databricks

Ingestion Alternatives

LECTURE

Ingesting into Existing Delta Tables

Ingesting into existing Delta tables is streamlined with the MERGE INTO command, enabling atomic updates, inserts, and deletes from a source—all in a single operation for robust incremental data loads and change data capture.

Ingesting into Existing Delta Tables

Updates, Inserts, and Deletes on Delta Tables with MERGE INTO

- Merges a set of **updates, insertions, and deletions** from a **source** table into a **target** Delta table.
- MERGE INTO supports **schema enforcement** or **schema evolution** and allows different actions depending on whether a row is matched between a source and target table:
  - Matched rows: **UPDATE** or **DELETE**
  - Unmatched rows by target: **INSERT**
  - Unmatched rows by source: **UPDATE** or **DELETE**
- This functionality makes MERGE INTO ideal for handling **slowly changing dimensions (SCDs)**, **incremental loads**, and complex **change data capture (CDC)** scenarios.

The MERGE INTO command allows you to apply updates, inserts, and deletes from a source table into an existing Delta table, all in a single, atomic operation.

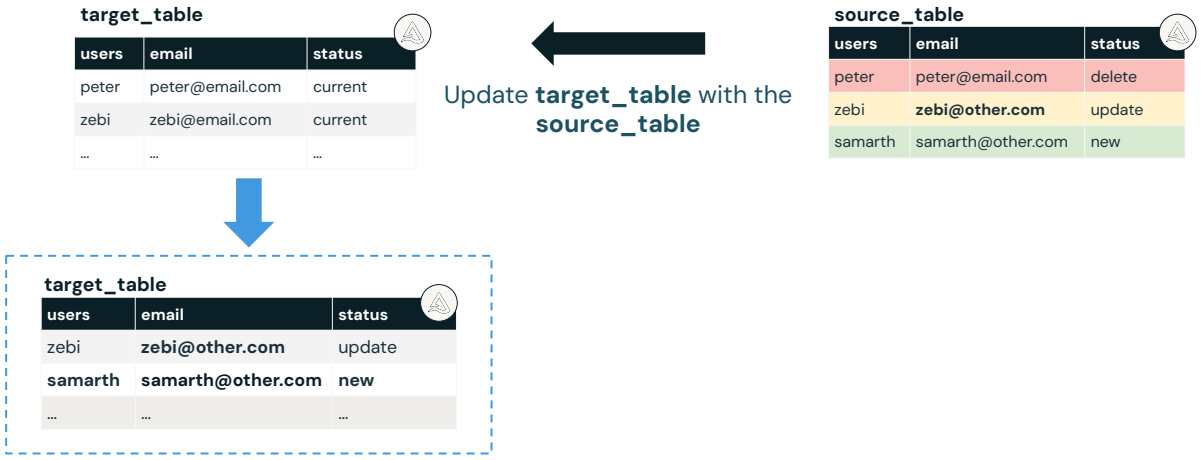
MERGE INTO supports schema enforcement or schema evolution and allows different actions depending on whether a row is matched between a source and target table:

- **Matched rows:** UPDATE or DELETE
- **Unmatched rows by target:** INSERT
- **Unmatched rows by source:** UPDATE or DELETE

This functionality makes MERGE INTO ideal for handling slowly changing dimensions (SCDs), incremental loads, and complex change data capture (CDC) scenarios.

Ingesting into Existing Delta Tables

Updates, Inserts, and Deletes on Delta Tables with MERGE INTO



There are situations where you need to update, insert, or delete records in a target table based on information from another table.

In this case, we have a target table called **target_table**, which contains a list of users and their statuses, and a table named **source_table**, which holds updated user information. The **target_table** table indicates that we need to delete the user Peter, update Zebi's email, and add a new user named Samarth to the **users_target** table.

The goal is to update the **users_target** table with the most up to date information.

Ingesting into Existing Delta Tables

MERGE INTO

source_table

id	users	email	status
1	peter	peter@email	delete
2	zebi	zebi@email	update
3	samarth	samarth@email	new

Original target_table

id	users	email	status
1	peter	peter@email	current
2	zebi	zebi@email	current
4	matt	matt@email	current

```
MERGE INTO target_table target
USING source_table source
```

Target table (table to update)

Source table (table to merge into target)

MERGE INTO **target_table** target - This declares the target table, the table you want to update, insert into, or delete from. You're giving it the alias **target** to reference it later in the query.

