TOWARDS AN EMPIRICAL ANALYSIS OF THE MAINTAINABILITY OF CRAN PACKAGES

UMONS Université de Mons

Maëlick Claes

Tom Mens & Philippe Grosjean

Software Engineering Lab & Numerical Ecology of Aquatic Systems Lab

Faculté des Sciences

17th December 2013, BENEVOL

INTRODUCTION

R & CRAN

- http://www.r-project.org
- Statistical environment based on the S language
- Package system
- Packages contain code (R, C, Fortran,...), documentation, examples, tests, datasets,...
- Dependency relations between packages

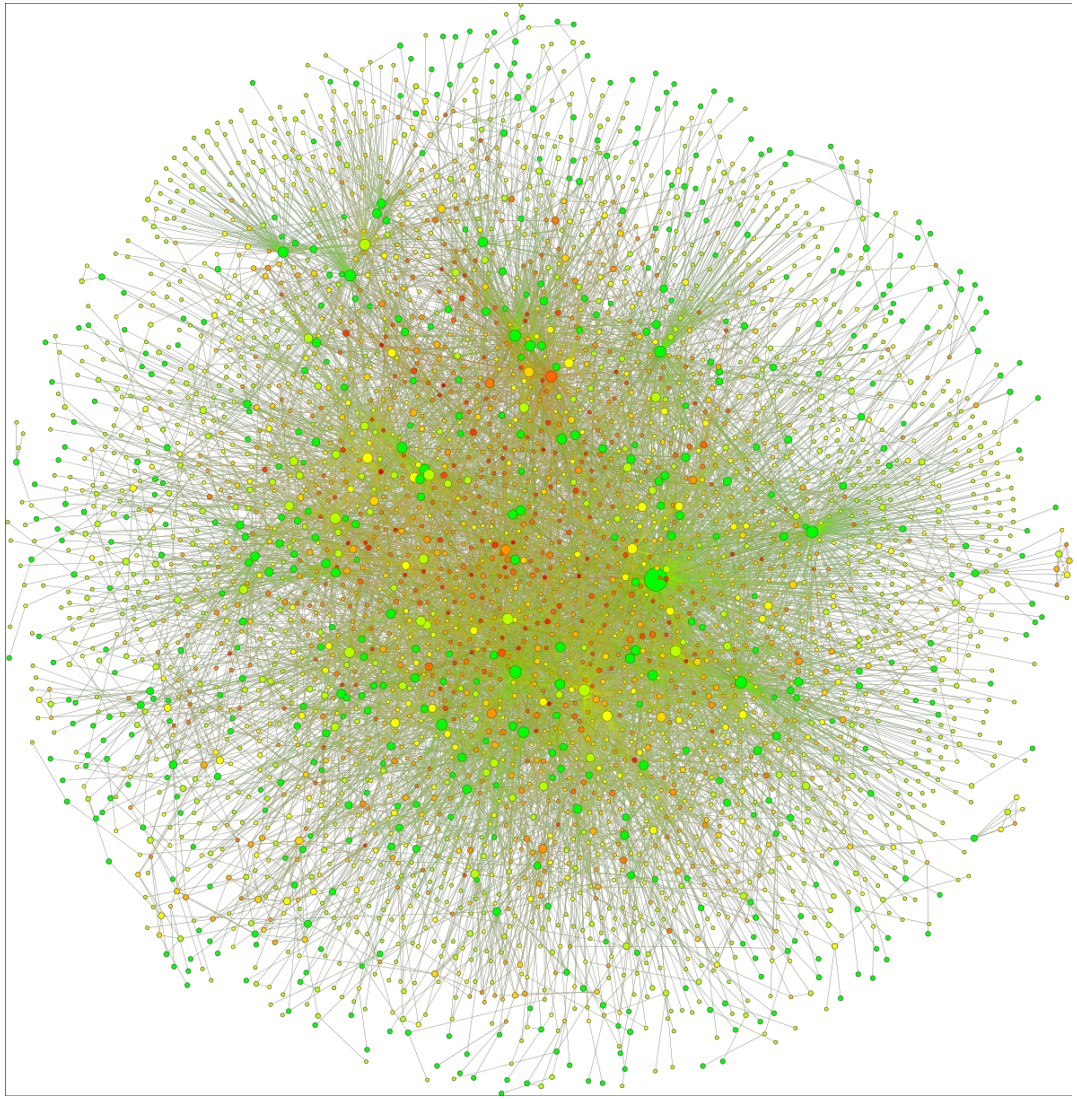
CRAN

- http://cran.r-project.org
- "Official" repository containing more than 5000 packages
- Strict policy for package acceptance
- Package quality regularly checked & archive process.
- Complaints in the community Hornik 2012, *Are there too many R packages?*

