



**Snowflake**

**Procolombia - Synthetikus**

**02-08-2024**

# Etiquetas para objetos (Object tagging)

- ¿Cómo monitorear el consumo de créditos por concepto de cómputo, almacenamiento y transferencia de datos?

# Etiqueta

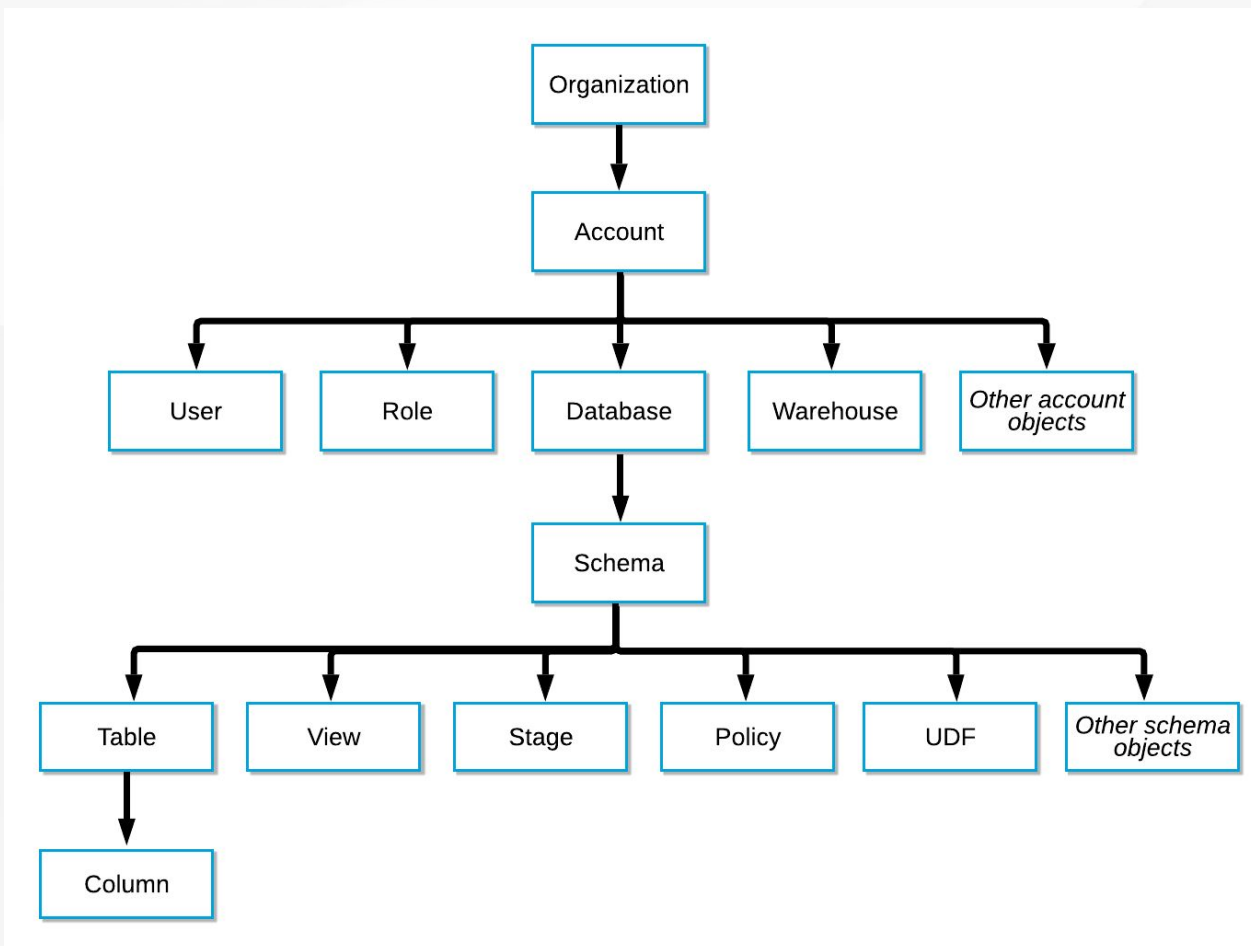
Es un objeto a **nivel de esquema** que es único y puede asignarse a otro objeto Snowflake. Consiste en un par **clave-valor** compuesto por una cadena (de hasta 256 caracteres, con la opción de especificar valores permitidos), además del objeto que representa a la etiqueta.

