

Appendix 4

Weather

Using weather data in your automation solution can have many benefits. Without weather data, your home automation happens regardless of environmental conditions. Some things you can use weather data for are:

- Don't run sprinkler systems when too hot or too windy
- Retract awnings when too windy
- When very hot or very cold adjust the thermostat setpoint to be a bit higher or a bit lower. In doing this you trade off a bit of comfort for what can be substantial energy savings
- Close remote control skylights when it's raining
- Adjust indoor lighting to make the house appear warmer when it's cold and cooler when it's hot. Your perception of the temperature varies with many environmental factors

This appendix describes how to work with weather data and covers these topics:

- What weather providers HCA supports
- Configuring HCA to use Weather Underground as a weather provider
- Configuring HCA to use the output of another program for current weather data.
- Configuring HCA to use METAR observations
- Configuring HCA to add weather observation data to a database file. This database is used for tests of historical weather data – like the max temperature in the last 4 hours
- The Visual Programmer weather test element
- Weather program triggers
- Additional support for Weather Underground

Weather Providers

HCA accepts weather data from three main types of sources. These are:

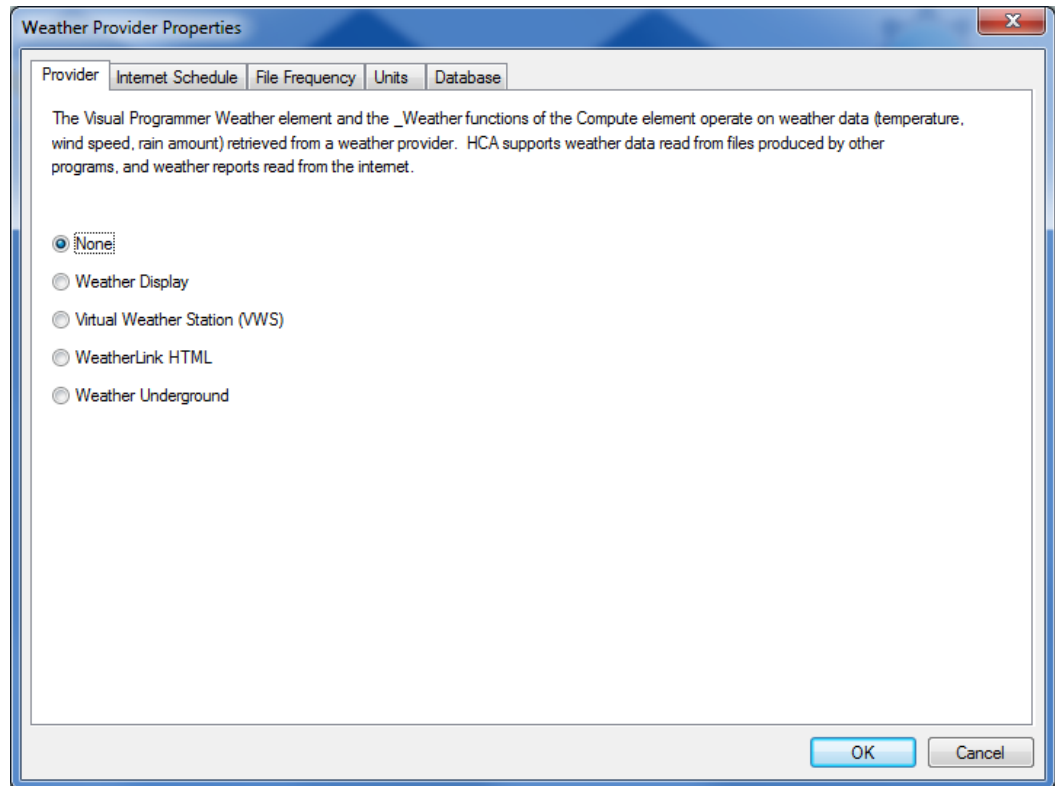
- HCA can retrieve weather data from the *Weather Underground* web site. You must first establish an account with them but once created, then you have access to not only current weather data – temperature, humidity - but also to forecasts and astronomical data.
- HCA reads a file that contains the weather data. These files are typically produced by another program. HCA can read these files in several formats. These are the formats produced by three programs that work with weather data: Virtual Weather Station, Weather Display, and the Davis WeatherLink program.
- HCA can retrieve weather data from the Internet using what are known as METAR reports. These are weather observations taken at airports across the country.

Once you have configured a weather provider, at periodic intervals weather data is retrieved. Each time this happens it is called an *Observation*. Contained in a weather observation are all the sensor data - temperature, barometer, wind speed, etc.

If you are new to working with weather data we recommend that you use the *Weather Underground* provider even if you own your own weather station. Your weather station can publish its data to Weather Underground – a feature of many weather programs - and then you can access that data from them. Working this way you also gain access to all that Weather Underground offers in addition to your own station data.

Weather Provider Setup

To setup a weather provider, press the *Weather Provider Setup* button in the ribbon *Interfaces* category. This weather provider dialog opens.