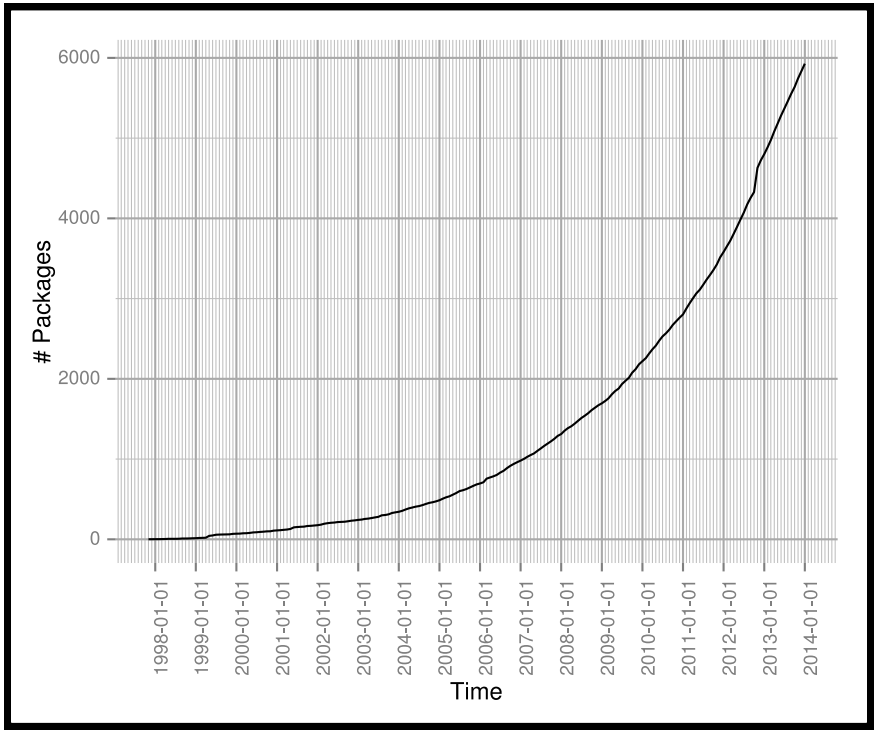
"DESCRIPTION" FILE

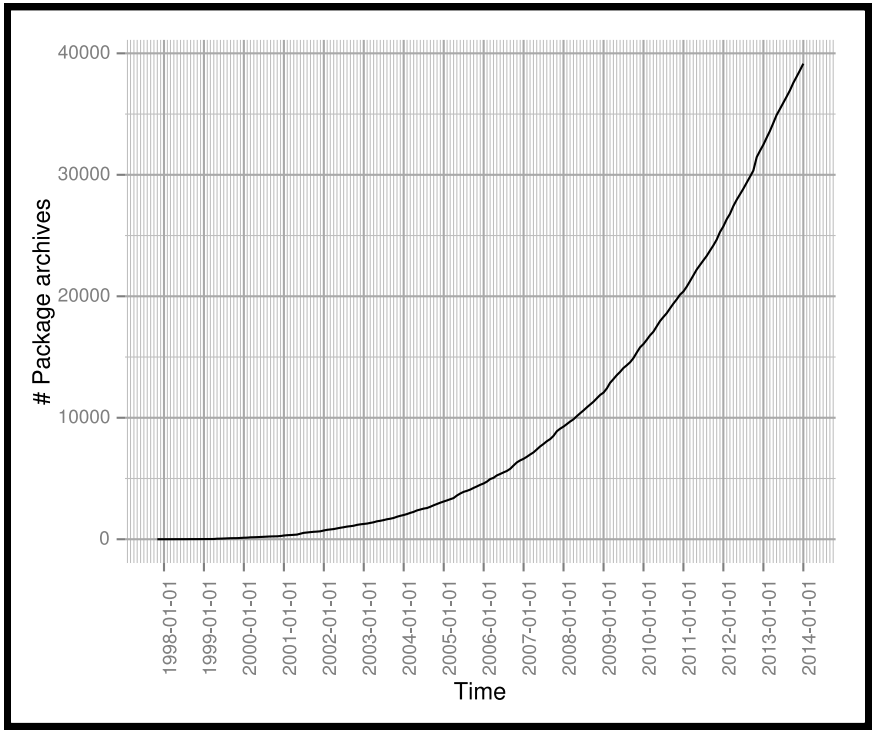
```
Package: SciViews
Type: Package
Title: SciViews GUI API - Main package
Imports: ellipse
Depends: R (>= 2.6.0), stats, grDevices, graphics, MASS
Enhances: base, stats
Description: Functions to install SciViews additions to R, and more
        (various) tools
Version: 0.9-4
Date: 2011-07-26
Author: Philippe Grosjean
Maintainer: Philippe Grosjean <phgrosjean@sciviews.org>
License: GPL-2
LazyLoad: yes
URL: http://www.sciviews.org/SciViews-R
BugReports: https://r-forge.r-project.org/tracker/?group_id=194
Packaged: 2011-07-26 07:20:19 UTC; phgrosjean
Repository: CRAN
Date/Publication: 2011-07-26 10:14:11
```

PACKAGE DEPENDENCIES



HAS GROWTH BECOME LINEAR?





