CI/CD in the {pharmaverse}

Paper: OS04
Ben Straub, GSK, USA
Dinakar Kulkarni, Roche, USA

ABSTRACT

A brief abstract at the beginning summarises and highlights the major points of your paper. Check your margins! They should print so that both the left and right margins are ¾ inch.

Continuous integration (CI) and continuous delivery (CD) are playing a pivotal role in ensuring that R packages in Pharma meet the highest standards. Focus is placed on ensuring that packages are fit for purpose for both internal systems as well as the various requirements for CRAN/BioConductor. In this paper, we discuss a few best practices that were adopted into making developer-friendly and efficient CI/CD pipelines and the impact that these pipelines have had in the open-source Pharma community and at Roche/Genetech. Two case studies of package and pipelines will be discussed - one on a beginner level and one on an advanced level. The first will be CI/CD workflows for the admiral R package, used for building ADaM datasets, and presented from the perspective of a newcomer to CI/CD. The second case will be regarding the NEST framework, which will have more advanced discussions presented by an experienced CI/CD developer.

INTRODUCTION

The introduction explains the purpose and scope of your paper and provides readers with any general information they need to understand your paper.

Both large and small software projects can greatly benefit from employing CI/CD into their processes. A lone software developer maintaining a project can benefit from continually checking that their software runs under current dependent packages and operating systems. A large open-source projects can benefit from continually checking that contributors code in Pull Requests/Merges meet standards set by the project.

Below we will discuss a few scenarios where CI/CD has great benefit the R packages admiral and nest.

{admiral}

{admiral} is an open-source R package that seeks to build a modularized toolbox to develop ADaM datasets. The package was initially started by a collaboration between GSK and Roche and has since burgeoned to multiple companies and extension packages. The author, Ben Straub, was a complete newcomer to the CI/CD processes when he stepped into the role of a developer for {admiral}. His eyes were opened to the possibilities of using CI to maintain the integrity of the codebase in a large open source project.

Below we will look at two CI processes employed in admiral:

1. Continuously checking template code that build standard ADaMs
2. Managing multiple CI processes across multiple packages/repositories

ADaM Template Codes

The community of developers building {admiral} have a strong desire to help users build ADaMs in R. This is accomplished by robust documentation for admiral's function as well as User Guides. However, documentation can only go so far in demonstrating the fitness of the product. Robust code examples that can build entire ADaM datasets like ADAE or ADLB is incredibly helpful and has been developed in the package and given the name templates. Unfortunately, these templates are not checked when building the R package as they would violate CRAN policies around examples runtime(citation). Enter a custom CI checks.

The following scenario is common on {admiral}. Users have identified an issue with a function in admira A developer on admiral would like to update a function specific to BDS-Findings. The developer intiaties a Pull Request on GitHub with their updated code.

As we developer admiral functions we continously check that the template code is working properfly.

```
---
name: Check Templates

on:
  workflow_dispatch:
  pull_request_review:
    types: [submitted]

jobs:
  templates:
    name: Check Templates
    uses: pharmaverse/admiralci/.github/workflows/check-templates.yml@main
    if: github.event.review.state == 'approved'
    with:
      r-version: "4.0"
```

The above code...do da

Common CI/CD framework

admiral has grown into multiple extension packages for specific disease areas. The team has also created a CDISC specific data package as well as a developer tool package.

```
---
# Source: https://github.com/pharmaverse/admiralci
# Common workflows designed for Admiral
# but can be easily used by any other R package
name: admiral CI/CD Workflows

on:
  # 'workflow_dispatch' gives you the ability
  # to run this workflow on demand, anytime
  workflow_dispatch:
  # 'push' events are triggered when commits
  # are pushed to one of these branches
  push:
```

```
      branches:
        - main
        - devel
        ...
```

Continuation of body – after source code.

{nest}

This is a main topic in the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. If you need to include source code:

```
data one;
set two;
if mix(var1, var2) > 0 then do;
```

Continuation of body – after source code.

SUBHEAD

This is subtopic for the above. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. If you need to include source code:

```
data one;
set two;
if mix(var1, var2) > 0 then do;
```

Continuation of body – after source code.

SUBHEAD

This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. This is the paper body. If you need to include source code:

```
data one;
set two;
if mix(var1, var2) > 0 then do;
```

Continuation of body – after source code.

CONCLUSION

The conclusion summarises your paper and ties together any loose ends. You can use the conclusion to make any final points such as recommendations predictions, or judgments.

REFERENCES

- admiral
- pharmaverse
- nest

ACKNOWLEDGMENTS

- GSK and Roche
- Huge shoutout to the developers of the pharmaverse building R packages.

RECOMMENDED READING

- Further Reading
  - GitHub Actions
  - GitLab CI
- Advanced Examples
  - r-lib/actions
  - {admiralci}
  - Docker
- Presentation built with Quarto
- This paper
- Presentation related to this paper

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name: Ben Straub

Company: GSK

Address: 1000 Black Rock Rd

City / Postcode: Collegeville, PA 19426

Work Phone: (610) 917-3493

Email: ben.x.straub@gsk.com

Web: www.gsk.com

Brand and product names are trademarks of their respective companies.