1. **Giới thiệu**

Unity Catalog cung cấp các khả năng kiểm soát quyền truy cập, kiểm tra đánh giá, theo dõi nguồn gốc dữ liệu và khám phá dữ liệu tập trung trên nền tảng Databricks.

Đặc điểm:

* **Định nghĩa một lần, bảo mật mọi nơi** : Unity Catalog cung cấp một nơi duy nhất để quản lý các chính sách truy cập dữ liệu áp dụng trên tất cả các không gian làm việc.
* **Mô hình bảo mật tuân thủ tiêu chuẩn** : Mô hình bảo mật của Unity Catalog dựa trên ANSI SQL tiêu chuẩn và cho phép quản trị viên cấp quyền trong hồ dữ liệu hiện có của họ bằng cú pháp quen thuộc, ở cấp catalog, database (còn gọi là lược đồ), table và view.
* **Tích hợp kiểm tra và đồ thị phát triển dữ liệu.** : Unity Catalog tự động ghi lại log kiểm tra cấp người dùng để ghi lại các lịch sử truy cập vào dữ liệu của bạn. Unity Catalog cũng thu thập dữ liệu phát triển của dữ liệu để theo dõi cách tạo và sử dụng nội dung dữ liệu trên tất cả các ngôn ngữ.
* **Khám phá dữ liệu** : Unity Catalog cho phép bạn gắn thẻ và ghi lại nội dung dữ liệu, đồng thời cung cấp giao diện tìm kiếm để giúp người tiêu dùng dữ liệu tìm thấy dữ liệu.
* **Bảng hệ thống**: Danh mục Unity cho phép bạn dễ dàng truy cập và truy vấn dữ liệu hoạt động của tài khoản của mình, bao gồm nhật ký kiểm tra, mức sử dụng phải trả phí và đồ thị phát triển dữ liệu.

1. **Tính năng của Unity Catalog**

A graphic of a diagram

Description automatically generated with medium confidence

Unity Catalog cung cấp 4 giải pháp cho doanh nghiệp

1. Data Access Control

Data Access Control quản lý quyền truy cập đến các file trong data lake, bảng, notebooks, dashboards và các mô hình học máy. Mô hình bảo mật của Unity Catalog đảm bảo cho mọi đối tượng bên trong metastore được bảo mật.

Có 3 loại danh tính có thể tương tác với đối tượng trong databricks:

* User: Người đăng nhập vào databricks bằng email và tương tác trực tiếp qua UI

A white background with colorful text

Description automatically generated

* Service Principles: Người dùng hệ thống được tạo để thực hiện các tác vụ tự động

A computer screen shot of a computer

Description automatically generated

* Groups: Nhóm có thể có nhiều users hoặc service principles để quản lý chung

A screenshot of a computer

Description automatically generated

Có 4 loại roles trong Unity Catalog:

* Account admin: Là role có vai trò cao nhất, có toàn quyền truy cập vào mọi kho lưu trữ được liên kết với tài khoản Databricks, vai trò duy nhất có quyền tạo metastore.
* Metastore admin: Mặc định ban đầu là người tạo metastore, có quyền quản lý, xóa metastore, sở hữu các đối tượng bên trong metastore và cấp quyền cho user khác thành metastore admin
* Object owner: Mặc định ban đầu là user tạo đối tượng, có thể cấp quyền quản lý đối tượng cho user khác, có toàn quyền truy cập vào đối tượng
* Object users: Là phần lớn người dùng, cần được cấp quyền để truy cập vào đối tượng

A diagram of a software company

Description automatically generated

Unity Catalog hoạt động theo mô hình không gian ba cấp, để user có quyền sử dụng view, table, function (như SELECT, MODIFY) ngoài việc cấp quyền sử dụng view, table, function bắt buộc cần quyền sử dụng catalog USE CATALOG và quyền sử dụng schema USE SCHEMA. Việc cấp quyền hoạt động theo cơ chế kế thừa, quyền được cấp ở cấp trên sẽ tự động được kế thừa xuống cấp dưới trong mô hình.

1. Data Audit

Data Audit cho phép xem log người dùng, thấy được cách người dùng sử dụng dữ liệu, tần suất truy cập dữ liệu.