The setup options vary with the chosen weather provider. Refer to the next sections for setup of each weather provider.

In addition to choosing the weather provider and its setup options, there are three other items to configure:

- The units to use for the sensor data – F or C, inches or mm, etc.
- How often a weather observation is taken
- The location of the database files that contain the historical weather data

These are covered in sections below.

Weather Underground

The Weather Underground web site (<http://www.wunderground.com/>) provides weather data to applications and that data can be accessed by HCA for your use.

Before you can use Weather Underground as a weather provider you need to sign up as a developer. What Weather Underground is doing is making weather data available to people creating applications, but since each of you are a “developer” of your own home design you should have your own key.

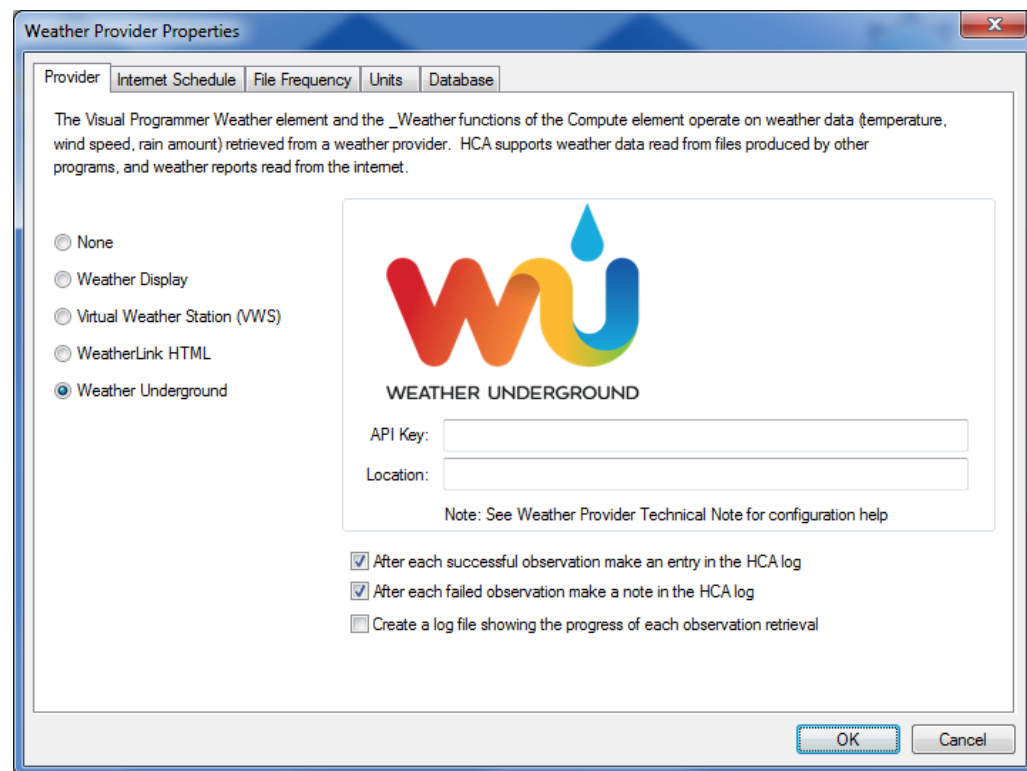
Signup for the key is here: <http://www.wunderground.com/weather/api/>

Complete the signup for the API Key and save the API Key they send in a safe place. During the signup process you may also want to do a quick review of the documentation that Weather Underground provides. This is important if you want to use Weather Underground beyond the standard weather data retrieval that HCA automatically performs.

Once you have your API Key, the next thing to do is to determine a location. Go here:

<http://www.wunderground.com/weather/api/d/docs?d=data/index>

Look at the section on “Query”. There are lots of different ways to specify your location. The easiest is to use your zip-code. Once you have a key and have determined a location you are ready to configure HCA.



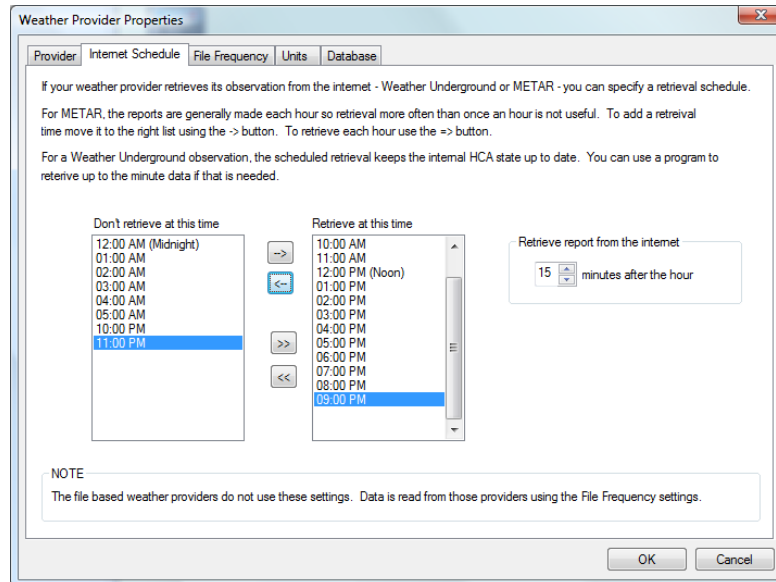
Enter your API key and location. No verification is done on either so enter them carefully.

