres_02m:long_name = "Vapor pressure at 2 m" ;
vap_pres_02m:units = "hPa" ;
vap_pres_02m:valid_min = 0.f ;
vap_pres_02m:valid_max = 100.f ;
vap_pres_02m:missing_value = -9999.f ;
vap_pres_02m:standard_name = "water_vapor_partial_pressure_in_air" ;
vap_pres_02m:ancillary_variables = "qc_vap_pres_02m source_vap_pres_02m" ;
vap_pres_02m:cell_transform = "" ;
int qc_vap_pres_02m(time) ;
qc_vap_pres_02m:long_name = "Quality check results on variable: Vapor pressure at 2
m" ;
qc_vap_pres_02m:units = "1" ;
qc_vap_pres_02m:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vap_pres_02m:standard_name = "quality_flag" ;
qc_vap_pres_02m:flag_method = "bit" ;
qc_vap_pres_02m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
qc_vap_pres_02m:bit_1_assessment = "Bad" ;
qc_vap_pres_02m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_vap_pres_02m:bit_2_assessment = "Indeterminate" ;
qc_vap_pres_02m:bit_3_description = "temp_02m assessment is Bad, value set to
missing_value." ;
qc_vap_pres_02m:bit_3_assessment = "Bad" ;
qc_vap_pres_02m:bit_4_description = "Value is less than valid_min, value set to
missing_value." ;
qc_vap_pres_02m:bit_4_assessment = "Bad" ;
qc_vap_pres_02m:bit_4_comment = "This test is only applicable when the source is
maws.b1:atmospheric_dew_point." ;
qc_vap_pres_02m:bit_5_description = "Value is greater than valid_max, value set to
missing_value." ;
qc_vap_pres_02m:bit_5_assessment = "Bad" ;
qc_vap_pres_02m:bit_5_comment = "This test is only applicable when the source is
maws.b1:atmospheric_dew_point." ;
int source_vap_pres_02m(time) ;
source_vap_pres_02m:long_name = "Source for variable: Vapor pressure at 2 m" ;
source_vap_pres_02m:units = "1" ;
source_vap_pres_02m:description = "This variable contains integer values which should
be interpreted as listed." ;
source_vap_pres_02m:flag_method = "integer" ;

```

```

source_vap_pres_02m:flag_0_description = "no_source_available" ;
source_vap_pres_02m:flag_1_description = "sgpmetE13.b1:vapor_pressure_mean" ;
source_vap_pres_02m:flag_2_description = "sgpmawsC1.b1:atmospheric_dew_point" ;
source_vap_pres_02m:flag_2_comment = "If maws.b1 is used as input, the vapor
pressure is calculated from atmospheric_dewpoint" ;
float vap_pres_02m_preferred(time) ;
vap_pres_02m_preferred:long_name = "Vapor pressure at surface from MET" ;
vap_pres_02m_preferred:units = "hPa" ;
vap_pres_02m_preferred:missing_value = -9999.f ;
vap_pres_02m_preferred:standard_name = "water_vapor_partial_pressure_in_air" ;
vap_pres_02m_preferred:ancillary_variables = "qc_vap_pres_02m_preferred" ;
vap_pres_02m_preferred:cell_transform = "time: TRANS_INTERPOLATE (range: 59)"
;
vap_pres_02m_preferred:source = "sgpmetE13.b1:vapor_pressure_mean" ;
int qc_vap_pres_02m_preferred(time) ;
qc_vap_pres_02m_preferred:long_name = "Quality check results on variable: Vapor
pressure at surface from MET" ;
qc_vap_pres_02m_preferred:units = "1" ;
qc_vap_pres_02m_preferred:description = "This variable contains bit-packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vap_pres_02m_preferred:standard_name = "quality_flag" ;
qc_vap_pres_02m_preferred:flag_method = "bit" ;
qc_vap_pres_02m_preferred:bit_1_description = "Transformation could not finish (all
values bad or outside range, etc.), value set to missing_value." ;
qc_vap_pres_02m_preferred:bit_1_assessment = "Bad" ;
qc_vap_pres_02m_preferred:bit_2_description = "Transformation resulted in an
indeterminate outcome." ;
qc_vap_pres_02m_preferred:bit_2_assessment = "Indeterminate" ;
qc_vap_pres_02m_preferred:bit_3_description = "temp_02m_preferred assessment is
Bad, value set to missing_value." ;
qc_vap_pres_02m_preferred:bit_3_assessment = "Bad" ;
float vap_pres_02m_alternate(time) ;
vap_pres_02m_alternate:long_name = "Vapor pressure at surface from
THWAPS/MAWS" ;
vap_pres_02m_alternate:units = "hPa" ;
vap_pres_02m_alternate:valid_min = 0.f ;
vap_pres_02m_alternate:valid_max = 100.f ;
vap_pres_02m_alternate:missing_value = -9999.f ;
vap_pres_02m_alternate:standard_name = "water_vapor_partial_pressure_in_air" ;
vap_pres_02m_alternate:ancillary_variables = "qc_vap_pres_02m_alternate" ;
vap_pres_02m_alternate:cell_transform = "" ;
vap_pres_02m_alternate:source = "sgpmawsC1.b1:atmospheric_dew_point" ;
int qc_vap_pres_02m_alternate(time) ;
qc_vap_pres_02m_alternate:long_name = "Quality check results on variable: Vapor
pressure at surface from THWAPS/MAWS" ;

```

