Starburst

Starburst Workshop: Modern table formats

Subtitle Goes Here

Starburst Academy
Data Universe 2024

Workshop objectives

- Move beyond Hive to understand the benefits of modern table formats such as Apache Iceberg.
- Run ACID compliant transactions to modify your data and understand how this creates new table versions.
- Explore time-travel queries and table rollbacks.
- Modifying the partitioning strategy without rebuilding your table.

For today...

- POIs
    - The lab guide has it ALL!
    - The S3 credentials will expire after the weekend
    - *Possible* copy/n/paste "issues"
        - if SQL errors out, verify your editor has it right
- Approach
    - I'm going to perform the labs myself & you can…
        - do them along with me
        - or just watch (and maybe do later)
    - We ALL come from DIFFERENT experiences & backgrounds, so…
        - **ASK QUESTIONS**

Starburst

Table formats

Moving beyond Hive

Starburst

Table formats

What are they?

Open-source mechanisms that manage and track all the files and metadata that make up a table.

Essentially, an abstraction layer between the physical files and how they are represented as tables accessible via SQL.

Were developed to address limitations with Hive.

Isn’t that Hive??

Yes, Hive’s approach of maintaining a table’s schema along with the location of where the files are, plus the file format being used, does just that.

It was only when alternatives were developed that the “Table Format” terminology surfaces.

Hive is the first-generation table format.

Table format vs file format

Table formats

A table format specifies the way in which a schema is enacted when recorded as actual files.

Examples:

- Apache Hive
- Apache Iceberg
- Delta Lake
- Apache Hudi

File formats

File formats control the way that each individual file stores data inside it.

Examples:

- Apache ORC
- Apache Parquet
- Apache AVRO
- JSON

Starburst

Hive limitations

Metadata

The Hive Metastore (HMS) impacts scalability & performance

Data manipulation

Inconsistent and limited INSERT, UPDATE, DELETE, MERGE abilities

Structural rigidity

Difficult to evolve the schema and impossible to change partitioning

Table history

No inherent table history concepts for rollback or “as of” querying

Table formats

Comparing & contrasting alternatives

Starburst

Overview

	HIVE	ICEBERG	DELTA LAKE	Apache hudi
Started at	Facebook	Netflix	Databricks	Uber
First OSS commit	2010	2017	2019	2016
Project governance	Apache Software Foundation (diverse PMC)	Apache Software Foundation (diverse PMC)	Linux Foundation (all-Databricks TSC)	Apache Software Foundation (diverse PMC)
Starburst integration	✓	✓	✓	read-only

ACID transactions

	HIVE	ICEBERG	DELTA LAKE	Apache hudi
INSERT	Statement available on all tables, but transactional via Hive ACID tables only**	✓	✓	via Spark, not supported in Trino/Starburst
UPDATE	Hive ACID tables only** (no changes to partition or bucket columns)	✓	✓	via Spark, not supported in Trino/Starburst
DELETE	Hive ACID tables** *Additionally, insert-only partitioned tables when filter matches full partition*	✓	✓	via Spark, not supported in Trino/Starburst
MERGE	Hive ACID tables only**	✓	✓	via Spark, not supported in Trino/Starburst

** Hive ACID tables created with Hive Streaming Ingest are not supported

Starburst

Key features

	HIVE	ICEBERG	DELTA LAKE	Apache hudi
File format support	Parquet, ORC, AVRO, RCText, RCBinary, SequenceFile, JSON, TextFile, CSV, RegEx	Parquet, ORC, AVRO	Parquet	Parquet, ORC
Time-travel		✓	✓	✓
Schema evolution	Limited support and only updates metadata	✓	✓	via Spark, not supported in Trino/Starburst
Partition evolution		✓		

