The Journal of Open Source Software

Managing Larger Data on a GitHub Repository

Carl Boettiger[1]

1 University of California, Berkeley

DOI: 10.21105/joss.00970

Software

- Review
- Repository
- Archive

Submitted: 21 September 2018
Published: 22 September 2018

Summary

Because larger (> 50 MB) data files cannot easily be committed to git, a different approach is required to manage data associated with an analysis in a GitHub repository. This package provides a simple work-around by allowing larger (up to 2 GB) data files to piggyback on a repository as assets attached to individual GitHub releases. `piggyback` provides a workflow similar to Git LFS ("Git lfs," 2018), in which data files can be tracked by type and pushed and pulled to GitHub with dedicated commands. These files are not handled by git in any way, but instead are uploaded, downloaded, or edited directly by calls through the GitHub API ("GitHub api version 3," 2018). These data files can be versioned manually by creating different releases. This approach works equally well with public or private repositories. Data can be uploaded and downloaded programmatically from scripts. No authentication is required to download data from public repositories.

References

Git lfs. (2018). https://git-lfs.github.com/. Retrieved from https://git-lfs.github.com/

GitHub api version 3. (2018). https://developer.github.com/v3/. Retrieved from https://developer.github.com/v3/