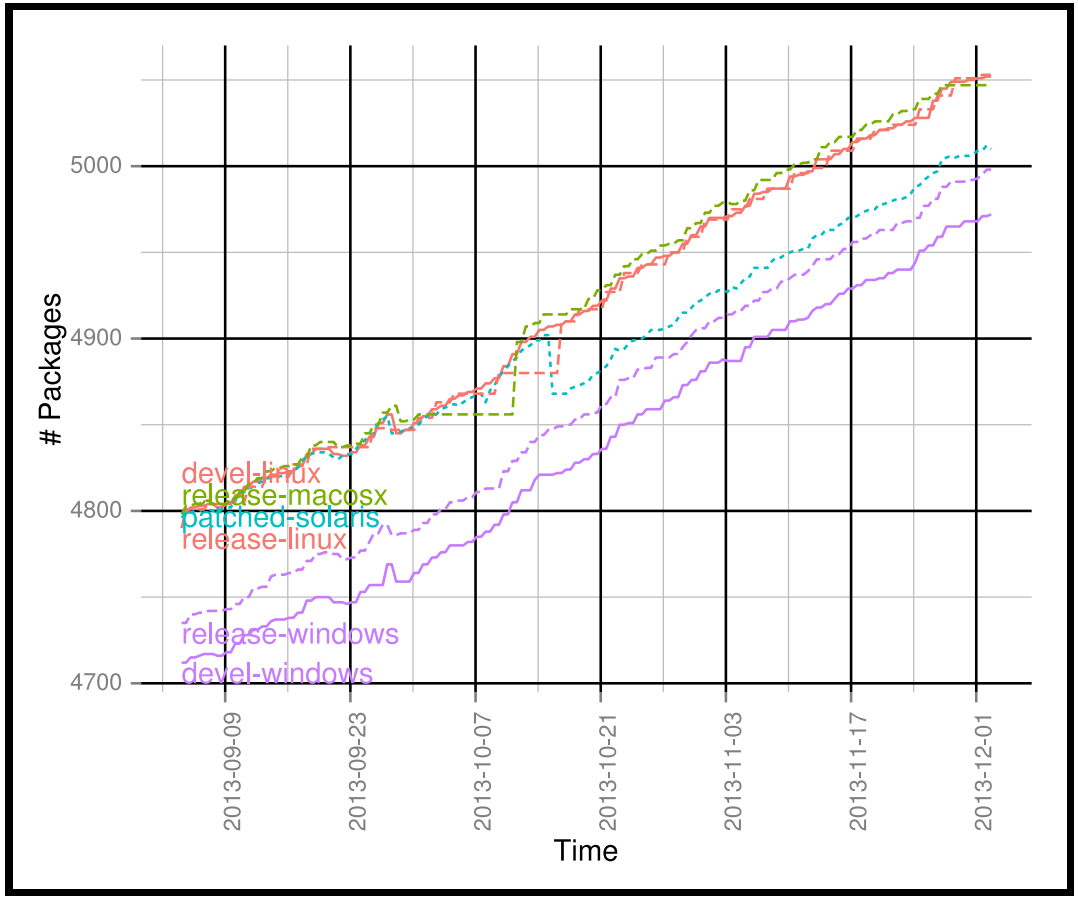
CRAN CHECKS

- *R CMD check* command tool for checking package correctness and quality
- Package status: *OK*, *NOTE*, *WARNING* or *ERROR*
- Every package release must pass the test on two OS to be accepted on CRAN
- *R CMD check* rerun daily on the whole set of packages
- Packages with inactive maintainer and ERROR status are archived when the next non-minor version of R is released
- A combination of R version, OS, architecture and C compiler forms a *flavor*