Also in this tab are additional options. After each weather observation is taken a program can be started. You can also make entries in the HCA log (always the first of the three possible logs) after each observation.

If you choose to create a log file – last checkbox on this tab – it is located in the *Logs* folder and is called “WeatherLog.txt”

The *Internet Schedule* tab sets the schedule to be used to retrieve observations from Weather Underground. The *File Frequency* tab is used for file based providers and has no effect on Weather Underground.

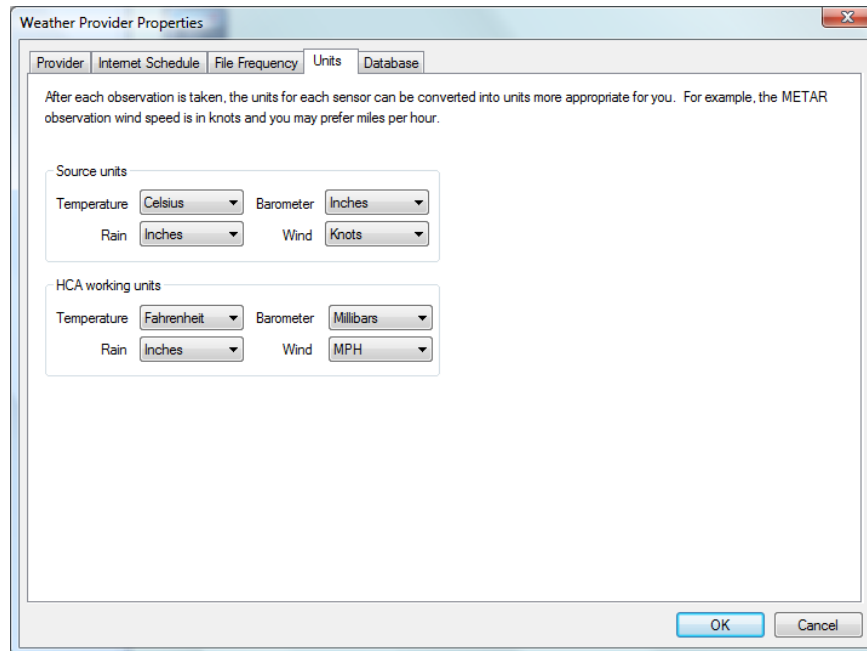
Using the schedule you can set the times during the day that weather observations are taken.



As the text in this tab explains, HCA goes out to the internet at most once an hour to retrieve the current weather observation. There are items to configure in this dialog:

- When during the hour to read the data. This is given as the number of minutes past the hour. In this example, the weather data is retrieved at, for example, 9:15 or 11:15, etc.
- What hours of the day to retrieve the data. In the above example, the data is retrieved every hour except between 10pm and 6am.

The Units tab is where you specify the units of measurement the data from the provider are in, as well as the units that you want to work in your programs. For example, while the provider may report temperatures in Fahrenheit you may want to work in Centigrade. HCA translates the source units to the working units as needed.



In the dialog above, the weather provider supplies temperature in degrees Fahrenheit, rain in inches, barometric pressure in inches and wind speed in knots. In this example, when working with weather data in the *Weather* element and weather triggers, temperature values are in degrees Fahrenheit, rain in inches, barometric pressure in millibars and wind speed in miles per hour.

When using Weather Underground, on the units tab make sure that the “source” units are configured as:

- Temperature: F
- Rain: Inches
- Barometer: Inches
- Wind: MPH

For other providers, you will have to determine the source unit choices based upon documentation or on examination of the data..

When the schedule tells HCA to take an observation – in this case retrieval of data from Weather Underground - an XML file containing the data is saved in the *Temp* folder in the HCA documents area.

After each observation is taken HCA extracts the data for these variables:

- Outside temperature
- Relative humidity
- Barometer (inches)
- Barometer Trend
- Wind speed (mph)
- Wind Dir (degrees)
- Wind Direction (string)
- Wind Chill
- Daily rain (precipitation today inches)

- Rain rate (1 hour Precipitation)
- Dewpoint
- Wind Gust (mph)
- WeatherCondition (string)

All these various values are extracted and are available to the *Weather* program element – described later in this appendix.. You can also use the `_Weather` function –described in the *Expressions* chapter - to operate on these values.

Reading data from a file

HCA can read data from three different file formats produced by three different programs that run on the same computer as HCA or on a different computer but create the file in a network location that HCA can access. You must configure these programs to produce the file that is read by HCA.

