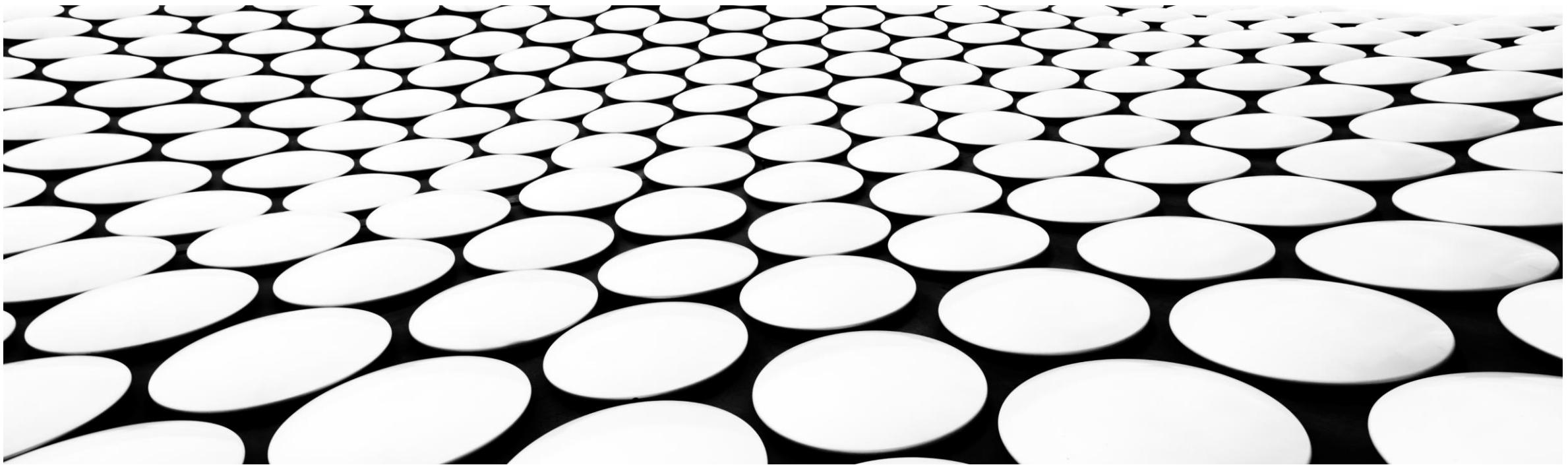




SNOWFLAKE

CLOUD DATA WAREHOUSE



WHAT IS SNOWFLAKE?



Built for the Cloud

Built from scratch, optimized for cloud, storage & compute decoupled



Software as a Service

No software, infrastructure or upgrades to manage



Pay only for used compute & storage

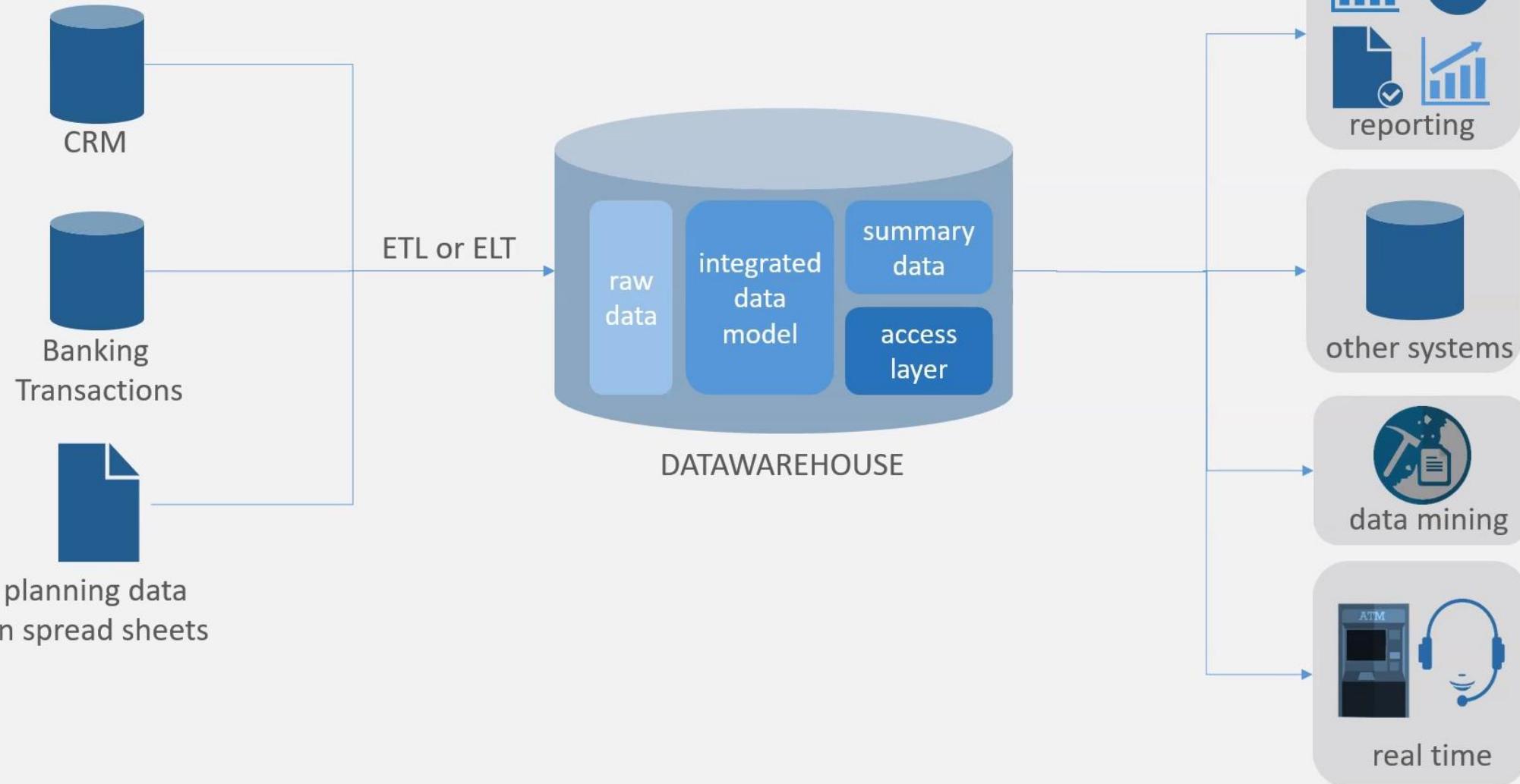
Storage & Compute charged independently, only for use



