Setting up your system for Geo-scripting with R

Loïc Dutrieux

October 16, 2013

1 Introduction

Having your system properly set-up is a cricial step before before starting any geo-processing with R. This short tutorial will guide you through some important steps that you should follw in order to have the required software installed and appropriately set-up. Figure 1 provides an overview of the different components of the system and how they are connected with each others.

2 On windows

R and R Studio

TODO Refer to R manual and R Studio Manual

Rtools

TODO Provide download page. Has to fit the version of R you are using. Important to select the option "add the following directories to your system path".

GDAL/OGR/PROJ4 libraries

FWTools, a bit outdated. Installation via OSGeo4W installer. Possibly the bin directory path has to be appended to system PATH.

Version control software

Installing these software is not an absolute necessity, however it might come handy if you plan to implement some of the good practices discussed in Lesson 3. Refer to Lesson 3 for more details.

Git and SlickSVN

Most likely you need one OR the other, depending on whether your repository is hosted on a git or Subversion version control system.

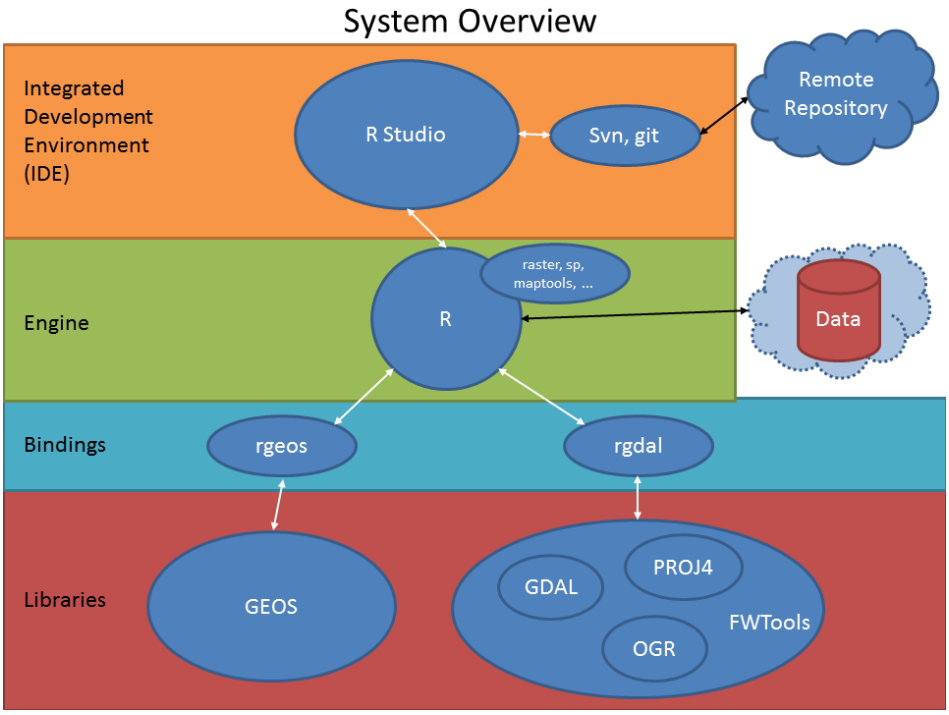

Figure 1: System architecture overview[1]

Integration with the R Studio IDE

It is possible that R Studio automatically detects Git and SlickSVN on your system, however this is not systematic. In case the Git/Svn buttons in R Studio are innactive, after installing either Git or SlickSVN, follow the steps described bellow.
Once SlikSVN (or git) has been installed, in order to allow it to interact with RStudio, its path has to be set. Conveniently that can be done directly from within RStudio. In R Studio, navigate to Tools - Options - Git/SVN - SVN (or Git) - Browse and select the SVN.exe (or git.exe) executable in C:/Program Files/SlikSVN/bin/ (or C:/Program Files/git/bin/) Restart RStudio.

Specific R packages

Not all packages are hosted on the CRAN package repository, this is the case for instance of the MODIS package that is maintained on R Forge, and requires an extra argument in the `install.packages()` function to be installed. The install line, to be copy pasted in the R shell is often given on the package R forge home page.

[1]Note that FWTools is one example of binary distribution for windows, you can also install gdal/ogr/proj.4 from OSGeo4W, (linux: by compiling it yourself from source or from a package archive)