WeatherLink Program

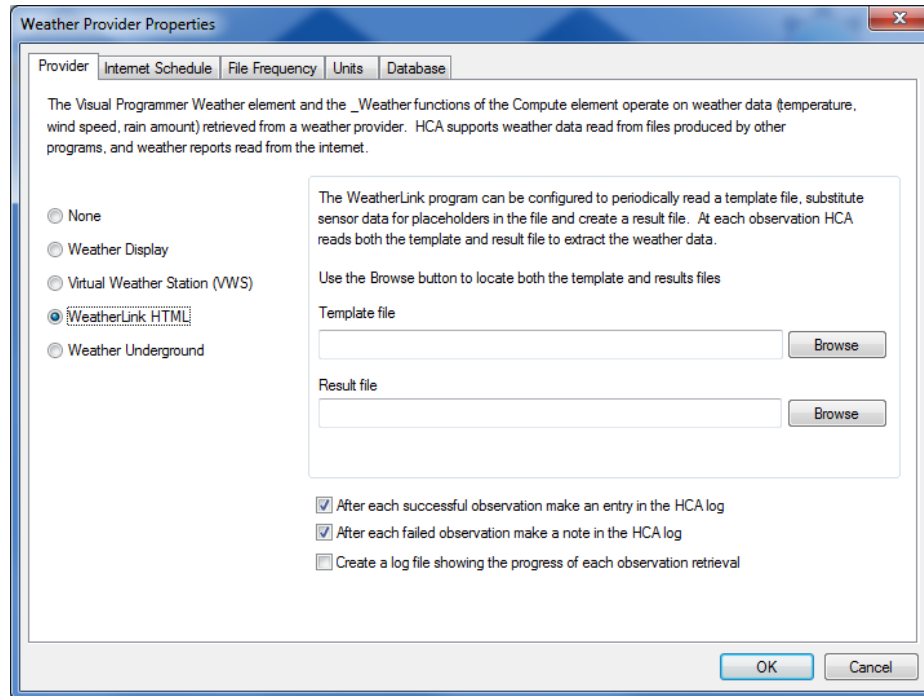
The WeatherLink program from Davis Instruments can be configured to produce a file on a schedule you set. That file is read by HCA and the weather data values extracted.

In a very similar manner as the HCA *Status Export* feature, WeatherLink reads a template file and produces a result file while replacing placeholders with data read from the weather station data.

The list of placeholders available in WeatherLink template files is very large. There are over 250 possible placeholders.

HCA uses both the template and results file produced to extract the weather data.

To setup the WeatherLink HTML as the weather data provider, select WeatherLink HTML and the setup dialog shows those settings:



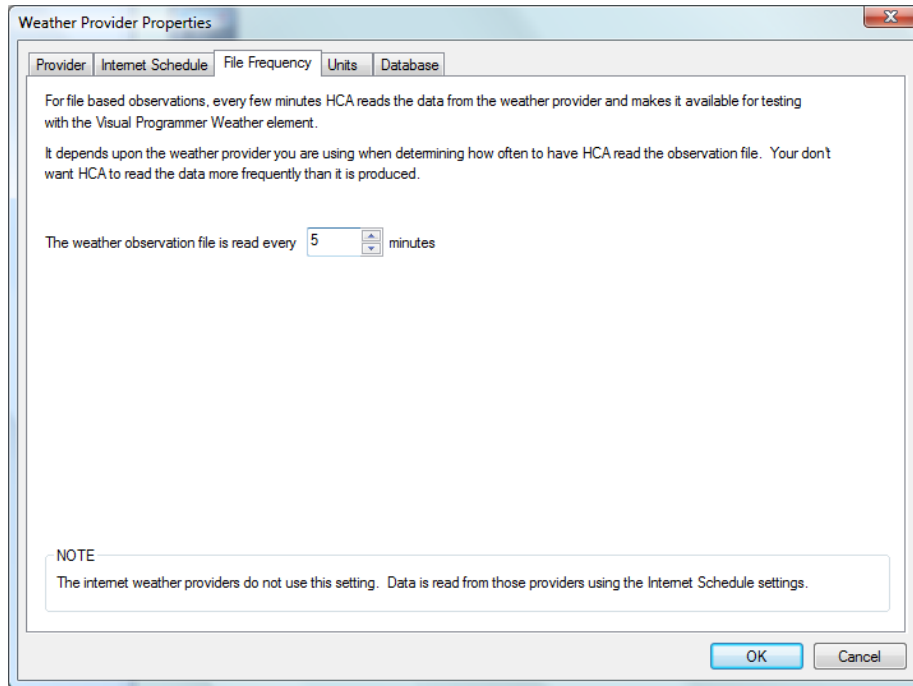
The two setup parameters are the paths to the template file and the result files.

To configure WeatherLink to produce the HTML files and what the placeholder keys are, see the WeatherLink documentation. Look in the *Internet Settings* sections.

Optionally, after the file is read a program can be started. You can also make entries in the HCA log (always the first of the three possible logs) after each observation.

Tip: Extracting the data from the file produced by WeatherLink can be difficult. In the HCA installation is a Windows program called *WTest* that lets you enter the path to the two files and then performs and displays the decoded information. This can be a good way to check the extraction of the data.

On the *File Schedule* tab is specified how often the data file is read.