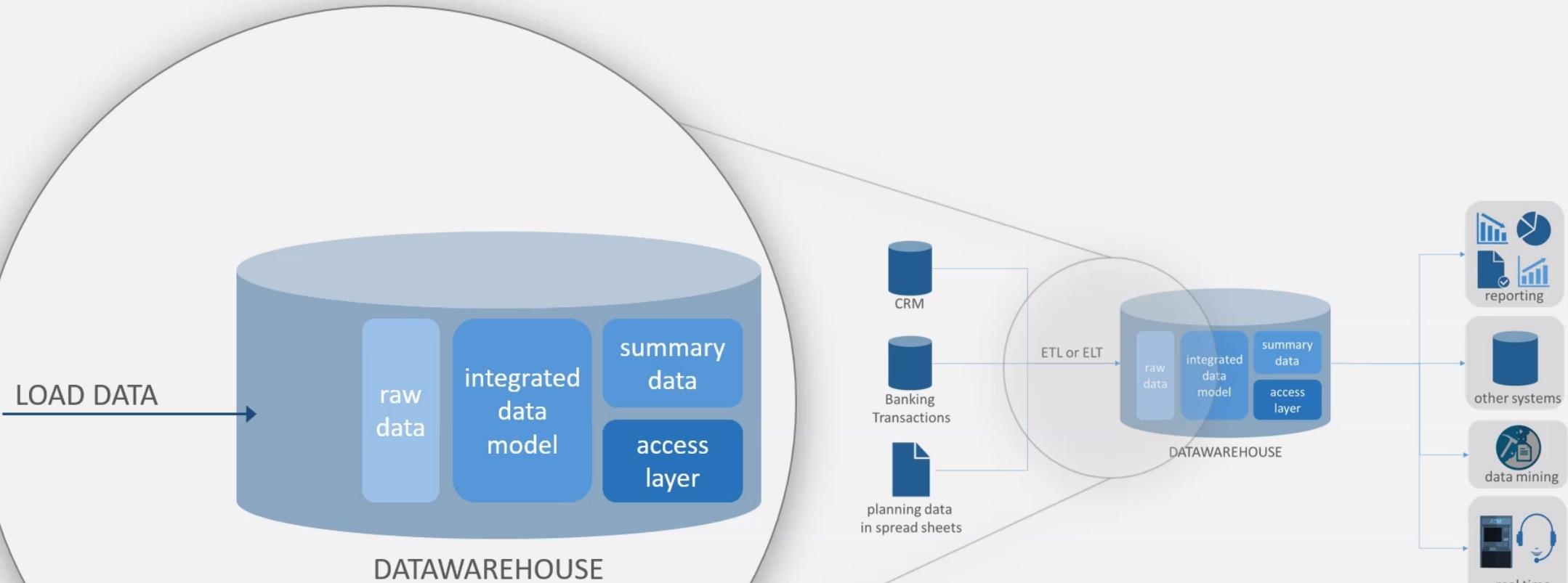
Scalable

Virtual warehouse enable compute scaling

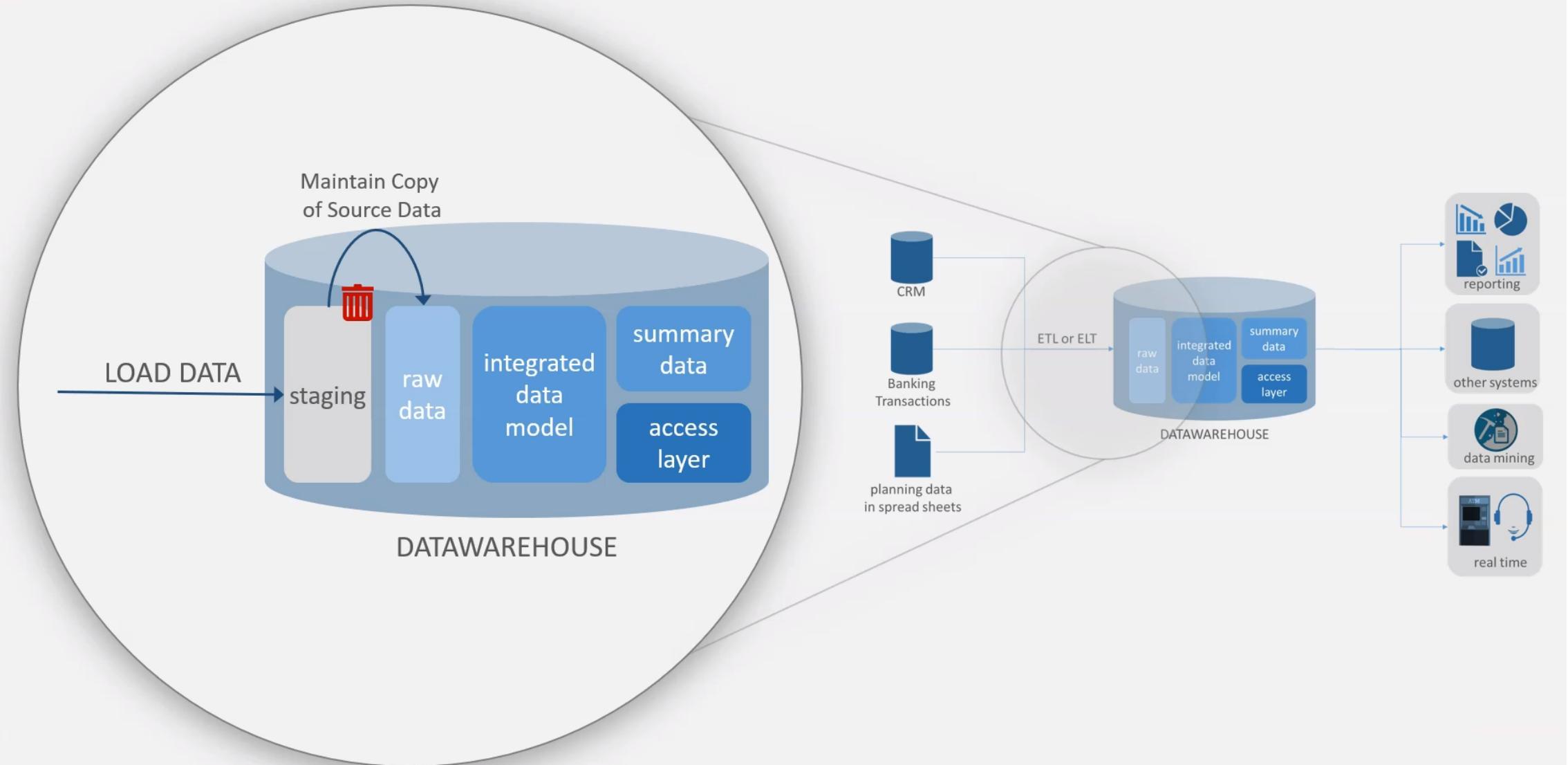
What is a data warehouse?



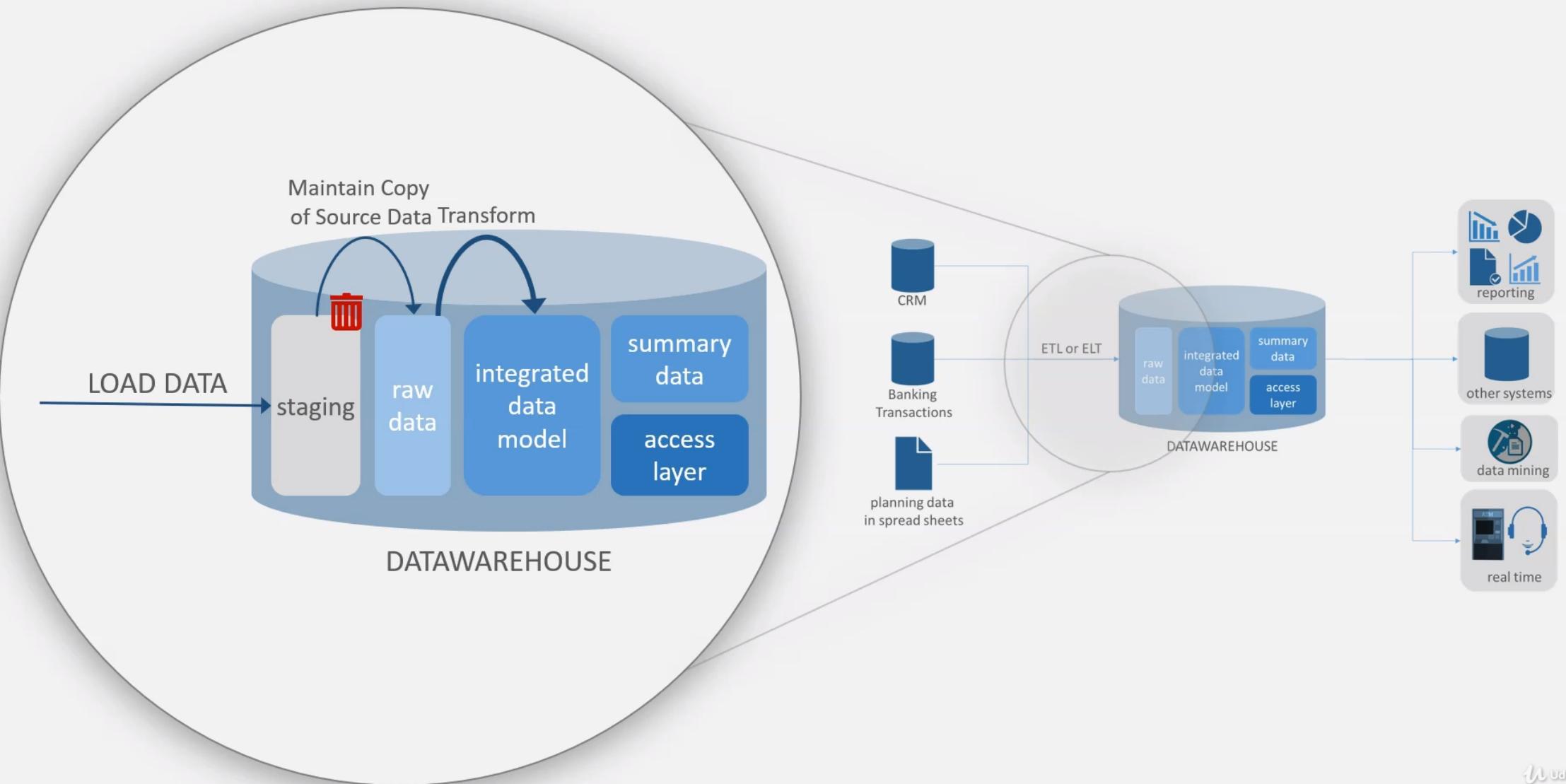
What is a data warehouse?



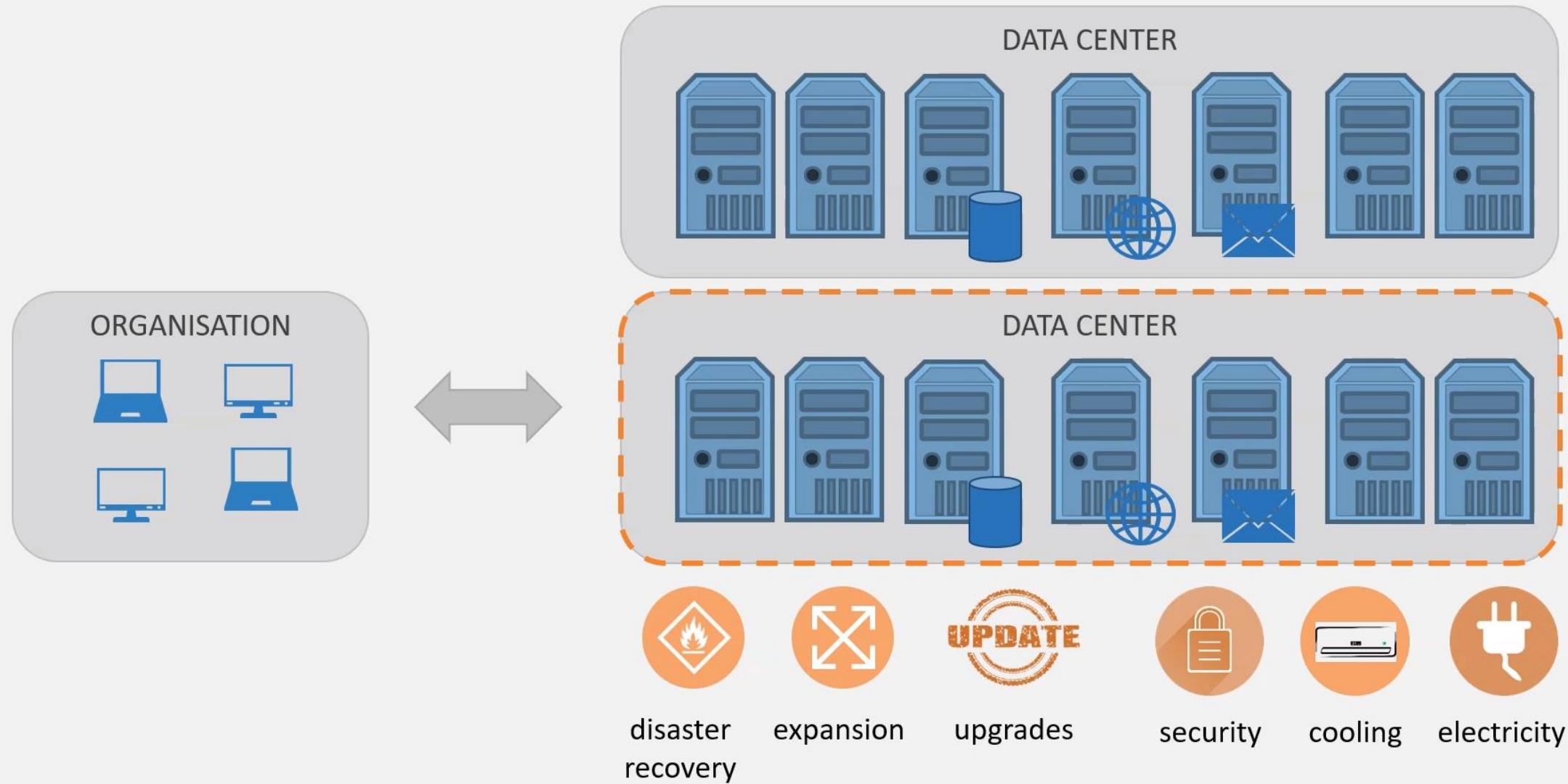
What is a data warehouse?