Sirven para agrupar y organizar recursos, para supervisar el uso de los objetos y facilitar las operaciones de gobernanza de datos, como la auditoría y la elaboración de informes.



Su uso **NO** acarrea costos.

# Objetos soportados y jerarquía



# Cuotas de etiquetas

- » **Tabla/vista:** Hasta 50 etiquetas únicas.
- » **Columnas:** Hasta 50 etiquetas únicas combinadas para todas las columnas de una tabla o vista.

Si alcanza el límite de etiquetas para una tabla y sus columnas, no se podrán añadir más etiquetas a menos que se eliminen algunas de las existentes (**DROP**). En caso de querer recuperarlas se puede usar el *statement* de **UNDROP**.

**Para el resto de los objetos de Snowflake:** se pueden establecer hasta 50 etiquetas únicas en un único objeto, con un máximo de 100 etiquetas especificadas en una única sentencia CREATE o ALTER.

# ¿Cómo crear y asignar una etiqueta?

1. Debido a que las etiquetas son objetos a **nivel de esquema**, lo primero que se debe hacer es crear una base de datos con un esquema en donde se puedan almacenar de manera organizada los tags que se vayan a crear.

```
--1. Centralized tagging db to store object tags  
CREATE OR REPLACE DATABASE TAGGING_DB;  
  
--2. Centralized schema within tagging db to store object tags  
CREATE OR REPLACE SCHEMA TAGGING_DB.TAGGING_SCH;
```

1. Para que un usuario pueda crear y asignar etiquetas es necesario que tenga el privilegio **CREATE TAG**, además de los globales otorgados al aplicar el *statement* de **APPLY TAG**. Por lo general, se toma un **enfoque centralizado** en el cual se establece un **rol personalizado**, como por ej. tag\_admin, a quién se le asignan los anteriores permisos. Posteriormente, dicho rol es dado al usuario que será responsable de crear y asignar etiquetas a los objetos Snowflake.

```
USE ROLE USERADMIN;  
CREATE ROLE tag_admin;  
USE ROLE ACCOUNTADMIN;  
GRANT CREATE TAG ON SCHEMA mydb.mysch TO ROLE tag_admin;  
GRANT APPLY TAG ON ACCOUNT TO ROLE tag_admin;
```

2. Conceder el rol personalizado tag\_admin a un usuario que actúa como administrador de etiquetas.

```
USE ROLE USERADMIN;
GRANT ROLE tag_admin TO USER jsmith;
```

3. Ejecutar una sentencia CREATE TAG para crear una etiqueta.

```
USE ROLE tag_admin;
USE SCHEMA mvdb.mvsch;
CREATE TAG departments ;
```

```
create tag departments
  allowed_values 'finance', 'engineering';
```

4. Asignar la etiqueta, se puede por interfaz (se puede crear también) o a través de comandos.


Navigate to the desired table, view, or column using the object explorer (i.e. **Data » Databases**).

Select the **More** menu (i.e. **...**) » **Edit**, and select **+ Tag**. Follow the prompts to manage the tag assignment.

```
USE ROLE tag_admin;
CREATE WAREHOUSE mywarehouse WITH TAG ( departments 'sales');

USE ROLE tag_admin;
ALTER WAREHOUSE wh1 SET TAG departments = 'sales';
```

[See Tag DDL statements](#)