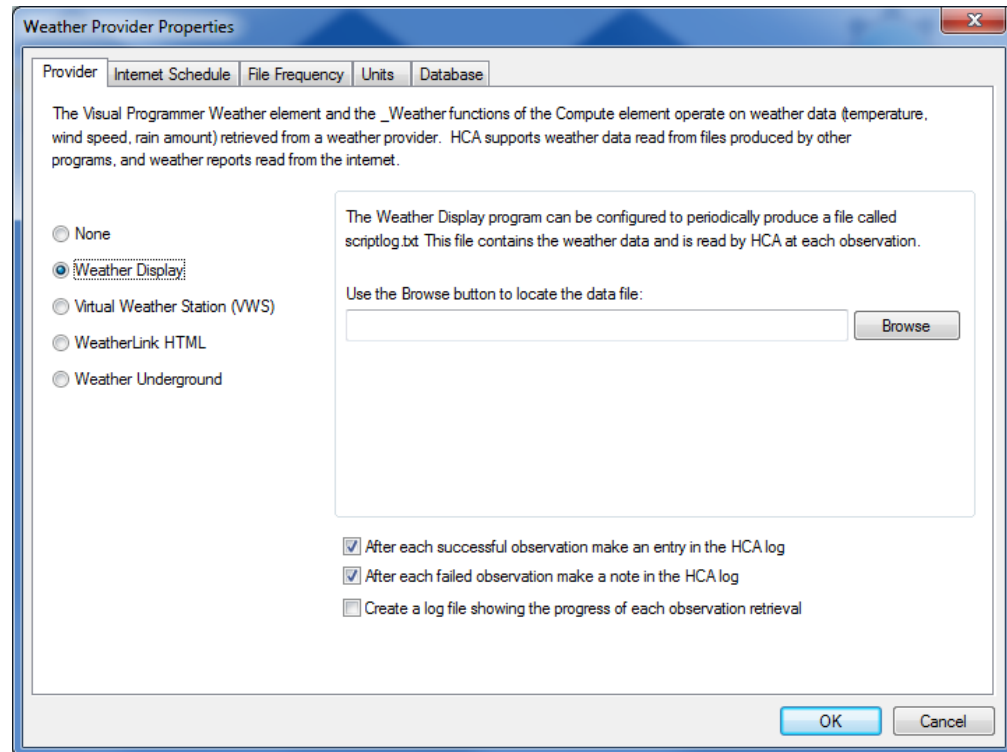


How often the file is read should approximate how often the file is produced by the program creating it.

When working with the *WeatherLink* program you should also set the observation units and working units on the *Units* tab as described above in the Weather Underground section.

Weather Display Program

The Weather Display program periodically writes a file containing the current weather sensor data. When you select this option the setup dialog settings are:



All that is needed is the path to the file produced by *Weather Display*.

To configure *Weather Display* to produce the export file, it is all done from Weather Display menu selections. Select from the *Weather Display* menu:

```
Setup
Log file recording
Produce a 1 line log output for scripting (scriptlog.txt)
Yes
```

The exported file, scriptlog.txt, is placed in the *Weather Display* installation log files folder.

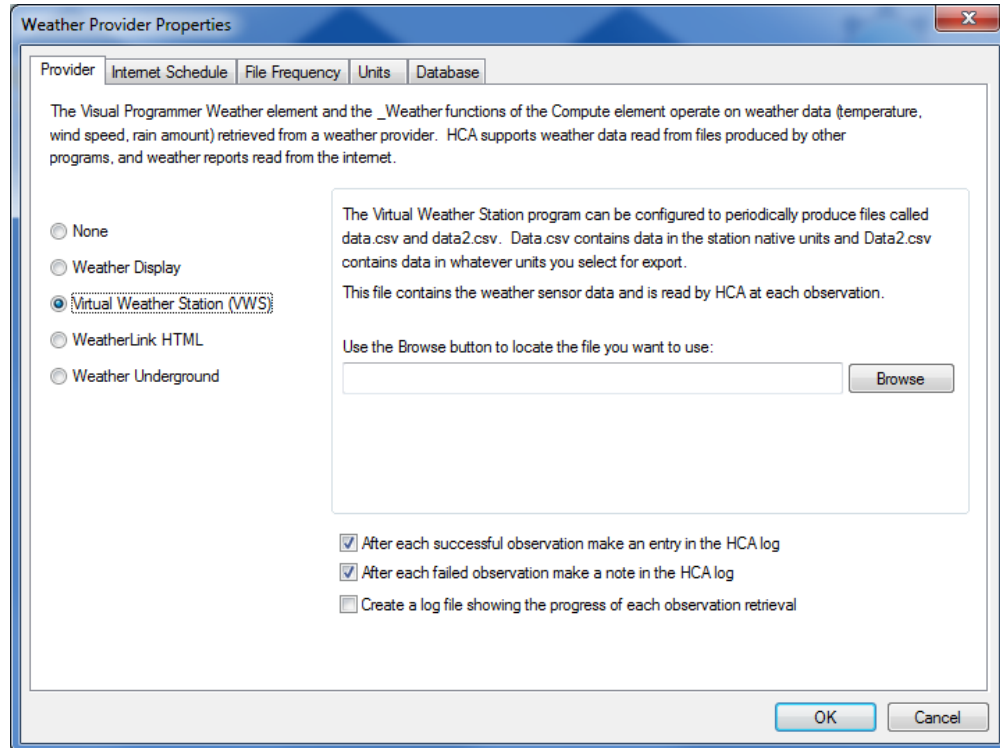
Hint: The method for producing scriptlog.txt may have changed in the current version of Weather Display. Refer to the *Weather Display* documentation for current instructions.

