

# OSGeo/R interfaces and upstream contributions

---

Roger Bivand

## Spatial data analysis in R

---

# The R package management system

- R is a free software environment for statistical computing and graphics.
- Packages add functionality to R, and can contain interpreted R code, compiled C, C++ and/or Fortran code, and can link to external libraries
- The Comprehensive R Archive Network (CRAN) not only distributes packages, but importantly lets R developers see what packages might get broken as R develops (arm64, Windows Universal C RunTime UCRT)
- It also checks all > 18,000 CRAN packages daily on multiple platforms, also detecting changes in packages that break other packages, and warns package maintainers of the need to help each other
- So the R package management system is tightly integrated, and this is seen as important by the R community

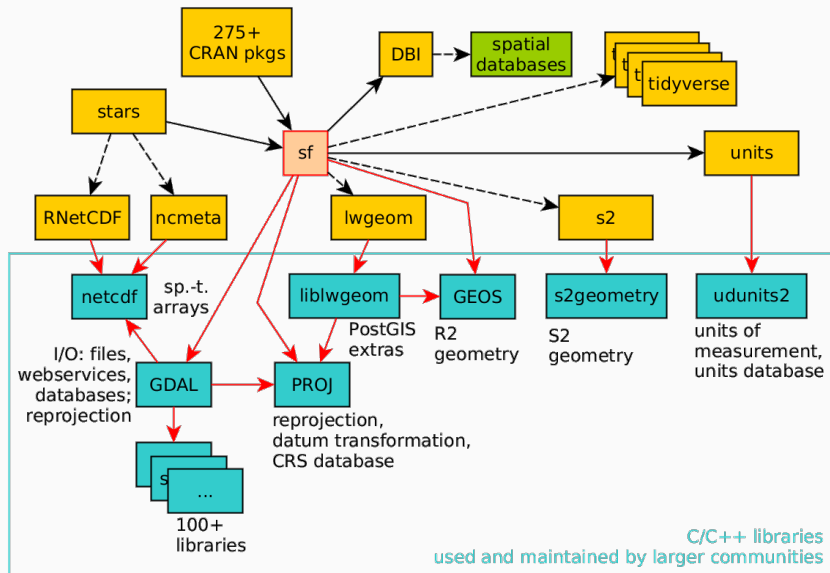
## Binary packages for macOS and Windows

- Source packages can be installed by all users across platforms, but binary packages are provided for macOS and Windows to reduce install times, and to work around security hurdles (like permissions for `rJava.dylib` on macOS)
- These binary packages are built static when they include compiled code for these platforms; this includes external libraries, which then need to be easy to build as clean, static libraries
- Static linking should ensure that distributions are complete (Brian Ripley).
- This is the way that the R package management system attempts to stabilize its software environment, rather than choosing containers or environments of dynamically linked libraries

## Reproducibility is important in the R community

- In R and S (R inherited a lot from S), reproducing statistics textbook examples was always central; even if bug-fixes change the output, we want to know what caused the change
- Journal of Statistical Software and the R Journal focus on reproducibility, and CRAN checking and testing tries to detect changes caused by changes in R, other R packages or external libraries
- Explicit and consistent version and dependency declaration is important in order to reconstruct the software environment underlying results
- Even setting locales can be important to see where changes are coming from; sometimes changes are many steps away in the dependency graph just in R packages, let alone in external software

# OSGeo libraries supporting spatial data analysis in R



## Upstream OSGeo libraries

- PROJ is really important, and has been used to define coordinate reference systems and carry out coordinate operations (projection, transformation) for almost twenty years
- GEOS was first interfaced to an R package ten years ago, and its use has increased since then
- GDAL bindings were first created as an R package twenty years ago, initially just for raster data, but quickly extended to vector data
- R packages for spatial data have been updated as PROJ, GEOS and GDAL have evolved, quite fast since the GDAL barn-raising occurred
- Following the development email lists: **proj**, **geos-devel**, **gdal-dev**, **grass-user**, **grass-dev**, **grass-stats** has been crucial