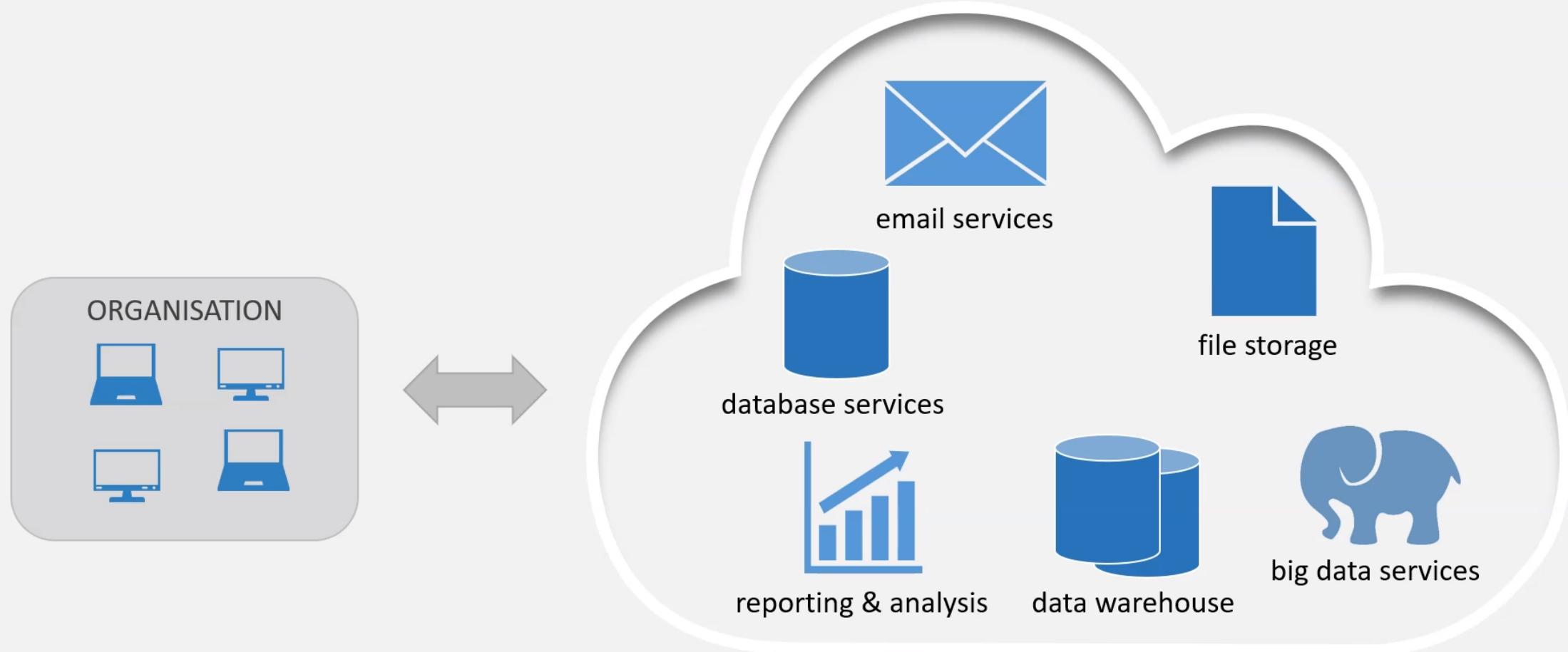
What is a data warehouse?



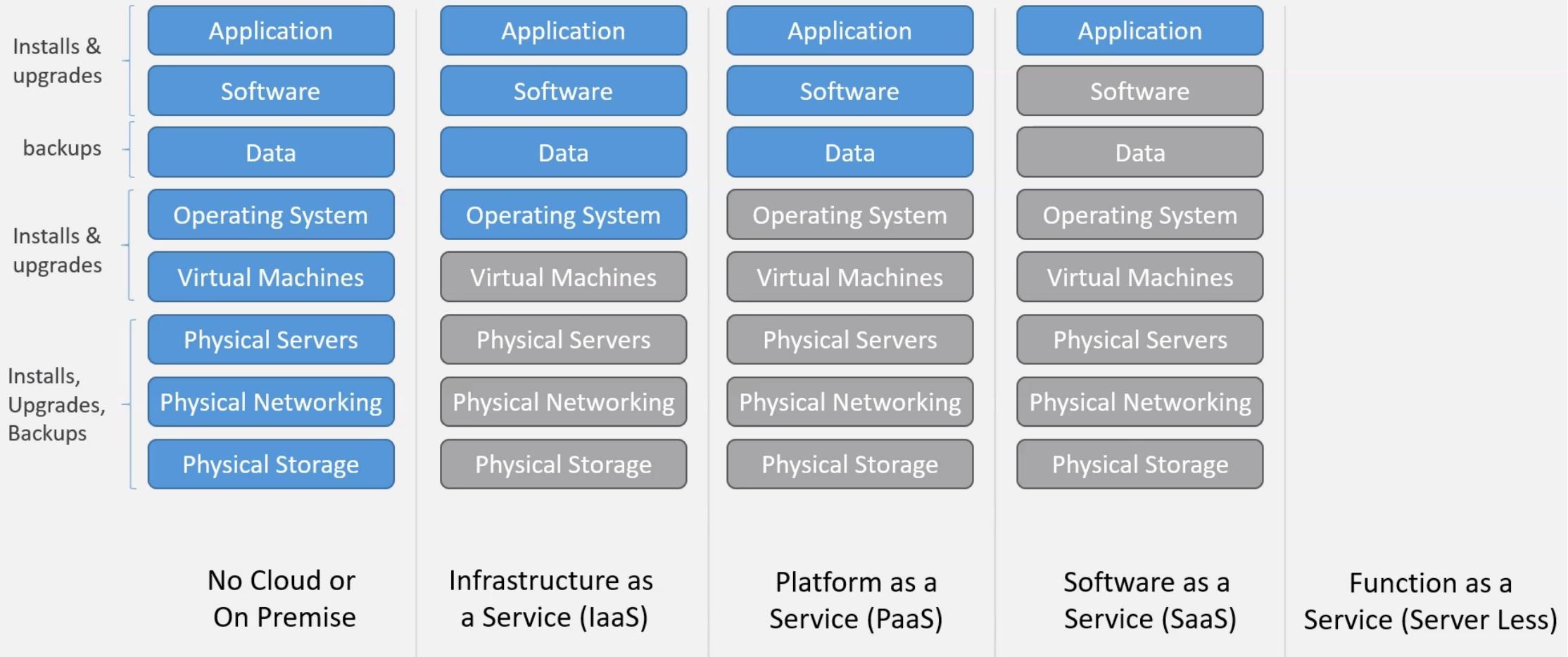
Why cloud (computing)?



Why cloud (computing)?



