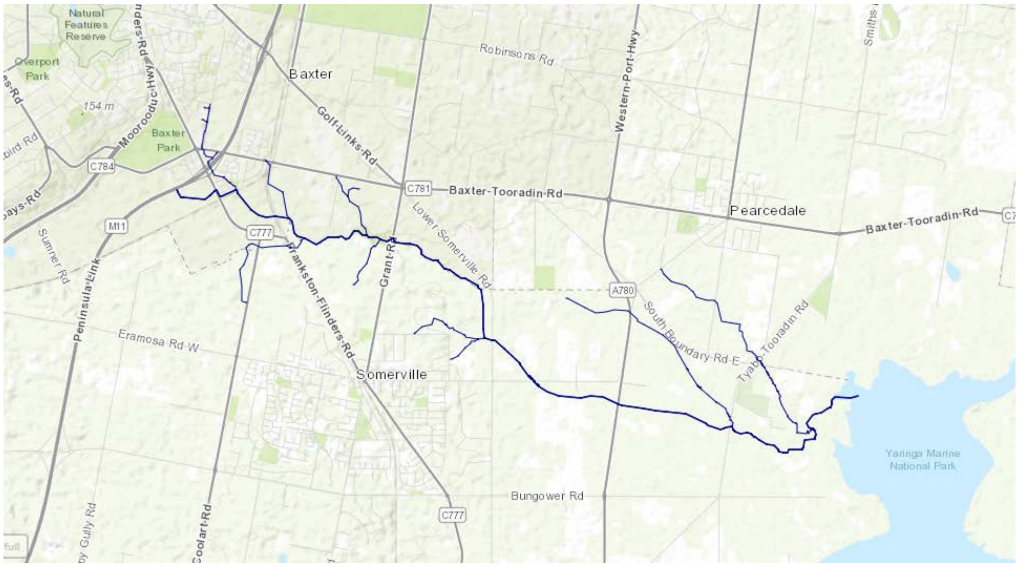
Melbourne Waterway Research-Practice Partnership

www.mwrpp.org

Melbourne Water Stream Network Version 1.0

Joshphar Kunapo, Christopher J Walsh &
Michael J Sammonds



THE UNIVERSITY OF MELBOURNE

Waterway Ecosystem Research Group

Melbourne Water®

Technical Report

19.4

Melbourne Waterway Research-Practice Partnership
Technical Report 19.4

Melbourne Water Stream Network. Version 1.0

Published: December 2019

Cite as: *Kunapo, J., Walsh, C.J., Sammonds, M.J. (2019) The Melbourne Water Stream Network. Version 1.0. Melbourne Waterway Research-Practice Partnership Report 19.4. School of Ecosystem and Forest Sciences, The University of Melbourne, Melbourne. URL: https://urbanstreams.net/tools/mwstr/.*

Contact author: Chris Walsh

Cover photo: Watsons Creek

Summary

The Melbourne Water Stream Network represents the flow paths and catchments of the streams and rivers of the Greater Melbourne Region (the region managed by Melbourne Water). The Network is a spatial dataset that :

- uses existing stream line data from several sources, ensuring downstream connectivity of lines, correcting delineation errors and augmenting the network to include small headwater streams missing from earlier stream data;
- standardises names of streams across the region, giving each stream a unique name and code;
- splits the stream lines into reaches at confluences and between confluences so that reaches are almost all less than 500 m long;
- provide a unique meaningful code for each reach, linked to existing codes such as Melbourne Water's asset identifiers;
- delineates a subcatchment boundary derived from a 5-m digital elevation model for each reach;
- is stored as two shapefiles (streams and subcatchments), and also as a spatial database structured to permit (with associated R functions) rapid calculation of upstream catchment statistics, such as land-use fractions, and rapid derivation of catchment boundaries. These functions can also be used through an application on the web.

Citation (for this report and for the use of any data downloaded from https://urbanstreams.net/tools/mwstr/: this site provides a direct interface to the data)

Kunapo, J., Walsh, C.J., Sammonds, M.J. (2019) The Melbourne Water Stream Network. Version 1.0. Melbourne Waterway Research-Practice Partnership Report 19.4. School of Ecosystem and Forest Sciences, The University of Melbourne, Melbourne. URL: https://urbanstreams.net/tools/mwstr/.

