Automating Instrument Data at Scale

Rachana Ananthakrishnan
Vas Vasiliadis
Data processing pipeline pattern

At the minimum, make data available to collaborators

Set permissions for access

Processing and storage

Local policy

Commercial cloud

Entreprise storage

Research computing/HPC

Research lab

Instrument facility

Bioinformatics core

Data source

Access by collaborators
Data processing pipeline pattern

**Data source**
- Instrument Facility
- Research Lab
- Bioinformatics Core

**Processing and storage**
- Enterprise Storage
- Commercial Cloud
- Research Computing/HPC

**Curation/Approvals**
- Metadata extraction
- Persistent Identifiers
- Set access and publish for discovery

**Access by collaborators**
Requirements for such pipeline

- Reliable, near-real time data access
- Uniformed policy for data access, based on local policy
- Delegation of data access management to PI
- Ability to compute on data across storage classes
- Apply best practices with data processing pipeline
- Support for data organization to facilitate FAIR data
- ...

Automation to scale
How do we use the Globus platform for full automation?
1. Preparing acquisition machine for data access
2. Configuring data collections for automated data transfer
3. Credentials for automation
4. Create and deploy flow for automation
5. Create trigger scripts
1. Preparing acquisition machine for data access
Install Globus Connect

• For Linux machines, install and use Globus Connect Server

• Acquisition machines often are Windows machine → Globus Connect Personal deployment
  – Installed as a local user account
  – PI login vs GCP always running on a shared account
  – Outbound connections only
2. Configuring data collections for automated data transfer
Using guest collections at instrument facilities

• Create guest collections on all storage systems for automation
  – Acquisition machine, storage mounted on compute etc.

• Creation of guest collection cannot generally be automated *

• Permission and role management on guest collections can be fully automated
Pattern 1: Guest collection use

• Create a guest collection at top level directory
  – This is done by a user who has local account

• For each experiment/project/modality
  – Create a folder
  – Set permissions PI/collaborators to read data from the folder

• Can automate permission management by using local policy store
Pattern 2: Guest collection use

- Create a guest collection for each experiment/project/modality
  - Grant PI role to manage access to the guest collection (Access Manager role)
  - Set permissions collaborators to read data from the folder

- Can automate role and permission management by using local policy store
3. Credentials for automation
Managing service accounts/app credentials

• Application Identity:
  appclientid@clients.auth.globus.org

• These are confidential apps with client id and secret

• Ensure application is on a secure device

• Set up policy for rotation of secret

• Assign project admins to manage the registration
Registering a service account

- **Webapp - Settings**
  - app.globus.org/settings/developers

Register an...

**App**

Register a service account or application credential for automation

Applications that authenticate and act as the application itself. These applications are used for automation and as service or community accounts, and do NOT act on behalf of other users.
Get app credentials at
app.globus.org/settings/developers
Get app credentials at app.globus.org/settings/developers
Set permission for the service account

- Use the web app to create guest collection on guest collection you created
Set permission for the service account

Assigned Roles

- Rachana Ananthakrishnan
  - ranantha@uchicago.edu
  - Administrator

- [Assign New Role]

Assign To

- Techex Test (6afa2dc5-d219-4078-9a06-ba37a32c739@clients.auth.globus.org)

Role

- Administrator
  - modify endpoint definition, delete the endpoint, manage roles, perform file system operations and transfers, and all capabilities of the Access Manager and Activity Manager roles

- Access Manager
  - view, add, and delete all access rules on the endpoint; implicitly gives read/write access to the root of the endpoint

- Activity Manager
  - view and control tasks and other endpoint activity

- Activity Monitor
  - view tasks and other activity to or from the endpoint

[Add Role] [Cancel]
4. Create and deploy flow for automation
Create a flow for your use case

- Start with flow definitions that are published:
  - github.com/globus/globus-flows-trigger-examples
  - docs.globus.org/api/flows/authoring-flows/examples/

- Manage flow definitions in a version controlled system

- Validation tools
  - Flows IDE: https://globus.github.io/flows-ide/
  - Globus CLI: globus flows validate
Flow definition

```
"StartAt": "TransferFiles",
"States": {
  "TransferFiles": {
    "Comment": "Transfer to a guest collection",
    "Type": "Action",
    "ActionUrl": "https://actions.automate.globus.org/transfer/transfer",
    "Parameters": {
      "source_endpoint_id.$": "$.'input.source.id",
      "destination_endpoint_id.$": "$.'input.destination.id",
      "transfer_items": [
        {
          "source_path.$": "$.'input.source.path",
          "destination_path.$": "$.'input.destination.path",
          "recursive.$": "$.'input.recursive_tx"
        }
      ]
    },
    "ResultPath": "$.'TransferFiles",
    "WaitTime": 60,
    "Next": "SetPermission"
  },
  "SetPermission": {
      ....
      "End": True
  }
}
```
"DetermineOutcome": {
  "Type": "Choice",
  "Choices": [
    {
      "Variable": "$.curator_decision",
      "StringEquals": "approve",
      "Next": "SetPermissionForAccess"
    },
    {
      "Variable": "$.curator_decision",
      "StringEquals": "approveEmbargo",
      "Next": "SetPermissionForEmbargo"
    }
  ],
  "Default": "SubmissionRejected"
},
Run context

- Flow and user properties as read-only values
- Available in $.context
  - Flow id
  - Run id
  - All identities of the user invoking the flow
  - Email id of the user invoking the flow
  - Token information (issued time, and expiration information)
Permissions to run the flow