Cloud – software approaches

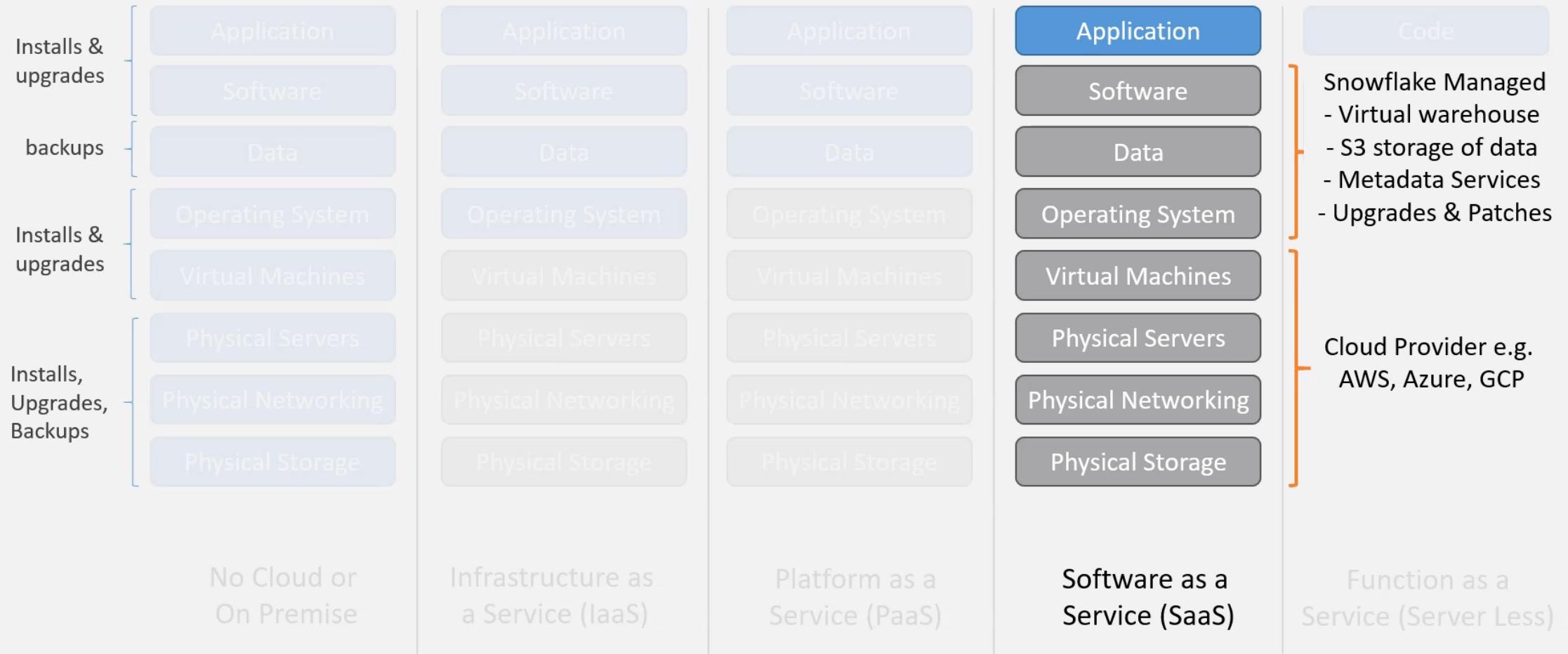


Cloud Provider Manages

You Manage



Cloud – software approaches

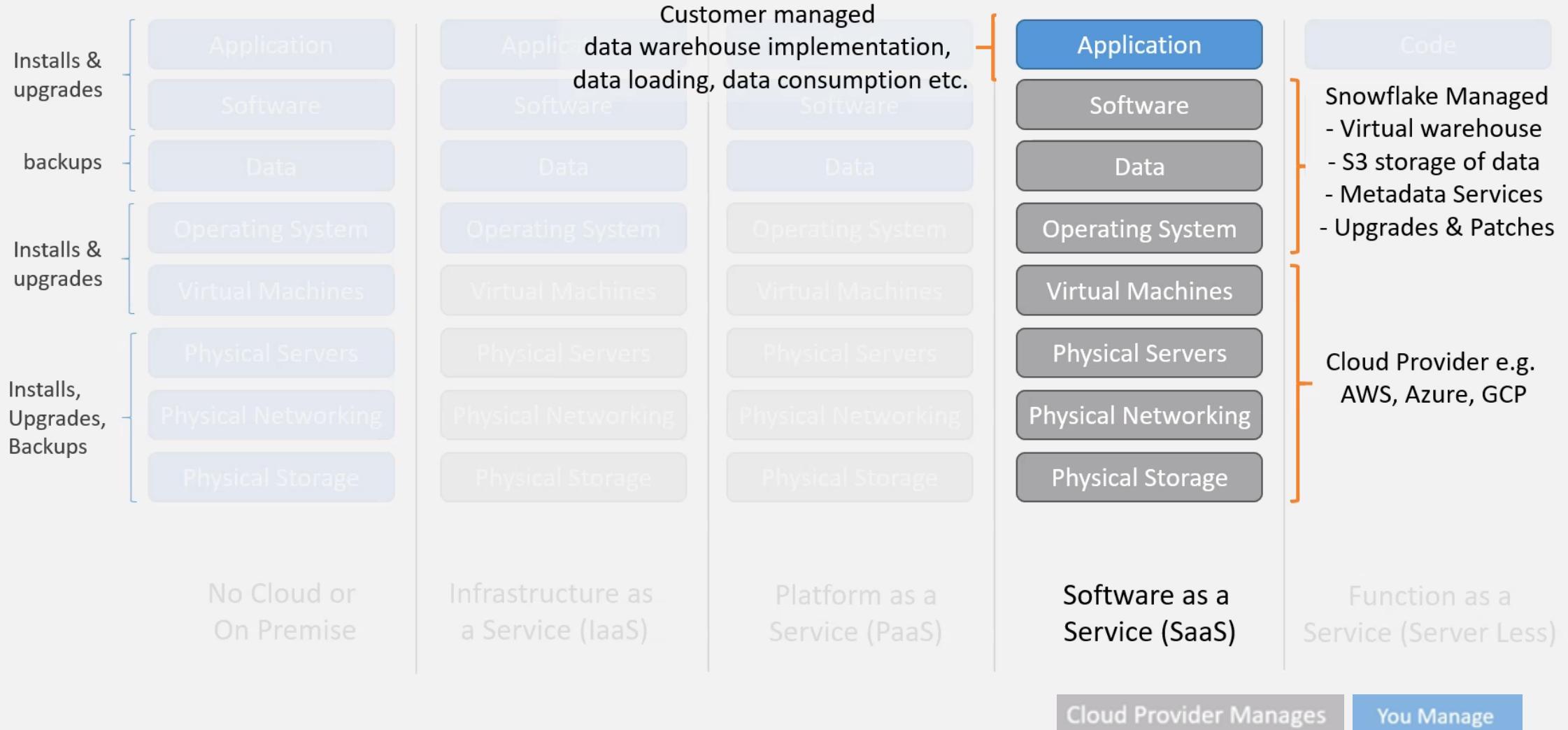


Cloud Provider Manages

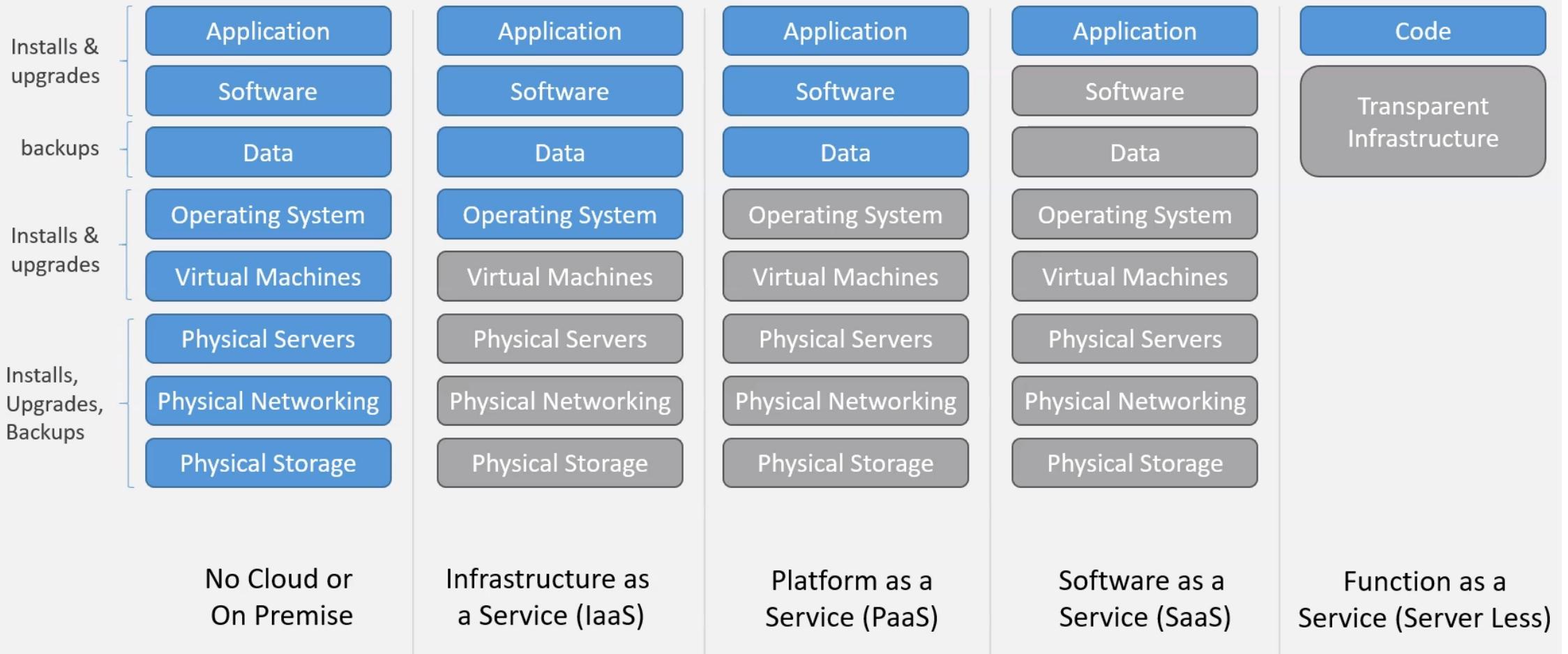
You Manage



Cloud – software approaches



Cloud – software approaches



Cloud Provider Manages

You Manage



SETTING UP SNOWFLAKE ACCOUNT

LET'S CREATE A FREE TIER SNOWFLAKE ACCOUNT





START YOUR 30-DAY FREE TRIAL

- Gain immediate access to the Data Cloud
- Enable your most critical data workloads
- Scale instantly, elastically, and near-infinitely across public clouds
- Snowflake is HIPAA, PCI DSS, SOC 1 and SOC 2 Type 2 compliant, and FedRAMP Authorized



Start your 30-day free Snowflake trial which includes \$400 worth of free usage

ucsc ✓
snowflake ✓
ucscsnowflake@yahoo.com ✓
ucsc ✓
Sri Lanka ▾

No, I do NOT want Snowflake to send me e-mails about products, services, and events that it thinks may interest me.

By clicking the button below you understand that Snowflake will process your personal information in accordance with its [Privacy Notice](#)

CONTINUE

or sign in to an existing account

Start your 30-day free Snowflake trial which includes \$400 worth of free usage

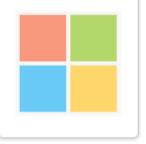
Choose your Snowflake edition*

Standard
A strong balance between features, level of support, and cost.

Enterprise
Standard plus 90-day time travel, multi-cluster warehouses, and materialized views.

Business Critical
Enterprise plus enhanced security, data protection, and database failover/fallback.

Choose your cloud provider*

 Microsoft Azure

 Amazon Web Services

 Google Cloud Platform

Asia Pacific (Mumbai) 

Check here to indicate that you have read and agree to the terms of the [Snowflake Self Service On Demand Terms](#).

GET STARTED

Start your 30-day free Snowflake trial which includes \$400 worth of free usage

YOU'RE NOW SIGNED UP! 

An email to activate your account has been sent to ucscsnowflake@yahoo.com (it may take a few minutes to arrive).

Why did you signup for a Snowflake account today?

Company is considering Snowflake

Virtual hands-on lab or demo

Training or certification

Attending an in-person event

Personal learning and development

Other [Tell us about your use case](#)

Submit

Start your 30-day free Snowflake trial which includes \$400 worth of free usage

YOU'RE NOW SIGNED UP! 

An email to activate your account has been sent to ucscsnowflake@yahoo.com (it may take a few minutes to arrive).

Thank you for your response!

➤ **GETTING STARTED VIDEO**
A quick tour of Snowflake's Platform.
[Watch Video](#)

➤ **VIRTUAL HANDS-ON LAB**
Instructor-led, online lab guiding you through key product features.
[Reserve Your Seat](#)

➤ **FULL SNOWFLAKE DOCUMENTATION**
A comprehensive document covering all aspects of Snowflake's Platform.
[Go To Documentation](#)



Hi ucsc,

Congratulations on taking the first step to become a data-driven organization by signing up for Snowflake. Click the button below to activate your account.

CLICK TO ACTIVATE

Please note, your activation link is temporary and will expire in 72 hours. Once you activate your account, you can access it at <https://QW84992.ap-south-1.aws.snowflakecomputing.com/console/login>.

Be sure to bookmark your login link to easily access your account going forward. If you experience any problems logging into your account or you forgot your username or password, please contact support@snowflake.com.

Best regards,

The Snowflake team

Welcome to Snowflake!

ucsc snowflake, please choose a username and password to get started

Username

Username can contain only letters and numbers.