Introduction

This document describes the structure of the Melbourne Water Stream Network dataset and provides illustrations of how to extract data from the dataset. The data are available in several forms (Table 1), with the most flexible for data extraction being a relational spatial database that can be accessed through the software program R using a library of functions. A web application (in development) provides a simple interface for extracting data from the database. All data can be accessed through the web application at urbanstreams.net/tools/mwstr/.

The tables of the database and their relationships are illustrated in Fig. 1. Each of the tables is described below.

Fig. 1. Table relationship diagram for the mwstr database, showing all fields in each table and connections between tables. Those with a geometry field are spatial tables. The 'coast' table is a polyline spatial table for adding a coast line to any stream-line maps.

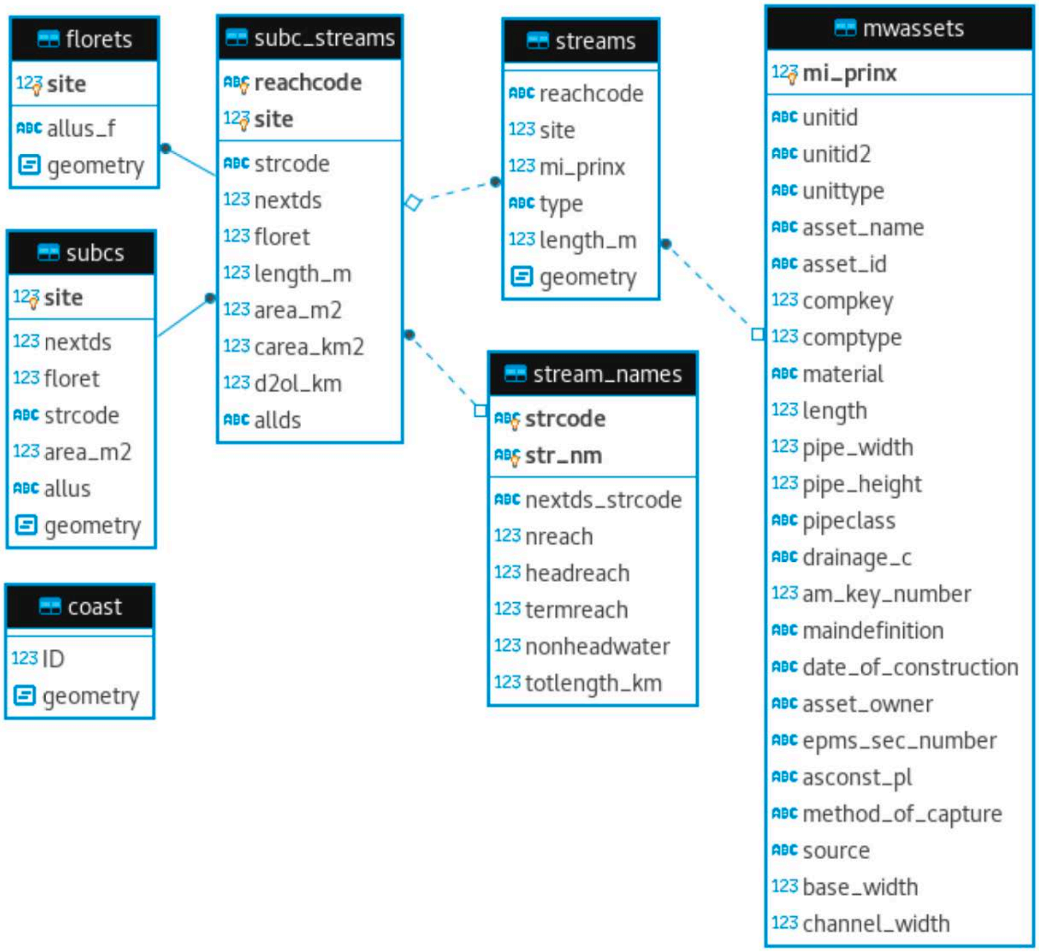


Table 1. Melbourne Water stream network data components, structures and locations for access.