A screenshot of a computer

Description automatically generated

Khi người dùng tạo workspace, 1 catalog mặc định tên system sẽ được tạo, chứa schema information\_schema, các bảng thông tin về hành vi người dùng tương tác với catalog và các đối tượng bên dưới catalog trong cây mô hình như schema, table, view, function

Khi người dùng tạo catalog, 1 schema mặc định tên information\_schema sẽ được tạo bên trong chứa các bảng thông về hành vi người dùng tương tác với schema và các đối tượng bên dưới schema trong cây mô hình như table, view, function

1. Data Lineage

Data Lineage cung cấp đồ thị để người dùng có thể thấy được toàn bộ quá trình biến đổi trước và sau của các bảng ở cả cấp độ bảng và cột, các trường dữ liệu, hỗ trợ tìm kiếm nguyên nhân khi vấn đề xảy ra.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

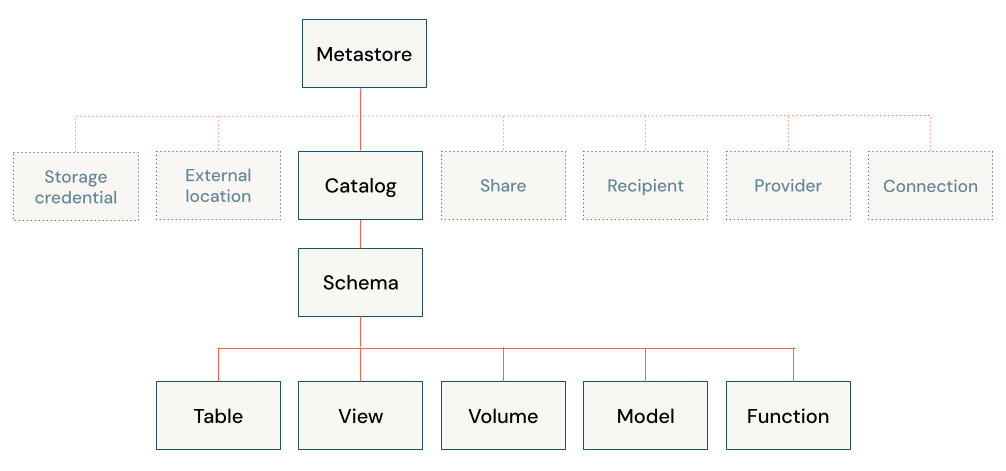
1. Data Discovery

Data Discovery cung cấp khả năng tìm kiếm dễ dàng giữa hàng ngàn object trong data lake, cung cấp danh mục các object với khả năng tìm kiếm ngay trên giao diện. Người dùng có thể tìm kiếm tất cả các đối tượng trong mô hình phân cấp của Unity Catalog bằng tên dưới thanh tìm kiếm, cũng có thể tìm kiếm cột ở bên trong các bảng

A screenshot of a computer

Description automatically generated

1. **Hệ thống phân cấp các đối tượng dữ liệu trong Unity Catalog**



* **Metastore** : Nơi chứa metadata cấp cao nhất. Mỗi metastore hiển thị một không gian tên ba cấp độ ( catalog. schema. table) để sắp xếp dữ liệu của bạn.
* **Catalog** : Lớp đầu tiên của hệ thống phân cấp đối tượng, được sử dụng để tổ chức các data assets.
* **Schema** : Còn được gọi là cơ sở dữ liệu, schema là lớp thứ hai của hệ thống phân cấp đối tượng và tables, views, volumes.
* **Tables, views and volumes**: Ở mức thấp nhất trong hệ thống phân cấp đối tượng dữ liệu là tables, views and volumes. Volumes cung cấp khả năng quản trị cho dữ liệu không phải dạng bảng.
* **Models**: Các mô hình học máy cũng có thể được quản lý trong Danh mục Unity và nằm ở cấp thấp nhất trong hệ thống phân cấp đối tượng.