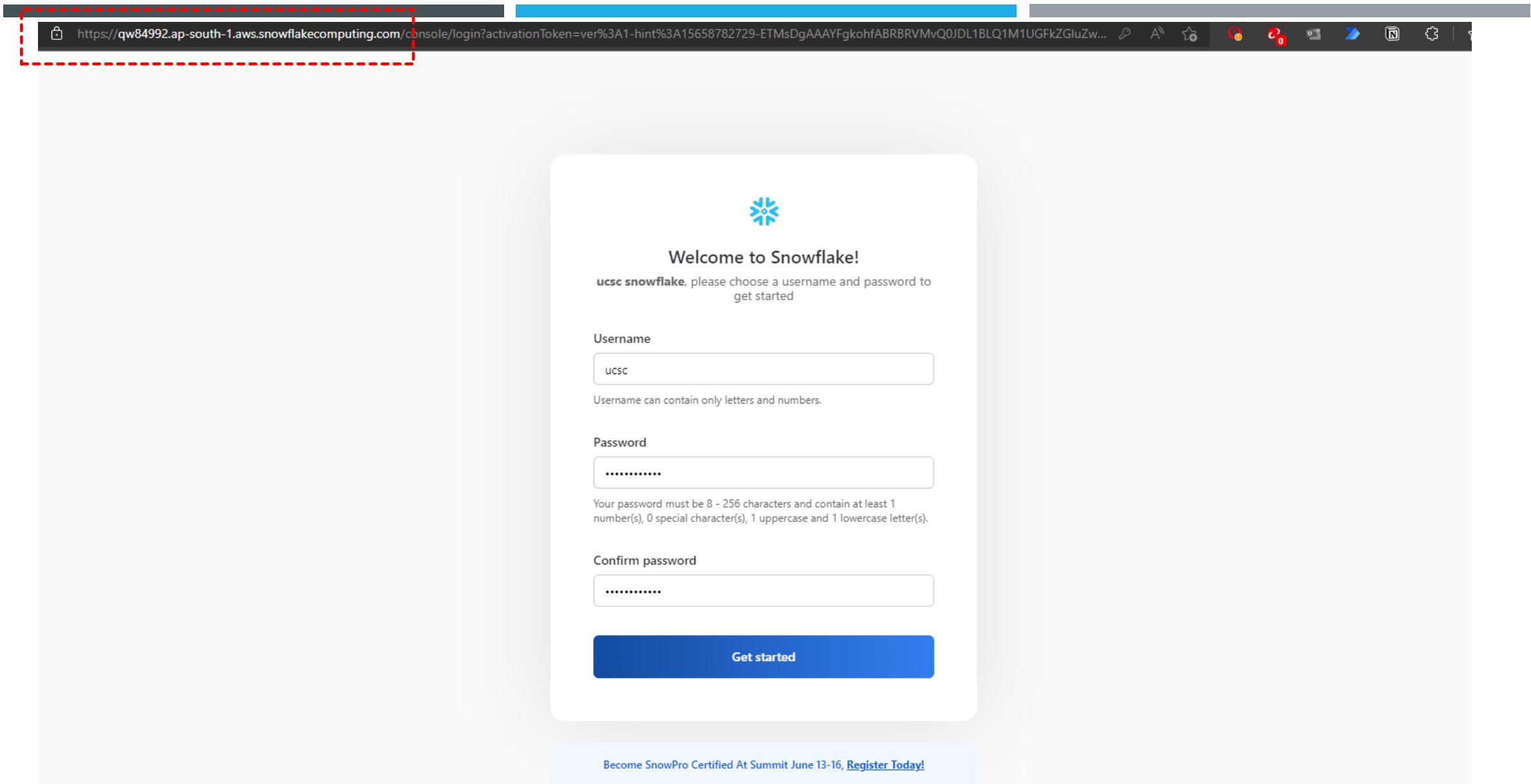
Password

Your password must be 8 - 256 characters and contain at least 1 number(s), 0 special character(s), 1 uppercase and 1 lowercase letter(s).

Confirm password

Get started

Become SnowPro Certified At Summit June 13-16, [Register Today!](#)





Sign in to Snowflake

Username
UCSC

Password
.....

Sign in

Become SnowPro Certified At Summit June 13-16, [Register Today!](#)

← → 🔍 https://app.snowflake.com/learn

ucsc snowflake ACCOUNTADMIN

- Worksheets
- Dashboards
- Data
- Marketplace
- Activity
- Admin
- Help & Support
- Learn**
- Support

Welcome to our new navigation!

We've made the features you need most easier to find. Visit **Activity** to see recent queries and copies; go to **Admin** to manage usage, warehouses, permissions, and more; explore the **Marketplace** or get **Help** in one click.

Got it!

Learn

 **Snowflake Trial**
Understand your Snowflake Trial

 **Snowflake Data Marketplace**
Explore immediately queryable data in the Marketplace

Foundations

Snowflake 101

Learn the basics of Snowflake, from terminology to concepts

 Snowflake Architecture Key concepts and architecture	 Warehouses Snowflake's parallel compute technology used for queries
 Credits Understand how Snowflake calculates your usage costs	 Key Features Snowflake's differentiators and key features
 Users Understand how to create and manage users	 Access Controls Understand how to manage access for users and roles
 Snowflake Editions Understand the differences between Snowflake editions	 SnowSQL Access Snowflake via CLI using SnowSQL
 Setup Cost Controls Hands on guides to understand cost controls	

Loading Data



SNOWFLAKE WALKTHROUGH OF CONSOLE

LET'S DIVE DEEP INTO ARCHITECTURE OF SNOWFLAKE PLATFORM



New Console - Worksheets

The screenshot shows the Snowflake New Console interface for managing worksheets. The top navigation bar includes tabs for 'Recent', 'Shared with me', 'My Worksheets', and 'Folders'. A search bar and a 'Worksheet' creation button are also present. The main content area displays a table of worksheets, sorted by viewed status (descending). The table columns are 'TITLE', 'VIEWED ↓', 'UPDATED', and 'ROLE'. The 'VIEWED ↓' column is highlighted with a blue arrow icon. The 'UPDATED' column is also sorted, indicated by a blue arrow icon. The 'ROLE' column shows all entries as 'ACCOUNTADMIN'. The table lists seven worksheets, all created on '2022-06-14 2:48pm' and last updated '1 day ago'. The worksheets are titled: '2022-06-14 2:48pm', 'Tutorials', 'Tutorial 1: Sample queries on TPC-H data', 'Tutorial 2: Sample queries on TPC-DS data', 'Tutorial 3: TPC-DS 10TB Complete Query Test', and 'Tutorial 4: TPC-DS 100TB Complete Query Test'. A sidebar on the left contains links for 'Dashboards', 'Data', 'Marketplace', 'Activity', 'Admin', 'Help & Support', and 'Classic Console'. A trial reminder box indicates '45 days left in trial' with an 'Upgrade' button. The bottom navigation bar includes icons for account management and session details.

TITLE	VIEWED ↓	UPDATED	ROLE
2022-06-14 2:48pm	1 day ago	1 day ago	ACCOUNTADMIN
Tutorials	1 day ago	1 day ago	ACCOUNTADMIN
Tutorial 1: Sample queries on TPC-H data	1 day ago	1 day ago	ACCOUNTADMIN
Tutorial 2: Sample queries on TPC-DS data	1 day ago	1 day ago	ACCOUNTADMIN
Tutorial 3: TPC-DS 10TB Complete Query Test	1 day ago	1 day ago	ACCOUNTADMIN
Tutorial 4: TPC-DS 100TB Complete Query Test	1 day ago	1 day ago	ACCOUNTADMIN

Classic Web Console – Worksheets

Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions.

New Worksheet

Find database objects Starting with...

Run All Queries Saved 6 seconds ago

ACCOUNTADMIN ENGINEERINGWH (M) Select Database Select Schema

Databases Shares Marketplace Warehouses Worksheets History Account

Partner Connect Help Notifications Snowsight UCSC ACCOUNTADMIN

New Worksheet

1 CREATE WAREHOUSE EXERCISE_WH
2 WAREHOUSE_SIZE = XSMALL
3 AUTO_SUSPEND = 600 -- automatically suspend the virtual warehouse after 10 minutes of not being used
4 AUTO_RESUME = TRUE
5 COMMENT = 'This is a virtual warehouse of size X-SMALL that can be used to process queries.';
6
7
8 SELECT * FROM "SNOWFLAKE_SAMPLE_DATA"."TPCH_SF1"."CUSTOMER";

Results Data Preview Open History

Query_ID SQL 2.64s 150,000 rows

Filter result... Copy Columns

Row	C_CUSTKEY	C_NAME	C_ADDRESS	C_NATIONKEY	C_PHONE	C_ACCTBAL	C_MKTSEGMENT	C_COMMENT
1	30001	Customer#000030001	Ui1b,3Q71CILTJn4MbVp,,YC...	4	14-526-204-4500	8848.47	MACHINERY	frays wake blithely enticingly...
2	30002	Customer#000030002	UVBoMtILkQu1J3v	11	21-340-653-9800	5221.81	MACHINERY	he slyly ironic pinto beans w...
3	30003	Customer#000030003	CuG19fwKn8JdR	21	31-757-493-7525	3014.89	BUILDING	e furiously alongside of the r...
4	30004	Customer#000030004	tkR93ReOnf9zYeO	23	33-870-136-4375	3308.55	AUTOMOBILE	ssly bold deposits. final req
5	30005	Customer#000030005	pvg4uD0D8pEwpAE01aesCt...	9	19-144-468-5416	-278.54	MACHINERY	ructions behind the pinto be...
6	30006	Customer#000030006	eFynZc,XeoKmPMZ	18	28-754-658-5648	2427.85	HOUSEHOLD	haggle after the furiously reg...
7	30007	Customer#000030007	H,Q8PvzCrZor	5	15-383-236-6110	3912.67	FURNITURE	nod fluffy final requests. reg...
8	30008	Customer#000030008	ZCVPLnDjfo9r8Hr0eeoyXxW...	5	15-325-560-7160	248.26	MACHINERY	carefully final requests detec...
9	30009	Customer#000030009	aEeWtUSm2lyKH46TIP1X,C8...	12	22-682-831-9594	1471.93	FURNITURE	ly bold packages about the s...
10	30010	Customer#000030010	m1XDhp3FIQBxmEFxujEp3r...	21	31-529-816-9872	8599.71	BUILDING	. accounts cajole after the sl...
11	30011	Customer#000030011	Y5Mj4Jrg1zies2v3iC1Vh2NN...	1	11-808-821-2617	4442.02	HOUSEHOLD	g furiously even, ironic theod...
12	30012	Customer#000030012	8Vz0kDL0lvolAboX,eU9 RRhl...	3	13-153-555-8557	9027.69	HOUSEHOLD	lithely: carefully silent packa...
13	30013	Customer#000030013	wePhjLpaFGyyK	4	14-929-544-3265	3438.09	MACHINERY	ntiments. even excuses hagg...
14	30014	Customer#000030014	nP,zOdCccNjc84xPG4N4dG...	24	34-560-171-5542	-970.23	HOUSEHOLD	olites sleep quickly furiously ...
15	30015	Customer#000030015	NJW2MuymnqNToraET iW	5	1E-520-112-0124	1212.60	IDENTITY	u crucial packages are caref...

Account Details

https://app.snowflake.com/ap-south-1.aws/qw84992/worksheets

ucsc snowflake ACCOUNTADMIN

ucsc snowflake ACCOUNTADMIN

Switch Role > ACCOUNTADMIN

Profile

Partner Connect

Documentation ↗

Support ↗

Sign Out

Classic Console

45 days left in trial

Upgrade

Worksheets

Recent Shared with me My Worksheets Folders

TITLE

- 2022-06-14 2:48pm
- Tutorials
 - Tutorial 1: Sample queries on TPC-H data Tutorials
 - Tutorial 2: Sample queries on TPC-DS data Tutorials
 - Tutorial 3: TPC-DS 10TB Complete Query Test Tutorials
 - Tutorial 4: TPC-DS 100TB Complete Query Test Tutorials

ucsc snowflake ACCOUNTADMIN

Worksheets

Recent Shared with me My Work

TITLE

- 2022-06-14 2:48pm
- Tutorials
 - Tutorial 1: Sample queries on TPC-H data Tutorials
 - Tutorial 2: Sample queries on TPC-DS data Tutorials
 - Tutorial 3: TPC-DS 10TB Complete Query Test Tutorials
 - Tutorial 4: TPC-DS 100TB Complete Query Test Tutorials

Dashboards

Data

Marketplace

Activity

Admin

Help & Support

Classic Console

45 days left in trial

Upgrade

VLIZWXP

EW54843 - UCSC AWS - Asia Pacific (Mumbai)

Sign Into Another Account

EW54843

Admin Section – Usage Details

Screenshot of the Snowflake Admin Section – Usage Details page.