Format	Description	File	Location
ESRI shapefiles	Stream layer	mwstr_v1.0.shp	urbanstreams.net/resources/mwstr/mwstr_v1.0.zip
	Subcatchment layer	mwsubc_v1.0.shp	
	Subcatchment floret layer	mwsubc_v1.0.shp	
Excel tables	Reachcode table	mwstr_tables.v1.0.xlsx	
	Stream Name table		
	MW asset table		
	Metadata table		
R objects	Simple features (sf) tables that inlcude all-upstream information	mwsubc mwsubc_florets	urbanstreams.net/resources/mwstr/mwsubc_v1.0.rda urbanstreams.net/resources/mwstr/mwsubc_florets_v1.0.rda
	List of all-upstream vectors for catchment calcs	allupstream_list	urbanstreams.net/resources/mwstr/allupstream_list.rda
Postgres spatial database	Relational database with library of R functions for data extraction and manipulation	mwstr database mwstr_functions.R	urbanstreams.net (external access proposed) urbanstreams.net/resources/mwstr/mwstr_functions.R
Web application	Shiny web application permitting extraction from the database	downloadable shapefiles	urbanstreams.net/tools/mwstr/
Documentation	This document in HTML, Word and Rmarkdown formats, the last containing source code used in producing this document	MWstreamNetworkManual.html MWstreamNetworkManual.docx MWstreamNetworkManual.Rmd	urbanstreams.net/resources/mwstr/MWstreamNetworkManual.html urbanstreams.net/resources/mwstr/MWStreamNetworkManual.docx urbanstreams.net/resources/mwstr/MWstreamNetworkManual.Rmd

The stream lines: table *mwstr*

The stream layer (table streams in the database) is made up of 137,709 lines, each with a unique combination of:

- reachcode, one of 127,977 unique reach/subcatchment codes made up of a three-character stream code and a (usually) four-digit code equal to 1000 times $\log_{10}$(catchment area in 100-m^2), so that codes beginning with 2 indicate total upstream catchment areas between 0.1 and 1 km^2, codes beginning with 3 indicate 1-10 km^2, codes beginning with 4 indicate 10-100 km^2, etc. (see below for more details) Each reach also has a unique numeric code called site, which is used for more efficient data extraction than is possible with subc.

- mi_prinx, one of 14,959 unique identifiers that link to Melbourne Water's asset ID table. 82,133 reaches have no mi_prinx value because they are small streams that augment the original Melbourne Water stream layers. 122,029 reaches have a single MI_PRINX. It is usual for a single MI_PRINX to extend over multiple subcs, however, 5,948 subcs have more than one MI_PRINX.

- type, five types of streams derived mainly from the source data for the stream lines (Table 2).

- The primary data sources listed in Table 2 were corrected and augmented to produce the layer shown in Fig. 2.

- While the Melbourne Water layer DR_Waterway_Centreline was the base source for stream lines in the dataset, we corrected many lines using LIDaR (see appendix for more details), through manual checks (e.g. deletion of duplicate/overlappiing lines) and in some cases filled in missing lines (sourceLayer "GraceGIS Links" in earlier versions of the stream layer).

Table 2. The five stream types in the Melbourne Water stream layer showing the total length of each type in the dataset, and the primary data source for each type. Sources beginning with "DR_" are spatial layers used by Melbourne Water. For all types, we have made corrections to the original data where appropriate.

Type	Length (km)	Primary data source
stream	8,737	DR_Waterway_Centreline
stream extensions	12,699	Derived by us (See Appendix 1 methods)
channel or drain	1,443	DR_Channel_Centreline
estuary	194	Our estimates informed by Melbourne Water's estuary layer
connecting line through waterbody	196	Added by us during assembly of the network
pipe	297	DR_UGround_Centreline

