**Discover Snowflake Database Design Standards and Procedures**

Contents

[Purpose 2](#_Toc526758494)

[Snowflake Table Design Standards 2](#_Toc526758495)

[Data Migration – Backfill Considerations 2](#_Toc526758496)

[DBA Checkpoint – Post Backfill Migration 2](#_Toc526758497)

[Snowflake Data Optimization Features 3](#_Toc526758498)

[S3 Micro-Partitions 3](#_Toc526758499)

[Query Optimization 3](#_Toc526758500)

# Purpose

The purpose of the document to outline Snowflake database features and provide standards for optimal database design, settings for load jobs and outline procedures to effectively manage data with cost saving and achieve optimal performance.

# Snowflake Table Design Standards

* Primary Key - All tables must have Primary Key constraint defined
* Clustering Keys - Clustering keys should be defined based on the Table access paths.
  + 1. Frequently used predicate columns are best candidates for Clustering Keys
    2. Ordering of Clustering Keys should be low cardinality column to higher. ( Eg. Perf\_Mthly\_Dt , Acct\_Key )
    3. For tables with frequent Join Conditions, Join columns should be taken into consideration as part of Clustering Keys.

Exception to Define Clustering Keys

* For tables size less than 10GB , clustering keys need not be defined.
* If the Clustering key is a single column and loading of data occurs naturally in that order, then Snowflake default behavior would takes care of pruning.

# Data Migration – Backfill Considerations

Data (usually Avro format) Files in S3 buckets are copied into Snowflake table using the copy command.

Following must be adhered part of the all COPY Commands

* Copy commands must specify the column names and values to avoid loading wrong values into wrong columns.
* Order of the columns is not maintained is not guaranteed in Snowflake.
* Load the data in the order of the prefix Clustering key column wherever possible.

# DBA Checkpoint – Post Backfill Migration

DBA checkpoint process is to determine and perform Recluster operations for tables necessary.

Depends on the size of the table and amount of un-sorted data, DBAs perform multiple Recluster operation or Duplicate table with Order by.

Procedures to run Reclustering and Duplicate table are documented outside of this document.

# Snowflake Data Optimization Features

The following topics are to provide overview of the Snowflake database features involved in effective database design.

## S3 Micro-Partitions

Persistent Data in Snowflake is stored in S3 and automatically divided into micro-partitions, which are contiguous units of storage. Each micro-partition contains between 50 MB and 500 MB of uncompressed data.

<https://docs.snowflake.net/manuals/user-guide/tables-micro-partitions.html>

Clustering Keys

This “clustering” is a key factor in query performance because table data that is not sorted or is only partially sorted may impact query performance, particularly on very large tables.

<https://docs.snowflake.net/manuals/user-guide/tables-micro-partitions.html#data-clustering>

## Query Optimization

Snowflake does not enforce Primary and Foreign Key constraints , but Snowflake query engine uses the information to generate an optimal execution plan.

Its strongly recommended to define primary key for all tables , unless all columns in the table constitute a unique row.