The URL is https://app.snowflake.com/ap-south-1.aws/qw84992/account/usage?preset=PRESET_LAST_7_DAYS&type=relative&relative=%7B%22tense%22%3A%22past%22%2C%22value%22%3A%227%22%2C%22unit%22%3A%22day%22%2C%22e...

The sidebar shows the user is logged in as **ucsc snowflake ACCOUNTADMIN**. The **Usage** tab is selected, highlighted with a blue background. Other tabs include Worksheets, Dashboards, Data, Marketplace, Activity, Admin, Usage (selected), Warehouses, Resource Monitors, Users & Roles, Security, Billing, Contacts, Accounts, Partner Connect, Help & Support, and Classic Console. A red dashed box highlights the Admin, Usage, Warehouses, Resource Monitors, Users & Roles, Security, Billing, Contacts, Accounts, Partner Connect, and Help & Support tabs.

The main content area displays the following information:

- 1.38 Spent** (Total cost)
- A bar chart showing spending by day from Jun 8 to Jun 14. The total cost is \$1.38.
- Spend per Account from 6/8/2022 to 6/14/2022**

ACCOUNT	REGION	SERVICE LEVEL	USAGE TREND	COST ↓
EW54843	AWS_AP_SOUTH_1	Enterprise		\$1.38

45 days left in trial (Upgrade button available)

Classic Web Console – History

Snowflake – The Complete Mast MS-700 Managing Microsoft Te Master DevOps Monitoring with History

https://qw84992.ap-south-1.aws.snowflakecomputing.com/console#/monitoring/queries

Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions.

Databases Shares Marketplace Warehouses Worksheets History Account Partner Connect Help Notifications Snowsight UCSC ACCOUNTADMIN

Last refreshed 12:44:11 PM Auto refresh

History Hide Filters View SQL Abort...

Display queries that meet all of the following criteria:

User is UCSC

Include client-generated statements
 Include queries executed by user tasks

Status	Query ID	SQL Text	User	Warehouse	Clust...	Size	Session ID	Start Time	End Time	Total Duration	Bytes Scanned	Client Info	Rows	Query Tag
✓	01a4f649-...	CREATE WAREHOUSE EngineeringWH WITH WAREHOUSE_SIZE = '10000000000'...	UCSC	COMPUTE_...			15658819605	6/15/2022, 4:19:...	6/15/2022, 4:19:...	105ms		✓ Snowflake UI 202206		
✓	01a4f647-...	ALTER WAREHOUSE "FINANCEWH" SUSPEND;	UCSC	SALESWH			15658819601	6/15/2022, 4:17:...	6/15/2022, 4:17:...	50ms		✓ Snowflake UI 202206		
✓	01a4f646-...	CREATE WAREHOUSE SalesWH WITH WAREHOUSE_SIZE = '5000000000'...	UCSC	FINANCEWH			15658819601	6/15/2022, 4:16:...	6/15/2022, 4:16:...	165ms		✓ Snowflake UI 202206		
✓	01a4f645-...	CREATE WAREHOUSE FinanceWH WITH WAREHOUSE_SIZE = '10000000000'...	UCSC	COMPUTE_...			15658819601	6/15/2022, 4:15:...	6/15/2022, 4:15:...	106ms		✓ Snowflake UI 202206		
✓	01a4f625-...	with active_contracts as (select contract_number, max(expira...	UCSC	COMPUTE_...	1	X-Small	15658823685	6/15/2022, 3:43:...	6/15/2022, 3:43:...	1.3s		✓ Go 1.1.5	1	
✓	01a4f053-...	select * from daily_14_total limit 1000;	UCSC	COMPUTE_...	1	X-Small	15658795057	6/14/2022, 2:53:...	6/14/2022, 2:53:...	3.1s	178.8MB	✓ Go 1.1.5	1K	
✗	01a4f050-...	select * from daily_14_total;	UCSC	COMPUTE_...	1	X-Small	15658795057	6/14/2022, 2:50:...	6/14/2022, 2:52:...	2min 35s	1.4GB	✓ Go 1.1.5	4.2M	
✓	01a4f04f-...	describe table daily_14_total;	UCSC	COMPUTE_...			15658795057	6/14/2022, 2:49:...	6/14/2022, 2:49:...	78ms		✓ Go 1.1.5		
✗	01a4f04f-...	describe daily_14_total;	UCSC	COMPUTE_...			15658795057	6/14/2022, 2:49:...	6/14/2022, 2:49:...	25ms		✓ Go 1.1.5		
✓	01a4f04f-...	show tables;	UCSC	COMPUTE_...			15658795057	6/14/2022, 2:49:...	6/14/2022, 2:49:...	68ms		✓ Go 1.1.5		
✓	01a4ef85-...	SHOw DATABASES;	UCSC	COMPUTE_...			15658799121	6/14/2022, 11:2...	6/14/2022, 11:2...	57ms		✓ SnowSQL 1.2.21		
✗	01a4ef85-...	SHOW_DATABASES ;	UCSC	COMPUTE_...			15658799121	6/14/2022, 11:2...	6/14/2022, 11:2...	19ms		✓ SnowSQL 1.2.21		
✓	01a4ef84-...	ROLLBACK	UCSC	COMPUTE_...			15658795037	6/14/2022, 11:2...	6/14/2022, 11:2...	30ms		✓ SnowSQL 1.2.21		
✓	01a4ef6c-...	SHOW GRANTS TO USER identifier('UCSC');	UCSC	COMPUTE_...			15658795033	6/14/2022, 11:0...	6/14/2022, 11:0...	39ms		✓ Snowflake UI 202206		
✓	01a4ef64-...	with active_contracts as (select contract_number, max(expira...	UCSC	COMPUTE_...	1	X-Small	15658803229	6/14/2022, 10:5...	6/14/2022, 10:5...	3.0s		✓ Go 1.1.5	1	
✓	01a4ef62-...	with active_contracts as (select contract_number, max(expira...	UCSC	COMPUTE_...	1	X-Small	15658803213	6/14/2022, 10:5...	6/14/2022, 10:5...	2.7s		✓ Go 1.1.5	1	
✓	01a4ef49-...	GRANT ROLE IDENTIFIER('ORGADMIN') to user IDENTIFIER('FI...	SNOWFLAKE				15658795013	6/14/2022, 10:2...	6/14/2022, 10:2...	101ms		JavaScript 1.6.8		

Classic Web Console – Account Details / Usage / Warehouses

Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions.

Databases Shares Marketplace Warehouses Worksheets History Account

Partner Connect Help Notifications Snowsight UCSC ACCOUNTADMIN

Account

Usage Billing Users Roles Policies Sessions Resource Monitors Reader Accounts

QW84992 - Usage June 2022 (local) ▾

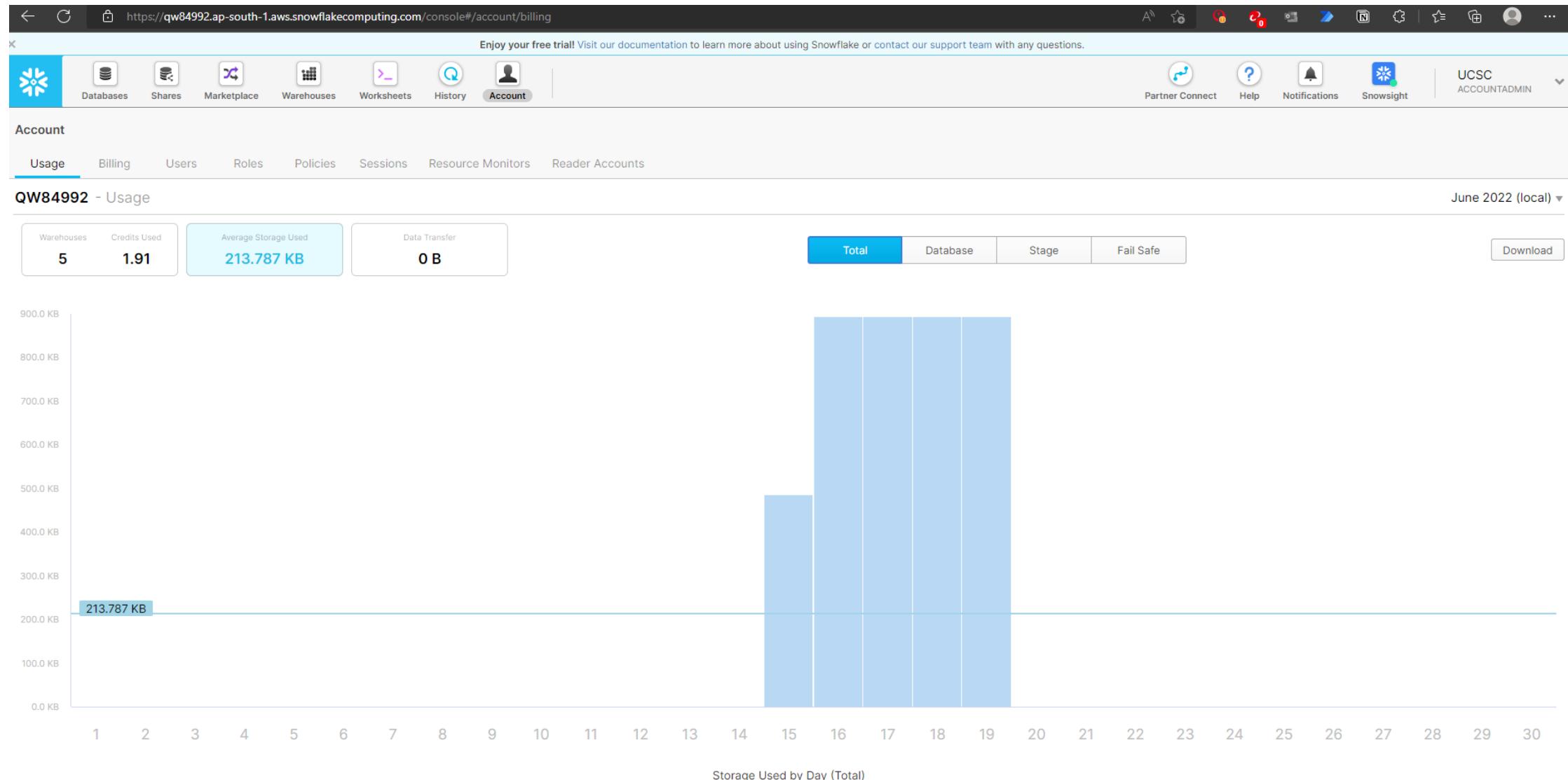
Warehouse Name	Credits Used
ENGINEERINGWH	0.73
COMPUTE_WH	0.60
SALESWH	0.37
FINANCEWH	0.21
CLOUD_SERVICES_ONLY	0.00