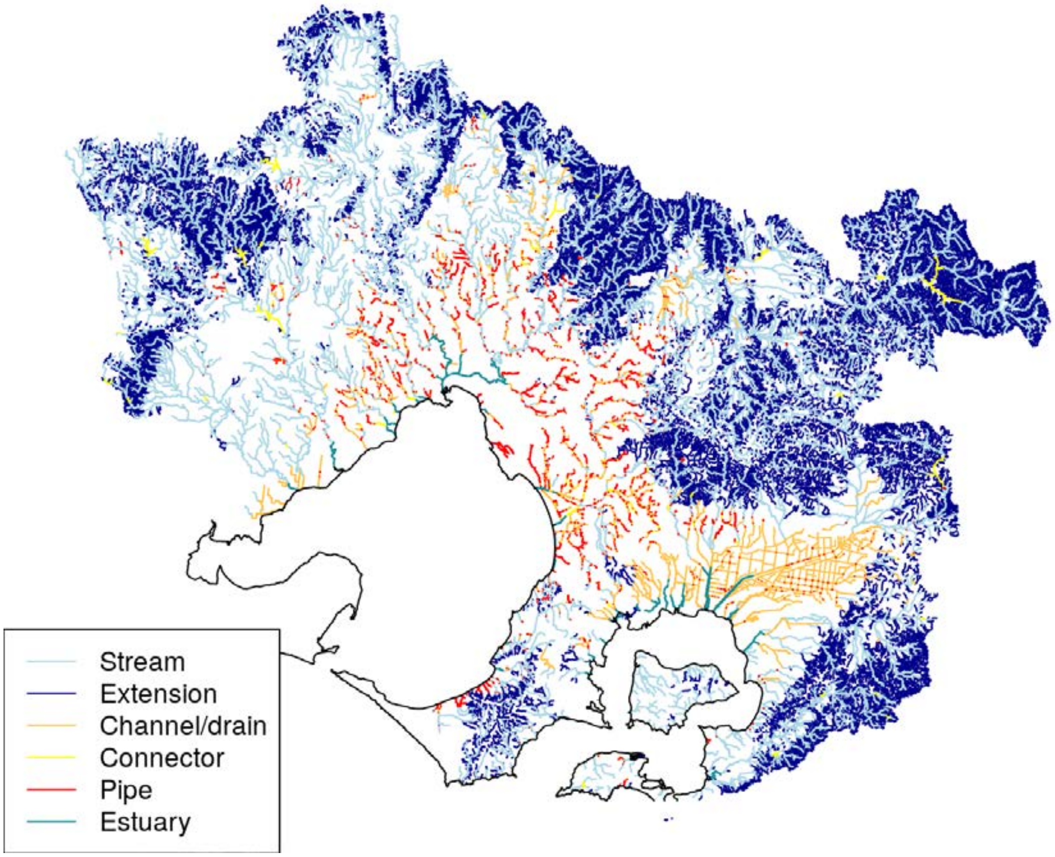
Stream extensions were added from a fine-scale stream network derived from LIDaR using hill-shade to identify the head of gullies as the most upstream point (see Appendix).

“Connecting lines through waterbody” lines should not be interpreted as stream channels, but serve only to maintain the hydrologic connectivity of the network through large waterbodies, such as reservoirs.

We have excluded most of the network of underground urban stormwater pipes in Melbourne Water’s DR_UGround_Centreline layer. However, we have retained all pipes that connect upland sections of streams with either their outlet to the sea or with more downstream sections of streams or channels. We have also retained small segments of pipes at their outlets to the sea or to streams. These collectively make up 297 km of pipes in the dataset (Table 1). There are 7,546 subcatchments in the mwsubc layer (see below) that do not have a stream line through them. Most of these are stormwater pipe catchments, which lack a line at their outlet.

We estimated estuary extent by visually inspecting aerial imagery and contours, cross-checking with Melbourne Water’s estuary layer (HWS 2018 estuaries layer) and our on-ground experience. Several of the lines in Melbourne Water’s layer extended too far into the catchments. The estuary extent in Fig. 1 is more correct, but likely retains some errors and omissions.

Fig. 2. Melbourne Water stream network colour-coded by type.



The subcatchments: table *mwsubc* (and *mwsubc_floret*)

The subcatchments layer (mwsubc table in the database) contains 135,523 subcatchments (Fig. 3) of which 127,977 contain stream lines (mwsub_streams table: see below).

Subcatchments were derived from a 5-m digital elevation model, by selecting 'pourpoints' just upstream of each confluence of stream lines and of major stormwater drains with stream lines using an automated process. In some cases pourpoints were automatically selected in at confluences of drainage paths on the DEM that were neither stream nor drains mapped on the DR_UGround_Centreline layer. These are the subcatchments without stream lines flowing through them. Many of these were large council drains, and are effectively large tributary inputs. Some, however, were small rural drainage lines that are of less importance. Most of these less important subcatchments were removed through manual checks, but some remain. These do not affect catchment calculations, but we note them here to explain the mwsubc_streams table (see below) having fewer rows than the mwsubc table.