When upstream OSGeo libraries  
change



---



## First steps

- Using the development email lists is really important, because it is there most changes are discussed, and issues especially around releases are highlighted
- Just yesterday for example, Greg Troxel commented on **geos-devel** WRT the forthcoming 3.10 release: *What really hurts in packaging is needing to synchronize things. The upcoming postgis release needs to work with a bunch of postgres versions, not necessarily IMHO the released-tomorrow one, and it should work with all geos versions released a year ago or more recently.*
- This is what R spatial packages need too: CentOS may be stuck with very old PROJ, GDAL and GEOS versions, but lots of labs use it
- Use repository issues/tickets; most of mine have been about version numbering and similar packaging issues

# PRØ issues/tickets

 Search or jump to...  Pull requests Issues Marketplace Explore

OSGeo / PROJ Public Watch 74 Star 1.1k Fork 502

<> Code Issues 40 Pull requests 3 Actions Projects Wiki Security Insights

Filters is:issue rsbivand Labels 28 Milestones 2 New issue

Clear current search query, filters, and sorts

1 Open 8 Closed

Author Label Projects Milestones Assignee Sort

✓	Fedora RPM build via cmake has no static libraries	bug	3
#2753 by rsbivand was closed on 23 Jun			
○	Raw LAS WKT seem not to render with projinfo	bug	15
#2022 opened on 5 Mar 2020 by rsbivand			
✓	NOTE: rgdal::checkCRSArgs: no proj_defs.dat in PROJ.4 shared file		13
#351 by lreng was closed on 13 Feb 2016			
✓	Where is TRAC bug tracker?		5
#293 by rsbivand was closed on 8 Sep 2015			
✓	different behaviour across platforms: pj_init_plus	bug C: default P: major	1
#274 by proj4-bot was closed on 9 Nov 2015 ↗ 4.9.1			
✓	Align CMake SOVERSION with autoconf	bug C: default P: major	8
#263 by proj4-bot was closed on 22 May 2015 ↗ 4.9.1			
✓	pj_init_plus() with init and other parms fails in 4.9.0beta	bug C: Core P: major	15
#229 by proj4-bot was closed on 22 May 2015 ↗ 4.9.0			
✓	The +no_uoff flag for omerc has been renamed to +no_off in trunk.	bug C: Core P: minor	6
#128 by proj4-bot was closed on 17 Oct 2016			
✓	pj_get_release() string bug	bug C: default P: critical wontfix	3
#99 by proj4-bot was closed on 22 May 2015			

## Handling upstream changes

- Sometimes issues raised by R package maintainers, in concert with other packagers, lead over time to improved version declarations
- Installing beta and release candidate libraries, running `R CMD check` on maintained packages, and adapting early helps (though with many version checks in `configure.ac` and 58 `#if` in `sf`'s compiled code source for PROJ, GEOS and GDAL versions)
- Once R packages interfacing external libraries have been adapted if necessary, it is time for reverse dependency checks on downstream R packages using the R interface packages
- Typically, now about 1,000 packages need checking; 1 time 5-core revdep check runs to about 5 hours, so a day+night on one core; these checks pick up usages not tested for or even dreamt of in interface packages

## Example: PROJ (correctly) bans UTM zone zero

```
> library(sf)
Linking to GEOS 3.9.1, GDAL 3.2.3, PROJ 7.2.1
> st_crs("+proj=utm")
Coordinate Reference System:
User input: +proj=utm
wkt:
PROJCRS["unknown",
  BASEGEOGCRS["unknown",
    DATUM["World Geodetic System 1984",
      ELLIPSOID["WGS 84",6378137,298.257223563,
        LENGTHUNIT["metre",1]],
      ID["EPSG",6326]],
    PRIMEM["Greenwich",0,
      ANGLEUNIT["degree",0.0174532925199433],
      ID["EPSG",8901]]],
    CONVERSION["UTM zone 0N",
      METHOD["Transverse Mercator",
        ID["EPSG",9807]],
        PARAMETER["Latitude of natural origin",0,
          ANGLEUNIT["degree",0.0174532925199433],
          ID["EPSG",8801]],
          PARAMETER["Longitude of natural origin",-183,
            ANGLEUNIT["degree",0.0174532925199433],
            ID["EPSG",8802]],
            PARAMETER["Scale factor at natural origin",0.9996,
              SCALEUNIT["unity",1],
              ID["EPSG",8805]],
              PARAMETER["False easting",500000,
                LENGTHUNIT["metre",1],
                ID["EPSG",8806]],
                PARAMETER["False northing",0,
                  LENGTHUNIT["metre",1],
                  ID["EPSG",8807]],
                  ID["EPSG",16000]],
                  CS[Cartesian,2],
                    AXIS["(E)",east,
                      ORDER[1],
                      LENGTHUNIT["metre",1,
                        ID["EPSG",9001]]],
                      AXIS["(N)",north,
                        ORDER[2],
                        LENGTHUNIT["metre",1,
                          ID["EPSG",9001]]]]]
> library(sp)
> CRS("+proj=utm")
CRS arguments:
+proj=tmerc +lat_0=0 +lon_0=-183 +k=0.9996 +x_0=500000 +y_0=0
+datum=WGS84 +units=m +no_defs
```