Average Storage Used 213.787 KB Data Transfer 0 B

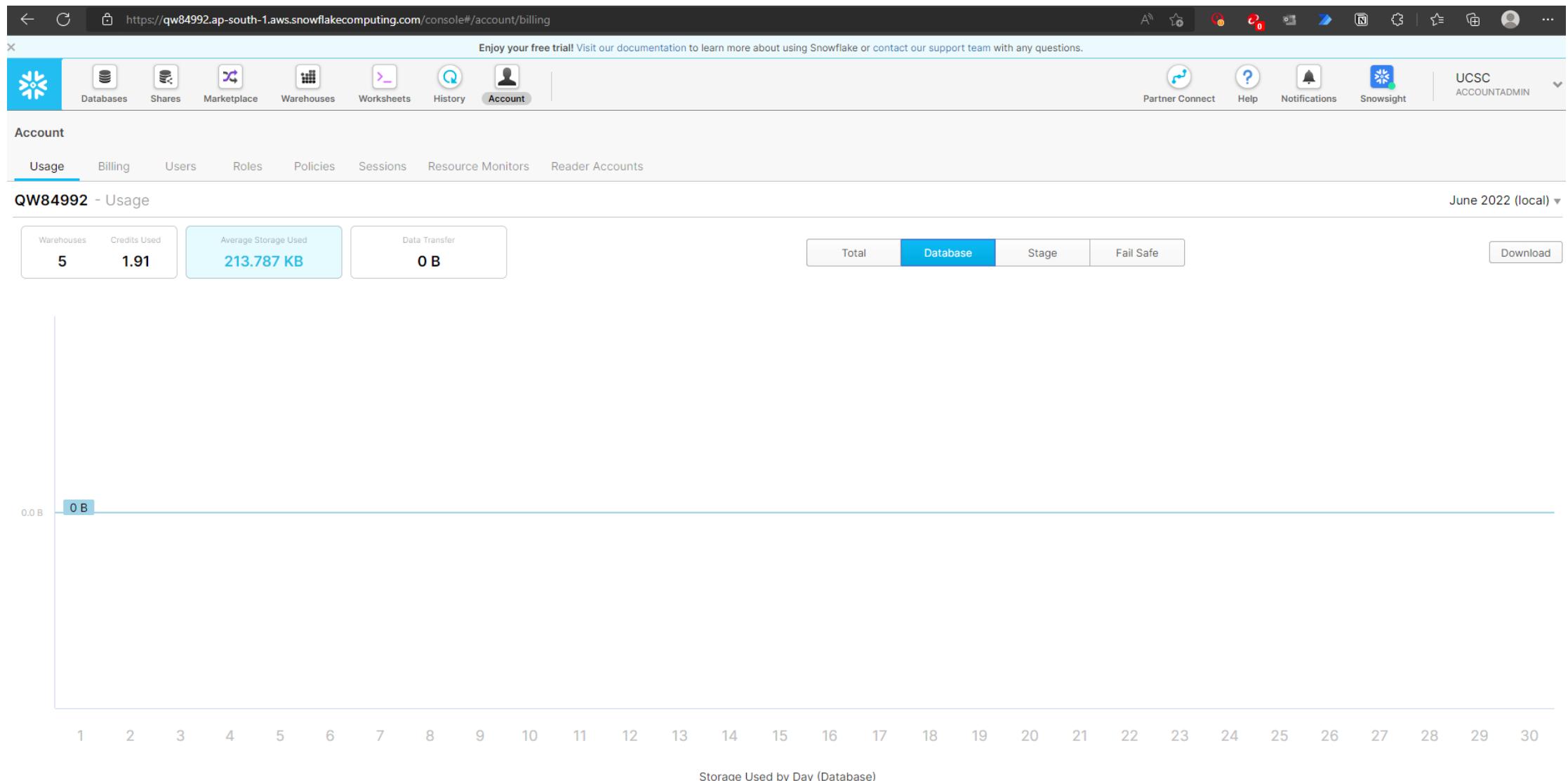
Usage by Day - All Warehouses Download

Date	Credits Used
Jun 1	0
Jun 2	0
Jun 3	0
Jun 4	0
Jun 5	0
Jun 6	0
Jun 7	0
Jun 8	0
Jun 9	0
Jun 10	0
Jun 11	0
Jun 12	0
Jun 13	0
Jun 14	0.42
Jun 15	1.50
Jun 16	0
Jun 17	0
Jun 18	0
Jun 19	0
Jun 20	0

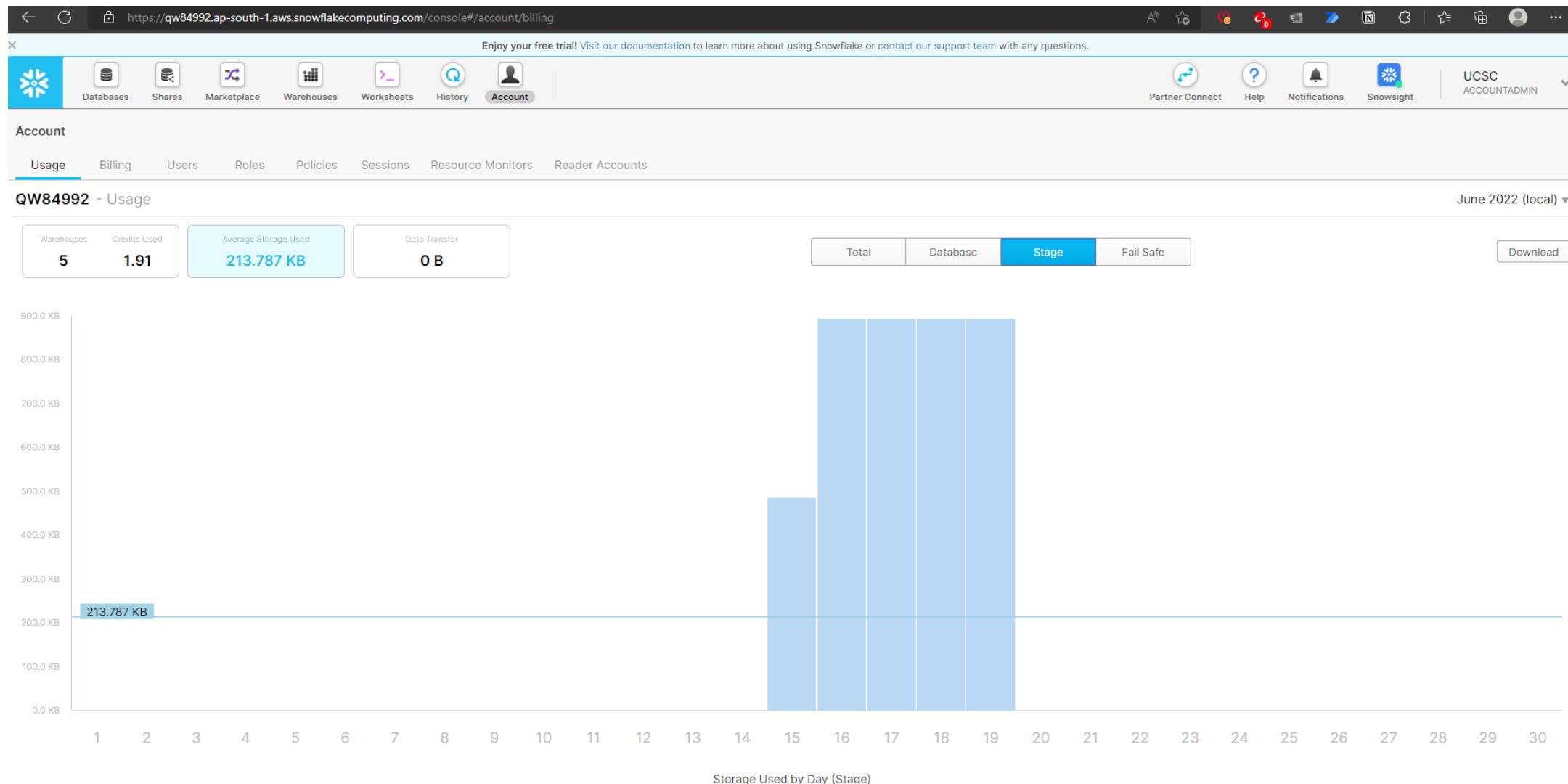
Classic Web Console – Account Details / Usage / Average Storage / Total



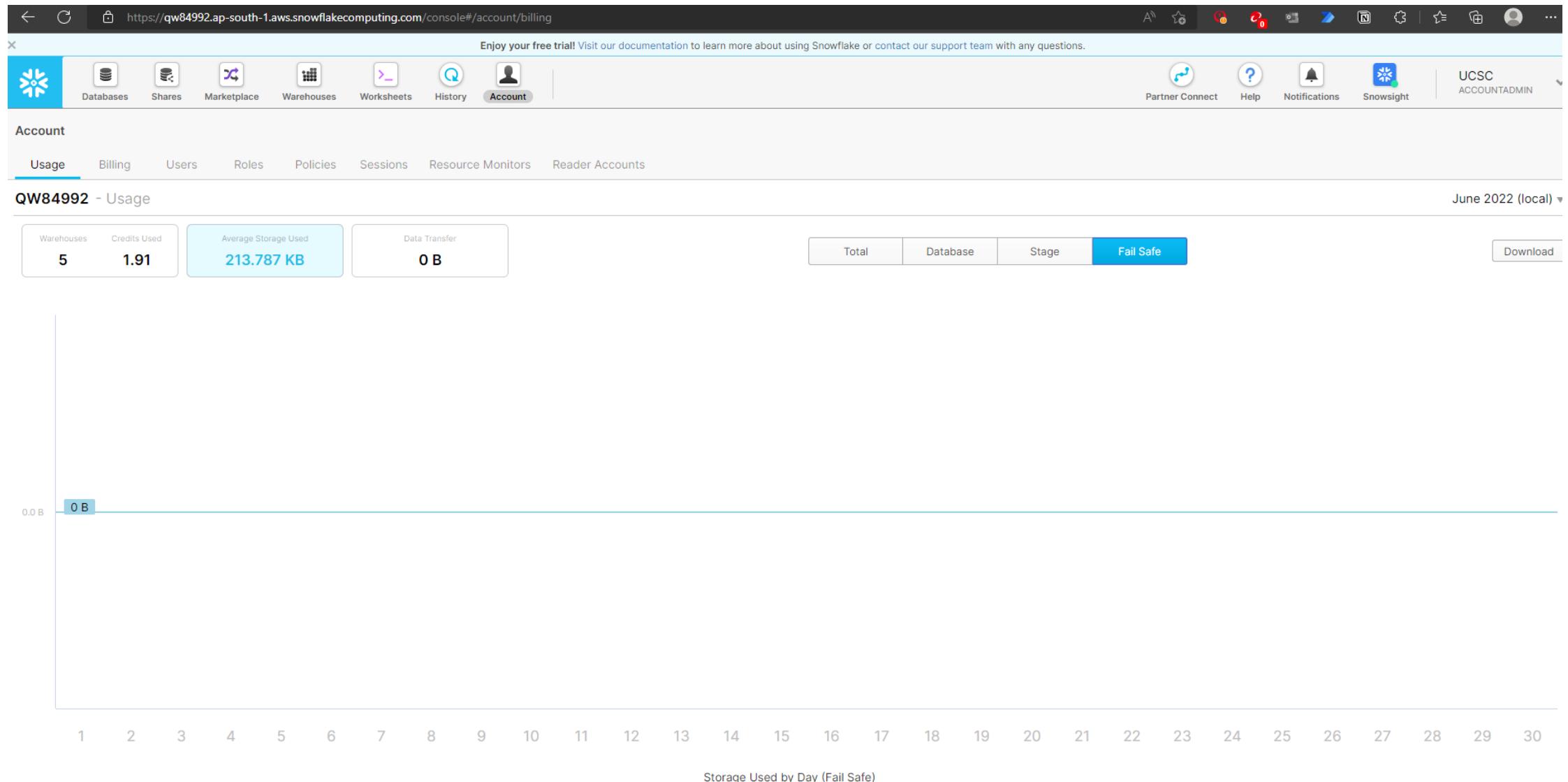
Classic Web Console – Account Details / Usage / Average Storage / Database



