**Water Use Data Assessment Part 2: Time-series Analysis and Visualisation**

**Aims**

Part 2 of the project aims to create analytical tools that can be used to analyse and visualise the entire water use time-series for selected WAPs. The tools are intended to support work such as catchment modelling, which requires detailed assessment of historic water use data.

**Requirements**

In order to run the associated Python programs the user will need to have installed:

* Python Anaconda (and the pandas, seaborn, matplotlib, xlsxwriter and datetime modules)
* The hilltop-py package developed by Mike Exner-Kitteridge.

Two of the Python programs (which deal with consent conditions) also require access to the CrcActSiteSumm table. This table is stored in the ConsentsReporting database on the edwprod01 server.

**Python Programs**

Four sets of programs have been developed that are intended to let the user explore water use data in differing levels of detail:

* Water meter list generator (created by HG & MEK Environment Canterbury)
* Quality Assessment programs
* Time-series Analysis programs
* Monthly Visualisation programs

|  |
| --- |
| **Water meter list generator** (created by HG & MEK Environment Canterbury) |

Three versions of this program have been produced depending on the users input (list of WAPs, list of consent numbers or catchment name. The programs generate a list of water meters which are then used as an input into the quality assessment, time-series analysis and monthly visualisation programs.

The required inputs for the WAP list generated water meter numbers are:

1. The user will need to specify the folder pathway the python packages are located (line 59) and the name of the .csv file (line 61)
2. The user needs to specify the list of consents in sites.csv using the heading "ExtSiteID" in cell A1

The required inputs for the consent number list generated water meter numbers are:

1. The user will need to specify the folder pathway the python packages are located (line 59) and the name of the .csv file (line 61)
2. The user needs to specify the list of consents in sites.csv using the heading "RecordNumber" in cell A1

The required inputs for the catchment defined generated water meter numbers are:

1. Line 57 of the script is where the user will define the catchment of interest. Example below:

site\_filter = {'SwazName': ['Hakataramea River']}

For all three water meter list generators the output will be a spreadsheet containing a list of water meters currently archived in Hilltop for the user defined WAP list/consent list/catchment.

|  |
| --- |
| **Quality Assessment Programs** |

These programs provide the most general assessment. They are intended to be run over a list of WAPs to give an initial assessment of the water use data that exists for each WAP, and the quality of that data.

There are two variations of the Quality Assessment programs.

***Quality Assessment - WAP List - Combined Series V2.py***

**Input:** A csv file containing a list of WAPs

**Output:** An Excel spreadsheet containing QA statistics

**Description:** Hilltop sometimes contains multiple measures of water use for a single WAP (eg, some combination of Compliance Volume, Volume, Volume [Flow], Volume [Average Flow], or Water Meter measurements). In order to produce time series plots it is necessary to stitch these different measures together while taking care to avoid overlaps. This program lets you perform a quality assessment on the ‘Combined Series’ that is used to produce the time-series plots.

**Notes on the ‘Combined Series’:** When a WAP has more than one measure of water use for a given period of time, it is necessary to include just one of these measures (so as to avoid double or triple counting water use). The method used in this program to create a ‘Combined Series’ is as follows:

1. Find the measurement type with the earliest start date (Measurement 1) and extract the entire time-series for Measurement 1.
2. If there is another measurement type with data that follows on from Measurement 1, extract data for this measurement type (Measurement 2). Avoid any overlap with Measurement 1 by using the end\_date for Measurement 1 to determine the start\_date for Measurement 2.
3. Repeat this process for any additional measurement types.

An example of this method – as applied to BY20/0088-M1 – is provided in the table below.

|  |  |  |
| --- | --- | --- |
| **Water use data in Hilltop** | **Water use data included in ‘Combined series’** | **Notes** |
| Compliance Volume  (01-07-2012 to 08-05-2016) | Water use data from 01-07-2012 to 08-05-2016 is included. | This is classified as Measurement 1 and the entire time-series is extracted. |
| Volume  (01-07-2015 to 30-06-2019) | Data from 09-05-2016 to 30-06-2019 is included. | The start date for the extraction (09-05-2016) is the day that follows the end of the Measurement 1 time-series. |

***Quality Assessment - WAP List – By Measurement Type V2.py***

**Input:** A csv file containing a list of WAPs

**Output:** An Excel spreadsheet containing QA statistics

**Description:** This variation produces a quality assessment for each type of water use measurement that a WAP has. For example, in the case of BY20/0088-M1 which has two measurement types, a quality assessment is provided for each of the two data series (Compliance Volume and Volume).

|  |
| --- |
| **Time-series Analysis programs** |

These programs enable the user to explore a time-series in more depth. When you run the programs you have the option to output:

* an Excel spreadsheet containing various monthly statistics
* a set of plots showing daily totals for each month in the time series, arranged by water year, or
* both the spreadsheet and plots.

