New directions in Globus: Collections, responsive storage, and safe data

Ian Foster
The University of Chicago and Argonne National Laboratory
Breaking down walls to yuge data sharing and analysis

Ian Foster
The University of Chicago and Argonne National Laboratory
Thesis: We enhance data sharing and analysis by eliminating barriers to navigation and flow
Notable barriers to data flow and navigation

• Moving data rapidly, securely, and reliably from lab to lab
• Accessing data at remote locations
• Controlling who can access data
• Tracking what data is where
• Discovering available data within a rapidly growing haystack
• Computing at large scale, including on distributed data
• Complying with rules on sensitive human data
• Data lifecycle for large and distributed data
Cloud: Outsourcing and automation

**Software as a service: SaaS**
(web & mobile apps)

**Platform as a service: PaaS**

**Infrastructure as a service: IaaS**
Cloud: Outsourcing and automation

**SaaS for science**

**Software as a service: SaaS**
(web & mobile apps)

**Platform as a service: PaaS**

**Infrastructure as a service: IaaS**
Research data management simplified.

195,511,375,344 MB TRANSFERRED

Researchers
Focus on your research, not IT problems. We make it easy to move, manage, and share big data.

Resource Providers
Globus gives you more control over your data infrastructure, while providing excellent ease-of-use for your researchers.

Our Users
Researchers and resource providers are our greatest inspiration and we love it when they say nice things about Globus.
Researcher initiates transfer request; or requested automatically by script, science gateway.

1. Transfer

Sequencing center

Globus transfers files reliably, securely

2. Share

Researcher selects files to share, selects user or group, and sets access permissions

Personal Computer

3. Publish

Researcher assembles data set; attaches metadata (Dublin core, domain-specific)

4. Discover

Peers, collaborators search and discover datasets; transfer and share using Globus

5. Compute facility

Globus controls access to shared files on existing storage; no need to move files to cloud storage!

6. Curator reviews and approves; data set published on campus or other system

Globus transfers files reliably, securely

7. Collaborator logs in to access shared files; no local account needed; download via Globus

Globus controls access to shared files on existing storage; no need to move files to cloud storage!

8. Publication repository

Peers, collaborators search and discover datasets; transfer and share using Globus

Personal Computer

Only Web browser required
• Use any storage system
• Access using any credential
How Globus adds value...

• Ease of use, consistent user interface across systems
• “Fire-and-forget” reliable file transfer
• Low-overhead external collaboration
• Secure access, multi-tier security model
• Maximized wide area network throughput
• Rapid deployment via standard packages
• Highly automatable: CLI, RESTful API
Globus has the best numbers!

- 4 major services
- 190 PB transferred
- 25 billion files processed
- 50,000 registered users
- 13 national labs
- 10,000 active endpoints
- 10,000 active users
- 99.9% uptime
- 35+ institutional subscribers
- 1 PB largest single transfer to date
- 3 months longest continuously managed transfer
- 130 federated campus identities
I just completed a 7680 files each of size ~57.7GB from ALCF to NCSA at a rate of 92.4 Gbits/s with no-verify-checksum on Globus. This rate is a little shy of the 1PB/day goal (at this rate it will take 1 day and 3 minutes).
Cloud: Outsourcing and automation

Software as a service: SaaS
(web & mobile apps)

Platform as a service: PaaS

Infrastructure as a service: IaaS
Transfer API Documentation

This API provides a REST-style interface to the Globus reliable file transfer service. The Transfer API supports monitoring the progress of a user's file transfer tasks, managing file transfer endpoints, listing remote directories, and submitting new transfer and delete tasks. The API is ideal for integration into Portals or Gateways to provide complex reliable file transfer capabilities without having to develop and support these features on your own. It is also easy to use for scripting, using any standard HTTPS or REST client library in scripting languages like Python and Ruby.

Contents

- **API Overview** - overview of API with authentication instructions and examples
- **Endpoint Activation** - associate user credentials with an endpoint
- **Task Submission** - submit transfer and delete tasks
- **Task Management** - monitor and cancel background transfer and delete tasks
- **File Operations** - foreground filesystem operations, including directory listing (ls), creating directories (mkdir), and renaming files (rename)
- **Endpoint Management** - create, update, and delete endpoint definitions and servers
- **Task Search** - find a task by matching on a set of criteria or display a summary of the task's field data
Globus SDK for Python (Beta)

This SDK provides a convenient Pythonic interface to Globus REST APIs, including the Transfer API and the Globus Auth API. Documentation for the REST APIs is available at https://docs.globus.org.

Two interfaces are provided - a low level interface, supporting only GET, PUT, POST, and DELETE operations, and a high level interface providing helper methods for common API resources.

Source code is available at https://github.com/globus/globus-sdk-python.

Installation

The Globus SDK requires Python 2.6+ or 3.2+. If a supported version of Python is not already installed on your system, see this Python installation guide.

The simplest way to install the Globus SDK is using the pip package manager (https://pypi.python.org/pypi/pip), which is included in most Python installations:

   pip install globus-sdk

This will install the Globus SDK and it's dependencies.

Bleeding edge versions of the Globus SDK can be installed by checking out the git repository and installing it manually:

   git checkout https://github.com/globus/globus-sdk-python.git
   cd globus-sdk-python
   python26.5 setup.py install

Basic Usage
Prototypical research data portal

Move portal storage into Science DMZ, with Globus endpoint

Leave portal web server behind firewall

Globus handles security and data heavy lifting
Sanger Imputation Service

This is a free genotype imputation and phasing service provided by the Wellcome Trust Sanger Institute. You can upload GWAS data in VCF or 23andMe format and receive imputed and phased genomes back. Click here to learn more and follow us on Twitter.