rsbivand commented 24 days ago

Author ...

The `+proj=utm` problem is caused by [OSGeo/PROJ#2672](#) from [OSGeo/PROJ#2671](#) and applies in PROJ  $\geq 8$ , probably backported to 8.0.1 and applies in 8.1.1. Previously a non-existing zone 0 was generated.

```
> library(sf)
Linking to GEOS 3.10.0dev, GDAL 3.3.2, PROJ 8.1.1
> st_crs("+proj=utm")
Error in st_crs.character("+proj=utm") : invalid crs: +proj=utm
In addition: Warning message:
In CPL_crs_from_input(x) :
  GDAL Error 1: PROJ: proj_create: Invalid zone number
> library(sp)
> CRS("+proj=utm")
proj_create: Invalid zone number
proj_create: Invalid zone number
Error in CRS("+proj=utm") : NA
> |
```

## Example: what happened to +proj=ob\_tran in PROJ 7.2.1RC

On Tue, 29 Dec 2020, Roger Bivand wrote:

> On Mon, 28 Dec 2020, Even Rouault wrote:

>

>> On lundi 28 d?cembre 2020 16:41:09 CET Kristian Evers wrote:

>

>>> Thanks for clearing this up, Even. The workflow is a bit cumbersome but  
>>> with the knowledge you've provided here it should be possible for Roger  
>>> to adapt the code to work around this.

>>>

>>> I agree that a function to determine if a CRS is derived would be a nice  
>>> addition.

>>

>> Queued in <https://github.com/OSGeo/PROJ/pull/2496> . I've targetted it for  
>> 8.0. Could be easily backported if deemed necessary.

>>

>

> Thanks! Handled with

>

> if (proj\_get\_type(target\_crs) == PJ\_TYPE\_GEOGRAPHIC\_2D\_CRS && use\_ob\_tran) {  
> if ((source\_crs = proj\_get\_source\_crs(PJ\_DEFAULT\_CTX, target\_crs)) ==  
> 0)

> ...

>

> where use\_ob\_tran is declared by the user as TRUE if the user wishes, and the  
> involved CRS include the proj=ob\_tran string.

>

> Tested OK with 7.2.1 RC1 and 7.2.0; now need to check for earlier releases.

> Current code on R-Forge SVN, revisions 1086-1088.

Check with PROJ 6.3.1/GDAL 3.0.4 OK.

No further problems with RC1.

Roger

## Example: multiple proj.db across the EPSG 9/10 upgrade

In `rgrass7`, `getLocationProj()` calls GRASS `g.proj -w` if GDAL and PROJ are post-barn-raising, but on Windows, R packages and GRASS may bundle different `proj.db` versions:

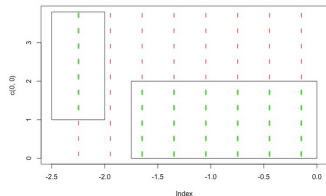
```
> getLocationProj()
ERROR 1: PROJ: proj_as_wkt: SQLite error on SELECT name, ellipsoid_auth_name,
ellipsoid_code, prime_meridian_auth_name, prime_meridian_code,
area_of_use_auth_name, area_of_use_code, publication_date,
deprecated FROM geodetic_datum WHERE auth_name = ? AND code = ?:
no such column: area_of_use_auth_name
[1] ""
> Sys.getenv("PROJ_LIB")
[1] "C:/Users/RB/Documents/R/win-library/4.1/rgdal/proj"
```

