Weather Station:
(–PDR–)

Peter Teuben

Version 2.2

Updated: February 3, 2005

1 Functionality and Scope

1.1 Overview

The weather station provides the observatory with basic weather information (temperature, pressure, wind speed and direction), and publishes the variables (and perhaps some derived ones) to the standard CARMA monitor system. The ambient temperature is important to calculate T_{sys}, and must thus be accurate. Equally so, the pressure determines the amount of refraction (several arcmin at low elevation), and depending on required accuracy the temperature, humidity level and observing frequency are needed.

The current SZA weatherstation has been in operation for a while, and an identical copy has been used to sample data at the high site. We plan to use this. In addition, there is a separate dew point sensor for very accurate measurements of the ambient and dewpoint temperature (most simple weatherstations have questionable humidity sensors).

The weather station control program can run on any Linux box, as long as a serial port is available (but see also comments on CAN bus below). The same holds for the dew point sensor.

The opacity monitor (tipper) is part of another package.

Note there is no corresponding hardware WP on weather.

1.2 Weather station variables

The following variables are read from the weather station:

```
WX_STN_ID,     /* The station ID character (110) */
WX_YEAR,       /* The year number (eg. 1998) */
WX_DAY,        /* The day number (eg. 232) */
WX_TIME,       /* The time-of-day (eg. 1934 [ie. 19 hours, 34 minutes]) */
WX_INT_TEMP,   /* The internal temperature of the weather station (C) */
WX_BATTERY,    /* The battery voltage (V) */
WX_AIR_TEMP,   /* The outside temperature (C) */
WX_HUMIDITY,   /* The relative humidity (%) */
WX_WIND_SPEED, /* The wind speed (m/s) */
WX_WIND_DIR,   /* The wind direction (degrees East of North) */
WX_PRESSURE,   /* The atmospheric pressure (mb) */
```

1.3 A little weather background

At low altitudes our atmosphere has roughly an exponential scaleheight of 8.3km, thus the barometric pressure, to first order, can be given by as a function of the altitude as follows:

$$P(alt) \approx 1000 \quad e^{-alt[m]/8300}[mbar]$$

The dew point temperature is defined as the temperature to which the air would have to cool (at constant pressure and constant water vapor content) in order to reach saturation.

1.4 Control Input

The weather station is an autonomous unit and does not need to be set into an operational mode. Hence no IDL code is needed.

1.5 Output

Software output consists of monitor data and occasional logging. The monitor data include temperature, pressure etc. The monitor points are described in detail in Section 2.2. Logging can occur if the weather station looses power (e.g. low voltage battery) Also, dangerous weather conditions should flag alarms and possible put the array into a "safe mode".

1.6 Administrative Summary

1.6.1 Estimated FTE Effort

Design:	0.25	
Implementation:	0.5	
Testing and integration:	0.5	(numbers are in man-months)
	—-	
TOTAL:	1.25	

1.6.2 Schedule

Package status:	open	
Work Package Analysis:	completed:	08/27/2002
Conceptual design:	completed:	n/a
Preliminary design:	completed:	02/03/2005
Critical design:	completed:	03/31/2005
simulated data		02/04/2005
peter visit OVRO?		03/13/2005

2 Design

Monitoring functions for the weather station will be realized through a simple RS232 serial port. New code is needed for publishing the associated CORBA distributed object (DO) that external clients contact to activate its control functionality.

2.1 Control API

The weather station has no controls, polling occurs on the RS232 port, though the firmware is said to be enablable in an asynchronous mode

2.2 Monitor Points

The following is the complete set of monitor points in the `carma.weather` hierarchy (or should this be `carma.environment.weatherStation`). All monitor points should be published at a 2 Hz rate.

- `ambientTemperature` (type: *float*; units: *K*)
  Outside ambient temperature.

- `stationTemperature` (type: *float*; units: *K*)
  Internal temperature of the weather station.
- `dewpointTemperature` (type: *float*; units: *K*)
  Dewpoint temperature, has to be derived.
- `pressure` (type: *float*; units: *mbar*)
  Barometric pressure, in mbar.
- `windSpeed` (type: *float*; units: *m/s*)
  Wind speed, in m/s .
- `windDirection` (type: *float*; units: *degrees*)
  Degrees, East of North. Runs between -360 and 360, but is normalized to 0..360 as a monitor point.
- `relHumidity` (type: *float*; units: *%*)
  Relative Humidity, between 0 and 100%.
- `mmh2o` (type: *float*; units: *m*)
  precipitable water vapor. Has to be derived. Note the carma units, despite that we commonly refer to this is "mm of water".

3 Implementation

User-level access to the weather system is provided by a C++ class named Weather, which uses the SZA weather code.

3.1 Code Reuse

SZA is using the same instrument (two identical systems were baught in year-????). Erik reported that in theory the firmware can be modified to have the weatherstation asynchronously send weather info at a pre-programmed rate (via firmware settings, not something we would want to control via IDL). This was never working according to advertised interfaces. However it's working fine by polling the serial port.

Some of the old code on the old OVRO WindowsNT3.5 based weather station box probably has some algorithms for the derived quantities. Otherwise no code reuse expected here.

4 Unresolved Issues/Notes

- battery : it can also be monitored, returns battery voltage.
- straight RS232 vs. CAN micro (via an XAC3 module?)
- 2HZ monitoring on RS232 (9600baud) should be ok.
- conversion of RS232 to fiber to avoid lightning damage to the computer (wouldn't you think these are protected against that?)
- dewpoint sensor has not been discussed
- rapid variable wind vs. slowly varying temp/press
- averaging here, and publish as a monitor point, or let the monitor system do this?
- water vapor radiometer: need the old bulky manuals back to Maryland? visit OVRO?
- wx200d-1.3 (wx200d.sourceforge.net, $200; open2300.sourceforge.net)

- differential refraction; override if bad weather info?
- overall weather quality?