

Name : Parth Nandedkar

Date : 21 Feb 2024

Topics : Azure Databricks

Batch : Data Engineering Batch-1

## **Q2.Explain Overview of 3 level namespace and creating Unity Catalog objects.**

The term "3 level namespace" typically refers to a hierarchical structure for organising data within a distributed file system like UnityFS. UnityFS is a highly scalable and parallel file system designed for use in high-performance computing (HPC) and big data environments. Here's an overview of the 3 level namespace concept and creating Unity Catalog objects within this context:

Hierarchy Levels:

The 3 level namespace consists of three hierarchical levels: root, directory, and file.

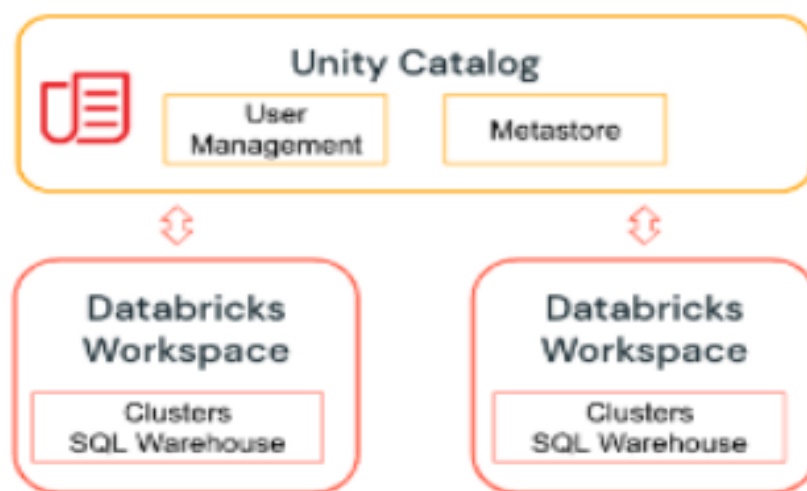
**1.Root Level:** At the top level, there is a single root directory that serves as the starting point for the entire file system.

**2.Directory Level:** Directories (or folders) exist beneath the root and can contain both subdirectories and files.

**3.File Level:** Files reside at the lowest level of the hierarchy and represent the actual data stored within the file system.

## Unity Catalog:

In the context of UnityFS, a catalog is a metadata repository that stores information about the file system's structure, attributes, and location of data. The Unity Catalog maintains mappings between logical file paths (namespaces) and physical data locations within the storage system. Catalog objects represent various entities within the file system, such as directories, files, and attributes associated with them.



**Define once, secure everywhere:** Unity Catalog offers a single place to administer data access policies that apply across all workspaces. Standards-compliant security model: Unity Catalog's security model is based on standard ANSI SQL and allows administrators to grant permissions in their existing data lake using familiar syntax, at the level of catalogs, databases (also called schemas), tables, and views.

**Built-in auditing and lineage:** Unity Catalog automatically captures user-level audit logs that record access to your data. Unity Catalog also captures lineage data that tracks how data assets are created and used across all languages.

**Data discovery:** Unity Catalog lets you tag and document data assets, and provides a search interface to help data consumers find data.

**System tables (Public Preview):** Unity Catalog lets you easily access and query your account's operational data, including audit logs, billable usage, and lineage.

### **Creating Unity Catalog Objects:**

To create Unity Catalog objects, you typically interact with the UnityFS API or command-line tools provided by the file system.

The exact method for creating catalog objects may vary depending on the specific implementation of UnityFS and the tools available.

Generally, you would use commands or API calls to create directories, files, and set attributes as needed within the file system.

Catalog objects can be created programmatically through API calls or manually using administrative tools provided by UnityFS management interfaces.

### **Usage and Benefits:**

The 3 level namespace and Unity Catalog provide a structured way to organise and manage data within the file system.

By creating catalog objects, administrators can efficiently manage file system resources, set permissions, and track metadata associated with files and directories.

Users can navigate the file system, access files, and perform operations such as reading, writing, and deleting data using standard file system interfaces while the underlying Unity Catalog manages the metadata and data placement transparently.

In summary, the 3 level namespace and Unity Catalog play crucial roles in organising and managing data within a distributed file system like UnityFS. Understanding these concepts helps administrators and users effectively interact with the file system and utilise its capabilities for storing and accessing large volumes of data.