+ Create

Home

Search

Projects

Data

Data Products

AI & ML

Monitoring

Query History

Copy History

Task History

Dynamic Tables

Trust Center

Governance

### Governance

Dashboard Tagged Objects

Data was refreshed 10 hours ago • Only directly assigned tags are visible

1

Tagged table

0

Tables with a row access policy

0

Tagged columns

0

Columns with a masking policy

Coverage of 20 tables

95%

Tags & policies

Tags only

Policies only

No tags & policies

Coverage of 184 columns

100%

Most Used Tags for Tables

COST\_CENTER

1

### Governance

Dashboard Tagged Objects

All All

Tables

All Tags

All Policy

20 tables

NAME ↑		TAGS ⓘ
ALERT_HISTORY		MON —
CONFIGURATION_DEFA...	SNOWFLAKE	COST_CENTER... —
CONFIGURATION_LOG	SNOWFLAKE	PRIVACY_CATEGORY... —
CONFIGURATION_REGIS...	SNOWFLAKE	SEMANTIC_CATEGORY... —
CONFIGURATION_TABLE	SNOWFLAKE	TRUST_CENTER... —

Tags

All Tags

Has Tags

No Tags

COST\_CENTER

PRIVACY\_CATEGORY

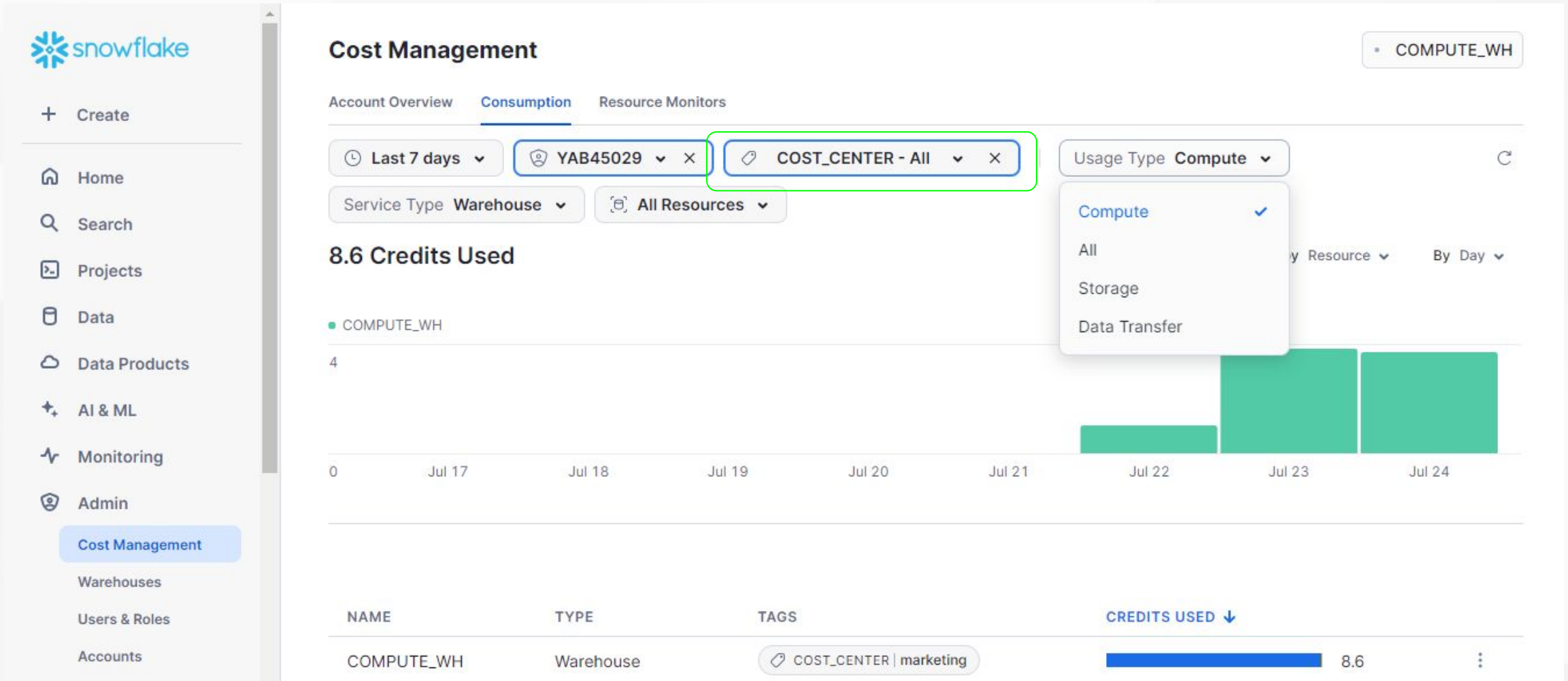
SEMANTIC\_CATEGORY

Cancel Apply



# ¿Cómo ayudan a determinar el consumo?

quantil



# Etiquetas para consultas (Query Tags)

- ¿Cómo llevar un seguimiento de las consultas que se realizan?

# Etiquetas de consulta (Query tags)

Son parámetros de Snowflake que pueden configurarse a **nivel de cuenta, usuario o incluso sesión** en Snowflake. Estas permiten a los usuarios asignar una etiqueta de cadena descriptiva de hasta 2000 caracteres de longitud (y que contenga cualquier carácter) a cualquier consulta que ejecuten.

A partir de ellas, los usuarios pueden nombrar fácilmente las consultas, agruparlas y realizar un mejor seguimiento del rendimiento de las mismas.

# ¿Cómo asignarlas?

## A nivel de cuenta

```
ALTER ACCOUNT DYZKCFJ.YAB45029 SET query_tag = 'HighPriority';  
ALTER ACCOUNT DYZKCFJ.YAB45029 UNSET QUERY_TAG;
```