```

qc_vap_pres_02m_alternate:units = "1" ;
qc_vap_pres_02m_alternate:description = "This variable contains bit-packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vap_pres_02m_alternate:standard_name = "quality_flag" ;
qc_vap_pres_02m_alternate:flag_method = "bit" ;
qc_vap_pres_02m_alternate:bit_1_description = "Transformation could not finish (all
values bad or outside range, etc.), value set to missing_value." ;
qc_vap_pres_02m_alternate:bit_1_assessment = "Bad" ;
qc_vap_pres_02m_alternate:bit_2_description = "Transformation resulted in an
indeterminate outcome." ;
qc_vap_pres_02m_alternate:bit_2_assessment = "Indeterminate" ;
qc_vap_pres_02m_alternate:bit_3_description = "temp_02m_alternate assessment is Bad,
value set to missing_value." ;
qc_vap_pres_02m_alternate:bit_3_assessment = "Bad" ;
qc_vap_pres_02m_alternate:bit_4_description = "Value is less than valid_min, value set
to missing_value." ;
qc_vap_pres_02m_alternate:bit_4_assessment = "Bad" ;
qc_vap_pres_02m_alternate:bit_4_comment = "This test is only applicable when the
source is maws.b1:atmospheric_dew_point." ;
qc_vap_pres_02m_alternate:bit_5_description = "Value is greater than valid_max, value
set to missing_value." ;
qc_vap_pres_02m_alternate:bit_5_assessment = "Bad" ;
qc_vap_pres_02m_alternate:bit_5_comment = "This test is only applicable when the
source is maws.b1:atmospheric_dew_point." ;
float vap_pres_25m(time) ;
vap_pres_25m:long_name = "Vapor pressure at 25 m" ;
vap_pres_25m:units = "hPa" ;
vap_pres_25m:missing_value = -9999.f ;
vap_pres_25m:standard_name = "water_vapor_partial_pressure_in_air" ;
vap_pres_25m:ancillary_variables = "qc_vap_pres_25m source_vap_pres_25m" ;
vap_pres_25m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_vap_pres_25m(time) ;
qc_vap_pres_25m:long_name = "Quality check results on variable: Vapor pressure at 25
m" ;
qc_vap_pres_25m:units = "1" ;
qc_vap_pres_25m:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vap_pres_25m:standard_name = "quality_flag" ;
qc_vap_pres_25m:flag_method = "bit" ;
qc_vap_pres_25m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
qc_vap_pres_25m:bit_1_assessment = "Bad" ;
qc_vap_pres_25m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;

```



