DME Delete Archive Link Design

Design Document

Version 1.0

7/11/2025

# Version History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version Number | Implemented By | Revision Date | Approved By | Approval Date | Description of Change |
| 1.0 | Sarada Chintala |  |  |  | Initial Draft |

Table of Contents

[Version History 2](#_Toc203472815)

[1 Introduction 4](#_Toc203472816)

[1.1 Purpose of this document 4](#_Toc203472817)

[2 Software Design 4](#_Toc203472818)

[2.1 High Level Design 4](#_Toc203472819)

[2.2 Detailed Design 4](#_Toc203472820)

[2.2.1 Archive Link Identification 4](#_Toc203472821)

[2.2.2 Archive Link Deletion 5](#_Toc203472822)

[2.2.3 Check to see if the file related to the Archive link exists 5](#_Toc203472823)

[2.2.4 Rules based on User Role for Archive Links 5](#_Toc203472824)

[2.2.5 Deletion of all associated Links 6](#_Toc203472825)

[2.2.6 Removal of Physical file 6](#_Toc203472826)

[2.2.7 Deletion from IRODS and Soft Delete 6](#_Toc203472827)

[2.3 Migration 7](#_Toc203472828)

# Introduction

## Purpose of this document

The purpose of this document is to record the detailed design of the Deletion of an Archive Link.

# Software Design

## High Level Design

The new requirement is that an Archive Link should not physically delete the file. The table below summarizes the existing deletion process for Data Files and Registered Links. It also shows the new requirement of the deletion of Archive Links which should occur by default. The detailed design section will describe that scenarios in detail.

DELETION FOR DIFFERENT ENTITIES

|  |  |  |  |
| --- | --- | --- | --- |
|  | Soft Delete | Delete from IRODS | Delete file Physically |
| Regular Data File | Badge Tick1 with solid fill | Badge Tick1 with solid fill | Badge Tick1 with solid fillif force=true  Badge Cross with solid fill if force=false |
| Registered Link | Badge Cross with solid fill | Badge Tick1 with solid fill | Badge Cross with solid fill |
| Archive Link  (new requirement) | Badge Cross with solid fill | Badge Tick1 with solid fill | Badge Cross with solid fill |

## Detailed Design

### Archive Link Identification

The DataManagementConfiguration structure will be used to save and retrieve information to identify an Archive Link. Two fields will be added to the DataManagementConfiguratio. In the file hpc-domain-model/--/HpcDataManagement.xsd, two new fields will be added to HpcDataManagementConfiguration: 1) isExternalSystem 2) pathForPosix

### Archive Link Deletion

The delete dataObject API contains an optional boolean parameter ***force***. In the API Specification document, the definition of force is:

***force*** – If set to true, the API will perform a hard delete. This is an optional parameter and the default is false.

By default, this parameter should always set to true for Archive Links. These will enable the Hard Delete from IRODS with the existing code logic.

If ***Archive Link*** {

***force*** = true

}

To prevent the deletion of the physical file, the code logic needs to be slightly modified. It is described in Section 2.2.6.

DELETION OF ARCHIVE LINK

|  |  |  |  |
| --- | --- | --- | --- |
|  | Soft Delete | Delete from IRODS | Delete file Physically |
| Archive Link | Badge Cross with solid fill | Badge Tick1 with solid fill | Badge Cross with solid fill |

### Check to see if the file related to the Archive link exists

The current logic checks if the corresponding physical file exists for regular data files. Registered Links bypass this check and Archive Links will do the same by default.

The pseudo code will be:

If it is not a ***RegisteredLink*** OR if it is not an ***ArchiveLink*** then {

Check if file exists

If it does not exist {

Throw error "The data object was not found in the archive. S3 Archive ID: "

}

}

### Rules based on User Role for Archive Links

1. Validate if the invoker is the data owner of the DataObject. This check will be performed for Archive Links.
2. Hard delete is performed by System Admins only for regular Data files. This check does not need to be performed for Archive Links.
3. Group Admin Rule currently is: If the User is a Group Admin, check if file is less than 90 days old. This check is not necessary for Archive Links.

ROLE BASED CHECKS SUMMARY

|  |  |
| --- | --- |
| Rule | Archive  Link |
| Validate if the invoker is the owner of the data  Object | Badge Tick1 with solid fill |
| Hard delete is performed by System Admins only. | Badge Cross with solid fill |
| Group Admin - 90 day check | Badge Cross with solid fill |

### Deletion of all associated Links

Currently for regular Data Files and Registered Links, the associated Links associated with the DataObject are deleted. This will be done for Archive Links too.

### Removal of Physical file

The deletion of the physical file for the Archive Link should be prevented.

The current logic for the deletion of physical file is:

If not ***RegisterLink*** AND ***force*** {

Delete physical file

}

This logic will change to:

If NOT ***RegisterLink*** AND NOT ***ArchiveLink*** AND force {

Delete physical file.

}

### Deletion from IRODS and Soft Delete

The current logic for Soft Deletion and IRODS deletion is:

If ***force*** {

Delete the file from IRODS

} else {

// Soft Delete

Move DataObject to a Destination(Deleted Archive)

}

For Archive Links, we do not need to move the DataObject to a Destination Archive.

The metadata in IRODS can be deleted for links and for regular files if force param is set to true.

The logic will change to:

If (***registeredLink*** or ***archiveLink*** or ***force***) {

Delete the file from IRODS

}

/\* if a regular file and force is set to false \*/

If not a ***registerLink*** and not an ***archiveLink*** and not ***force*** {

// Soft Delete

Move DataObject to a Destination(Deleted Archive)

}

## Migration

The changes described in the above detailed design section will only affect the Rest API layers. No changes are needed in the Migration process which automatically deletes the Archive Links upon completion.