

WIDataRipper: A simple quick way of getting NSW Hydrological data from within **R**.

Jason Lessels

August 29, 2011

Contents

1	Introduction	3
2	Installation of WIDataRipper	3
3	The main functons.	3
4	The added bonus functions	6
4.1	Sites within geographical boundaries	6
5	Further upgrades	10

1 Introduction

WIDataRipper is designed to provide a simple and quick method to get data from the NSW water info website (Real-time water data). The main aim of this package is to provide the ability of the direct importation of data from the web server. Additional functions have been added to the package to allow for more complicated searches and meta data queries.

2 Installation of WIDataRipper

To install the WIDataRipper package, both the Rcurl and the rjson packages are required.

```
> install.packages("Rcurl")  
> install.packages("rjson")
```

To install the WIDataRipper package the R-Forge repository must be provided.

```
> install.packages("WIDataRipper", repos="http://R-Forge.R-project.org")
```

3 The main functions.

WIDataRipper has several functions designed to be used in conjunction to get the desired data off the server. The main work flow assumes the user knows the desired site number. In upcoming releases a site name query will be added. My current work involves a site near Coolac, south west of Canberra. The following examples will provide an example of how to use this package to obtain desired data.

The first stage in obtaining the data, is to first get some meta-data about the site. Using the function `getSiteInfo`.

```
> library(WIDataRipper)  
> getSiteInfo(410044)
```

```
$siteName  
[1] "MUTTAMA CREEK AT COOLAC"
```

```
$siteShortName  
[1] "MUTTAMA CK @ COOLAC"
```

```
$latitude  
[1] -34.9304
```

```
$longitude  
[1] 148.1628
```

```
$elevation  
[1] 234.234
```

```
$geoDatum
[1] "GDA94"
```

```
$comments
[1] "Site location was fixed using a Silvia Navigator handheld GPS in October 2003. Point of r
```

```
$stillActive
[1] TRUE
```

With the results of this function, we now have the site location and the elevation and any comments about the site, and the data recording process at the site. The next important piece of meta-data is the available variables at the site and the length of time they have been collected for. However, due to the setup of the server, there are potentially several data sources for each site. Below the available data sources for the site are obtained.

```
> getSiteDataSources(410044)
```

```
$site
[1] "410044"
```

```
$dataSources
[1] "A"      "PROV"
```

From my current understanding data source ‘PROV’ are any samples that have not undergone proper quality coding. That is to say that no one from the department has looked at these values in any real detail. There is two important things to note about this. There is overlap between some of these values from each data source. The second thing to note is that the latest bleeding edge values from each site tend to be within the ‘PROV’ data source.

The next stage is to find out what variables are within each data source for the site. The first time will be for the ‘A’ data source.

```
> getSiteVariables(410044,data_source="A")
```

```
$siteName
[1] "MUTTAMA CREEK AT COOLAC"
```

```
$siteShortName
[1] "MUTTAMA CK @ COOLAC"
```

```
$siteNumber
[1] 410044
```

```
$variables
      startingDate      endingDate      subdesc variable
1 1938-05-05 12:00:00 2011-07-12 12:30:00      100.00
```

2	1975-10-26 11:00:00	1996-10-01 12:00:00	Externally supplied peak	100.09
3	1938-06-01 09:00:00	1975-07-12 08:30:00		101.00
4	1938-01-01	2012-01-01	monthly max	141.01
5	1938-01-01	2012-01-01	Monthly Min	141.02
6	1938-01-01	2012-01-01		151.00
7	1938-01-01	2012-01-01	Monthly Tot	151.01
8	1801-01-01	2101-01-01	Yearly Total	151.02
9	2001-10-25 11:10:00	2011-07-12 12:15:00		2010.00
10	2010-04-30 08:00:00	2011-07-12 12:15:00		2012.00
11	2001-10-25 11:10:00	2011-07-12 12:15:00		2080.00

	units	name
1	Metres	Stream Water Level
2	Metres	Stream Water Level
3	Feet	Stream Water Level
4	Megalitres/Day	Stream Discharge
5	Megalitres/Day	Stream Discharge
6	Megalitres	Discharge Volume
7	Megalitres	Discharge Volume
8	Megalitres	Discharge Volume
9	microsiemens/cm	Electrical Conductivity @ 25deg. C
10	microsiemens/cm	Electrical Conductivity (Uncompensated)
11	Degrees Celsius	Water Temperature

