



PP03: Bringing devops to the pharmaverse

the PHUSE PharmaDevOps working group

Working group members (alphabetical): Ben Straub, Dinakar Kulkarni, James Black

Purpose

To help R package authors implement continuous integration/continuous delivery (CI/CD) for pharma-related packages.

Our deliverable

A website (phuse-org.github.io/devops), that helps to orientate you on why CI/CD is important, and the different types of CI/CD we expect to see on pharmaverse packages.

Why CI/CD?

CI/CD (Continuous Integration/Continuous Delivery) allows teams to automate testing, integration, and deployment processes, leading to faster, more reliable software releases with fewer manual errors and reduced risk.

Our website helps to summarise key CI/CD that can help generate more robust code and documentation, stratified into 3 tiers based on how important these tools are to your R package.

We include examples of these CI/CD tools in existing pharmaverse packages, as well as copy and paste code for how you can add them to your package.

CI/CD Tiers

Minimum: Run package checks before updating the main branch, generate documentation to ensure code stays in sync with your pkgdown website and automate code coverage checks.

Recommended: Automatically check any and all links in your documentation, to ensure broken links are identified quickly.

Enhanced: Generate a PDF that that links documented functionality (roxygen) to unit tests (testthat) of that functionality.

Check
out the
site!
Contact details
listed online



Pharma DevOps

https://phuse-org.github.io/devops/

PHARMADEVOPS

DISCUSS ABOUT PHARMADEVOPS

Welcome to the home of PharmaDevOps

This PHUSE working group is focussed on ensuring R package authors new to concepts like CI/CD can easily add these useful automations to pharmaverse R packages and related biostatistical packages.

We hope to both convince package developers of the value of CI/CD, provide a minimum bar of what is expected, and provide a set of tools to make it easy to implement.

Getting started with CI/CD

In this guidance, you'll see a number of [gh-actions](#) that you can use to implement CI/CD on your project. We have categorised these into three groups:

- Minimum CI/CD:** These are the minimum CI/CD tools we recommend for all packages, including those in the [pharmaverse](#) and [openstatsware](#). These CI/CD tools are focussed on ensuring your package works, and the documentation users see matches the latest release of your code.
- Recommended CI/CD:** These are tools we feel make your life developing and releasing robust code easier.
- Enhanced CI/CD:** These are tools are option tools for power users, or those with specific needs.

Each CI/CD action is described via a common pattern, with an emphasis on why we recommend it, and how to use it.

A recommended plan for success in implmenting CI/CD on your project is to:

- Read [this post](#) on the general anatomy of a [gh-action](#)
- Browse our recommended [gh-action](#)'s below
- Implement CI/CD on your git repo!

We've also collected together [this list of external learning resources](#).

MINIMUM CI/CD

The following CI/CD is what we consider a minimum bar for a package on the pharmaverse. These CI/CD tools are focussed on ensuring your package works, and documentation matches the code.

There is no need to use the specific code we suggest, if you are happy getting the same functionality from another source, like the ones built into [usethis](#).

R CMD check results

Duration: 0m

> checking installed package size ... NOTE

Installed size is 17.1Mb

sub-directories of 1Mb or more:

R 1.4Mb

help 1.1Mb

libs 14.2Mb

0 errors ✓ | 0 warnings ✓ | 1 note ✖

R CMD Check

TESTING

Ensure that when you release your package, all the unit tests are passing and it works in a clean environment.

AUG 23, 2023

Generate pkgdown documentation

DOCUMENTATION R

Automate creation and deployment of R package documentation

AUG 1, 2023

Check code coverage

TESTING R

This workflow measures code coverage for unit tests and reports the code coverage as a percentage of the total number of lines covered by unit tests vs. the total number of...

PHARMAVERSE AUG 1, 2023

RECOMMENDED CI/CD

The following CI/CD are tools we feel make your life developing and releasing robust code easier.

404

Check links

DOCUMENTATION LINKS

This workflow checks whether URLs embedded in code and documentation are valid. Invalid URLs result in workflow failures.

AUG 1, 2023

ENHANCED CI/CD

The following CI/CD are tools are optional tools for power users.

theValidator

TESTING DOCUMENTATION VALIDATION

This workflow will take your documentation, and link it to unit testing, creating a report that links requirements (documentation) to tests of those requirements.