```

qc_vap_pres_25m:bit_2_assessment = "Indeterminate" ;
qc_vap_pres_25m:bit_3_description = "temp_25m assessment is Bad, value set to
missing_value." ;
qc_vap_pres_25m:bit_3_assessment = "Bad" ;
int source_vap_pres_25m(time) ;
source_vap_pres_25m:long_name = "Source for variable: Vapor pressure at 25 m" ;
source_vap_pres_25m:units = "1" ;
source_vap_pres_25m:description = "This variable contains integer values which should
be interpreted as listed." ;
source_vap_pres_25m:flag_method = "integer" ;
source_vap_pres_25m:flag_0_description = "no_source_available" ;
source_vap_pres_25m:flag_1_description =
"sgptowermetC1.b1:vapor_pressure_SE_25m_avg" ;
source_vap_pres_25m:flag_2_description =
"sgptowermetC1.b1:vapor_pressure_W_25m_avg" ;
float vap_pres_25m_SE(time) ;
vap_pres_25m_SE:long_name = "Vapor pressure at 25 m from SE elevator" ;
vap_pres_25m_SE:units = "hPa" ;
vap_pres_25m_SE:missing_value = -9999.f ;
vap_pres_25m_SE:standard_name = "water_vapor_partial_pressure_in_air" ;
vap_pres_25m_SE:ancillary_variables = "qc_vap_pres_25m_SE" ;
vap_pres_25m_SE:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
vap_pres_25m_SE:source = "sgptowermetC1.b1:vapor_pressure_SE_25m_avg" ;
int qc_vap_pres_25m_SE(time) ;
qc_vap_pres_25m_SE:long_name = "Quality check results on variable: Vapor pressure at
25 m from SE elevator" ;
qc_vap_pres_25m_SE:units = "1" ;
qc_vap_pres_25m_SE:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vap_pres_25m_SE:standard_name = "quality_flag" ;
qc_vap_pres_25m_SE:flag_method = "bit" ;
qc_vap_pres_25m_SE:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_vap_pres_25m_SE:bit_1_assessment = "Bad" ;
qc_vap_pres_25m_SE:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_vap_pres_25m_SE:bit_2_assessment = "Indeterminate" ;
qc_vap_pres_25m_SE:bit_3_description = "temp_25m_SE assessment is Bad, value set
to missing_value." ;
qc_vap_pres_25m_SE:bit_3_assessment = "Bad" ;
float vap_pres_25m_W(time) ;
vap_pres_25m_W:long_name = "Vapor pressure at 25 m from W elevator" ;
vap_pres_25m_W:units = "hPa" ;
vap_pres_25m_W:missing_value = -9999.f ;
vap_pres_25m_W:standard_name = "water_vapor_partial_pressure_in_air" ;

```

```

vap_pres_25m_W:ancillary_variables = "qc_vap_pres_25m_W" ;
vap_pres_25m_W:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
vap_pres_25m_W:source = "sgptowermetC1.b1:vapor_pressure_W_25m_avg" ;
int qc_vap_pres_25m_W(time) ;
  qc_vap_pres_25m_W:long_name = "Quality check results on variable: Vapor pressure at
25 m from W elevator" ;
  qc_vap_pres_25m_W:units = "1" ;
  qc_vap_pres_25m_W:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
  qc_vap_pres_25m_W:standard_name = "quality_flag" ;
  qc_vap_pres_25m_W:flag_method = "bit" ;
  qc_vap_pres_25m_W:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
  qc_vap_pres_25m_W:bit_1_assessment = "Bad" ;
  qc_vap_pres_25m_W:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
  qc_vap_pres_25m_W:bit_2_assessment = "Indeterminate" ;
  qc_vap_pres_25m_W:bit_3_description = "temp_25m_W assessment is Bad, value set to
missing_value." ;
  qc_vap_pres_25m_W:bit_3_assessment = "Bad" ;
float vap_pres_60m(time) ;
  vap_pres_60m:long_name = "Vapor pressure at 60 m" ;
  vap_pres_60m:units = "hPa" ;
  vap_pres_60m:missing_value = -9999.f ;
  vap_pres_60m:standard_name = "water_vapor_partial_pressure_in_air" ;
  vap_pres_60m:ancillary_variables = "qc_vap_pres_60m source_vap_pres_60m" ;
  vap_pres_60m:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
int qc_vap_pres_60m(time) ;
  qc_vap_pres_60m:long_name = "Quality check results on variable: Vapor pressure at 60
m" ;
  qc_vap_pres_60m:units = "1" ;
  qc_vap_pres_60m:description = "This variable contains bit-packed integer values, where
each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the description
for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
  qc_vap_pres_60m:standard_name = "quality_flag" ;
  qc_vap_pres_60m:flag_method = "bit" ;
  qc_vap_pres_60m:bit_1_description = "Transformation could not finish (all values bad or
outside range, etc.), value set to missing_value." ;
  qc_vap_pres_60m:bit_1_assessment = "Bad" ;
  qc_vap_pres_60m:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
  qc_vap_pres_60m:bit_2_assessment = "Indeterminate" ;
  qc_vap_pres_60m:bit_3_description = "temp_60m assessment is Bad, value set to
missing_value." ;
  qc_vap_pres_60m:bit_3_assessment = "Bad" ;

```

