Transfer Format for Water Meter Data

M W Rodgers
January 2015

Hawke's Bay Regional Council accepts water meter data into their FTP server and then software behind the server reads the data and saves it into an archive. The councils uses Hilltop Software and has a number of options for the data transfer format, but there is a preference to use Hilltop's XML schema. This is because the identification information travels with the data and so processing the data is simpler.

The XML schema described is that used by Hilltop programs. This means you can use Hilltop Manager to write an XML file, or use the DataInput or DataCopy objects in HydroLib. The trick is to write to a data file with a filename extension of ".xml" or ".zip". The easiest way to describe the XML schema is through a worked example. A farmer has a water meter installed on his pump and the data logger creates a record every 10 cubic-metres. The data in XML format would be:

```
<?xml version="1.0"?>
<Hilltop>
<Agency>Aqua Fetch Ltd</Agency>
<Measurement SiteName="123456M1">
<DataSource Name="Compliance Volume" NumItems="1">
<Interpolation>Incremental</Interpolation>
</DataSource>
<Data DateFormat="Calendar">
<V>10-Apr-2008 11:06:15 10</V>
<V>10-Apr-2008 11:25:56 10</V>
<V>10-Apr-2008 11:46:02 10</V>
<V>10-Apr-2008 12:05:15 10</V>
</Data>
</Measurement>
</Hilltop>
```

The SiteName attribute on the Measurement element is a unique identifier for the water meter and the site name will be supplied by council staff. The information in the DataSource node tells the XML reader about the data it is going to read, and you must provide the elements and attributes shown in the example. Having said that, Hilltop can accept your names for the site and measurement and so do its own translation. This is done through the Agency element placed before the Measurement nodes.

```
<?xml version="1.0"?>
<Hilltop>
<Agency>Aqua Fetch Ltd</Agency>
<Measurement SiteName="ID42">
<DataSource Name="Volume" NumItems="1">
<Interpolation>Incremental</Interpolation>
</DataSource>
<Data DateFormat="Calendar">
<V>10-Apr-2008 11:06:15 10</V>
<V>10-Apr-2008 11:25:56 10</V>
<V>10-Apr-2008 11:46:02 10</V>
<V>10-Apr-2008 12:05:15 10</V>
</Data>
</Measurement>
</Hilltop>
```

In the example above, the Agency tag identifies the company sending the data. Regional Council staff will populate a table in Hilltop that converts site names and measurements to their requirements.

Actual Meter Readings

The examples above are where a logger computes the volume by counting the clicks from the water meter across a time interval. The logger resets the counter at the end of the interval. Some loggers don't reset the counter and so they report a number that is analogous to the dial on the water meter. Hilltop can accept this data but needs an extra element in the XML to tell it the data is like a real water meter. Add the DataType element into the DataSource node, as follows:

```
<DataSource Name="Water Meter" NumItems="1">
<Interpolation>Incremental</Interpolation>
<DataType>MeterReading</DataType>
</DataSource>
```

Notice the data source name is different. It is Water Meter in this example, but may not be the same for all councils. However, Water Meter is a good choice and the council staff can use the Hilltop Transfers table to change from your data name to their one, provided you use an Agency element.

Data Format

The Data node has the actual data and is more like a CSV format than XML. Each data value element has a date, time and value, separated by one or more spaces. The DateFormat attribute specifies how you are sending the date and time, and the "Calendar" value means you are sending the date and time in a readable form. The example shows a particular style of dates and times, but you can choose a format convenient to you. Valid formats for the date are:

Format	Example	Format	Example
dd/mm/yy	1/3/02	dd/mm/yyyy	1/3/2002
yyyy/mm/dd	2002/3/1	dd/mmm/yyyy	1/mar/2002
dd-mmm-yy	1-mar-02	dd-mmm-yyyy	1-mar-2002
dd-mm-yy	1-03-02	dd-mm-yyyy	1-03-2002
yyyy-mm-dd	2002-03-02		
Yyymmdd	1020301	Yyyymmdd	20020301

Two digits years are put into the 21st century if the number is between 0 and 49 inclusive, and the 20th century if the number is greater than or equal to 50.

Valid time of day formats are:

Format	Example
hh:mm	14:15
hh:mm:ss	14:15:00
Hhmmss	141500

It is assumed that times are given in a 24 hour format. However, the software will check for the presence of "a.m.", or "p.m." after the time, and so accept a 12 hour clock.

You can combine the date and time into a single string using the ISO 8601 format and this is useful when extracting data from SQL Server. There are two formats and these are:

Format	Example
yyyymmddThhmmss	20020301T141500
yyyy-mm-ddThh:mm:ss	2002-03-01T14:15:00

Time Zones

The DateFormat="Calendar" attribute assumes the date and time are in New Zealand Standard Time. It is not good practice to allow the dataloggers to change to daylight time, but you will get manual readings from farmers in daylight time during the summer. The XML parser will accept two other date formats to indicate the time zone.

The DateFormat="Local" attribute tells the parser the date and time will be in local time (i.e. clock time), and it will convert the date and time to NZST as it reads the data. The DateFormat="UTC" attribute tells the parser the date and time are in UTC (i.e. Greenwich Mean Time), and it will add 12 hours to the date and time to get to NZST. UTC does not have the concept of daylight time, so the software always adds 12 hours.

Logger Styles

There are a number of different data loggers in use, and the Hilltop Schema can accept data from these loggers without undue data manipulation. The example showed a logger/meter combination where by a time stamp was generated every 10 cubic metres. Hilltop can accept any such combination, so it is possible to provide a timestamp every cubic metre or intermediate totals of say five cubic metres. The data value element has the date and time, and then the number of cubic metres assigned to that time stamp.

