# Reverse engineering the RUBICON efficiency report

This report summarises the processes of how the hydrology group have developed tools to calculate water balances within a pool in the water network. A ground up approach has given us the ability to audit the branch efficiency report that RUBICON can supply.

## Generating topological awareness of pools

To the extent that the hydrology team have investigated, no accurate and concise linkage between asset codes is available in a format easy to work with. If one wanted to understand the relationship between a meter on a property e.g. M1893/1 and its nearest regulator (in this case Andretta’s Reg.), one would have to look up that meter on network visualiser or MI maps and manually explore the relationship through laterals to the nearest upstream regulator, then find the object numbers [[1]](#footnote-1) of each asset to get the required data from a production database, e.g. SC\_EVENT\_LOG . If one was wanting to find all the meters in an entire pool it can easily be seen that this method does not scale.

A branch summary can be exported as a comma separated file (csv). This file contains information about a branch and the objects in that branch. Unfortunately, it lacks a lot of features. It does not link laterals to a pool, nor provide object numbers and contains inconsistencies. By manually working through the csv and comparing to network visualiser the laterals can be defined, and inconsistencies removed. This csv is still not in a format to perform routine large searches on a branch to get data.

The next step is to parse through the csv and generate a more hierarchical link of all the topology in a branch. A program was written to do just this. It moves though the csv and builds up a structure of the network that makes links between any point in a branch and its upstream and downstream neighbours. It finds the object numbers of that asset and links the object numbers in a table, so that any relationship required for analysis can be easily extracted late. See output below.

|  |  |  |
| --- | --- | --- |
| Upstream objects…  └─── OVERS REGULATOR - Regulator (30840)  ├─── M2615A/1 - D/S Meter (67510)  ├─── M2038/2 - D/S Meter (64930)  ├─── N665/P - D/S Meter (70693)  ├─── M2615/D - D/S Meter (70048)  ├─── M2038C/1 - D/S Meter (64933)  ├─── M1631H/1 - D/S Meter (67513)  ├─── OT L185 - D/S Offtake (30862)  │ ├─── M2038B/1 - D/S Meter (64936)  │ ├─── M1997/2 - D/S Meter (64939)  │ ├─── M1997/3 - D/S Meter (64942)  │ ├─── M1985/1 - D/S Meter (64954)  │ └─── SV 185-1 - D/S Scour Valve (57141)  ├─── M1631D/1 - D/S Meter (68062)  ├─── MC10 TEMPORALIS ESCAPE - D/S Escape (55372)  └─── QUARY RD REGULATOR - Regulator (30868)  └─── Downstream objects… | | **OBJECT LINK LINK TYPE**  30822 30840 Regulator  30840 67510 D/S Meter  30840 64930 D/S Meter  30840 70693 D/S Meter  30840 70048 D/S Meter  30840 64933 D/S Meter  30840 67513 D/S Meter  30862 64936 D/S Meter  30862 64939 D/S Meter  30862 64942 D/S Meter  30862 64954 D/S Meter  30862 57141 D/S Scour Valve  30840 30862 D/S Offtake  30840 68062 D/S Meter  30840 55372 D/S Escape  30840 30868 Regulator |
|  |  | |

This is an example output of the parsing program. Note the output has been

## Working with metering data

## Summary of results

## Conclusion

1. The unique identifier for each object in the MI databases. The object number ties all the information about that asset; e.g. flow, gate position, battery voltage, etc. [↑](#footnote-ref-1)