Pourpoints were also placed between confluences to avoid reaches exceeding 500 m long. The mean reach length is 171 m As a result of manual corrections 229 stream reaches have lengths > 500 m (0.2% of streams and stream extensions, maximum stream reach length is ~ 1 km).

Every subcatchment has a unique code named 'site' (see above). The field 'nextds' contains the next downstream subcatchment's 'site' code. (Subcatchments of reaches that flow to the sea have a nextds value of -1.) The relationships between 'site' and 'nextds' permit network calculations such as aggregating statistics for all upstream subcatchments, or dissolving all upstream catchments to derive the catchment boundary for each reach.

We have structured the data to speed up the catchment calculations by:

a) deriving catchment boundaries for independent groups of between 250 and 900 subcatchments, which we term 'florets' (Fig. 3);

b) compiling a vector of all upstream 'site' codes for each subcatchment. For subcatchments that are downstream of one or more floret subcatchments, the vector contains the site code for the florets, but none of the subcatchments contained in the floret. As a result no 'all-upstream' vector is longer than ~1000 'site' codes. For calculation of catchment statistics, a recursive algorithm is used to concatenate the site codes upstream of each floret to compile a complete vector upstream site codes.

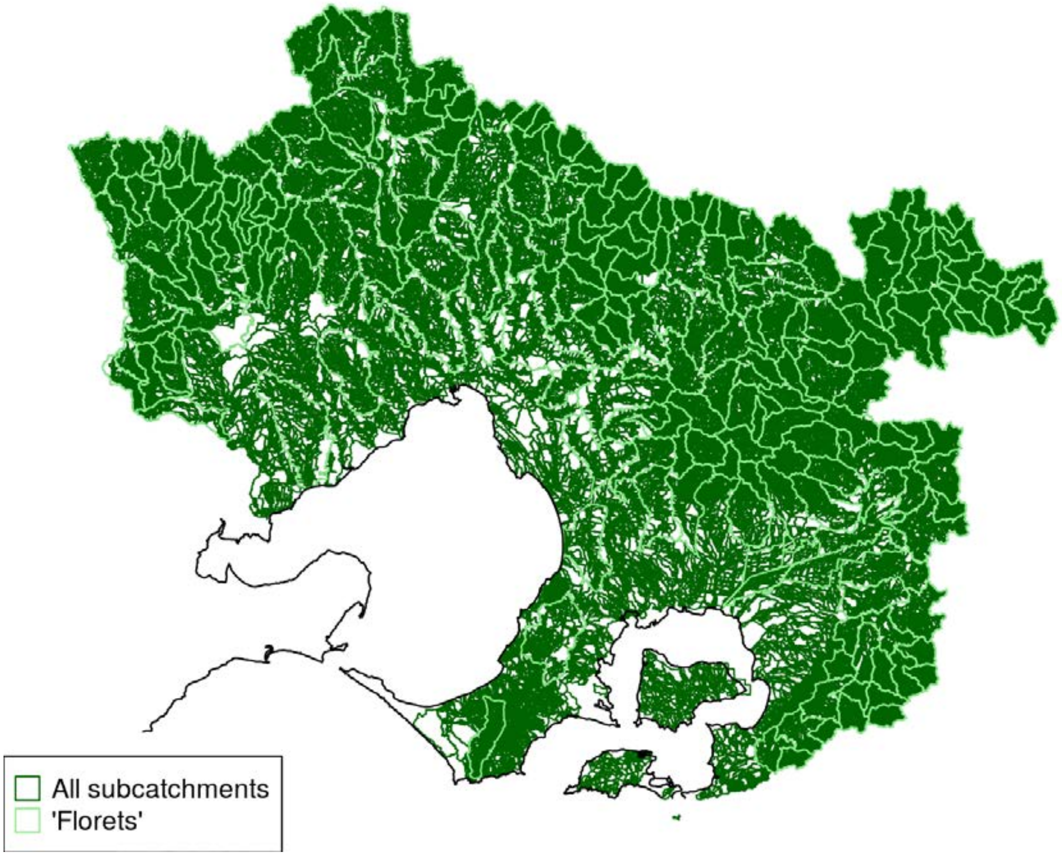
c) compiling a vector of all downstream 'site' codes for each subcatchment, to allow rapid calculation of distances between reaches

