

Singularity Registry: Open Source Registry for Singularity Images

Vanessa Sochat¹

DOI: [10.21105/joss.00426](https://doi.org/10.21105/joss.00426)

¹ Stanford University Research Computing

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Licence

Authors of JOSS papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Summary

Singularity Registry is a non-centralized free and Open Source infrastructure to facilitate management and sharing of institutional or personal containers.

A container is the encapsulation of an entire computational environment that can be run consistently if the platform supports it. It is an aid in reproducibility (Moreews et al. 2015, Belmann et al. (2015), Boettiger (2014), Santana-Perez and Pérez-Hernández (2015), Wandell et al. (2015)) because different researchers can run the exact same software stack on different underlying (Linux Intel) systems. Docker (Merkel 2014) has become popular as a general container system because it allows software to be bundled with administrator privileges, however, it poses great security risks if installed on a multi-user, shared computational resource, and thus Docker has not had admission to these high performance computing (HPC) environments. Singularity (Kurtzer, Sochat, and Bauer 2017) offers similar features to Docker for software deployment but does not allow the user to escalate to root, and so it has been easy to accept in HPC environments. Since its introduction, Singularity has been deployed in over 50 HPC resources across the globe.

Sharing of Containers for Reproducible Science

Essential to the success of Singularity is not just creation of images, but sharing of them. To address this need, a free cloud service, Singularity Hub (Sochat, n.d.), was developed to build and share containers for scientists simply by way of building containers from a specification file in a version controlled Github repository. This setup is ideal given a small number of containers, each belonging in one Github repository, but is not optimal for institutions that wanted to build at scale using custom strategies.

Singularity Registry (sregistry) (Sochat, n.d.) was developed to empower an institution or individual to build at scale, and push images that can be private or public to their own hosted registry. It is the first user and institution installable, non-centralized Open Source infrastructure to facilitate the sharing of containers. While Singularity Hub works with cloud builders and object storage, Singularity Registry is optimized for storage on a local filesystem and any choice of builder (e.g., continuous integration (Travis, Circle), cluster or private node, or separate server). A Registry is also customized with a center's name, and links to appropriate help contacts.

The Registry, along with native integration into the Singularity software, includes several tools for organization, analysis, and logging of image metrics and usage. Administrators can control the ability for users or the larger community to create accounts, and give finely tuned access (e.g., an expiring token) to share containers. An application programming interface (API) exposes metadata such as container sizes, versioning, and build times. Every time a container is used, or starred by a user, the Registry keeps a record. This

discovery in science.

References

- Belmann, Peter, Johannes Dröge, Andreas Bremges, Alice C McHardy, Alexander Sczyrba, and Michael D Barton. 2015. “Bioboxes: Standardised Containers for Interchangeable Bioinformatics Software.” *Gigascience* 4 (October). [giga-science.biomedcentral.com: 47. doi:10.1186/s13742-015-0087-0](https://doi.org/10.1186/s13742-015-0087-0).
- Boettiger, Carl. 2014. “An Introduction to Docker for Reproducible Research, with Examples from the R Environment,” October. [doi:10.1145/2723872.2723882](https://doi.org/10.1145/2723872.2723882).
- Kurtzer, Gregory M, Vanessa Sochat, and Michael W Bauer. 2017. “Singularity: Scientific Containers for Mobility of Compute.” *PLoS One* 12 (5). Public Library of Science: e0177459. [doi:journal.pone.0177459](https://doi.org/10.1371/journal.pone.0177459).
- Merkel, Dirk. 2014. “Docker: Lightweight Linux Containers for Consistent Development and Deployment.” *Linux J*. Houston, TX: Belltown Media.
- Moreews, François, Olivier Sallou, Hervé Ménager, Yvan Le Bras, Cyril Monjeaud, Christophe Blanchet, and Olivier Collin. 2015. “BioShaDock: A Community Driven Bioinformatics Shared Docker-Based Tools Registry.” *F1000Res*. 4 (December). [ncbi.nlm.nih.gov: 1443. doi:10.12688/f1000research.7536.1](https://doi.org/10.12688/f1000research.7536.1).
- Santana-Perez, Idafen, and María S Pérez-Hernández. 2015. “Towards Reproducibility in Scientific Workflows: An Infrastructure-Based Approach.” *Sci. Program*. 2015 (February). Hindawi Publishing Corporation. [doi:http://dx.doi.org/10.1155/2015/243180](https://doi.org/10.1155/2015/243180).
- Sochat, Vanessa. n.d. *Singularity Registry Documentation*. <https://singularityhub.github.io/sregistry/>.
- . n.d. *Singularity Registry Github*. <https://github.com/singularityhub/sregistry>.
- . n.d. *Singularity-Hub*. <https://singularity-hub.org/>.
- Wandell, B A, A Rokem, L M Perry, G Schaefer, and R F Dougherty. 2015. “Data Management to Support Reproducible Research,” February.