```

int source_vap_pres_60m(time) ;
    source_vap_pres_60m:long_name = "Source for variable: Vapor pressure at 60 m" ;
    source_vap_pres_60m:units = "1" ;
    source_vap_pres_60m:description = "This variable contains integer values which should
be interpreted as listed." ;
    source_vap_pres_60m:flag_method = "integer" ;
    source_vap_pres_60m:flag_0_description = "no_source_available" ;
    source_vap_pres_60m:flag_1_description =
"sgptowermetC1.b1:vapor_pressure_SE_60m_avg" ;
    source_vap_pres_60m:flag_2_description =
"sgptowermetC1.b1:vapor_pressure_W_60m_avg" ;
float vap_pres_60m_SE(time) ;
    vap_pres_60m_SE:long_name = "Vapor pressure at 60 m from SE elevator" ;
    vap_pres_60m_SE:units = "hPa" ;
    vap_pres_60m_SE:missing_value = -9999.f ;
    vap_pres_60m_SE:standard_name = "water_vapor_partial_pressure_in_air" ;
    vap_pres_60m_SE:ancillary_variables = "qc_vap_pres_60m_SE" ;
    vap_pres_60m_SE:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
    vap_pres_60m_SE:source = "sgptowermetC1.b1:vapor_pressure_SE_60m_avg" ;
int qc_vap_pres_60m_SE(time) ;
    qc_vap_pres_60m_SE:long_name = "Quality check results on variable: Vapor pressure at
60 m from SE elevator" ;
    qc_vap_pres_60m_SE:units = "1" ;
    qc_vap_pres_60m_SE:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
    qc_vap_pres_60m_SE:standard_name = "quality_flag" ;
    qc_vap_pres_60m_SE:flag_method = "bit" ;
    qc_vap_pres_60m_SE:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
    qc_vap_pres_60m_SE:bit_1_assessment = "Bad" ;
    qc_vap_pres_60m_SE:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
    qc_vap_pres_60m_SE:bit_2_assessment = "Indeterminate" ;
    qc_vap_pres_60m_SE:bit_3_description = "temp_60m_SE assessment is Bad, value set
to missing_value." ;
    qc_vap_pres_60m_SE:bit_3_assessment = "Bad" ;
float vap_pres_60m_W(time) ;
    vap_pres_60m_W:long_name = "Vapor pressure at 60 m from W elevator" ;
    vap_pres_60m_W:units = "hPa" ;
    vap_pres_60m_W:missing_value = -9999.f ;
    vap_pres_60m_W:standard_name = "water_vapor_partial_pressure_in_air" ;
    vap_pres_60m_W:ancillary_variables = "qc_vap_pres_60m_W" ;
    vap_pres_60m_W:cell_transform = "time: TRANS_INTERPOLATE (range: 59)" ;
    vap_pres_60m_W:source = "sgptowermetC1.b1:vapor_pressure_W_60m_avg" ;
int qc_vap_pres_60m_W(time) ;

```

```

qc_vap_pres_60m_W:long_name = "Quality check results on variable: Vapor pressure at
60 m from W elevator" ;
qc_vap_pres_60m_W:units = "1" ;
qc_vap_pres_60m_W:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_vap_pres_60m_W:standard_name = "quality_flag" ;
qc_vap_pres_60m_W:flag_method = "bit" ;
qc_vap_pres_60m_W:bit_1_description = "Transformation could not finish (all values
bad or outside range, etc.), value set to missing_value." ;
qc_vap_pres_60m_W:bit_1_assessment = "Bad" ;
qc_vap_pres_60m_W:bit_2_description = "Transformation resulted in an indeterminate
outcome." ;
qc_vap_pres_60m_W:bit_2_assessment = "Indeterminate" ;
qc_vap_pres_60m_W:bit_3_description = "temp_60m_W assessment is Bad, value set to
missing_value." ;
qc_vap_pres_60m_W:bit_3_assessment = "Bad" ;
float mixing_ratio_02m(time) ;
mixing_ratio_02m:long_name = "Water vapor mixing ratio at 2 m" ;
mixing_ratio_02m:units = "g/kg" ;
mixing_ratio_02m:missing_value = -9999.f ;
mixing_ratio_02m:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_02m:ancillary_variables = "qc_mixing_ratio_02m" ;
int qc_mixing_ratio_02m(time) ;
qc_mixing_ratio_02m:long_name = "Quality check results on variable: Water vapor
mixing ratio at 2 m" ;
qc_mixing_ratio_02m:units = "1" ;
qc_mixing_ratio_02m:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_02m:standard_name = "quality_flag" ;
qc_mixing_ratio_02m:flag_method = "bit" ;
qc_mixing_ratio_02m:bit_1_description = "pres_02m or vap_pres_02m have a Bad
assessment, value set to missing_value." ;
qc_mixing_ratio_02m:bit_1_assessment = "Bad" ;
qc_mixing_ratio_02m:bit_2_description = "pres_02m or vap_pres_02m have an
Indeterminate assessment." ;
qc_mixing_ratio_02m:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_02m_preferred(time) ;
mixing_ratio_02m_preferred:long_name = "Water vapor mixing ratio at surface from
MET" ;
mixing_ratio_02m_preferred:units = "g/kg" ;
mixing_ratio_02m_preferred:missing_value = -9999.f ;
mixing_ratio_02m_preferred:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_02m_preferred:ancillary_variables = "qc_mixing_ratio_02m_preferred" ;
int qc_mixing_ratio_02m_preferred(time) ;

```