The floret field in mwsubc contains the floret in which each subc sits (NA if it does not sit in a floret). The mwsubc_floret layer contains polygons for each floret (Fig. 3) with a site field identifying the most downstream subcatchment in each floret. The postgres database version of the floret table contains a field allus_f, which lists the florets upstream of each floret. Similarly the postgres version database version of the mwsubc table contains a field 'allus', which lists all the 'site' codes upstream of each reach. And the subc_streams table (see below) contains a field 'allds', which lists all the 'site' codes downstream of each reach. The allus and allds vectors are too long to be included as a field in an ESRI shapefile. If you wish to use the all-upstream information and do not have access to the database version of the data, the fields are contained in simple features (sf) objects that can be used in R (Table 1). The allds field is contained in the mwsubc_streams worksheet in mwstr_tables.v1.0.xlsx.

The mwsubc table also contains a strcode field containing the three-character stream code for the line flowing through the subc (NA if there is no line). See the streamNames table below for more detail on this field, which links to that table.

Finally, the table contains a field named area_m2, the area of the subcatchment polygon in m^2. This is used to calculate upstream catchment areas (see below). In future versions, libraries of environmental variables such as land cover, climatic and hydrologic variables will be developed linking to this table that will similarly be used to calculate upstream catchment statistics.

Fig. 3. All subcatchments (their large number makes them mostly indistinguisable), and the 'floret' subcatchments that were compiled to improved the speed of catchment calculations.



Stream names: table *stream_names*

The table stream_names contains summary information about each of the 31,395 streams of the region.

Past stream map data for the Melbourne region has lacked a systematic, unambiguous approach to naming streams. We have ensured, in compiling the stream network data, that every stream (a continuous collection of lines from a headwater reach to a most downstream reach before it flows into another stream or to the sea) has a unique name that is consistent with its 'official' name, if it has one.

For instance, there are eight 'Deep Creeks' in the region, and each of them have been given a unique name describing their locality (str_nm) and a unique three-digit stream code (strcode, Table 3).

The stream_names table also lists:

- nextds_strcode, the strcode of the stream to which each stream flows (nextds_strcode, = "sea" if its outlet is marine);
- nreach, the number of reaches making up each stream;
- headreach, the most upstream reachcode (linked to tables mwsubc, mwsubc_streams) of each stream;
- termreach, the most downstream reachcode of each stream;
- nonheadwater, = 0 for most streams, but = 1 for those which headreach is not a headwater. For instance, the Maribyrnong River is a non-headwater stream because it is the product of Deep and Jacksons Creeks (their termreaches flow to the headreach of the Maribyrnong);
- totlength_km, the total length of all reaches of each stream.
- Stream names can be used to quickly plot (and save) the lines of a stream, and optionally its tributaries and catchment boundary. For instance, the following code produced the three plots of the moderately large catchment of Deep Creek (Trib of Maribyrnong) shown in Fig. 4. Fig. 4A, showing just the main stem of Deep Creek took less than 1 s to extract and plot, Fig. 4B, including all of its tributaries took 4 s, and Fig. 4C, including tributaries and catchment boundary took 15 s.

Table 3. A subset of the stream_names table showing details of the seven streams variously referred to as "Deep Creek".

strcode	str_nm	nextds_strcode	nreach	headreach	termreach	nonheadwater	totlength_km
DPE	DEEP CREEK (KOOWEERUP)	sea	116	134544	98032	0	33.4
DP2	DEEP CREEK SOUTH D.S.	DPE	1	92507	92507	0	0.0
USJ	DEEP CREEK (WESTERNPORT	sea	20	138676	107643	0	4.8
IC6	DEEP CREEK CATCH DRAIN	DPE	26	94008	97702	0	9.5
DPW	DEEP CREEK (TRIB OF	MRB	578	111987	44243	0	154.4
XWS	DEEP CREEK (TRIB OF DRY IN	DRM	20	111773	1223	0	5.2
DPC	DEEP CREEK (TRIB OF	OSH	50	120247	29722	0	12.0
Y66	DEEP CREEK (TRIB OF	RUN	38	115748	25195	0	10.2

Most streams plot significantly faster than this: for instance, a similar plot of Deep Creek, trib of O'Shannassy (see Table 3) took A. 0.1 s, B. 0.3 s, and C. 3 s to plot. The longest such calculation in the dataset (mouth of the Yarra River downstream of Maribyrnong River, more than one-third of the entire region) took 0.4 s, 18 s and 8 min respectively.