When working with the *Weather Display* program you should also set the observation units and working units on the *Units* tab as described above in the Weather Underground section.

When working with the *Weather Display* program you should also set the observation frequency as described above in the WeatherLink section.

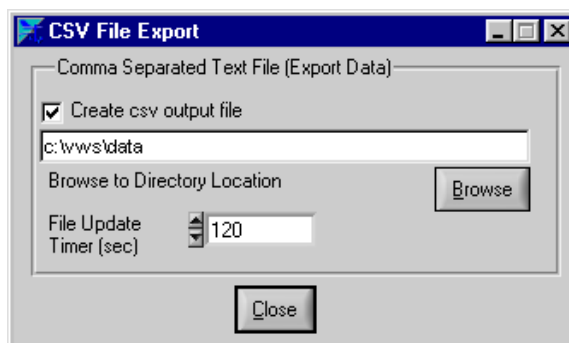
Virtual Weather Station Program

The *Virtual Weather Station* program periodically writes a file containing the current weather sensor data. When you select this option the setup options are:



All that is needed is the path to the file produced by VWS.

To configure VWS to periodically create the data.csv and data2.csv files, start VWS and select from the VWS menu: Settings then CSV Export. This dialog appears:



Make sure that you check the *Create csv output file* option and choose a folder for where the file is produced and how often. Don't use too short an update time or HCA will never get access to the file – VWS will always have it open. Then press *Close*.

Hint: The method for producing scriptlog.txt may have changed in later versions of VWS. Refer to the VWS documentation for current instructions.

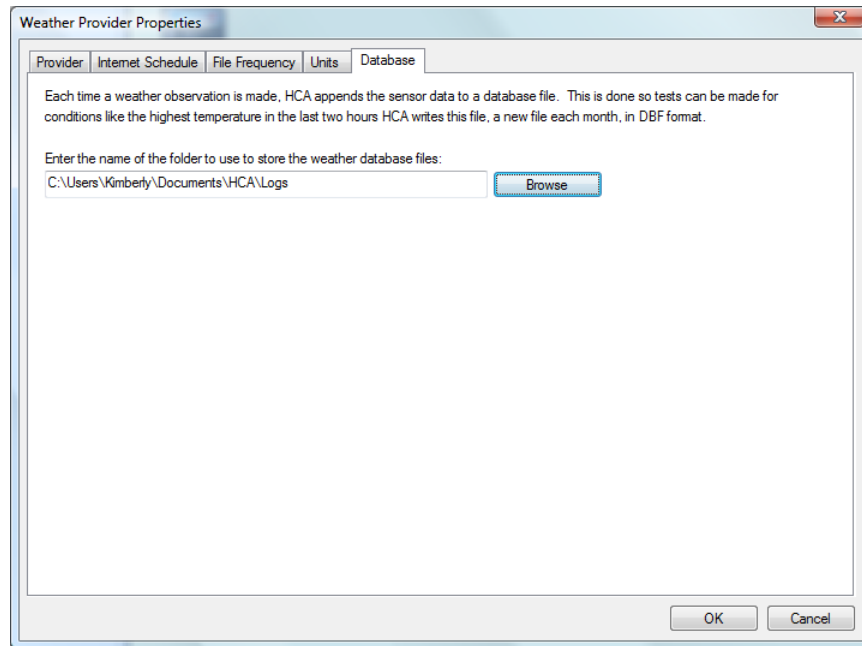
When working with the *VWS* program you should also set the observation units and working units on the *Units* tab as described above in the Weather Underground section.

When working with the *VWS* program you should also set the observation frequency as described above in the WeatherLink section.

Weather database

Based upon the configured schedule HCA take a weather observation and writes data to a file. This is necessary to support testing on weather data over time. For example, the database is used when testing to see if the average temperature in the last 24 hours has been above or below freezing.

To setup a weather database, choose the *Database* tab in the weather provider setup.