Some loggers total the volume over a period and then store the total at a regular interval. In this case, the data value element has the date and time of the end of the interval, plus the amount of water pumped during the interval.

Harvest loggers run a counter which is incremented each time the meter gives a pulse. The counter is like the dials on the meter and represents the total amount of water pumped since the unit was commissioned. A Harvest unit running with its default settings will store the value of the counter every 10 minutes. It is necessary to compute the difference between the counter values in order to compute the volume pumped in the ten minute period. Hilltop can accept the actual counter value without having to perform the difference calculation. The method is to use a DataSource name of "Meter Reading" and then write the actual counter values. An example is:

```xml
<?xml version="1.0"?>
<Hilltop>
<Measurement SiteName="123456M1">
<DataSource Name="Meter Reading" NumItems="1">
<Interpolation>Incremental</Interpolation>
```

```
</DataSource>
<Data DateFormat="Calendar">
<V>10-Apr-2008 11:00 24779</V>
<V>10-Apr-2008 11:10 24782</V>
<V>10-Apr-2008 11:20 24785</V>
</Data>
</Measurement>
</Hilltop>
```

The software reading the data can cope with the counter overflow.

Separate Elements for Timestamp and Value

The <V> data value element is simple to produce if you have software which is translating the raw data from a logger and writing the results to the XML file. The combination of timestamp and value is more problematic if you have already stored the data in a relational database because the database table will have one field for the timestamp and one for the value.

The Hilltop parser can accept separate elements. The date and time are in the <T> element and the value is in the <I1> element. The element names are kept short to keep the file size down, and the name "I1" refers to the fact that Hilltop can accept multiple data items at a time stamp, and the volume is therefore "Item 1". Each reading is contained in an <E> element. An example data node with separate elements is:

```
<Data DateFormat="Calendar">
<E><T>10-Apr-2008 11:00</T><I1>24779</I1></E>
<E><T>10-Apr-2008 11:10</T><I1>24782</I1></E>
<E><T>10-Apr-2008 11:20</T><I1>24785</I1></E>
</Data>
```

Multiple Meters

Many farms have more than one meter, and the transfer format can cope with this. Repeat the entire measurement node for the second meter and give the correct site name. The same applies for data processed by third party agents and there is no need to produce an XML file for each farmer. An XML document with two meters in it is:

```
<?xml version="1.0"?>
<Hilltop>
<Measurement SiteName="123456M1">
<DataSource Name="Abstraction Volume" NumItems="1">
<Interpolation>Incremental</Interpolation>
</DataSource>
<Data DateFormat="Calendar">
<V>10-Apr-2008 11:06:15 10</V>
<V>10-Apr-2008 11:25:56 10</V>
</Data>
</Measurement>
<Measurement SiteName="991234M1">
<DataSource Name="Abstraction Volume" NumItems="1">
<Interpolation>Incremental</Interpolation>
</DataSource>
<Data DateFormat="Calendar">
<V>10-Apr-2008 11:46:02 10</V>
<V>10-Apr-2008 12:05:15 10</V>
</Data>
</Measurement>
```

```
                 </Hilltop>
```

Location via GPS

Portable pumps can be fitted with GPS equipment and the logger can then report the pump's location. The easting and northing are transmitted as extra items in the data. For example:

```
        <?xml version="1.0"?>
        <Hilltop>
        <Measurement SiteName="123456M1">
        <DataSource Name=" Compliance Volume and Location" NumItems="3">
        <Interpolation>Incremental</Interpolation>
        </DataSource>
        <Data DateFormat="Calendar">
        <V>10-Apr-2008 11:06:15 10 -39.493165 176.196693</V>
        <V>10-Apr-2008 11:25:56 10 -39.493165 176.196693</V>
        </Data>
        </Measurement>
        </Hilltop>
```

The Name attribute on the Data Source element is different and will be set by the Regional Council. The NumItems attribute changes is set to three because there are three numbers at each timestamp. The easting and northing are in degrees as a number.

Data be sent from a relational database may have the easting and northing in separate XML elements, and an example layout is:

```
        <?xml version="1.0"?>
        <Hilltop>
        <Measurement SiteName="WS080123">
        <DataSource Name=" Compliance Volume and Location" NumItems="3">
        <Interpolation>Incremental</Interpolation>
        </DataSource>
        <Data DateFormat="Calendar">
          <E><T>10-Apr-2008 11:06:15</T><I1>10</I1><I2>-39.493165</I2>
               <I3>176.196693</I3></E>
          <E><T>10-Apr-2008 11:25:56</T><I1>10</I1><I2>-39.493165</I2>
               <I3>176.196693</I3></E>
        </Data>
        </Measurement>
        </Hilltop>
```

File Names

The file name may be important if the Regional Council receiving the data says so. However, Hawkes Bay Regional Council use the Hilltop Telemetry System and it does not care about the file name.

Hawkes Bay Regional Council will issue you a user name and password for their FTP site and this will take you to a private directory on their server. Any file with a filename extension of ".xml" will be read by the telemetry system and new data will be loaded into the archive. The data file will then be copied into another directory on their system and the filename changed to include a timestamp of when the file was processed. The file on the FTP server is then deleted. This system means you can choose a file name and use the same name for each data transfer.

There is an issue with always sending the data with the same file name in that the Council's software must read the file before you send the next lot of data. Data sent weekly will be fine, but it would be best if files sent daily or less than daily have a sequence number or the date built into the filename. That way, all the files will be available if the data processing software stops for a short period.

Third party providers may wish to batch their data before sending it, and the files may get quite large. Hilltop will accept a zip file and will extract the data from the compressed file. The zip file must have a filename extension of ".zip", but does not have to have a particular name. There can be one or more files inside the zip file, and these can also have any name, but must have a filename extension of ".xml".