1. Metastore

Metastore là nơi chứa các đối tượng cấp cao nhất trong Unity Catalog. Nó chứa metadata về data và AI assets cũng như các quyền quản lý quyền truy cập vào chúng. Databricks account admins chỉ được tạo một metastore cho mỗi khu vực mà họ hoạt động và chỉ định chúng cho các không gian làm việc của Databricks trong cùng khu vực. Để một không gian làm việc sử dụng Unity Catalog, không gian làm việc đó phải được đính kèm Unity Catalog metastore.

1. Catalog

Danh mục là cấp cao nhất trong hệ thống phân cấp dữ liệu (catalog > schema > table/view/volume) được quản lý bởi kho lưu trữ Unity Catalog. Chúng được coi là đơn vị cô lập dữ liệu chính trong mô hình quản trị dữ liệu Databricks điển hình.

Catalog đại diện cho một nhóm logic của các schema theo các yêu cầu truy cập dữ liệu. Catalog thường đại diện cho các đơn vị tổ chức hoặc phạm vi vòng đời phát triển phần mềm. Ví dụ: bạn có thể chọn có một danh mục cho dữ liệu sản xuất và một danh mục cho dữ liệu phát triển hoặc một danh mục cho dữ liệu không phải của khách hàng và một danh mục cho dữ liệu khách hàng nhạy cảm.

Catalog có thể được lưu trữ ở cấp độ metastore hoặc bạn có thể định cấu hình một catalog để được lưu trữ riêng biệt với phần còn lại của metastore gốc. Nếu workspace chưa được cấu hình UnityCatalog thì người dùng phải tự cấu hình địa chỉ cho catalog khi tạo.

Theo mặc định, quyền truy cập đối với một đối tượng có thể bảo mật được kế thừa bởi các đối tượng con của đối tượng đó, catalog ở đầu hệ thống phân cấp. Điều này giúp dễ dàng hơn trong việc thiết lập các quyền truy cập mặc định cho dữ liệu của bạn và chỉ định các quyền khác nhau ở mỗi cấp của hệ thống phân cấp.

Người dùng cần được cấp quyền USE CATALOG để sử dụng catalog. Quyền được cấp ở catalog sẽ tự động kế thừa xuống các schema bên dưới.

A screenshot of a computer

Description automatically generated

Overview: Chứa thông tin các schema trong catalog

Detail: Chứa thông tin về catalog

Permission: Cấp quyền cho user sử dụng catalog

Workspaces: Thông tin về workspace có thể truy cập đến catalog

1. Schema

Schema, còn được gọi là cơ sở dữ liệu, là các nhóm logic của dữ liệu dạng bảng (table và view), dữ liệu không phải dạng bảng (volume), function và mô hình học máy. Chúng cung cấp cho bạn cách tổ chức và kiểm soát quyền truy cập vào dữ liệu chi tiết hơn danh mục.

Các schema có thể được lưu trữ trong cùng bộ lưu trữ vật lý với catalog cha hoặc bạn có thể định cấu hình một schema để được lưu trữ riêng biệt với phần còn lại của catalog cha.

Người dùng cần được cấp quyền USE CATALOG và USE SCHEMA để sử dụng schema. Quyền được cấp ở schema sẽ tự động kế thừa xuống các table,view,volume và function bên dưới.

A screenshot of a computer

Description automatically generated

Overview: Chứa thông tin về table, view, function và mô hình học máy bên trong schema

Detail: Chứa thông tin schema

Permission: Cấp quyền cho user sử dụng schema

1. Table

Nằm trong tầng thứ ba trong không gian tên ba cấp của Unity Catalog, chứa các hàng dữ liệu. Chứa các hàng dữ liệu. Để tạo bảng , users phải có quyền CREATE, USE SCHEMA  trong schema cha và USE CATALOG  trong catalog cha.Để query bảng, users phải có quyền SELECT của bảng, USE SCHEMA  trong schema cha và USE CATALOG  trong catalog cha

A screenshot of a computer

Description automatically generated

Overview: chứa các hàng trong bảng

Sample Data: Xem trước dữ liệu của bảng

Details: Thông tin chi tiết bảng

Permission: Quyền cho user sử dụng bảng

History: Lịch sử thay đổi của bảng

Lineage: Thông tin các bảng trong luồng phát triển và đồ thị

Insights: Dashboard được tổng hợp từ dữ liệu tương tác với bảng

Có 2 loại là managed và external

Managed tables

Managed tables được dùng nhiều hơn trong các cách tạo table của Unity Catalog. Unity Catalog quản lý vòng đời và file cho các bảng này, cũng tối ưu hiệu năng tự động, luôn dùng Delta table format