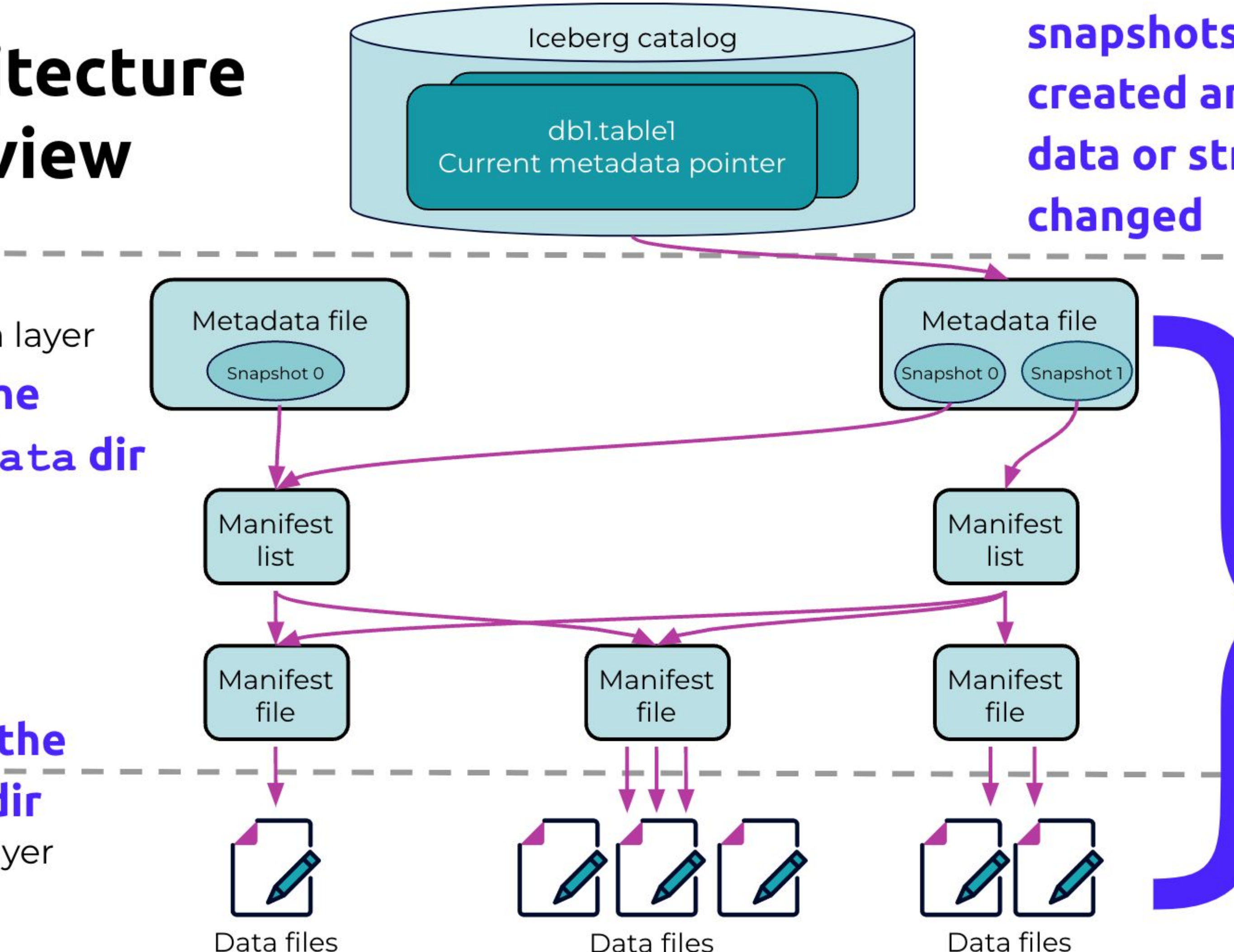
Apache Iceberg

Creating an open data lakehouse

Starburst

Architecture overview

snapshots are created anytime data or structure is changed



Metadata layer

lives in the `./metadata` dir

saved in the `./data` dir

Data layer

metadata and data persisted together on the data lake

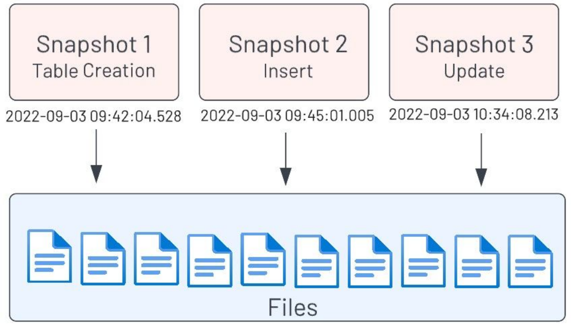
Starburst

Modifying table contents

Data Manipulation Language (DML)

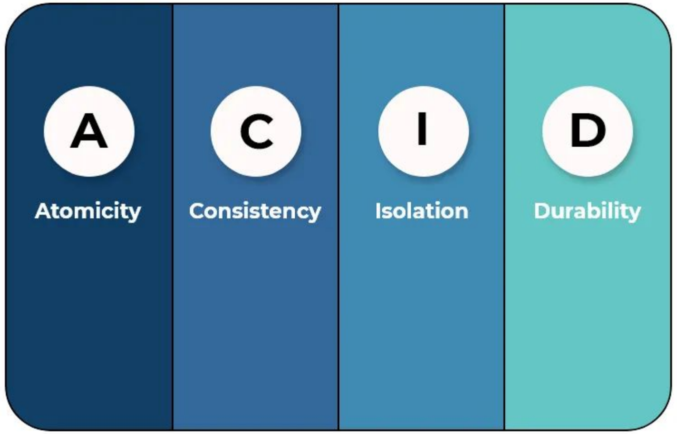
DML operations create new versions and utilize a merge-on-read strategy for the changes

- `INSERT, UPDATE, DELETE`
- and `MERGE`



ACID transactions

Single statement, single table, operations meet the "ACID test" for transactional integrity



Starburst

Starburst

HANDS-ON LABS!!

Starburst Academy