EXAMPLES

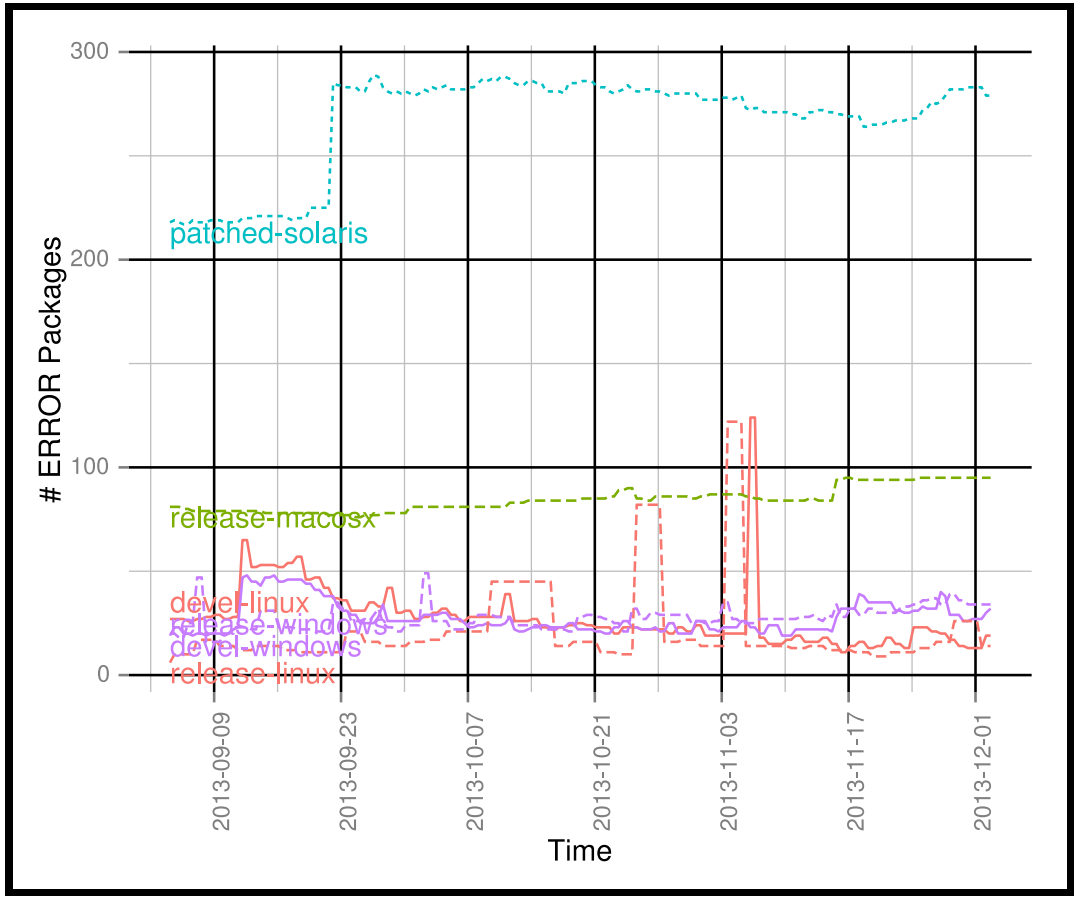
- Package listing
- Package with ERROR status

NUMBER OF PACKAGES FOR EACH FLAVOR

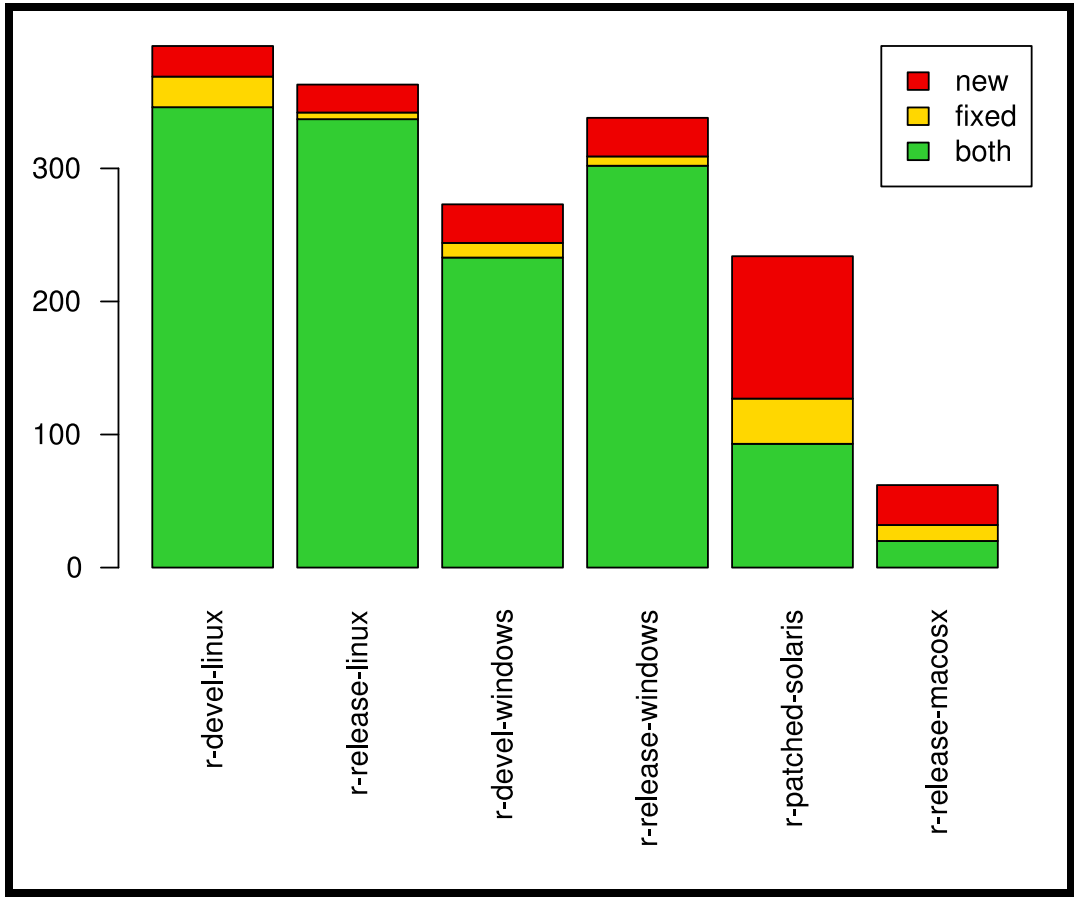


HOW DO FLAVORS IMPACT ERRORS?