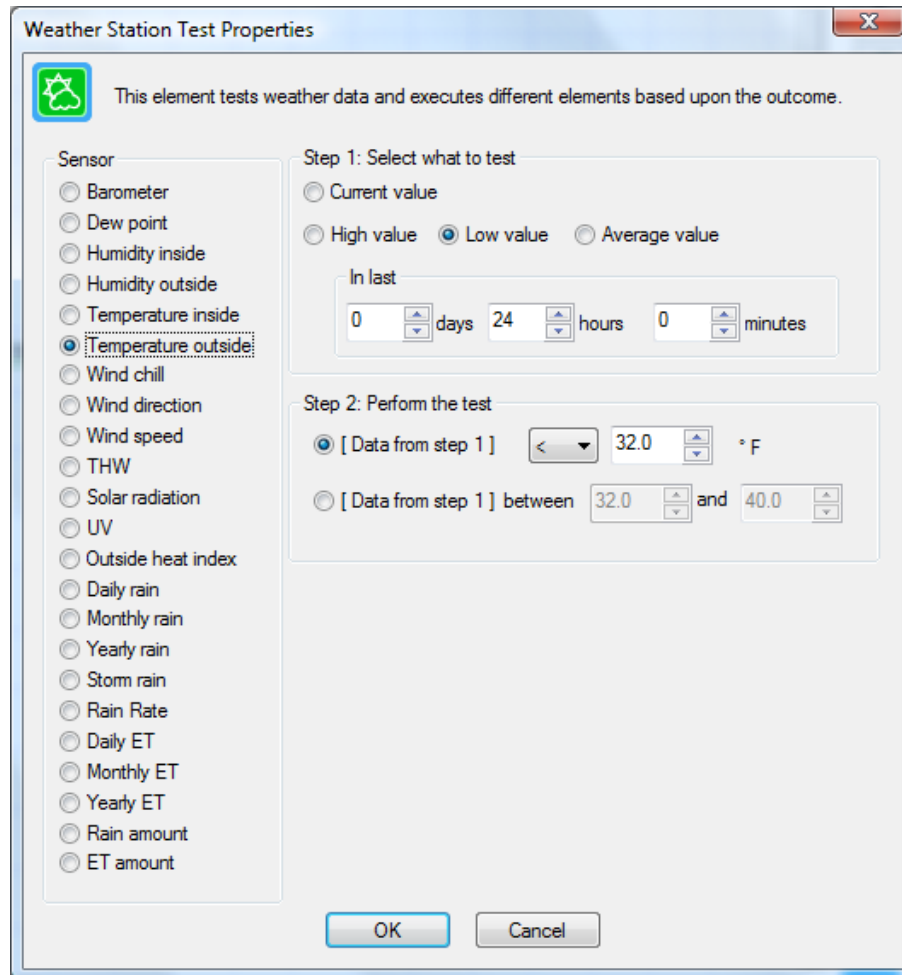


The only information to enter is the path to a folder that contains the monthly log files. What you enter here is the path to a folder and **not** the path to a file. Each monthly log file is given a name by HCA.

Tip: Not all weather data is saved in the weather database. On the HCA support web site is a technical note on Weather providers. The technical note describes exactly what parts of an observation are saved in the database and which can be tested with the historical weather tests.

Visual Programmer

To use weather data to make decisions in your HCA programs, the *Weather Test* element is used. This visual programmer element makes it possible to test both current weather data and historical weather data. When added to a program, the element properties are:



The parts of this dialog are:

- What data item to test. Not all weather providers support all data items.
- What to test. Either the current value taken from the last weather observation or a value determined from the weather database.
- The test. You can test the data to be <, <=, >, >=, =, <> to a value or within a range of values.

In the above example the test is to see if the low outside temperature over the last 24 hours is less than 32 degrees F.

Like other Test elements, execution continues following the path marked *Yes* if the test is true, otherwise execution follows the *No* path.

Weather Triggers

Weather data can be used to trigger a program when an observation is taken and the values of weather data meet certain criterion.

To create a weather trigger for a program, select the *Triggers* tab of the program's properties, select *Weather Condition Change* and press *Add*.

The trigger creation dialog shows these options.

This dialog looks a lot like the properties for the visual programmer *weather* element described above. As described in the User Guide *Programs* chapter, HCA evaluates all weather triggers every minute and when the condition evaluates to Yes, the program starts. Before the program can start again, the weather condition must evaluate to No and then to Yes again. For more information on how these kinds of triggers are evaluated and programs started see the User Guide *Programs* chapter.

Weather Data and the Compute and Compute Test elements

In addition to examining weather data with the Visual Programmer *Weather* element and with weather triggers, there are also functions that can be used in expressions in the Compute, Compute Test, and in expression placeholders in elements that work with text strings. Refer to the User Guide expressions chapter for information on the `_Weather` functions.

Weather Underground

In addition to the *Weather Test* Visual Programmer element and *_Weather* function in expressions, an additional function has been added in support of Weather Underground. The function allows you to get deeper into all the data provided by Weather Underground.

The function is called *_WUnderground* and it takes a minimum of 4 arguments and a max of 10. The format is:

```
_Wunderground ("feature", "location", "path1", "path2", ... "pathn", "tag")
```

The “feature” – 1st argument - is one of these on this list:

<http://www.wunderground.com/weather/api/d/docs?d=data/index>

The location – 2nd argument - is any acceptable location as described on that page in the “query” section. You can use “” for the feature and it defaults to “*conditions*”. You can use “” for location and it defaults to whatever you entered as the location in the weather provider dialog setup for Weather Underground.

