**BCBSRI-logo**

Snowflake Data Warehouse

Object Naming Standards

Version # : 1.0

Released on : 11/29/2022

Author : Ramesh Karnati, Sunil Rangineni

NTT Data

Blue Cross Blue Shield of Rhode Island Account

500 Exchange Street

Providence, RI 02903

NTT Data Confidential

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

2. Snowflake Naming Conventions 3

2.1 Snowflake Naming Conventions for Account-Level Objects 4

*2.1.1 User Level:* 4

*2.1.2 Role Level:* 4

*2.1.3 Warehouse Level:* 4

*2.1.4 Database Level:* 5

2.2 Snowflake Naming Conventions for Schema-Level Objects 5

2.3 Snowflake Naming Conventions Constraints 6

# Introduction

## Purpose

Snowflake object naming convention is a framework for naming objects in a way that describes what they contain and how they are used, following a standard naming convention helps to formalize expectations and to promote consistency with its users. It’s a good practice to introduce these naming conventions when starting a new implementation.

## Scope

# [Snowflake Naming Conventions](https://hevodata.com/learn/snowflake-naming-conventions/#t1)

Objects in Snowflake can be classified into account level objects and schema level objects. Here are some common account and schema level objects

Timeline

Description automatically generated

## Snowflake Naming Conventions for Account-Level Objects

Account Level Objects are the non-database objects, i.e., users, roles, warehouses, databases, etc. Commonly, we might be working with different environments in a single account. Hence, as a good practice in Snowflake Naming Conventions, we can add the environment prefixes, such as DEV, TEST, PROD, etc., to the object name as a differentiator. For instance, you can add DEV, PRD, and TEST for development, production, and testing environments. It is suggested to include prefixes for your database names.

### *User Level:*

A username is a unique identifier for a user. There are 2 types of accounts, namely the user account used by real, and the service account required by system services.

|  |  |  |  |
| --- | --- | --- | --- |
| **Account Type** | **Naming Convention** | **Description** | **Example** |
| User | [Email] | Email addresses within an organization | [user@company.com](mailto:user@company.com) |
| Service | [ENV]\_[PROJECT/PROGRAM] \_SA | Environment\_ project/program short code\_SA\_USER postfix | DEV\_PYTHON\_SA\_USER |

### *Role Level:*

**Roles** are privileges granted to users on securable objects that allows them to execute certain actions needed for business functions in your organization. For USER Roles, environment prefixes are generally not required as we will control access to multiple environments using grants. However, in cases of Service Roles, we would need prefixes so that access is not shared between environments.

|  |  |  |  |
| --- | --- | --- | --- |
| **Account Level Object** | **Naming Convention** | **Description** | **Example** |
| Role | [PROJECT/PROGRAM]\_ [ROLE\_NAME]\_ROLE | Project Name\_Role of user in the project\_ROLE postfix | EDR\_ANALYST\_ROLE EDR\_READONLY\_ROLE |

### *Warehouse Level:*

When creating a virtual warehouse with the desired resources such as CPU, memory, and temporary storage for performing several operations, we can go along with the following Snowflake Naming Conventions:

|  |  |  |  |
| --- | --- | --- | --- |
| **Account Level Object** | **Naming Convention** | **Description** | **Example** |
| Virtual Warehouse | WH\_[ENV]\_[ROLE\_NAME]\_ [WH\_SIZE] | WH\_Environment\_User Role\_Warehouse Size | WH\_DEV\_USERS\_S WH\_DEV\_ETL\_XL |

### *Database Level:*

A database is a logical grouping of schemas. Each database belongs to a single Snowflake account. We can follow the below Snowflake naming conventions for our database:

|  |  |  |  |
| --- | --- | --- | --- |
| **Account Level Object** | **Naming Convention** | **Description** | **Example** |
| Database | [ENV]\_[PROJECT/PROGRAM]  \_DB | Environment\_Project name\_DB postfix | EDR\_DEV\_DB DEV\_FDA\_DB TEST\_EDR\_DB |

## Snowflake Naming Conventions for Schema-Level Objects

When we name schema-level objects, we can leave out the environment prefix as there aren’t multiple environments inside a single database. A schema is a logical grouping of database objects (tables, views, etc.). Each schema belongs to a single database. As a standard, you can go with the following Snowflake Naming Conventions for schemas:

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Type** | **Naming Convention** | **Description** | **Example** |
| Schema | [DATA\_GROUPING] | Groupings by source name/table types | DEV\_EDR\_DB.REF DEV\_FDA\_DB.CLAIM TEST\_EDR\_DB.NONCLAIMS |

Below are the suggested naming conventions for schema-level objects:

|  |  |  |
| --- | --- | --- |
| **Object Type** | **Naming Convention** | **Example** |
| Table | [TABLE\_NAME] | MEMBER |
| View | VW\_[TABLE\_NAME] | VW\_MEMBER |
| File Format | [NAME]\_[FORMAT]\_FF | EDR\_CSV\_FF, EDR\_XML\_FF |
| Stage | [TABLE\_NAME]\_[TYPE]\_STAGE | MEMBER\_INT\_STGAGE, MEMBER\_BLOB\_STAGE |
| Snowpipe | [TABLE\_NAME]\_PIPE | MEMBER\_PIPE |
| Stored Procedure | [PROCESS\_NAME]\_SP | GLOBALCAP\_MEMBER\_SP |
| Stream | [TABLE\_NAME]\_STREAM | MEMBER\_STREAM |
| Task | [TABLE\_NAME]\_TASK | MEMBER\_TASK |

## Snowflake Naming Conventions Constraints