EVOLUTION OF ERRORS FOR EACH FLAVOR



NUMBER OF ERRORS EXTRACTED

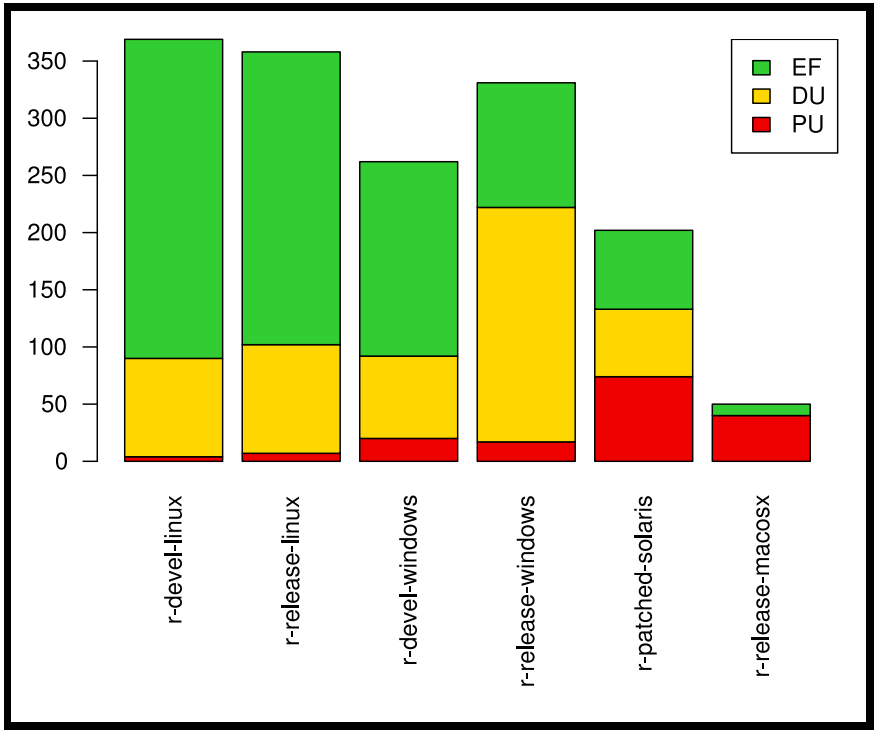


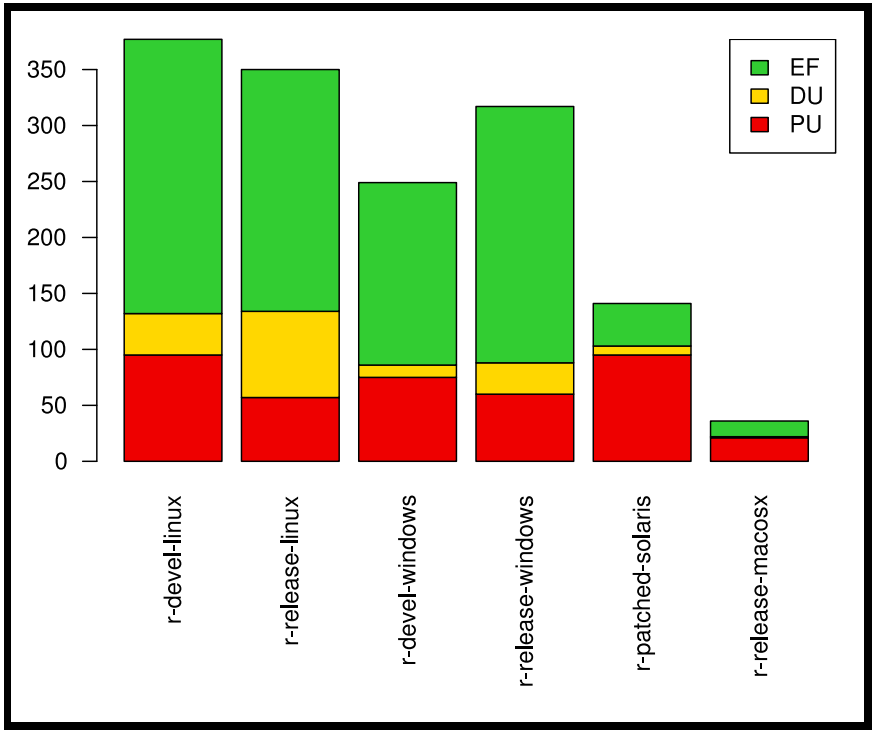
WHAT IS THE CAUSE OF ERRORS IN CRAN AND HOW ARE THEY FIXED?

