# Metadata Catalog API – Software Design Document

HPC-DME stores the metadata for each collection and data objects into the iRODS tables.

A new requirement is to provide a new API to retrieve a subset of metadata identified as Catalog metadata based on DOC or base path. Whether the subset of metadata for that collection shall be returned as a part of the Catalog metadata, is controlled using the metadata attribute with key “access”. If the value of “access” metadata is “Controlled access” or “Open access”, the subset of metadata identified as a part of the Catalog metadata for that catalog entry will be returned base on the DOC or base path specified by the user.

This document provides the software design to enable this capability.

## Metadata catalog

### Materialized view Design

We shall design a new materialized view to retrieve the catalog metadata. The materialized view will utilize the existing hierarchical metadata view r\_coll\_hierarchy\_meta\_main.

We shall populate only the predefined set of catalog metadata where the collection has the “access” attribute set to “Controlled access” or “Open access”.

Currently, the metadata attributes identified for the catalog is:

* Collection: PI\_Lab, Attribute name: pi\_name
* Collection: PI\_Lab, Attribute name: affiliation
* Collection: Project, Attribute name: origin
* Collection: Project, Attribute name: project\_title
* Collection: Project, Attribute name: project\_description
* Collection: Project, Attribute name: method
* Collection: Project, Attribute name: start\_date
* Collection: Project, Attribute name: access
* Collection: Project, Attribute name: project\_affiliation
* Collection: Project, Attribute name: project\_poc
* Collection: Project, Attribute name: publication\_status
* Collection: Project, Attribute name: origin\_repository
* Collection: Project, Attribute name: origin\_accession\_id
* Collection: Project, Attribute name: study\_id
* Collection: Project, Attribute name: study\_design
* Collection: Project, Attribute name: study\_disease
* Collection: Project, Attribute name: assembly\_name
* Collection: Project, Attribute name: cell\_line\_name
* Collection: Project, Attribute name: number\_of\_cases
* Collection: Project, Attribute name: summary\_of\_samples
* Collection: Project, Attribute name: organism

These predefined set of attributes will be stored in a static table: **HPC\_CATALOG\_ATTRIBUTE**.

The materialized view will have the following columns:

View name: **r\_catalog\_meta\_main**

* DOC text
* BASE\_PATH text
* object\_id bigint
* object\_path varchar
* meta\_id bigint
* meta\_attr\_name varchar
* meta\_attr\_value varchar
* meta\_attr\_unit varchar

### API Design

A new catalog metadata API will be provided to retrieve the catalog metadata based on user provided doc or base path input. The input JSON request will be the following:

{

"doc": "CCR\_SBL",

"basePath" : "/CCR\_SBL\_Admin\_Archive",

"page": 1,

"pageSize": 10,

"totalCount": true

}

Note: If neither “doc” nor “basePath” is specified, it will return all.

### Detailed Design

The following additions to the code base are needed to implement the metadata catalog API:

#### New HpcCatalogTypes.xsd

* Create a complex type HpcCatalogCriteria with attributes: doc, basePath, page, pageSize and totalCount.
* Create a complex type HpcCatalogEntry with attributes: attribute and value.
* Create a complex type HpcCatalog with attributes: doc, basePath, path and “catalogEntries” – list of type HpcCatalogEntry
* Create a complex type HpcCatalogMetadataEntry with attributes: doc, basePath, path, attribute and value

#### New HpcCatalog.xsd

* Create a complex type HpcCatalogRequestDTO with attributes: doc, basePath, page, pageSize and totalCount.
* Create a complexType HpcCatalogsDTO which includes attribute “catalogs” - list of type HpcCatalog, page, pageSize and totalCount.

#### HpcDataSearchRestServiceImpl.java

* Add queryCatalog() POST API to retrieve catalog entries

#### HpcDataSearchBusServiceImpl.java

* Add getCatalog() service to retrieve catalog entries and convert to HpcCatalogsDTO

#### New HpcCatalogServiceImpl.java

* Create getCatalog() API to be called from the Business layer
* Create getCatalogCount() API to be called from the Business layer

#### New HpcCatalogDAOImpl.java

* Create getCatalog() to retrieve data from the new catalog metadata view that matches the user specified criteria.
* Create getCatalogCount () to get the total count of objects from the new catalog metadata view that matches the user specified criteria.