USING **source_table** source - This defines the source table, the table that contains the new data or changes you want to apply to the target. It's given the alias **source**.

Ingesting into Existing Delta Tables

source_table

id	users	email	status
1	peter	peter@email	delete
2	zebi	zebi@email	update
3	samarth	samarth@email	new

Updated target_table

id	users	email	status
1	peter	peter@email	current
2	zebi	zebi@email	current
4	matt	matt@email	current

```
MERGE INTO target_table target
USING source_table source
ON target.id = source.id
```

Specify the condition for merging

ON **target.id = source.id** - Defines the matching condition between the source and target tables, typically using a unique key like **id**.

Ingesting into Existing Delta Tables

source_table

id	users	email	status
1	peter	peter@email	delete
2	zebi	zebi@email	update
3	samarth	samarth@email	new

Updated target_table

id	users	email	status
1	peter	peter@email	current
2	zebi	zebi@email	update
4	matt	matt@email	current

```
MERGE INTO target_table target
USING source_table source
ON target.id = source.id
WHEN MATCHED AND source.status = 'update' THEN
 UPDATE SET
   target.email = source.email,
   target.status = source.status
```

First WHEN clause for merge

WHEN MATCHED AND source.status = 'update' THEN

If a row exists in both tables and the source indicates it's an update...

UPDATE SET - Updates the target table with new values from the source:

- email
- status

Ingesting into Existing Delta Tables

source_table

id	users	email	status
1	peter	peter@email	delete
2	zebi	zebi@email	update
3	samarth	samarth@email	new

Updated target_table

id	users	email	status
~~1~~	~~peter~~	~~peter@email~~	~~current~~
2	zebi	zebi@email	update
4	matt	matt@email	current

```
MERGE INTO target_table target
USING source_table source
ON target.id = source.id
WHEN MATCHED AND source.status = 'update' THEN
 UPDATE SET
   target.email = source.email,
   target.status = source.status
WHEN MATCHED AND source.status = 'delete' THEN
 DELETE
```

Second WHEN clause for merge

WHEN MATCHED AND source.status = 'delete' THEN DELETE

If the row exists in both tables and the source says it should be deleted, it’s removed from the target.

Ingesting into Existing Delta Tables

source_table

id	users	email	status
1	peter	peter@email	delete
2	zebi	zebi@email	update
3	samarth	samarth@email	new

Updated target_table

id	users	email	status
2	zebi	zebi@email	update
4	matt	matt@email	current
3	samarth	samarth@email	new

```
MERGE INTO target_table target
USING source_table source
ON target.id = source.id
WHEN MATCHED AND source.status = 'update' THEN
 UPDATE SET
   target.email = source.email,
   target.status = source.status
WHEN MATCHED AND source.status = 'delete' THEN
 DELETE
WHEN NOT MATCHED THEN
 INSERT (id, first_name, email, sign_up_date,
status)
 VALUES (source.id, source.first_name,
source.email, source.sign_up_date,
source.status);
```

WHEN NOT MATCHED clause for merge

If the row exists in the source but not in the target...

INSERT (...) VALUES (...) - A new row is inserted into the target table with values from the source.

Ingesting into Existing Delta Tables

source_table

id	users	email	status
1	peter	peter@email	delete
2	zebi	zebi@email	update
3	samarth	samarth@email	new

Fully Updated target_table

id	users	email	status
2	zebi	zebi@email	current
4	matt	matt@email	current
3	samarth	samarth@email	new

```
MERGE INTO target_table target
USING source_table source
ON target.id = source.id
WHEN MATCHED AND source.status = 'update' THEN
 UPDATE SET
   target.email = source.email,
   target.status = source.status
WHEN MATCHED AND source.status = 'delete' THEN
 DELETE
WHEN NOT MATCHED THEN
 INSERT (id, first_name, email, sign_up_date,
status)
 VALUES (source.id, source.first_name,
source.email, source.sign_up_date,
source.status);
```


This single MERGE INTO command:

- Updates rows when status = 'update'
- Deletes rows when status = 'delete'
- Inserts new rows not present in the target

Perfect for incremental data loads, CDC, or SCD Type 1 patterns.

Thank you for completing this lesson and continuing your journey to develop your skills with us.