SOURCES OF STATUS CHANGES

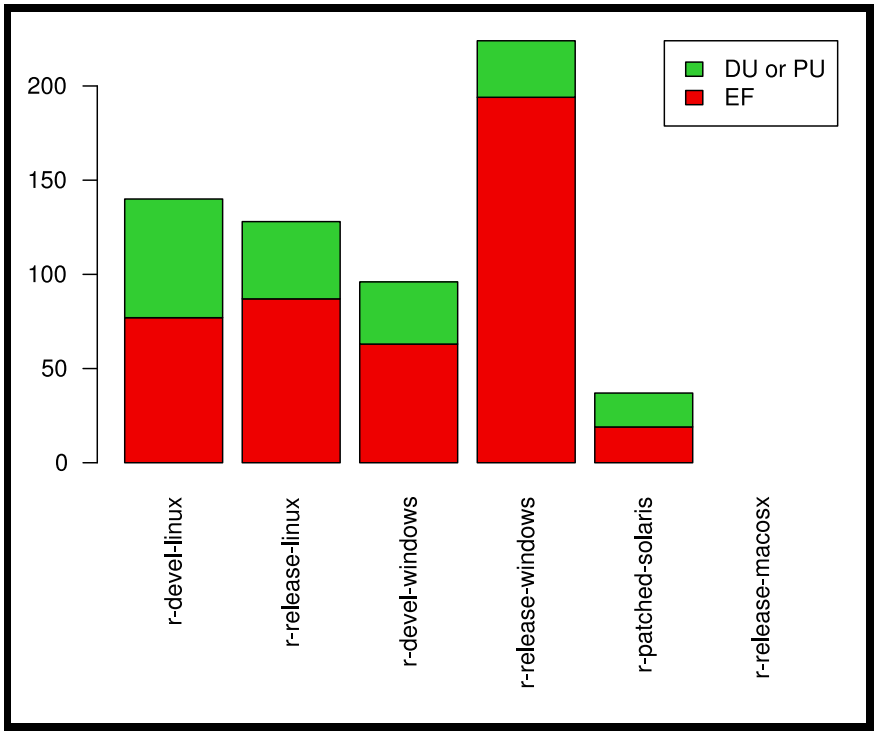
- **Package Update (PU)**
  - ERROR status change coincides with the release of a new package version
- **Dependency Update (DU)**
  - ERROR status change coincides with the release of a new package version of a dependency
- **External Factors (EF)**
  - ERROR status change without a new package version or a new version of any dependency