**Note:** Only an **ACCOUNTADMIN** can set an account-level query tag.

## A nivel de sesión

```
ALTER SESSION SET QUERY_TAG = 'selectAllOrders';  
ALTER SESSION UNSET QUERY_TAG;
```

**Note:** Session-level Snowflake query tags are unique to the current session and replace user-level Snowflake query tags. So, by utilizing session-level Snowflake query tags, you enhance your ability to efficiently identify queries related to specific tasks or issues, which can be incredibly valuable when debugging queries or any issue.

## A nivel de usuario

```
ALTER USER Pramit SET QUERY_TAG = 'TeamA';  
ALTER USER Preeti SET QUERY_TAG = 'TeamB';  
ALTER USER Happy SET QUERY_TAG = 'TeamC';  
ALTER USER Alice SET QUERY_TAG = 'TeamD';  
ALTER USER John SET QUERY_TAG = 'TeamE';
```

```
ALTER USER Pramit UNSET QUERY_TAG;
```



+ Create

Home

Search

Projects

Data

Data Products

AI & ML

Monitoring

Query History

Copy History

Task History

Dynamic Tables

Trust Center

Governance

## Query History

Status All

User All

Last 7 days

Filters

549 Queries

Columns

Columns

SQL TEXT

SELECT system\$GET\_NPS\_FE

drop USER IDENTIFIER('K

SELECT system\$GET\_NPS\_FE

SELECT system\$GET\_NPS\_FE

drop RESOURCE MONITOR ID

SHOW RESOURCE MONITORS L

ALTER RESOURCE MONITOR R

SHOW RESOURCE MONITORS L

ALTER RESOURCE MONITOR R

SHOW RESOURCE MONITORS L

SHOW RESOURCE MONITORS L

SQL Text

Query ID

Warehouse

Statement Type

Duration

Session ID

Query Tag

Enter query tag

Client-generated statements

Queries executed by user tasks

Clear all

Apply Filters

STATUS

USER

WAREHOUSE

DURATION

Success

KLOPEZ

—

69ms

Success

KLOPEZ

—

346ms

Success

KLOPEZ

—

44ms

Success

KLOPEZ

—

68ms

Success

KLOPEZ

—

65ms

Success

KLOPEZ

—

52ms

Success

KLOPEZ

—

59ms

Success

KLOPEZ

—

64ms

Success

KLOPEZ

—

57ms

Success

KLOPEZ

—

52ms

Success

KLOPEZ

—

51ms