There are four variations of the Time-series analysis programs.

***Analyse Time Series – Single WAP V3.py***

**Input:** On running this program the user is asked to specify:

1. a single WAP of interest
2. whether they want to export Statistics [s], Plots [p] or Both [b]

**Output:** Depending on the export option selected, the program will output an Excel spreadsheet containing monthly statistics, a PDF file containing time-series plots, or both.

***Analyse Time Series – WAP List V2.py***

**Input:** On running this program the user is asked to specify:

1. a csv file that contains a list of WAPs
2. whether they want to export Statistics [s], Plots [p] or Both [b]

**Output:** The program iterates through the WAPs included in the csv file. For each WAP the program will output an Excel spreadsheet containing monthly statistics, a PDF file containing time-series plots, or both (depending on export option that has been selected).

***Analyse Time Series – With Consent Conditions - Single WAP V1.py***

This variation adds extra features to the time-series plots, based on consent conditions. The consent conditions are extracted from the CrcActSiteSumm table, stored in the ConsentsReporting database on the epwprod01 server.

**Input:** On running this program the user is asked to specify:

1. a single WAP of interest
2. whether they want to export Statistics [s], Plots [p] or Both [b]

**Output:** Depending on the export option selected, the program will output an Excel spreadsheet containing monthly statistics, a PDF file containing time-series plots, or both.

***Analyse Time Series – With Consent Conditions - Single WAP with date V1.py*** (modified from AA, Whiterock Consulting, script by HG & MEK, Environment Canterbury)

This variation adds extra features to the time-series plots, based on consent conditions and a user defined date range. The consent conditions are extracted from the CrcActSiteSumm table, stored in the ConsentsReporting database on the epwprod01 server. Line 528 is where the date range is defined. An example of this is below for the 2018/19 water year:

mslist2 = process\_measurement\_list(mslist, '2018-07-01', '2019-06-30')

**Input:** On running this program the user is asked to specify:

1. a single WAP of interest
2. whether they want to export Statistics [s], Plots [p] or Both [b]

**Output:** Depending on the export option selected, the program will output an Excel spreadsheet containing monthly statistics, a PDF file containing time-series plots, or both.

***Analyse Time Series – With Consent Conditions - WAP List V1.py***

This variation adds extra features to the time-series plots, based on consent conditions. The consent conditions are extracted from the CrcActSiteSumm table, stored in the ConsentsReporting database on the epwprod01 server.

**Input:** On running this program the user is asked to specify:

1. a csv file that contains a list of WAPs
2. whether they want to export Statistics [s], Plots [p] or Both [b]

**Output:** The program iterates through the WAPs included in the csv file. For each WAP the program will output an Excel spreadsheet containing monthly statistics, a PDF file containing time-series plots, or both (depending on export option that has been selected.

***Analyse Time Series – With Consent Conditions - WAP List and Date V1.py*** (modified from AA, Whiterock Consulting, script by HG & MEK, Environment Canterbury)

This variation adds extra features to the time-series plots, based on consent conditions and a user defined date range. The consent conditions are extracted from the CrcActSiteSumm table, stored in the ConsentsReporting database on the epwprod01 server. Line 545 is where the date range is defined. An example of this is below for the 2018/19 water year:

mslist2 = process\_measurement\_list(mslist, '2018-07-01', '2019-06-30')

**Input:** On running this program the user is asked to specify:

1. a csv file that contains a list of WAPs
2. whether they want to export Statistics [s], Plots [p] or Both [b]

**Output:** The program iterates through the WAPs included in the csv file. For each WAP the program will output an Excel spreadsheet containing monthly statistics, a PDF file containing time-series plots, or both (depending on export option that has been selected).

