Storage Media Federation for Galaxy
How to distribute data on user-owned cloud-based resources, serving two goals:

[essentially] unlimited storage
joint data analysis
Storage resources configuration:

Where to store data?

[Advanced] How to distribute data?
Configure Galaxy to distribute data on multiple persistence media

```xml
<?xml version="1.0"?>
<object_store type="hierarchical">
    <backends>
        <object_store type="disk" id="primary" order="0">
            <files_dir path="..."/>
        </object_store>
        <object_store type="s3" id="secondary" order="1">
            <auth access_key="..." secret_key="..."/>
            <bucket name="..."/>
        </object_store>
        <object_store type="azure_blob" id="tertiary" order="2">
            <auth account_name="..." account_key="..."/>
            <container name="..."/>
        </object_store>
    </backends>
</object_store>
```
Persistence media setup is **transparent** to an end-user.
Persistence media setup is transparent to an end-user.
Two challenges with this model that you’ll face … sooner or later … guaranteed!!

1. Genomical data is competing with astronomical data for the biggest big data problem of mankind title … and … genomics is performing promisingly!!

Two challenges with this model that you’ll face ... sooner or later ... guaranteed!!

1. Genomical data is competing with astronomical data for the *biggest big data problem of mankind* title ... and ... genomics is performing promisingly!!


1. Joint data analysis is difficult with data scattered on disconnected storages.
Aha! Solution!

Federated Storage Resources!

Step #1: Integrate with a user-owned cloud-based storage
A Galaxy Instance

Storage resources

User 1

- 40%
- User-owned cloud-based storage

Upload data from a Galaxy history to cloud

Download data from cloud to a Galaxy history
Back-end from 10km

CloudBridge

Galaxy

API

Upload

Download

Payload

History ID, Provider, Bucket, Credentials, Dataset IDs

Payload

History ID, Provider, Bucket, Credentials, Object
1. Validate payload
2. Establish a connection to the specified provider
3. Cache the object
4. Persist the object
5. Create a dataset for the download object
6. Add the dataset to the history
7. Delete cached object

CloudBridge

Azure BLOB
AWS S3
OpenStack Swift

Galaxy

API

Download

Payload
History ID, Provider, Bucket, Credentials, Object

The info of the created dataset in JSON
Back-end from 5km

Validate payload

Establish a connection to the specified provider

Any dataset IDs given?

Yes

Upload the specified datasets

No

Upload all the datasets in the specified history

Payload

History ID, Provider, Bucket, Credentials, Dataset IDs

CloudBridge

Azure BLOB

AWS S3

OpenStack Swift

Galaxy

API

Upload

A message of successful upload
Do NOT share your credentials!

&

We will NOT ask for your credentials!

Payload

History ID, Provider, Credentials, Bucket, Dataset IDs

Payload

History ID, Provider, Bucket, Credentials, Object
At the moment, all features are accessible via APIs, and UI is under development.

- Merged PR #4474
- Open PR #5835
- Open PR #5903
- & more to be PRed
- Open PR #6078
Conclusion

Feature:
Upload and Download your Galaxy datasets to and from cloud-based storages without sharing your credentials.

Bonus:
Conclusion

Applications:
- Theoretically unlimited storage.
- Simplified joint data analysis.
- Simplified data sharing across different Galaxy instances and third-party applications.
Future work

Step #2: Plug-your-own-media (User-Based ObjectStore)

[WIP] Open PR #4840
Future work

Step #2: Plug-your-own-media (User-Based ObjectStore)

Corresponds to a dataset in DB

User-owned cloud-based storage

Uploaded

Download

Online

S3

A Galaxy Instance

Storage resources

DB

Upload

[Blob]
Thanks

Vahid Jalili
Enis Afgan
Nuwan Goonasekera
Dannon Baker
Jeremy Goecks
The “Core” Galaxy team and the community

Supported by the NHGRI (HG005542, HG004909, HG005133, HG006620), NSF (DBI-0850103, DBI-1661497),
Penn State University, Johns Hopkins University, Oregon Health and Science University, and the
Pennsylvania Department of Public Health