The mwstr app provides a web-based means of deriving such subsets of the data, allowing download of the derived catchment boundaries as GIS files.

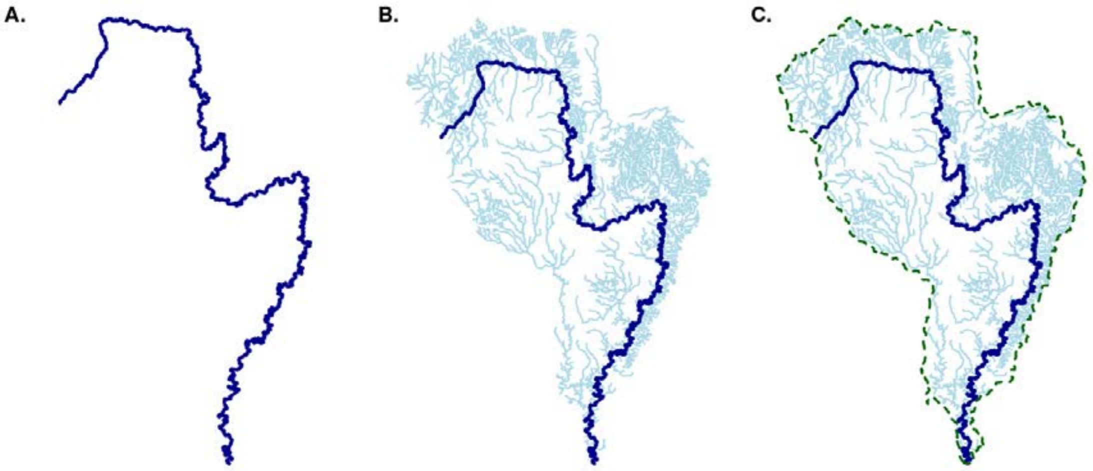
Fig. 4. Code and resulting maps of an example stream (Deep Creek, trib of Maribyrnong), showing A. the main stem; B. main stem and tributaries; C. main stem, tributaries and catchment boundary.

```r
source(paste(wd,"/data_v1.0/mwstr_functions.R",sep = ""))
strcodei <- "DPW"
termreachi <- stream_names$termreach[stream_names$strcode == strcodei]
par(mar = c(0,0,0,0), mfrow = c(1,3))

  plot_allus(site, plotTribs = FALSE, plotCatchment = FALSE)
  title(main = " A.", adj = 0, line = -1)

  plot_allus(site, plotTribs = TRUE, plotCatchment = FALSE)
  title(main = " B.", adj = 0, line = -1)

  plot_allus(site, plotTribs = TRUE, plotCatchment = TRUE)
  title(main = " C.", adj = 0, line = -1)
```



Reach and subcatchment data: table mwsubc_streams

Derived data for reaches (i.e. the 127977 subcatchments with a stream line flowing through them) is stored in the (non-spatial) table mwsubc_streams. This table can be joined to the mwsubc and mwstr layers by the ‘site’ field, which links them all.

