

What is Full Load?

Full load refers to the process of completely replacing the target data with the new, transformed data on every ETL run.

In a full load scenario, all the data from the source system is extracted, transformed to meet the target system's requirements, and then loaded into the target system. This approach is commonly used when the source data has changed significantly or when there is a need to reset the target data to a specific point in time.

Disadvantage: While a full load may ensure that the target data is always consistent with the source data, it can be time-consuming and resource-intensive because all data is reloaded on every ETL run, regardless of whether it has changed or not. Additionally, a full load may cause data loss if any incremental changes made to the target data between ETL runs are overwritten.

What is incremental Load?

Incremental load refers to the process of updating only the changed or new data in the target system, rather than reloading all the data on every ETL run.

In an incremental load scenario, the first ETL run load all data like full load, and then the ETL process compares the source data with the target data to identify any changes or new data that need to be loaded into the target system. This approach can save time and resources by only processing the data that has changed or is new, rather than reloading all the data.

Advantage: The incremental load can be particularly useful in situations where the source data changes frequently or the data volumes are large. By updating only the changed or new data, the ETL process can reduce the amount of processing time and minimize the risk of data loss or inconsistency.

Mechanism: However, incremental load requires a mechanism to identify the incremental changes made to the source data, which can be challenging for some data sources. Additionally, care must be taken to ensure that the ETL process is designed and tested to accurately identify and process incremental changes to avoid data errors or inconsistencies.

Some mechanism are following

- Control table with ID: Here control table will look which last id in table is updated by when even next ETL runs it will match id from source table and only fill the columns after that ID

Name: Ahsan Bilal

- Control table with time stamp: Here control table will look for last time stamp and then it will match the timestamps in the source table if any time stamp is bigger then last time stamp saved in control table it will update that row and change its time stamp to that.

What is Historical load?

Historical load refers to the process of loading historical data into a target system. Historical load is often used in data warehousing and business intelligence applications, where it is important to analyze historical data to identify trends, patterns, and insights. Historical load involves extracting data from one or more source systems, transforming it into the desired format, and then loading it into the target system for analysis.

The historical data loaded into the target system can be used to create reports, dashboards, and other analytical tools to help business users make informed decisions based on past performance. Historical load may also be used to create historical baselines for comparison with current or future data, to identify changes or anomalies in data patterns, and to forecast future performance based on historical trends.

Historical load can be a resource-intensive process, particularly for large or complex data sets. Careful planning and execution are required to ensure that the ETL process is designed to efficiently handle the data volumes and complexity, while also ensuring data accuracy and consistency.

Date: 23/3/2023