Section 02 – Lakehouse (udemy)

Friday, October 13, 2023 12:27 PM

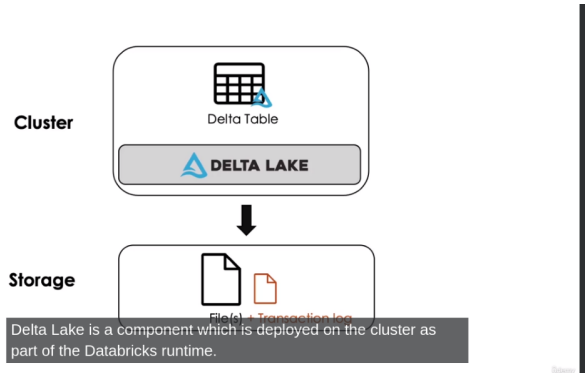
What is Delta lake?
It is a open source storage framework or layer that brings reliability to data lakes, it overcomes performance, inconsistency and limiation issues in data lake.IS

IS	IS not
Open source framework	Propriety which means handled by only one
Storage framework: it is broaded architecture concept that encompases overall structure design, principles for managing and stroing the data within a computing system.	Storage format, which is an specific format in which data is store such as json or csv etc.
Enable lake house platform unify both data warehouse and advanced analytics.	Data warehouse or database service

It is a component which is deployed on cluster as a part of databricks runtime, It is stored on the storage of respective cloud. It stores two types of file or information.

1. Data files (in parquet format)
   The data which is stored, extracted from any source or stored as results from any query
2. Transaction log(In json format)

- ALso known as delta log, it contains the information of every transaction performed on table since it created.
- Serve as single source of truth
- When you query, spark checks the transaction log and return the most recent version of data.
- It manages information in json format with detail of operation type, Predictes such as condition or filters used during the operation and other info.



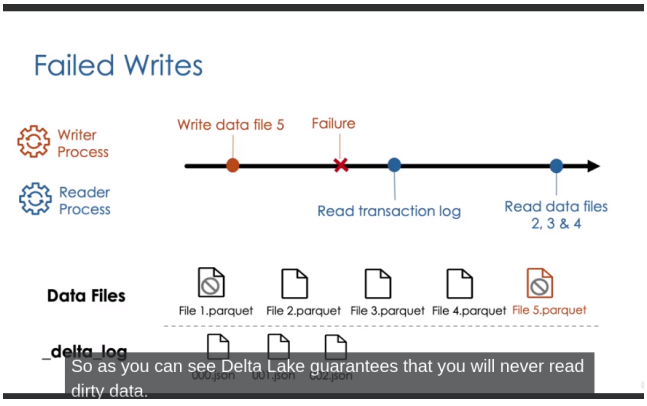
Write Process
When we perform write process, files are stored in parquet format, when process completed it add transaction log info. It create two fies (data files and transaction log)

Reader Process
It reads transaction log file and and read data files only which defined as latest version in transaction log.

Update Process
Whenever you update any record in delta lake, It directly create new file instead of updating existing file, new delta log file will be generated to mention updated files names or versions only.

Suppose another operation is happening, or in process of updating, it will give you second latest version until the operation gets updated. 😊 .

If file gets corrupted between the process, delta logs will not be updated until it completes the process properly. Databricks confirms you never feed dirty data.



Delta lake advantages

1. Bring ACID transactions to object storage
2. Handle scalable metadata
3. Full audit trial of all changes
4. Based upon standard data formats (parquet and json)

Hands On (Uploaded on git repo)

Advanced features

1. **Time travel**
2. **Compacting Small files and Indexing (Optimization)**
3. **Vaccum (Clean Up unused data files table direc)**

1. **Time Travel**
   You can audit changes using describe history command
   Can query older versions using a timestamp or version number
   It helps you to look back in case if required

   If your pipeline job had a bug that corrupted or made dirty data, using time travel you can easily fix errors.

   You can restore table using restore operation.

2. **Compacting files**
   Compacting small files, databricks can improve the speed of read quries from a table, compacting small files into large ones.
   Instead to reading records from multiple files, all data can be optimized to single files or less distributions basted on data.

   Further **Z order indexing,** also can be applied to which co or re locate and recognizing column information in the same set of files.