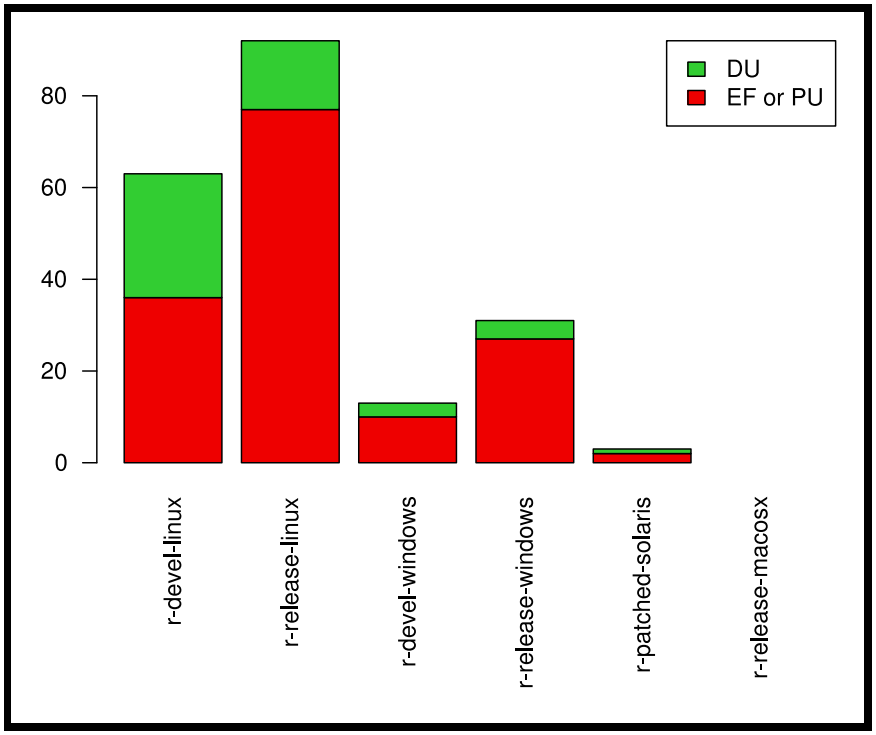
SOURCES OF NEW ERRORS AND ERROR FIXES



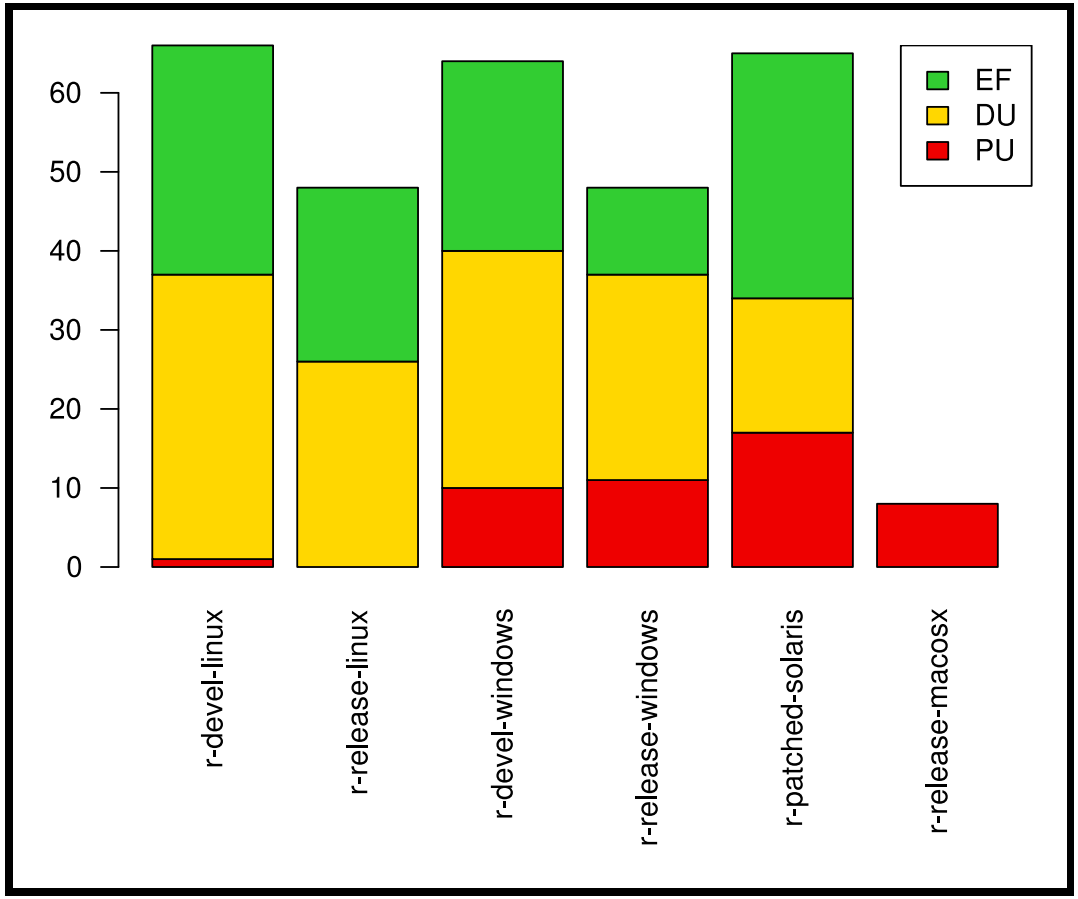


ARE ERRORS CAUSED AND FIXED BY DEPENDENCY UPDATE REALLY CAUSED AND FIXED BY IT?