Assign To
- User
- Group
- All Logged In Users
- Public (anonymous users)

Role
- Administered By
  - can start this flow, view this flow and associated activity, and modify this flow
- Runnable By
  - can start this flow and view associated activity
- Visible To
  - can view this flow and associated activity

Set permission for the service account to run the flow
Triggering flows on instruments ...and other resources
A simple, and very common, use case

1. Transfer raw instrument images
2. Set permissions for accessing the data

Actions
Creating and triggering runs of this flow

1. Log into the Globus CLI
2. Create (deploy) the transfer-and-share flow
3. Edit the monitor (trigger) script
4. Ensure GCP is running on the instrument
5. Run the monitor script
6. Trigger the flow

github.com/globus/globus-flows-trigger-examples
Create the flow

- Success returns the flow ID
- Inspect the flow using the web app

```bash
$ globus login
$ cd ~/globus-flows-trigger-examples/transfer_share
$ globus flows create FLOW_NAME \\> definition.json --input-schema schema.json
$ ~/globusconnectpersonal-3.2.5/globusconnectpersonal -start &
```
Run the monitor script and trigger the flow

```bash
$ source ~/.trigger/bin/activate
$ cd ~/globus-flows-trigger-examples
$ ./trigger_transfer_share_flow.py \
  > --watchdir /home/dev3/images \
  > --patterns .done
$ cp ~/test-data ~/images
$ touch ~/images/i.am.done
```

- Directory to monitor for file creation
- Flow is triggered when a new filename matches this expression
- Simulate instrument data creation
- Trigger the flow
Adding computation to our instrument flow

1. Transfer raw instrument images
2. Run a compute job to process raw image files
3. Move processed images to repository
4. Set permissions for accessing the data
Our instrument research environment

Sharing Repository (ALCF Eagle)

HPC System (UChicago RCC Midway3)

Compute Endpoint

Registered Compute Function

Storage Endpoint

“The Instrument” (just a VM in the cloud :)

Transfer Endpoint (using GCP)

Monitor script

Trigger flow run

1. Transfer raw files
2. Invoke image processing function
3. Transfer result files
4. Set permissions

Access result files
Key consideration: Be identity aware

• What identity is the flow running as?

• Does identity have access to target resources?
  – Collections (ideally, guest collections)
  – Compute endpoint
  – Compute function

• Does identity have the required role?
  – Access Manager, if granting/revoking permissions
Making Data Findable with Globus Search
Data description and discovery

- Metadata store with fine-grained visibility controls
- Schema agnostic dynamic schemas
- Simple search using URL query parameters
- Complex search using search request document

[docs.globus.org/api/search]
Data ingest with Globus Search

POST /index/{index_id}/ingest'

```json
{
    "ingest_type": "GMetaList",
    "ingest_data": {
        "gmeta": [
            {
                "id": "filetype",
                "subject": "https://search.api.globus.org/abc.txt",
                "visible_to": ["public"],
                "content": {
                    "metadata-schema/file#type": "file"
                }
            },
            ...
        ]
    }
}
```

- Bulk create and update
- Task model for ingest at scale
Data ingest with Globus Search

POST /index/{index_id}/ingest'

```
{
    "ingest_type": "GMetaList",
    "ingest_data": {
        "gmeta": [
            {
                "id": "weight",
                "subject": "https://search.api.globus.org/abc.txt",
                "visible_to": ["urn:globus:auth:identity:46bd0f56-e24f-11e5-a510-131bef46955c"],
                "content": {
                    "metadata-schema/file#size": "37.6",
                    "metadata-schema/file#size_human": "<50lb"
                }
            }
        ]
    }
}
```

Visibility limited to Globus Auth identity
- Single user
- Globus Group
- Registered client application
Data discovery with Globus Search

GET /index/{index_id}/search?q=type%3Ahdf5

```
{
  "@datatype": "GSearchResult",
  "@version": "2017-09-01",
  "count": 1,
  "gmeta": [
    {
      "@datatype": "GMetaResult",
      "@version": "2019-08-27",
      "entries": [
        ...
      ],
      "subject": "https://..."
    }
  ],
  "offset": 0,
  "total": 1
}
```
Data discovery with Globus Search

POST /index/{index_id}/search

```
{
    "filters": [
        {
            "type": "range",
            "field_name": "pubdate",
            "values": [
                {
                    "from": "*",
                    "to": "2020-12-31"
                }
            ]
        }
    ],
    "facets": [
        {
            "name": "Publication Date",
            "field_name": "pubdate",
            ...
        }
    ]
}
```
Making our instrument data FAIR

1. Transfer raw instrument images
2. Run a compute job to process raw image files
3. Move processed images to repository

- Transfer
- Compute
- Transfer

- Share
- Ingest protected metadata, searchable by one group
- Set permissions for accessing the data

- Search
- Ingest open metadata, searchable by all
End-to-end automation in practice: XPCS

Data capture

Globus Flows

FAIR data, ready for discovery!

Transfer
- Transfer HDF5 files

Transfer
- Transfer IMM

Compute
- Run Corr

Search
- Ingest to index

Share
- Set access controls

Transfer
- Move results to repo

Compute
- Gather metadata

Compute
- Plot results
Take a look…
acdc.alcf.anl.gov
Extending the ecosystem: Action Providers

• Action Provider is a service endpoint
  – Run
  – Status
  – Cancel
  – Release
  – Resume

  docs.globus.org/api/flows/hosted-action-providers

• Action Provider Toolkit
  action-provider-tools.readthedocs.io

Custom developed