**HPC DATA MANAGEMENT**

**Data Transfer Process**

**Version 1.0**

03/27/2019

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Implemented**  **By** | **Revision**  **Date** | **Description of Change** |
| 1.0 | Eran Rosenberg | 03/27/2019 | Initial Draft |

Table of Contents

[1. Introduction 4](#_Toc6396433)

[2. Upload (Data Registration) 5](#_Toc6396434)

[2.1. API Attachment -> Cleversafe 5](#_Toc6396435)

[2.2. AWS S3 -> Cleversafe 6](#_Toc6396436)

[2.3. Upload URL -> Cleversafe 6](#_Toc6396437)

[2.4. Globus -> Cleversafe (2-Hop Upload) 7](#_Toc6396438)

[2.5. API Attachment -> POSIX 7](#_Toc6396439)

[2.6. Globus -> POSIX 7](#_Toc6396440)

[3. Download 8](#_Toc6396441)

[3.1. Cleversafe -> API Attachment 8](#_Toc6396442)

[3.2. Cleversafe -> AWS S3 9](#_Toc6396443)

[3.3. Cleversafe -> Download URL 9](#_Toc6396444)

[3.4. Cleversafe -> Globus (2-Hop Download) 10](#_Toc6396445)

[3.5. POSIX -> API Attachment 10](#_Toc6396446)

[3.6. POSIX -> Globus 11](#_Toc6396447)

# Introduction

This document describes the process in which data is transferred (upload/download) to/from HPC-DME archive. The system supports multiple archive and source types through a single API. This document provides details on the process implemented to transfer in each combination of source and archive type.

HPC-DME is using 2 technologies to transfer data – Globus and S3.

At the integration layer, HpcDataTransferProxy defines a generic data transfer interface. This interface is implemented by 2 concrete classes, one for Globus and one for S3.

The following table shows the various types of archives, and the data transfer proxy implementation that is used to communicate with it:

|  |  |
| --- | --- |
| **Archive** | **Data Transfer Proxy Impl** |
| POSIX (File System) | Globus |
| Temporary (Server’s Globus Endpoint) | Globus |
| Cleversafe | S3 |

The following table shows the various types of user’s source/destination, and the data transfer proxy implementation that is used to communicate with it:

|  |  |
| --- | --- |
| **User’s Source/Destination** | **Data Transfer Proxy Impl** |
| Globus Endpoint | Globus |
| AWS S3 | S3 |

# Upload (Data Registration)

The process of file registration starts with registering the data object and metadata in iRODs. If this part completes successfully, the data transfer process to upload is kicked off. The upload data transfer process is controlled by 2 system metadata:

* data\_transfer\_type – The data transfer proxy that is engaged in upload (S3 or GLOBUS). Please note that in a particular case (will be discussed later) both data transfer proxies are used to upload a file.
* data\_transfer\_status – represents a ‘state’ the transfer is in (e.g. IN\_PROGRESS)

The following upload scenarios are supported:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Source** | **Sync / Async** | **Archive** |
| 2.1 | API attachment | Sync | Cleversafe |
| 2.2 | AWS S3 | Async | Cleversafe |
| 2.3 | Upload URL | Async | Cleversafe |
| 2.4 | Globus | Async | Cleversafe |
| 2.5 | API Attachment | Sync | POSIX |
| 2.6 | Globus | Async | POSIX |

For each scenario, a statechart shows the process and the values of the 2 system-metadata used to track and complete the upload