```

qc_mixing_ratio_02m_preferred:long_name = "Quality check results on variable: Water
vapor mixing ratio at surface from MET" ;
qc_mixing_ratio_02m_preferred:units = "1" ;
qc_mixing_ratio_02m_preferred:description = "This variable contains bit-packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_02m_preferred:standard_name = "quality_flag" ;
qc_mixing_ratio_02m_preferred:flag_method = "bit" ;
qc_mixing_ratio_02m_preferred:bit_1_description = "pres_02m_preferred or
vap_pres_02m_preferred have a Bad assessment, value set to missing_value." ;
qc_mixing_ratio_02m_preferred:bit_1_assessment = "Bad" ;
qc_mixing_ratio_02m_preferred:bit_2_description = "pres_02m_preferred or
vap_pres_02m_preferred have an Indeterminate assessment." ;
qc_mixing_ratio_02m_preferred:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_02m_alternate(time) ;
mixing_ratio_02m_alternate:long_name = "Water vapor mixing ratio at surface from
THWAPS/MAWS" ;
mixing_ratio_02m_alternate:units = "g/kg" ;
mixing_ratio_02m_alternate:missing_value = -9999.f ;
mixing_ratio_02m_alternate:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_02m_alternate:ancillary_variables = "qc_mixing_ratio_02m_alternate" ;
int qc_mixing_ratio_02m_alternate(time) ;
qc_mixing_ratio_02m_alternate:long_name = "Quality check results on variable: Water
vapor mixing ratio at surface from THWAPS/MAWS" ;
qc_mixing_ratio_02m_alternate:units = "1" ;
qc_mixing_ratio_02m_alternate:description = "This variable contains bit-packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_02m_alternate:standard_name = "quality_flag" ;
qc_mixing_ratio_02m_alternate:flag_method = "bit" ;
qc_mixing_ratio_02m_alternate:bit_1_description = "pres_02m_alternate or
vap_pres_02m_alternate have a Bad assessment, value set to missing_value." ;
qc_mixing_ratio_02m_alternate:bit_1_assessment = "Bad" ;
qc_mixing_ratio_02m_alternate:bit_2_description = "pres_02m_alternate or
vap_pres_02m_alternate have an Indeterminate assessment." ;
qc_mixing_ratio_02m_alternate:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_25m(time) ;
mixing_ratio_25m:long_name = "Water vapor mixing ratio at 25 m" ;
mixing_ratio_25m:units = "g/kg" ;
mixing_ratio_25m:missing_value = -9999.f ;
mixing_ratio_25m:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_25m:ancillary_variables = "qc_mixing_ratio_25m" ;
int qc_mixing_ratio_25m(time) ;
qc_mixing_ratio_25m:long_name = "Quality check results on variable: Water vapor
mixing ratio at 25 m" ;
qc_mixing_ratio_25m:units = "1" ;

```