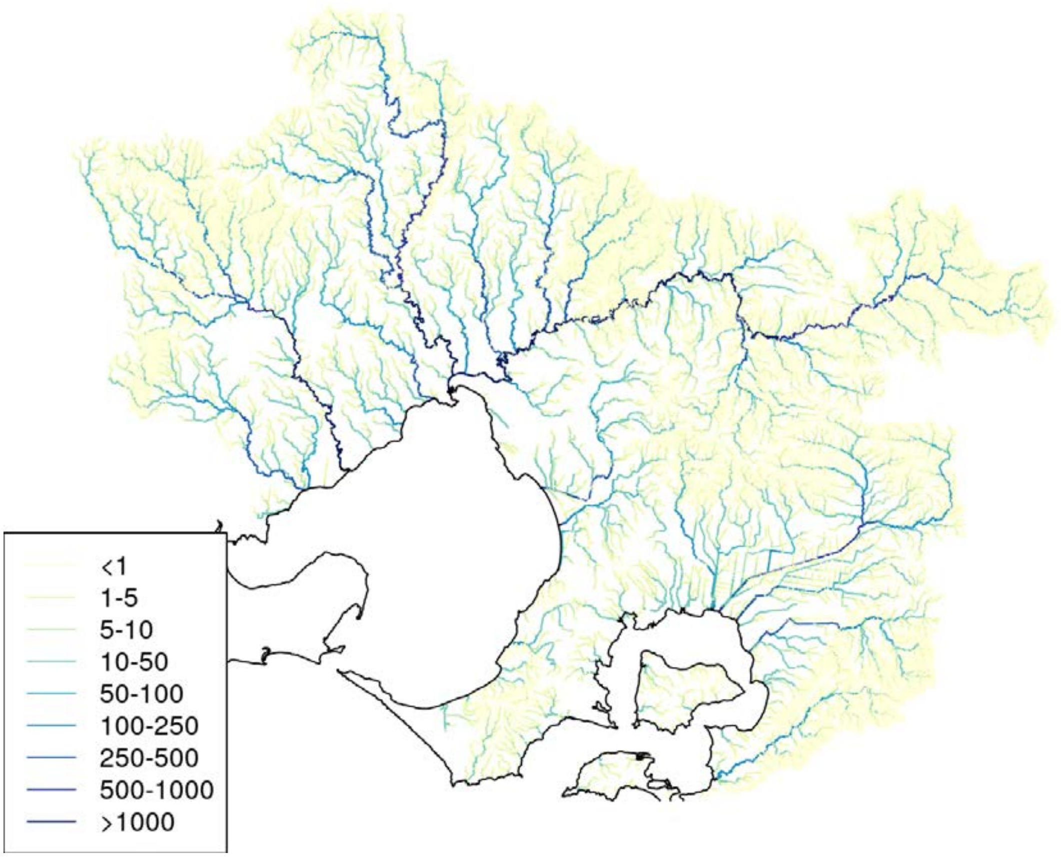
A library of environmental data, which will be added to this table, in preparation. For now, the table contains three fields that are not also in the msubc table:

- length_m, the total length of stream line in each reach in m;
- carea_km2, the total area of all subcatchments including and upstream of each reach in km^2 (Fig. 5). This was calculated using the allus and allus_f fields described above, and

took ~ 3 min to calculate for the all reaches (using the function allusSumVector, which needs further refinement);

- d2ol_km, the total length of stream lines downstream of each reach to the sea in km.
- allds, a comma-separated string of all 'site' codes downstream of each reach. This field is used by the mwstr app when 'show downstream' option is checked to plot downstream reaches.
- The catchment area values were used to derive unique, meaningful 7-character reachcodes for every reach. The reach codes consist of the three-character strcode (see streamNames, above) and a four-digit suffix related to the catchment size of the reach. To avoid any potential confusion with 7-character sitecodes used in earlier datasets (which used, in many cases, the same stream codes, but a different catchment area convention), the strcode and suffix are separated by a hyphen in the new reachcodes.

Fig. 5. Melbourne Water stream network colour-coded by catchment area in km^2.



The suffix is calculated as the rounded value of 1000*log10(catchment area in 100-m^2). Thus, for instance, the reach of the Yarra River at Whittons Reserve, Wonga Park (YAR-7350) has a catchment area of 2237 km^2.Table 4 summarises the ranges of suffix values and the equivalent catchment areas. For 4458 (~3%) of reaches, this scheme resulted in duplicate codes. In those cases, a letter was appended to duplicate codes. For instance, the four reaches of the Yarra River immediately upstream of Fitzsimons Lane, Templestowe have codes YAR-7440D, YAR-7440C, YAR-7440B, YAR-7440A, and YAR-7440.

Table 4. Ranges of suffix values used in the field reachcode and the equivalent catchment areas.

Code range	Catchment area (km2)
0301-1000	0.0002-0.001
1001-2000	0.001-0.01
2001-3000	0.01-0.1
3001-4000	0.1-1
4001-5000	1-10
5001-6000	10-100
6001-7000	100-1000
7001-7739	1000-5483

Appendices

Detailed methods: still in draft as at 9 Dec 2019.