## API Attachment -> Cleversafe

A drawing of a person

Description automatically generated

## AWS S3 -> Cleversafe

A screenshot of a cell phone

Description automatically generated

## Upload URL -> Cleversafe

A screenshot of a cell phone

Description automatically generated

## Globus -> Cleversafe (2-Hop Upload)

A screenshot of a cell phone

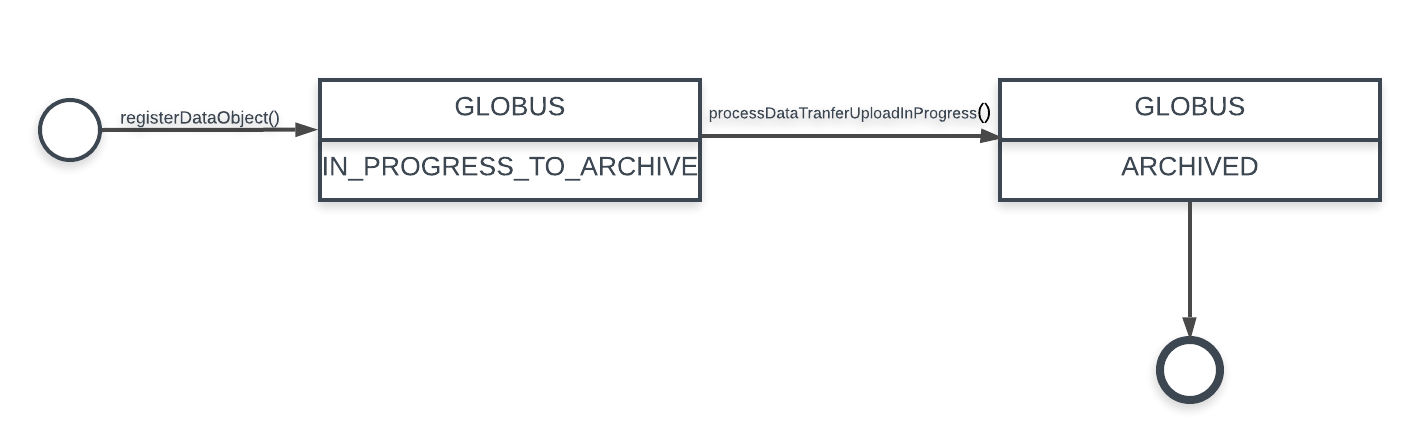
Description automatically generated

## API Attachment -> POSIX

A drawing of a person

Description automatically generated

## Globus -> POSIX



# Download

The process of file download is tracked in the DB table HPC\_DATA\_OBJECT\_DOWNLOAD\_TASK. Each row in this table represents an **active** download task of a single file. Once the task is completed (successfully or not), the results are recorded in HPC\_DOWNLOAD\_TASK\_RESULT, and the task is deleted.

The download data transfer process is controlled by 2 columns in the HPC\_DATA\_OBJECT\_DOWNLOAD\_TASK table:

* DATA\_TRANSFER\_TYPE – The data transfer proxy that is engaged in download (S3 or GLOBUS). Please note that in a particular case (will be discussed later) both data transfer proxies are used to download a file.
* DATA\_TRANSFER\_STATUS – represents a ‘state’ the transfer is in (e.g. IN\_PROGRESS)

The following download scenarios are supported:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Destination** | **Sync / Async** | **Archive** |
| 3.1 | API attachment | Sync | Cleversafe |
| 3.2 | AWS S3 | Async | Cleversafe |
| 3.3 | Download URL | Async | Cleversafe |
| 3.4 | Globus | Async | Cleversafe |
| 3.5 | API Attachment | Sync | POSIX |
| 3.6 | Globus | Async | POSIX |

For each scenario, a statechart shows the process and the values of the 2 columns used to track and complete the download

## Cleversafe -> API Attachment

This download is performed synchronously, and not tracked in the HPC\_DATA\_OBJECT\_DOWNLOAD\_TASK table. The user waits for the API call to complete, and the file is attached to the API response

## Cleversafe -> AWS S3

A screenshot of a cell phone

Description automatically generated

## Cleversafe -> Download URL

The request to generate a download URL is performed synchronously. The API returns the URL that the user can use to download the file directly from Cleversafe. This download is not tracked in the HPC\_DATA\_OBJECT\_DOWNLOAD\_TASK table.

## Cleversafe -> Globus (2-Hop Download)

A picture containing screenshot

Description automatically generated

## POSIX -> API Attachment

This download is performed synchronously, and not tracked in the HPC\_DATA\_OBJECT\_DOWNLOAD\_TASK table. The user waits for the API call to complete, and the file is attached to the API response

## POSIX -> Globus

A screenshot of a cell phone

Description automatically generated