Storage Media Federation for Galaxy

Vahid Jalili¹, Enis Afgan², Nuwan Goonasekera³, Dannon Baker², Jeremy Goecks¹
¹Oregon Health and Science University, Portland, OR, USA, jalili@ohsu.edu
²Johns Hopkins University, Baltimore, MD, USA
³University of Melbourne, Melbourne, Australia.

Project Website: https://usegalaxy.org
Source Code: https://github.com/galaxyproject/galaxy
License: Academic Free License version 3.0

Some public Galaxy servers are bound by available resources, including data storage limits applied on a per-user basis. With growing dataset sizes and number of biological samples, limited available resources can restrict researchers’ capacity to do large-scale research using these public servers. We are extending Galaxy to support the ability of users to plug-in a variety of storage resources in a federated fashion. This federated vision of storage resources integrates physically disconnected persistence resources from a Galaxy instance.

We are enhancing Galaxy to support two models of federated storage resources: instance-wide and user-based. The instance-wide configuration allows all the data for a Galaxy installation to be stored on a cloud computing platform (e.g., Amazon, Google) or other remote storage resources while Galaxy automatically manages data storage, caching, and persistence. The user-based option will allow each individual user of a Galaxy instance to associate external, additional storage resources with their account (e.g., Amazon S3). This storage media will be made available in addition to any storage already provided by the public server and the user will have an option to define rules (e.g., priority and quota) that determine what data gets stored on what remote resource. For example, a researcher can associate an AWS S3 bucket and an Azure Blob container with their account with a rule specifying that up to 200GB of data is stored in S3 while the rest is in Azure, or associate individual histories with a specific storage resource. Galaxy will then automatically support the specified distribution topology. This allows a user to aggregate resource from multiple sources while respecting resource limits.

Among other applications, the proposed federated storage can be advantageous in joint data analysis scenarios. For instance, a Galaxy user can execute an analysis pipeline using jointly private data hosted on Amazon S3 and a collaborator’s data persisted on Microsoft Azure Blob, while Galaxy handles the details of moving data for processing and persisting results.

Toward this end, we have extended Galaxy by making it an OpenID Connect relying party, so that users are not required to share their credentials with Galaxy, and can revoke or restrict privileges of a Galaxy instance at anytime. This is a significant advance that enables Galaxy users to log in using their external accounts (e.g., Google or home institution) and use these external identities to connect Galaxy to cloud computing resources. We have also extended the Galaxy ObjectStore to run on a per-user basis and partially support the above-described usage model. We have integrated the Galaxy ObjectStore with CloudBridge, which enables Galaxy to read/write from multiple cloud-based storage resources, including Amazon, Azure, Google, and OpenStack. At present, we are implementing methods to compute a checksum for each dataset, so that data integrity is guaranteed when transferring to and from attached storage. The talk will demonstrate the ability for Galaxy to store datasets on multiple cloud storage providers with minimal configuration.