**Notes relating to the Time-series analysis programs:**

1. The time-series data that features in the monthly statistics and plots is based upon the ‘Combined Series’ described earlier in this document.
2. When a WAP has an extensive time-series, the time-series plots can take some time to generate.
3. Any negative values in a time-series are filtered out prior to the calculation of monthly statistics and the creation of the time-series plots. A summary of any negative values is included in the output spreadsheet.
4. The monthly statistics include values for MinExtraction, MeanExtraction, and MaxExtraction. The concept of ‘extraction’ relates to datapoints where the volume extracted is greater than 0. In instances where no water is extracted for a given month, there will be no value for these extraction statistics.
5. The monthly statistics also include values for Spikes > 5sd, Spikes > 10sd and Spikes > 20 sd. These statistics convey the number of datapoints in a month that exceed certain thresholds (where the extracted value is greater than 5, 10 or 20 standard deviations from the mean).
6. Finally, please note that the extraction statistics (MinExtraction, MeanExtraction and MaxExtraction) and the spike statistics (Spikes > 5sd, Spikes > 10sd and Spikes > 20 sd) all relate to datapoints in the raw data rather than aggregated data for a fixed time period. So, for example, MaxExtraction represents the highest extracted value in the raw data, but it is possible that this value may relate to water extracted during 15 minutes, an hour or some other time period.

**Notes relating to Consent Conditions:**

1. All consent conditions are extracted from a SQL table (CrcActSiteSumm). It is important to note that these conditions relate to entire sites (eg, BZ19/0066) rather than the meter entities that are stored in Hilltop (eg, BZ19/0066-M1 and BZ19/0066-M2).
2. To obtain the consent conditions for a given site the following logic is applied:
   1. In the first instance, any active consents are extracted and aggregated
   2. If no active consents exist, the consent with the most recent “ToDate” is extracted.
3. In the plot titled “Daily volume extracted in m3”, the dotted reference line is derived from the ConsentedRate value in the SQL table. It represents the volume of water that would be extracted if the maximum extraction rate was applied for an entire day.
4. In the plot titled “Volume extracted (m3) – [] day moving average”, the moving average is based on the ConsentedMultiDayPeriod from the SQL table. The dotted reference line is derived by dividing the ConsentedMultiDayVolume by the ConsentedMultiDayPeriod. When the volume extracted is above the dotted reference line, this indicates that the ConsentedMultiDayVolume has been exceeded.
5. In the plot titled “Average daily extraction rate in L/s”, the dotted reference line is derived from the ConsentedRate value in the SQL table. It enables comparison of the average daily extraction rate with the maximum consented extraction rate.
6. In instances where a site has no consent conditions, the programs still generate a set of plots but without the extra consent features. The moving average plot is not generated at all.

|  |
| --- |
| **Monthly Visualisation program** |

This final program is intended to let the user explore detailed data for a month of interest, in instances where something unusual has been noticed in the time-series plots or monthly statistics.

***Plot Water Use - Single WAP - Specified Month V2.py***

**Input:** On running this program the user is asked to specify:

1. the WAP of interest
2. the start date for the plots in YYYY-MM-DD format

**Output:** The program generates a PDF file containing water use plots, with hourly totals, over a 35 day time period. If the WAP has more than one measurement type during the specified time period, a separate PDF file is generated for each measurement type.

**Notes relating to the Monthly Visualisation program:**

1. This program does not filter out any negative values that may be in the raw data.
2. Volume extracted is visualised using both a line plot and a bar plot as both of these are beneficial in different circumstances.

**Troubleshooting**

1. ***Temporary errors with the Hilltop web server***

Occasionally the Hilltop web server is offline and you may get an unusual error when running a program (eg, a message saying that a WAP doesn’t exist in the WaterUse.hts file, even though you know it does). This is a temporary issue and is normally resolved when you re-run the program.

1. ***Oddities in historic Hilltop water use data***

During testing of these programs, various oddities were observed in the historic Hilltop data. Examples of these oddities included:

1. WAPs with incomplete measurement lists in the Hilltop WaterUse.hts file.
2. WAPs with water use data stored using the wrong measurement type.

Please let me know if the program ever crashes, or returns warning messages, as this may indicate additional oddities that need to be considered.

**Contact details**

Alan Ambury

Email: [alan@whiterockconsulting.co.nz](mailto:alan@whiterockconsulting.co.nz) Cell: 0274 942 263