

Contributing to R Core

R Forwards

(updated on 2020-07-13)

Contents

	5
1 Initiatives	7
2 Documentation	9
3 Bugzilla	11
4 More open development	13
5 Help	15
5.1 Identify specific areas where help is needed	15
6 Mentoring	17
7 Funding	19
8 Openness	21
8.1 Openness of the Process itself	21
9 Contributing	23
9.1 Initiatives to encourage CRAN contributions	23

Chapter 1

Initiatives

1.0.1 Encouraging R Core contributions

Many of these ideas are inspired by the work of other developer communities as Heather Turner discussed here: <https://youtu.be/BbpkKzz71EY?t=1045>.

Example of a note box in case we want to alert the reader of something

Chapter 2

Documentation

Dev guide like devguide.python.org

- how to find issues in bugzilla
- how to propose a patch
- how to do patch reviews
- document process of “promotion” to R core

c.f. Contributing to GNU R, Bug Reporting, R can use your help reviewing bug reports, Bugzilla mirror on GitHub, CI (Continuous Integration) for R-devel, About Mirror of R svn server with Github actions CI for testing patches

Chapter 3

Bugzilla

- Use bugzilla for general issues tracking, not just bugs? (not really sure how this is done right now). Analogous to Python issue tracker. (FYI Python is moving issue tracker to GitHub.)
- Establish intermediate roles e.g. people able to triage issues on issue tracker? Analogous to becoming a Developer on Python issue tracker
- Core devs/bugzilla veterans could offer mentoring/tutorial
- Work with Forwards, R-Ladies, etc to reach potential contributors
- Blog post by Tomas and Luke could become part of more comprehensive Developer Guide

Chapter 4

More open development

- Move to git/GitHub
- Requests for proposals on perceived deficiencies in R
 - community teams propose solutions
- R-ideas mailing list (analagous to python-ideas mailing list)
- Analagous process to Python Enhancement Proposal (PEP)?
 - process would be described in the dev guide.
- Encourage more open developer-focused meetings
 - R core sessions at useR!
 - link to developer-focused meetings e.g. RIOT
 - open attendance e.g. at DSC
- Post about any initiatives on R blog
 - promote via Forwards, R-Ladies, MiR

Chapter 5

Help

5.1 Identify specific areas where help is needed

- Help needed on recommended packages: MASS, survival
- Testing pre-releases
 - write how-to?
 - set up virtual machines for people to test on?
- Responding on R-devel/R-package-devel
 - introduce moderation?

Chapter 6

Mentoring

- Establish core-mentorship mailing list (separate from R-devel).
- Use Zulip for more interactive discussion (and modern format)?
 - potential place for R-core/experienced contributors to offer office hours
- R-core/experienced contributors to offer bookable 1-to-1 office hours (e.g. via Zoom)
- Write a mentoring guide
 - c.f. R-Ladies mentoring guide
- Run contributor tutorials
 - will need to be online
- Develop code of conduct
 - for contribution to R project?
 - for mailing lists/Zulip
- Offer GSoC projects
 - scheme already established, could be used more by R Core
 - or fund some of our own as Julia does: <https://julialang.org/jsoc/archive/>
- Similarly Google Season of Docs
 - <https://developers.google.com/season-of-docs>
- Establish mentored projects (diversity scholarships)
 - mentees invest 2-5 hours per week over 3 months

- expenses paid scholarship to useR! conference following year: is there some other incentive we can offer while useR! is online?
- Outreachy projects
 - paid internship, 40 hours per week over 3 months
 - require mentor(s) to invest 5 hours per week
- Data umbrella sprints (similar to tidyverse developer days)
 - <https://www.dataumbrella.org/open-source/sprints>

Chapter 7

Funding

- Funding to support above efforts?
 - R Foundation likely to support some
 - R Consortium + other sources to support FOSS projects?

Chapter 8

Openness

8.1 Openness of the Process itself

- Get input from community on ideas via discuss forum like rOpensci?

Chapter 9

Contributing

9.1 Initiatives to encourage CRAN contributions

- create organization on GitHub for people to put/link their repos for feedback/review/help prior to CRAN release
 - e.g. “seeking review” tag?
 - a bit like how Bioconductor does it but in a open, community distributed way?
 - or more like ropensci model? This is more in-depth review, but could borrow some ideas:
 - * match people with similar interests
 - * don’t go back to same person too often
- create community of on-rampers
 - sharing error messages from CRAN to create more user-friendly checklists, pointers to relevant parts of docs
- Zulip chat as alternative to R-package-devel to provide ad-hoc help
- Step-by-step guide for people to review their own package before submission?
 - use some of tools from rOpenSci?
- Chatbot like rOpenSci one helping with package review, to help submitters?