Before you start

Be sure to read through the instructions.

You will need to set up a free account with Globus and have Globus Connect running at your institute or on your computer to transfer files to and from the service.

Ready to start?

If you are ready to upload your data, please fill in the details below to register an imputation and/or phasing job. If you need more information, see the about page.

Full name

Organisation

Email address

What is this?

Globus user identity

Next

News

11/05/2016
Thanks to EAGLE, we can now return phased data. The HRC panel has been updated to r1.1 to fix a known issue. See ChangeLog for more details.

15/02/2016
Globus API changed, please see updated instructions.

17/12/2015
New status page and reworked internals. See ChangeLog.

09/11/2015
Pipeline updated to add some features requested by users. See ChangeLog.
Integrate Globus for data downloads

Shared endpoint with subfolder per request

Single sign on via streamlined account provisioning
DMagic is an open-sourced Python toolbox to perform data management and data sharing for users of the Imaging Group of the Advanced Photon Source.

This guide is maintained on GitHub.

- About DMagic
- Install directions
- Development
- API reference
- Examples
- Frequently asked questions
RDP admin

RDP endpoint

/~/

...
# (1) Create directory to be shared

share_path = '~/' + shared_dir + '/'

tc.operation_mkdir(host_id, path=share_path)
# (1) Create directory to be shared
share_path = '~/' + shared_dir + '/'
tc.operation_mkdir(host_id, path=share_path)

# (2) Create the shared endpoint on that directory
shared_ep_data = {
    'DATA_TYPE': 'shared_endpoint',
    'host_endpoint': host_id,
    'host_path': share_path,
    'display_name': 'RDP ' + shared_dir,
    'description': 'RDP shared endpoint'
}
r = tc.create_shared_endpoint(shared_ep_data)
share_id = r['id']
# (3) Copy data into the shared endpoint

tc.endpoint_autoactivate(share_id)

tdata = TransferData(tc, host_id, share_id,
                     label='RDP copy to share',
                     sync_level='checksum')

tdata.add_item(source_path, '/~/', recursive=True)

r = tc.submit_transfer(tdata)

tc.task_wait(r['task_id'], timeout=1000,
             polling_interval=10)
# (4) Set access control to enable access by user

```python
r = ac.get_identities(ids=user_id)
email = r['identities'][0]['email']
rule_data = {
    'DATA_TYPE': 'access',
    'principal_type': 'identity',  # To whom is access granted?
    'principal': user_id,         # In this case, an individual user
    'path': '/',                  # Path to which access is granted
    'permissions': 'r',           # Grant read-only access
    'notify_email': email,        # Email invite to this address
    'notify_message':             # Include this message in email
        'The data that you requested from RDP is available.'
}

tc.add_endpoint_acl_rule(share_id, rule_data)
```
# (4) Set access control to enable access by user

```python
r = ac.get_identities(ids=user_id)
email = r['identities'][0]['email']
rule_data = {
    'DATA_TYPE': 'access',
    'principal_type': 'identity',  # To whom is access granted?
    'principal': user_id,  # In this case, an individual user
    'path': '/',  # Path to which access is granted
    'permissions': 'r',  # Grant read-only access
    'notify_email': email,  # Email invite to this address
    'notify_message': 'The data that you requested from RDP is available.'
}
tc.add_endpoint_acl_rule(share_id, rule_data)
```

# (5) Ultimately, delete the shared endpoint

tc.delete_endpoint(share_id)
What’s coming soon: Richer endpoints

HTTPS access to endpoints

• Enhanced use of research storage:
  • Asynchronous, bulk transfer: GridFTP
  • Synchronous remote access: HTTPS

• Enhanced Globus web app
  • Browser-based upload/download
  • Inline file viewer

• Integration with clients, web apps
What’s coming soon: Richer endpoints

HTTPS access to endpoints

• Enhanced use of research storage:
  • Asynchronous, bulk transfer: GridFTP
  • Synchronous remote access: HTTPS

• Enhanced Globus web app
  • Browser-based upload/download
  • Inline file viewer

• Integration with clients, web apps

Collections

• Groupings of files that are to be treated as logical units
• Can be named and described
What’s coming soon: Richer endpoints

**HTTPS access to endpoints**
- Enhanced use of research storage:
  - Asynchronous, bulk transfer: GridFTP
  - Synchronous remote access: HTTPS
- Enhanced Globus web app
  - Browser-based upload/download
  - Inline file viewer
- Integration with clients, web apps

**Data search**
- Automated metadata harvesting
  - From Globus endpoints
  - Event-driven extraction/synthesis
- Rich search capabilities
  - Free text, faceted, boosted

**Collections**
- Groupings of files that are to be treated as logical units
- Can be named and described
Thank you to our sponsors

And the Globus team at the University of Chicago and Argonne, in particular: Rachana Ananthakrishnan, Ben Blaiszik, Kyle Chard, Raj Kettimuthu, Ravi Madduri, Brigitte Raumann, Steve Tuecke, Vas Vasiliadis
We have constructed a **new global-scale data fabric** that accelerates discovery by streamlining scientific data sharing and analysis

- **Globus-enabled storage systems** enable robust, secure access
- **Globus cloud services** implement transfer, sharing, publication, discovery, and other capabilities

We are now working to extend this fabric to:

- Enable **distributed computation** as well as data movement
- Use distributed computation to **map data** without movement
- Work with **sensitive data**