   For example if you have numberical values in files such id, it will optimize the files according to the id columne, make large files 1-50 ids in separate files and 50-100 ids. By this information will be stored in log about which ids are stored in which file.
   Data skip algorithm will skip the files automatically, and he will know which files to read for aprticular ids based on give query. (readuce the amount of data needs to be read)

3. **Vaccum on delta table**
   Cleaning up unused files in delta table, files with older versions.
   Uncommitted files
   Files that are no longer in latest table

   Data bricks allows you to garbage collection and remove by using vaccum command, you just need to specify threshold of retention (number of days or duration of which data you want to have with you (e.g 7 days (all versions before 07 days will be removed).

   By default, threshold is 07 days.

   Vaccum table_name [retention period]

   Limitations
   Once you use vaccum, you lose ability to time travel because all your privious than retention threshold will be removed.

Relational Entities

1. Databses
   It is actually a schema in hive metastore, you can use crate database or schema db name, which is exactly the same. You can create both external and managed databses, to create managed databse you just use simply create dabase command where for external database you need to specify the location using Location keyword. 7
   4

2. Hive Metastore
   Repository of metadata, it stores data structures, such as databses, tables and partitions. Every databse has a central hive data metastore, accessible by all cluster to presist metadata.

3. Tables
   There are two types of tables in data Bricks.

Managed table	**External Table**
A managed table: which is created in storage under the database directory (Default table).	External table: Table created in storage out of databse directory specified by **location** keyword
Hive manages both metadata and table data. Which means it manages life cycle.	Hive only onws the metadata not the unerlying data files.
Droping the table will delete underlying files	Dropping the table will not delete the underlying files

   Hands On

Setup data table

Learning Outcomes
CTAS statements
Table Constraintsor
Cloning delta lake table

1. **CTAS statement**
   Create table as select statement, create and populate data tables using output of select statement.
   Automatically infer schema,
   Don't support manually schema creation

   Additional options
   You can comment
   Can be partitioned by columns in table
   Can be external table using Location keyword

   Normal table vs CTAS

Normal	**CTAS**
Manual Schema Declaration	Do not support Manul Schema declaration instead automatically infer the schema
Create Empty table	Table with data.

2. **Table Contstraints**
   Table contstrains are use to limit the type of data that can go into a table.
   Ensures accuracy and reliability of the data which is being stored in table.

a. Not null contraint
b. Check Constraint

You must ensure that there is no data already in table which is voilating the constraints.
Once you add the constraints in the table, data which is violating constraints, will results in data write failure.

Example
ALTER TABLE orders
ADD CONSTRAINTS valid_date CHECK (date > '2020-01-01'

3. **Clone Delta Lake**
   Clone or make a copy of delta lake,

a. Deep Clone
Fully copies data and meta data from source to the target,
Example:
CREATE TABLE clone_table
DEEP CLONE source_table;

Can sync changes, copies can occur incrementaly.
Take quite a while for large datasets, this is why you need **Shallow Clone**

b. **Shallow Clone**
Quickly create a copy of a table, just copy the delta transaction logs. No data moving/ sync during shallow clone, It is good option to test out applying changes on a table without the risk of modyfying.

Example :
Create table table_name
SHALLOW CLONE source_table

Cloning is a great way to copy production tables for testing your code in development mode.
In either case, data modification will not effect the source.

Views in Databricks
Logical Query against source tables
It is an virtual table that has no physical data, just a save SQL query againts actual tables.
This query is executed each time when a view is quired.

Types of Views
1. **Classical or Stored views**
   These views are presisted in databse,
   CREATE VIEW view_name
   AS Query (e.g select * from table_name)
2. **Temporary Views**
   These views are tied with spark session so It is dropped when session ends.
   Session-Scoped views,
   Create TEMP VIEW view_name
   AS query

Times when spark session is created:

- Opening a notebook
- Detaching and Attaching to a cluster
- Installing a python package

- Restarting a Cluster

3. **Global Temporary Views**
   Cluster Scoped views, as long as cluster is running, any notebook attached to cluster will access the view.

   CREATE GLOBAL TEMP VIEW view_name as query
   Select * from global_temp.view_name