## Example: multiple proj.db across the EPSG 9/10 upgrade

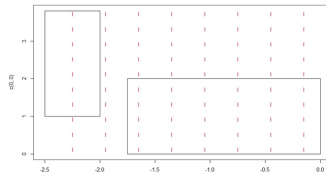
```
> packageVersion("rgdal")
[1] '1.5.27'
> rgdal::rgdal_extSoftVersion()
      GDAL GDAL_with_GEOS      PROJ      sp      EPSG
      "3.2.1"      "TRUE"      "7.2.1"      "1.4-5"      "v10.008"
> library(RSQLite)
> db <- dbConnect(SQLite(), dbname=file.path(
  "C:/Program Files/GRASS GIS 7.8/share/proj", "proj.db"))
> (metadata <- dbReadTable(db, "metadata"))
      key      value
1 EPSG.VERSION      v9.8.6
...
> dbDisconnect(db)
```

# Example: GEOS OverlayNG and GEOS 3.9.0/3.9.1 in dsims

Behaviour on mac (correctly identifies segments inside multipolygon):



Behaviour on windows (nothing):



Windows machine:

```
> packageVersion("sf")
[1] '1.0.2'
> sf::st_extSoftVersion()
      GEOS      GDAL      proj.4      GDAL_with_GEOS      USE_PROJ_H      PROJ
      "3.9.0"      "3.2.1"      "7.2.1"      "true"      "true"      "7.2.1"
```

Mac machine:

```
> packageVersion("sf")
[1] '1.0.2'
> sf::st_extSoftVersion()
      GEOS      GDAL      proj.4      GDAL_with_GEOS      USE_PROJ_H      PROJ
      "3.8.1"      "3.2.1"      "7.2.1"      "true"      "true"      "7.2.1"
```

de-jts commented 21 days ago

There is a bug in GEOS 3.9.0 which causes overlay to essentially drop straight lines if they are axis-parallel. It very much sounds like this is what you are seeing. The bug was [fixed](#) in GEOS 3.9.1.

If it's not possible to upgrade, two workarounds are:

- rotate the lines a very small amount
- use `Intersects` to detect if a line is wholly inside a polygon, and if so just include it in the result directly

rsbivand commented 20 days ago

@de-jts thanks very much for joining in, very helpful. @LHM Marshall, you could adapt your workflow to rotate slightly if GEOS == 3.9.0 but not otherwise. The rwinlib GEOS version needs to be updated @jeroen could you please replace GEOS 3.9.0 with GEOS 3.9.1 in rwinlib/gdal3? Reason is [locationtech/ja685](#) fixing [fgeos/geos#408](#), which was a regression from 3.8.\* and earlier that occurred in moving to the new and better overlay engine.

jeroen mentioned this issue 20 days ago

geos 3.9.1 r-windos/tools-packages#225

jeroen added a commit to rwinlib/gdal3 that referenced this issue 20 days ago

Update to GEOS 3.9.1

4425e33

jeroen commented 20 days ago

I have updated the [rwinlib/gdal3](#), and also ported to the v3.2.1 tag. So new builds of sf, geos, etc will automatically use geos 3.9.1 now.

The CRAN winbuilder rebuilds packages approximately once per week, so it may take a few days before the windows binaries on CRAN are fixed.



## Summary

- Without OSGeo libraries and the active and generous participation of their developers, there would be no R spatial
- Providing timely feedback upstream is essential for R spatial, and although irritating at times for upstream library developers, does not differ much from feedback from other packagers
- The broader R community is also interested in collaborating in efforts to clean up and test `pkg-config`, `configure.ac` and `cmake` especially for static builds, which will make handling new versions smoother
- We now have plenty of experience with Windows Universal CRT too; also cross-compilation (`spatialite` has `#include <Windows.h>` but the file in MinGW is `windows.h` and of course the cross-building platform (MXE) is case sensitive ..., Tomas Kalibera)