```

qc_mixing_ratio_25m:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_25m:standard_name = "quality_flag" ;
qc_mixing_ratio_25m:flag_method = "bit" ;
qc_mixing_ratio_25m:bit_1_description = "pres_25m or vap_pres_25m have a Bad
assessment, value set to missing_value." ;
qc_mixing_ratio_25m:bit_1_assessment = "Bad" ;
qc_mixing_ratio_25m:bit_2_description = "pres_25m or vap_pres_25m have an
Indeterminate assessment." ;
qc_mixing_ratio_25m:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_25m_SE(time) ;
mixing_ratio_25m_SE:long_name = "Water vapor mixing ratio at 25 m from SE
elevator" ;
mixing_ratio_25m_SE:units = "g/kg" ;
mixing_ratio_25m_SE:missing_value = -9999.f ;
mixing_ratio_25m_SE:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_25m_SE:ancillary_variables = "qc_mixing_ratio_25m_SE" ;
int qc_mixing_ratio_25m_SE(time) ;
qc_mixing_ratio_25m_SE:long_name = "Quality check results on variable: Water vapor
mixing ratio at 25 m from SE elevator" ;
qc_mixing_ratio_25m_SE:units = "1" ;
qc_mixing_ratio_25m_SE:description = "This variable contains bit-packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_25m_SE:standard_name = "quality_flag" ;
qc_mixing_ratio_25m_SE:flag_method = "bit" ;
qc_mixing_ratio_25m_SE:bit_1_description = "pres_25m_SE or vap_pres_25m_SE have
a Bad assessment, value set to missing_value." ;
qc_mixing_ratio_25m_SE:bit_1_assessment = "Bad" ;
qc_mixing_ratio_25m_SE:bit_2_description = "pres_25m_SE or vap_pres_25m_SE have
an Indeterminate assessment." ;
qc_mixing_ratio_25m_SE:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_25m_W(time) ;
mixing_ratio_25m_W:long_name = "Water vapor mixing ratio at 25 m from W elevator"
;
mixing_ratio_25m_W:units = "g/kg" ;
mixing_ratio_25m_W:missing_value = -9999.f ;
mixing_ratio_25m_W:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_25m_W:ancillary_variables = "qc_mixing_ratio_25m_W" ;
int qc_mixing_ratio_25m_W(time) ;
qc_mixing_ratio_25m_W:long_name = "Quality check results on variable: Water vapor
mixing ratio at 25 m from W elevator" ;
qc_mixing_ratio_25m_W:units = "1" ;

```

```

qc_mixing_ratio_25m_W:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_25m_W:standard_name = "quality_flag" ;
qc_mixing_ratio_25m_W:flag_method = "bit" ;
qc_mixing_ratio_25m_W:bit_1_description = "pres_25m_W or vap_pres_25m_W have a
Bad assessment, value set to missing_value." ;
qc_mixing_ratio_25m_W:bit_1_assessment = "Bad" ;
qc_mixing_ratio_25m_W:bit_2_description = "pres_25m_W or vap_pres_25m_W have
an Indeterminate assessment." ;
qc_mixing_ratio_25m_W:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_60m(time) ;
mixing_ratio_60m:long_name = "Water vapor mixing ratio at 60 m" ;
mixing_ratio_60m:units = "g/kg" ;
mixing_ratio_60m:missing_value = -9999.f ;
mixing_ratio_60m:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_60m:ancillary_variables = "qc_mixing_ratio_60m" ;
int qc_mixing_ratio_60m(time) ;
qc_mixing_ratio_60m:long_name = "Quality check results on variable: Water vapor
mixing ratio at 60 m" ;
qc_mixing_ratio_60m:units = "1" ;
qc_mixing_ratio_60m:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_60m:standard_name = "quality_flag" ;
qc_mixing_ratio_60m:flag_method = "bit" ;
qc_mixing_ratio_60m:bit_1_description = "pres_60m or vap_pres_60m have a Bad
assessment, value set to missing_value." ;
qc_mixing_ratio_60m:bit_1_assessment = "Bad" ;
qc_mixing_ratio_60m:bit_2_description = "pres_60m or vap_pres_60m have an
Indeterminate assessment." ;
qc_mixing_ratio_60m:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_60m_SE(time) ;
mixing_ratio_60m_SE:long_name = "Water vapor mixing ratio at 60 m from SE
elevator" ;
mixing_ratio_60m_SE:units = "g/kg" ;
mixing_ratio_60m_SE:missing_value = -9999.f ;
mixing_ratio_60m_SE:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_60m_SE:ancillary_variables = "qc_mixing_ratio_60m_SE" ;
int qc_mixing_ratio_60m_SE(time) ;
qc_mixing_ratio_60m_SE:long_name = "Quality check results on variable: Water vapor
mixing ratio at 60 m from SE elevator" ;
qc_mixing_ratio_60m_SE:units = "1" ;
qc_mixing_ratio_60m_SE:description = "This variable contains bit-packed integer
values, where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in
the description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;

```