Đối với các không gian làm việc đã được bật cho Unity Catalog theo cách thủ công, các bảng được quản lý sẽ được lưu trữ ở vị trí lưu trữ gốc mà bạn định cấu hình khi tạo metadata. Có thể tùy ý chỉ định các vị trí lưu trữ bảng được quản lý ở cấp catalog hoặc schema, ghi đè vị trí lưu trữ gốc.

Đối với các không gian làm việc được bật tự động cho Unity Catalog, vị trí lưu trữ gốc của metastore là tùy chọn và các bảng được quản lý thường được lưu trữ ở cấp catalog hoặc schema.

External tables

External tables là các bảng mà vòng đời dữ liệu và file không được quản lý bởi Unity Catalog. Dùng external tables để đăng ký dữ liệu đã tồn tại vào Unity Catalog hoặc cần truy cập vào dữ liệu dùng công cụ bên ngoài Databricks.

Khi xóa bảng, Unity Catalog không xóa data nằm phía dưới. Cách quản lý và query giống với managed tables.

Các file formats của External tables

* DELTA
* CSV
* JSON
* AVRO
* PARQUET
* ORC
* TEXT

1. Views

Là các đối tượng read-only được tạo từ tables và views trong metastores, nằm trong lớp thứ ba của không gian tên ba cấp của Unity Catalog.

1. Volumes

Volumes nằm trong lớp thứ ba của không gian tên ba cấp của Unity Catalog ngang cấp với tables,views và các objects khác dưới schema. Volumes chứa các thư mục, files dữ liệu lưu trữ ở bất kì dạng nào, cung cấp các kiểu truy cập non-tabular đến dữ liệu, có nghĩa là các files trong volumes không được đăng ký như tables

Để tạo volume phải có quyền CREATE VOLUME,USE SCHEMA trên schema cha và USE CATALOG trên catalog cha

Để đọc files và thư mục trong volume cần có quyền READ VOLUME, USE SCHEMA trên schema cha và USE CATALOG trên catalog cha.

Để thêm, xóa , sửa files và thư mục trong volum cần có quyền WRITE VOLUME, USE SCHEMA trên schema cha và USE CATALOG trên catalog cha.

Có 2 loại là external và managed

Managed volumes được lưu trữ trong địa chỉ kế thừa từ schema cha.

External volumes được đăng ký vào Unity Catalog external location and cung cấp quyền truy cập đến các file tồn tại trên kho lưu trữ cloud mà không yêu cầu di chuyển dữ liệu, cần có quyền CREATE EXTERNAL VOLUME trên external location để tạo external location. External volumes hỗ trợ việc files được tạo bởi hệ thống khác và truy cập vào databricks dùng công cụ và cách lưu trữ bên ngoài Databricks. Databrick không quản lý file của external volumes, khi xóa volume, dữ liệu bên dưới không bị xóa

A screenshot of a computer

Description automatically generated

Overview: Thông tin file trong volumes

Details: Thông tin chi tiết về volumes

Permission: Quyền cho user sử volume

1. Models

Models nằm trong lớp thứ ba của không gian tên ba cấp của Unity Catalog, là các mô hình học máy được đăng ký trong MLflow Model Registry. Để tạo model cần có quyên CREATE MODEL trong catalog hoặc schema,USE SCHEMA trên schema cha và USE CATALOG trên catalog cha.

1. Storage credentials and external locations

Để quản lý truy cập vào các kho lưu trữ đám mây cho external tables, external volumes và managed storage, Unity Catalog sử dụng các object:

* **Storage credentials** đóng gói thông tin xác thực đám mây cung cấp quyền truy cập vào bộ lưu trữ đám mây,
* **External location** chứa storage credentials và đường dẫn đến kho lưu trữ đám mây bên ngoài.

1. **So sánh Unity Catalog công nghệ khác**