>

The next enquiry will be for the 'PROV' data source.

```
> getSiteVariables(410044,data_source="PROV")
```

```
$siteName
```

```
[1] "MUTTAMA CREEK AT COOLAC"
```

```
$siteShortName
```

```
[1] "MUTTAMA CK @ COOLAC"
```

```
$siteNumber
```

```
[1] 410044
```

```
$variables
```

	startingDate	endingDate	subdesc	variable	units
1	2010-02-08 10:00:00	2011-08-30 07:00:00		100.00	Metres
2	2010-02-08 23:59:00	2011-08-30 07:00:00		300.00	Volts
3	2010-02-08 10:00:00	2011-08-30 07:00:00		2010.00	microsiemens/cm
4	2010-04-30 08:00:00	2011-08-30 07:00:00		2012.00	microsiemens/cm
5	2010-02-08 10:00:00	2011-08-30 07:00:00		2080.00	Degrees Celsius
6	2010-04-30 08:00:00	2010-06-07 09:00:00		2169.00	Milligrams/Litre

	name
1	Stream Water Level
2	Logger Battery Voltage

```

3      Electrical Conductivity @ 25deg. C
4 Electrical Conductivity (Uncompensated)
5      Water Temperature
6 Inst. Salinity (Total Dissolved Salts)

```

The main difference between the two results are the starting and ending dates. However The 'PROV' data source provides a few additional variables: Logger Battery Voltage and Inst. Salinity.

With all the above meta-data gathered it is now possible to get the desired data. The method for this is the following

```

> streamHeight <- getData(site_number=410044,start_time="20110101000000",
+ end_time="20110201000000",interval="day",variable_number=100)

```

Sending request to the server

Server responded, now just cleaning up the response.

Make sure you check the quality codes. 255 = missing data, but data is represented by 0's.

And the EC can also be obtained.

4 The added bonus functions

OK, so that is the it with the main functions now on to some other functions. Mainly designed for searching the data base. There is one additional function `getLatest` that provides the ability to get the last 7 days of observations. The options with this function are limited and is mainly designed to help users keep track of what it happening at their study site. It could be setup to run when R is started every day.

```

> latest <- getLatest(410044,variable_number=100)

```

Sending request to the server

Server responded, now just cleaning up the response.

Make sure you check the quality codes. 255 = missing data, but data is represented by 0's.

The remainder of the functions within the package provide the ability to search the data base. The function `getAllSites` queries the server for every site in the data base. The function returns a data.frame with every name of every site. There are a little over 3000 sites, so one might want to just believe me that it works.

4.1 Sites within geographical boundaries

There are currently two functions that allow for searching for sites within geographical boundaries `getSitesWithinCircle` and `getSitesWithinrectangle`. Both functions retrieve a list formatted in the same style as the `getSiteInfo` function. But contain all sites with either a specified circle or rectangle.

```
> plot(streamHeight$data[,1:2],type="l")
```

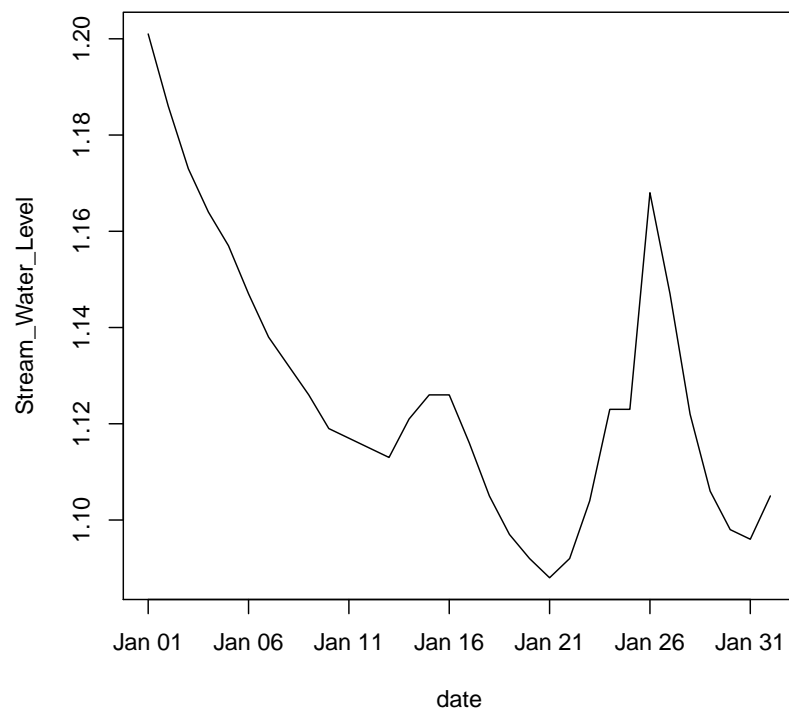


Figure 1: An example of daily stream height for one month

Sending request to the server
Server responded, now just cleaning up the response.
Make sure you check the quality codes. 255 = missing data, but data is represented by 0's.