```

qc_mixing_ratio_60m_SE:standard_name = "quality_flag" ;
qc_mixing_ratio_60m_SE:flag_method = "bit" ;
qc_mixing_ratio_60m_SE:bit_1_description = "pres_60m_SE or vap_pres_60m_SE have
a Bad assessment, value set to missing_value." ;
qc_mixing_ratio_60m_SE:bit_1_assessment = "Bad" ;
qc_mixing_ratio_60m_SE:bit_2_description = "pres_60m_SE or vap_pres_60m_SE have
an Indeterminate assessment." ;
qc_mixing_ratio_60m_SE:bit_2_assessment = "Indeterminate" ;
float mixing_ratio_60m_W(time) ;
mixing_ratio_60m_W:long_name = "Water vapor mixing ratio at 60 m from W elevator"
;

mixing_ratio_60m_W:units = "g/kg" ;
mixing_ratio_60m_W:missing_value = -9999.f ;
mixing_ratio_60m_W:standard_name = "humidity_mixing_ratio" ;
mixing_ratio_60m_W:ancillary_variables = "qc_mixing_ratio_60m_W" ;
int qc_mixing_ratio_60m_W(time) ;
qc_mixing_ratio_60m_W:long_name = "Quality check results on variable: Water vapor
mixing ratio at 60 m from W elevator" ;
qc_mixing_ratio_60m_W:units = "1" ;
qc_mixing_ratio_60m_W:description = "This variable contains bit-packed integer values,
where each bit represents a QC test on the data. Non-zero bits indicate the QC condition given in the
description for those bits; a value of 0 (no bits set) indicates the data has not failed any QC tests." ;
qc_mixing_ratio_60m_W:standard_name = "quality_flag" ;
qc_mixing_ratio_60m_W:flag_method = "bit" ;
qc_mixing_ratio_60m_W:bit_1_description = "pres_60m_W or vap_pres_60m_W have a
Bad assessment, value set to missing_value." ;
qc_mixing_ratio_60m_W:bit_1_assessment = "Bad" ;
qc_mixing_ratio_60m_W:bit_2_description = "pres_60m_W or vap_pres_60m_W have
an Indeterminate assessment." ;
qc_mixing_ratio_60m_W:bit_2_assessment = "Indeterminate" ;
float lat ;
lat:long_name = "North latitude" ;
lat:units = "degree_N" ;
lat:valid_min = -90.f ;
lat:valid_max = 90.f ;
lat:standard_name = "latitude" ;
float lon ;
lon:long_name = "East longitude" ;
lon:units = "degree_E" ;
lon:valid_min = -180.f ;
lon:valid_max = 180.f ;
lon:standard_name = "longitude" ;
float alt ;
alt:long_name = "Altitude above mean sea level" ;
alt:units = "m" ;
alt:standard_name = "altitude" ;

```



```
// global attributes:
:command_line = "twrmr -s sgp -f C1 -b 20210414 -R" ;
:Conventions = "ARM-1.3" ;
:process_version = "vap-twrmr-1.1-0.e17" ;
:dod_version = "1twrmr-c1-1.3" ;
:input_datastreams = "sgpmetE13.b1 : 4.45 : 20210414.000000\n",
                    "sgpmawsC1.b1 : 1.0 : 20210414.000007\n",
                    "sgptowermetC1.b1 : 1.1 : 20210414.000000" ;
:site_id = "sgp" ;
:platform_id = "1twrmr" ;
:facility_id = "C1" ;
:data_level = "c1" ;
:location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
:datastream = "sgp1twrmrC1.c1" ;
:averaging_interval = "1 min" ;
:doi = "10.5439/1027267" ;
:surface_selection_order = "MET used as primary surface obs, THWAPS as secondary,
MAWS as tertiary" ;
:tower_selection_order = "SE elevator used as primary tower obs, W elevator as
secondary" ;
:history = "created by user dsmgr on machine zinc at 2021-04-19 18:36:25, using vap-
twrmr-1.1-0.e17" ;
}
```



U.S. DEPARTMENT OF
ENERGY

Office of Science