Using the *_Wunderground* function you can get to **anything** that weather underground provides. For an example, enter this into a browser:

<http://api.wunderground.com/api/<your api key>/forecast/q/<your zipcode>.xml>

Substitute your API key and location into this URL. This is similar to what you will see:

```
▼<response>
  <version>0.1</version>
  ▼<termsofService>
    http://www.wunderground.com/weather/api/d/terms.html
  </termsofService>
  ▼<features>
    <feature>forecast</feature>
  </features>
  ▼<forecast>
    ▼<txt_forecast>
      <date>7:00 AM EST</date>
      ▼<forecastdays>
        ▼<forecastday>
          <period>0</period>
          <icon>rain</icon>
          <icon_url>http://icons-ak.wxug.com/i/c/k/rain.gif</icon_url>
          <title>Wednesday</title>
          ▼<fcttext>
            ▼<![CDATA[
              Overcast with rain. High of 45F. Breezy. Winds from the South at 15 to 20 mph. Chance of rain 90%.
            ]]>
          </fcttext>
          ▼<fcttext_metric>
            ▼<![CDATA[
              Overcast with rain. High of 7C. Windy. Winds from the South at 25 to 30 km/h. Chance of rain 90%.
            ]]>
          </fcttext_metric>
          <pop>90</pop>
        </forecastday>
      ></forecastday>...</forecastday>
```

As you can see this is forecast data for your location. To extract the forecast string for today the path though this XML is:

```
Response - forecast - txt_forecast - forecastdays - forecastday - fcttext
```

To make it easier to see, this image has that path highlighted.

```

▼<response>
  <version>0.1</version>
  ▼<termsOfService>
    http://www.wunderground.com/weather/api/d/terms.html
  </termsOfService>
  ▼<features>
    <feature>forecast</feature>
  </features>
  ▼<forecast>
    ▼<txt_forecast>
      <date>7:00 AM PST</date>
      ▼<forecastdays>
        ▼<forecastday>
          <period>0</period>
          <icon>rain</icon>
          <icon_url>http://icons-ak.wxug.com/i/c/k/rain.gif</icon_url>
          <title>Wednesday</title>
          ▼<fcttext>
            ▼<![CDATA[
              Overcast with rain. High of 45F. Breezy. Winds from the South at
            ]]>
          </fcttext>
          ▼<fcttext_metric>
            ▼<![CDATA[
              Overcast with rain. High of 7C. Windy. Winds from the South at 2:
            ]]>
          </fcttext_metric>
          <pop>90</pop>
        </forecastday>
      ></forecastday>...</forecastday>

```

Extracting the forecast string using the `_Wunderground` function uses these arguments:

```
Forecast = _wunderground ("forecast", "", "forecast", "txt_forecast",
                          "forecastdays", "forecastday", "fcttext")
```

The first argument to `_wunderground` is the *feature*. In this case “forecast”. The second argument is the location. Since it is set to “” the location as set in the weather provider setup is used. The remaining arguments are the path to the data using the highlighted names shown above.

Given the above XML, and the function with those arguments, it evaluates to this string:

```
“Overcast with rain. High of 45F. Breezy. Wind from the South at 15 to 20 mph. Chance of rain 90%.”
```

Instead of the forecast, if you wanted to get the “icon” the path would be:

```
Response - forecast - txt_forecast - forecastdays - forecastday - icon
```

In addition to weather data, Weather Underground offers lots of other data as well. For example, to get the current moon phase percent use:

```
Moon = _wunderground ("astronomy", "", "moon_phase", "percentIlluminated")
```

There is one complication with some keys: In this example XML below there are multiple “forecastDay” branches:

```

▼<forecast>
  ▼<txt_forecast>
    <date>7:00 AM PST</date>
    ▼<forecastdays>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
      ▶<forecastday>...</forecastday>
    </forecastdays>
  </txt_forecast>

```

As an example, to get data for a specific day, include a qualifier in the name of the tag – note the “:3” which requests the 3rd *forecastday*.

```
Forecast = _wunderground ("forecast", "", "forecast", "txt_forecast",
                          "forecastdays", "forecastday:3", "fctext")
```

This retrieves the 3rd forecast day. To get the first forecast day use “forecastday:1”. To find the second use “forecastday:2”, etc. If you leave off the qualifier it defaults to 1.

A good suggestion is to try out the various retrievals in your browser first then look at the data and figure out the path you need to get at that data in HCA. Pay close attention to any paths that will need qualifiers.

Two more important points when using Weather Underground data

To prevent multiple queries to Weather Underground each time a *_wunderground* function is evaluated, the file is retrieved, parsed, and saved internally along with the date/time that it was retrieved. Subsequent *_wunderground* functions don’t retrieve the data again from the internet if the XML for that feature was retrieved less than 15 minutes ago. This way you can extract lots of data and do it quickly since only one internet action is needed.

And finally, the interactions with the Weather Underground site are done using HTTP so there should be no firewall issues.