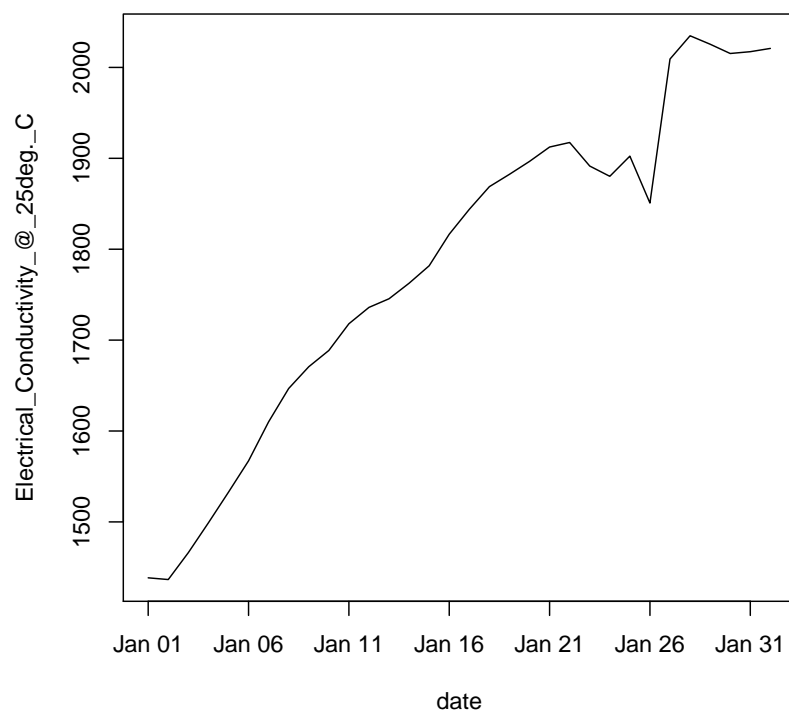


Figure 2: An example of daily EC for one month


```
> plot(latest$data[,1:2],type="l")
```

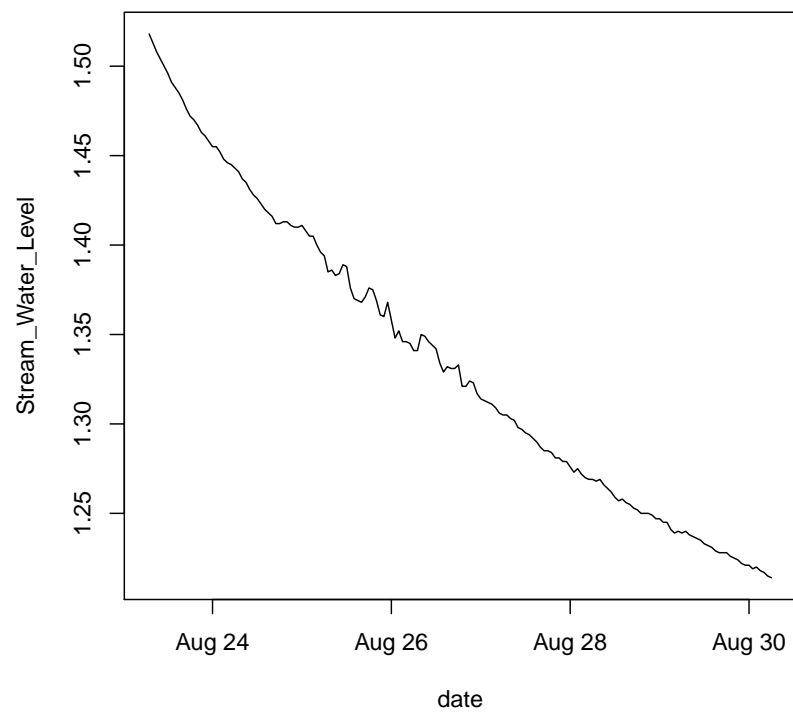


Figure 3: An example of the latest function.

5 Further upgrades

In upcoming releases two additional functions will be added. A search by site name and a search by town name. An export function for the geographical functions will be added allowing for the conversion to a **SpatialPointsDataFrame** from the **sp** package. I hope to include the ability to search for all sites within a given polygon, that accepts a **sp** formatted object.

Other search queries are possible, but I do not have any other needs personally. I am happy to add additional search queries on request.