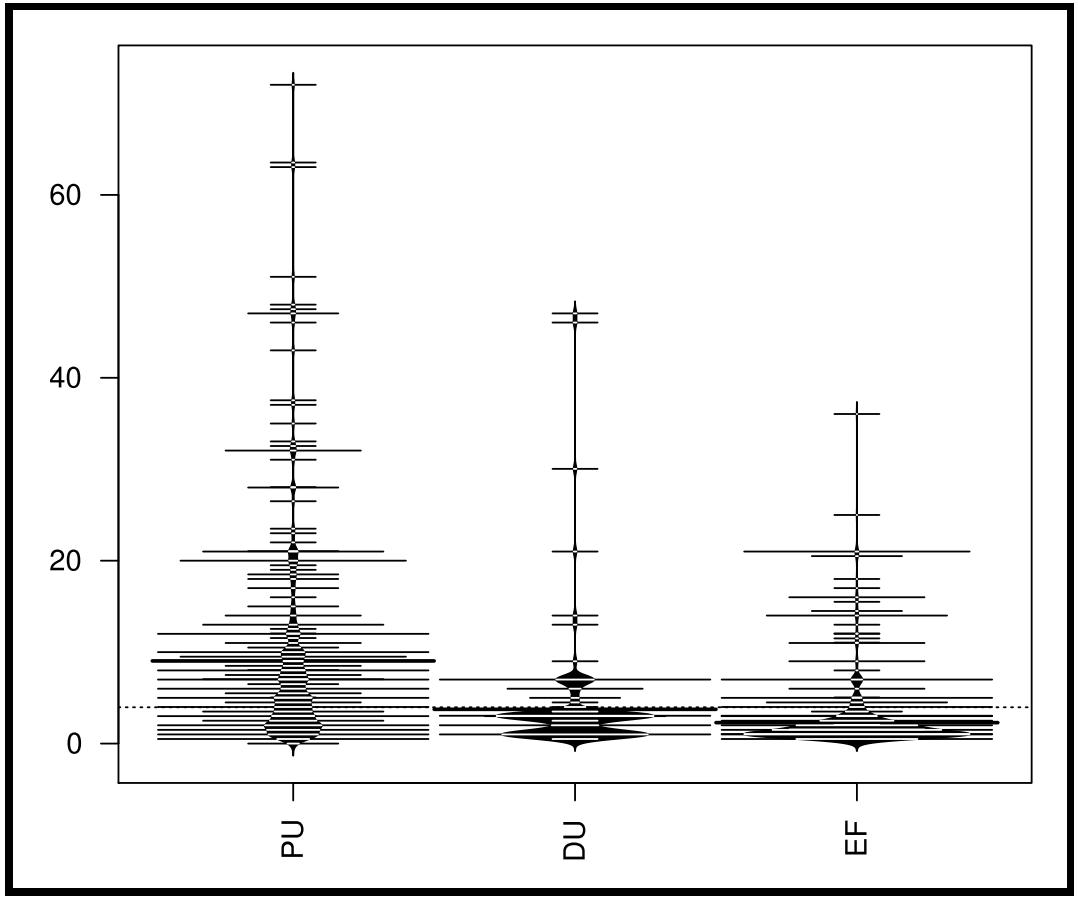


WHAT IS THE SOURCE OF ERRORS FIXED BY PACKAGE UPDATE?

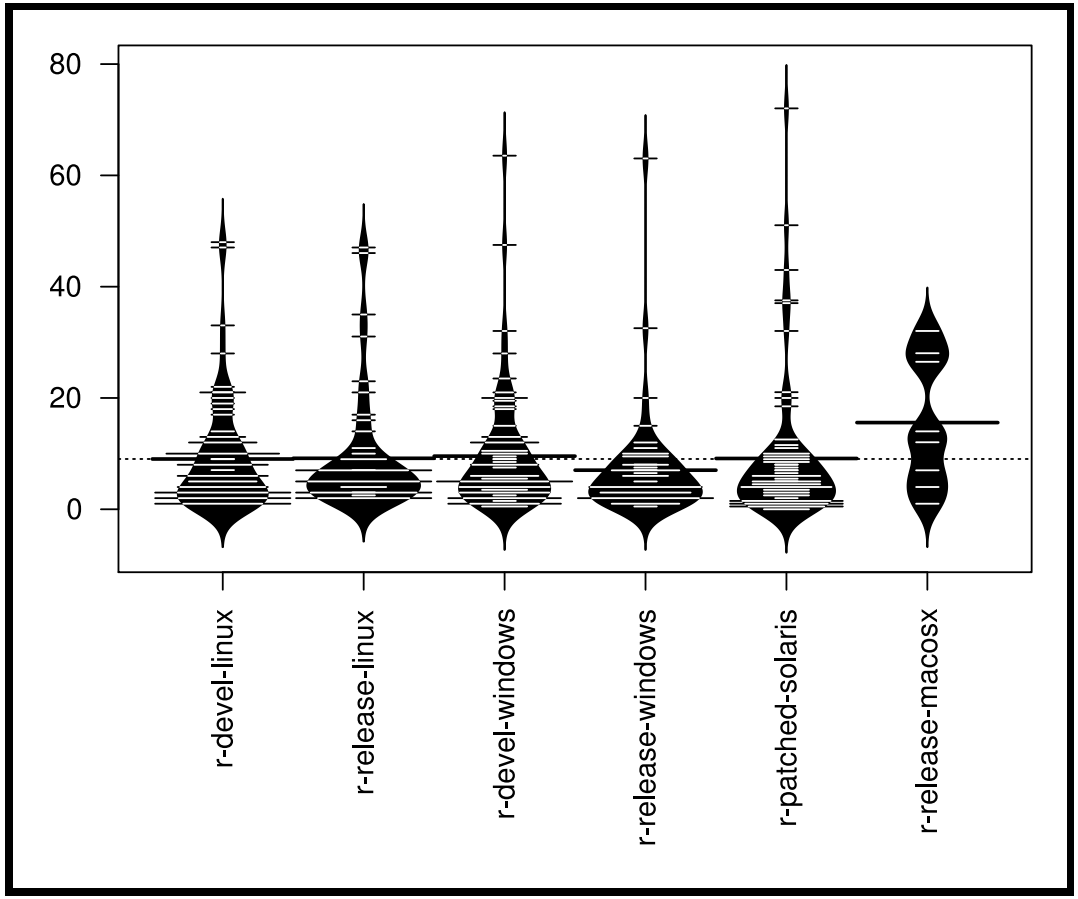


HOW LONG DOES IT TAKE TO FIX AN ERROR?

NUMBER OF DAYS NEEDED TO FIX ERRORS



NUMBER OF DAYS NEEDED TO FIX ERRORS WITH PACKAGE UPDATE



CONCLUSION

CONCLUSION

- Some flavors are more error prone than others
- High amount of errors caused by external factors
- Most errors fixed quickly without developer intervention
- Maintenance effort needs to be focused on fixing errors caused by others
- Need for a more specific tool for problems related to dependency changes

FUTURE WORK

- Refine errors by looking at package content
- Static analysis of the R code
- Function dependency graph
- Rerun some part of *R CMD check*
- Impact of the maintainers on the time required to fix packages and amount of errors for each flavor
- Do packages containing tests are more error prone?
- How does the dependency network evolve over time?

THANKS FOR YOUR ATTENTION

QUESTIONS?