Containershare: Open Source Registry to build, test, deploy with CircleCI

Vanessa Sochat[1]

1 Stanford University Research Computing

Summary

Containershare is an open source library of containers, both providing itself as a template, a library, and production application programming interface (API) for interested users. Specifically, it is a complete metadata registry that can be freely deployed directly from a Github repository to validate and serve tested, tagged, and version controlled containers, each maintained from independent Github repositories. The registry uses several open source and free to use solutions to accomplish this, and brings them together programatically with steps that are easy for the user to set up. Specifically, the user must connect the repository to the continuous integration service CircleCI ("Continuous integration and deployment," n.d.) and then turn on Github Pages ("GitHub pages," n.d.) from the repository web interface. After these steps, adding text files to describe other container repositories via pull requests will test the submissions, and programatically add them to all components of the library.

While several frameworks exist with a primary goal to provide a listing of container builds (e.g., biocontainers (Veiga Leprevost et al., 2017)) containershare is unique in its requirement for each container to serve more substantial metadata. For example, while biocontainers also takes advantage of services like Docker Hub to provision a build, it doesn't go out of its way to provide any more metadata than a standard image manifest. Each repository is required to serve a listing of tags, image inspections (files with sizes and package versions) and image manifests via Github pages, and also a LICENSE and README file from the master branch. Further, there is no additional software or server dependency on serving the API, because the entire operation happens via Github Pages and continuous integration. The manifests themselves and the container listings are served using a distributed strategy so that the primary containershare repository is not eventually overwhelmed with growing size, and a listing of tags are provided to discover container metadata for current and past builds, each associated with a Github commit. While biocontainers shines and succeeds to provide a listing of bioinformatics containers, it seems to require an additional registry backend and setup. In comparison, containershare is a more lightweight solution intended to empower other users to deploy similar initatives branded for their containers. Whereas the containershare Github repository works as an

Figure 1: https://vsoch.github.io/assets/images/posts/containershare/table.png

Figure 2: https://vsoch.github.io/assets/images/posts/containershare/share.png

API, biocontainers does not appear to have any metadata served from the respository directly, container versions are not obviously linked to Github commits, and previous tags are not directly accessible. Further, containershare takes a distributed strategy of serving metadata from each included repository, allowing for individual containers to have unique and distinct tests, if wanted.

Container Repository Templates

The individual container repository is responsible for serving its own metadata that the containershare can discover, validate, and provide for the user. It is also driven by a simple definition of a general CircleCI workflow, including building of a container, extraction of metadata, and then deployment of the container to Docker Hub and metadata files and user interface back to Github Pages. The workflow definitions to accomplish this are provided by containershare, and include:

- A table of version controlled containers, where each commit is associated with a tagged container for the user to interact with

- A complete list of tags for a program to discover and consistently query for the associated container metadata
- The traditional image manifests that are provided by Docker Hub

Figure 3: https://vsoch.github.io/assets/images/posts/containershare/inspect.png

- An "inspection" of the container that includes package manager packages and versions, along with a listing of files and sizes inside of the container, extracted during the build step using the open source Container Diff ("Container-diff," n.d.) tool provided by Google Open Source.

Containershare serves as a general skeleton that can be extended to several use cases and themes, including sharing of containers for behavioral paradigms, open source publications, or components to scientific workflows. The use of open source tools like Github and CircleCI means that the implementation is completely transparent and customizable for an individual or institutional needs. For a research scientist, submission of a container repository to a containershare gives confidence that the container can be discovered. For a service provider, deployment of a container share (and subsequent provision of containers using it) gives confidence that the service users have a central location to discover containers, and API to discover them programmatically. Containershare and links to container repository templates are provided for use from the containershare repository, contributions in the way of code or issues are encouraged, and for interested readers, a more descriptive writeup is available.

References

Container-diff. (n.d.). Github.

Continuous integration and deployment. (n.d.). https://circleci.com/dashboard.

GitHub pages. (n.d.). Github.

Veiga Leprevost, F. da, Grüning, B. A., Alves Aflitos, S., Röst, H. L., Uszkoreit, J., Barsnes, H., Vaudel, M., et al. (2017). BioContainers: An open-source and community-driven framework for software standardization. *Bioinformatics*, *33*(16), 2580–2582.