Classic Web Console – Account Details / Usage / Average Storage / Storage



Classic Web Console – Account Details / Usage / Average Storage / Fail Safe



Classic Web Console – Warehouses

The screenshot shows the Snowflake Classic Web Console interface for managing Warehouses. The top navigation bar includes links for Databases, Shares, Marketplace, Warehouses (which is the active tab), Worksheets, History, and Account. On the right side of the header, there are icons for Partner Connect, Help, Notifications, and Snowsight, along with the account name UCSC ACCOUNTADMIN.

The main content area is titled "Warehouses" and displays a message: "Manage your warehouses from this page. To operate on your data, you need to create one or more warehouses." It shows the last refresh was at 12:48:28 PM and includes an "Auto refresh" checkbox with a refresh icon.

Below this, there is a toolbar with buttons for Create..., Configure..., Suspend..., Resume..., Drop..., and Transfer Ownership.

The main table lists four warehouses:

Status	Warehouse Name	Size	Clusters	Scaling Poli...	Runn...	Que...	Auto Suspe...	Auto Resume	Created On ▾	Resumed On	Owner	Comment
Suspended	ENGINEERINGWH	Medium	min: 1, max: 2	Standard	0	0	10 minutes	Yes	6/15/2022, 4:19:22 ...	6/15/2022, 4:19:24 ...	ACCOUNTADMIN	
Suspended	SALESWH	Small	min: 1, max: 2	Standard	0	0	10 minutes	Yes	6/15/2022, 4:16:19 ...	6/15/2022, 4:16:19 ...	ACCOUNTADMIN	
Suspended	FINANCEWH	Large	min: 1, max: 2	Standard	0	0	10 minutes	Yes	6/15/2022, 4:15:45 ...	6/15/2022, 4:15:45 ...	ACCOUNTADMIN	
Suspended	COMPUTE_WH	X-Small	min: 1, max: 1	Standard	0	0	10 minutes	Yes	6/14/2022, 10:31:05...	6/15/2022, 3:43:14 ...	SYSADMIN	

New Web Console – Data Market Place

Snowflake – The Complete Master | MS-700 Managing Microsoft Teams | Master DevOps Monitoring with | Data Exchange | Snowflake Marketplace

ucsc snowflake ACCOUNTADMIN

Worksheets Dashboards Data Marketplace Activity Admin Help & Support

Classic Console

40 days left in trial Upgrade

EW54843

Search Snowflake Marketplace Categories Business Needs Providers My Requests

Ready to Query Free Weather Financial 360-Degree Customer View Demand Forecasting Japan Last month

Together, we can build a sustainable future

Join the Data Collaboration for the Climate initiative to inform the decisions we must make today to preserve our climate for tomorrow. [Learn more here...](#)

OAG Emissions Data OAG **GaiaLens Environmental Data...** GaiaLens **Actionable Weather Forecasts (Demo...)** AccuWeather **OECD Greenhouse Gas Emissions** Knoema

Featured Providers All Providers >

IPInfo **Heap** **Clootrack**

Most Recent More >

S&P Global Commodity Insights
HS Market Drilling & Development Activity
Streamline and grow your business with the most comprehensive data and market intelligence
Personalized

Acxiom
Acxiom Match Multiplier
Extend the reach of first party data by adding new touch points to customers before activation
Personalized

Innovaccer Inc.
InCare
AI-assisted care management solution, with patient-centered medical home (PCMH)-level care delivery.
Personalized

Innovaccer Inc.
InNote
Physician's digital assistant that surfaces health insights at point of care.
Personalized

Classic Web Console – Shares

The screenshot shows the Snowflake Classic Web Console interface. The top navigation bar includes links for Databases, Shares, Marketplace, Warehouses, Worksheets, History, Account, and various system icons. A banner at the top encourages users to "Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions." On the right, a user profile for "UCSC ACCOUNTADMIN" is visible. The main content area is titled "Secure Shares" and shows two inbound secure shares. The table has columns for "Secure Share Name", "Shared By", "Database", "Creation Time", "Owner", and "Comment". The first share is named "ACCOUNT_USAGE" and was created on 4/27/2020, 11:14:24 AM by SNOWFLAKE. The second share is named "SAMPLE_DATA" and was created on 4/23/2020, 6:33:10 AM by SFC_SAMPLES.

Secure Share Name	Shared By	Database	Creation Time	Owner	Comment
ACCOUNT_USAGE	SNOWFLAKE	SNOWFLAKE	4/27/2020, 11:14:24 AM		
SAMPLE_DATA	SFC_SAMPLES	SNOWFLAKE_SAMPLE_DATA	4/23/2020, 6:33:10 AM		

The screenshot shows the Snowflake Classic Web Console interface. The top navigation bar includes links for Databases, Shares, Marketplace, Warehouses, Worksheets, History, Account, and various system icons. A banner at the top encourages users to "Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions." On the right, a user profile for "UCSC ACCOUNTADMIN" is visible. The main content area is titled "Secure Shares" and shows zero outbound secure shares. The table has columns for "Secure Share Name", "Shared With", "Database", "Creation Time", "Owner", and "Comment".

Secure Share Name	Shared With	Database	Creation Time	Owner	Comment

Classic Web Console – Shares

The screenshot shows the Snowflake Classic Web Console interface. At the top, there's a navigation bar with links for Databases, Shares, Marketplace, Warehouses, Worksheets, History, Account, and various system icons like Partner Connect, Help, Notifications, and Snowsight. The user is logged in as UCSC ACCOUNTADMIN.

The main area is titled "Secure Shares" and shows an "Outbound" tab selected. There are buttons for "+ Create", "+ Add Consumers", "Edit", and "Drop". Below these are search fields for "Search Outbound Secure Shares" and "Shared With".

A prominent red dashed box surrounds a modal dialog titled "Create a Secure Share and add Database objects to it". Inside the dialog, there's a message about what a Secure Share is, a note about preparing data, and a "Create" section where the user can define the share. The "Create" section includes fields for "Secure Share Name" (set to "Hardware Issue Logs"), "Database" (a dropdown menu showing "Search for a Database..." and "SNOWFLAKE" which is currently selected), "Tables & Views" (a dropdown menu showing "SNOWFLAKE" and "SNOWFLAKE_SAMPLE_DATA"), and a "Comment" field. At the bottom of the dialog are "Show SQL", "Cancel", and "Create" buttons.

Classic Web Console – Databases / Create Database

The screenshot shows the Snowflake Classic Web Console interface. At the top, there is a navigation bar with links for Databases, Shares, Marketplace, Warehouses, Worksheets, History, Account, Partner Connect, Help, Notifications, and Snowsight. The user is signed in as UCSC ACCOUNTADMIN. A message at the top says "Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions."

The main area is titled "Databases" and shows a list of existing databases:

Database	Origin	Creation Time	Owner	Comment
SNOWFLAKE_SAMPLE_DATA	SFC_SAMPLES.SA...	6/14/2022, 10:27 AM	ACCOUNTADMIN	Provided by Snowflake during account provisioning
SNOWFLAKE	SNOWFLAKE.ACCO...	6/14/2022, 10:27 AM		

Below the table, there are buttons for "+ Create...", "Clone...", "Drop...", and "Transfer Ownership".

A search bar shows "Search Databases" and "2 databases".

In the center, a modal dialog box is open titled "Create Database". It contains fields for "Name*" (set to "USCS") and "Comment" (set to "UCSC BIG DATA ANALYTICS TRAINING"). There are also "Show SQL" and "Cancel" buttons, and a prominent blue "Finish" button.

Classic Web Console – Database / Grant Privileges

The screenshot shows the Snowflake Classic Web Console interface. The top navigation bar includes links for Databases, Shares, Marketplace, Warehouses, Worksheets, History, Account, and various system icons. The user is signed in as 'UCSC ACCOUNTADMIN'. The main left sidebar lists 'Databases' with options to Create, Clone, Drop, or Transfer Ownership. A search bar indicates 1/3 databases found. The table below lists three databases: 'USCS', 'SNOWFLAKE_SAMPLE_DATA', and 'SNOWFLAKE'. The 'USCS' database is selected. On the right, a modal dialog titled 'Grant privileges on database USCS' is open, allowing the user to grant privileges to a role. The dialog fields include 'Grant privileges on' set to 'USCS', 'Privileges to grant' set to 'Select privileges', 'Grant privileges to' set to 'Select a role', and a checkbox for 'with Grant Option'. The 'Grant' button is highlighted in blue.

Enjoy your free trial! Visit our documentation to learn more about using Snowflake or contact our support team with any questions.

Databases

UCSC ACCOUNTADMIN

USCS

SNOWFLAKE_SAMPLE_DATA

SNOWFLAKE

Grant privileges on database USCS

Grant privileges on **USCS**

Privileges to grant **Select privileges**

Grant privileges to **Select a role**

with Grant Option

Cancel **Grant**

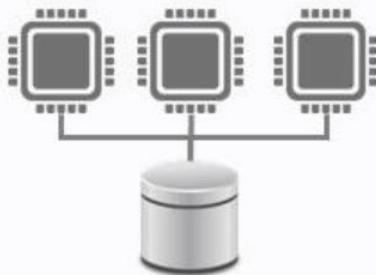


SNOWFLAKE ARCHITECTURE

LET'S DIVE DEEP INTO ARCHITECTURE OF SNOWFLAKE PLATFORM



TRADITIONAL ARCHITECTURES



Shared Disk

Storage is shared
Easier to manage storage
Performance affected by disk contention

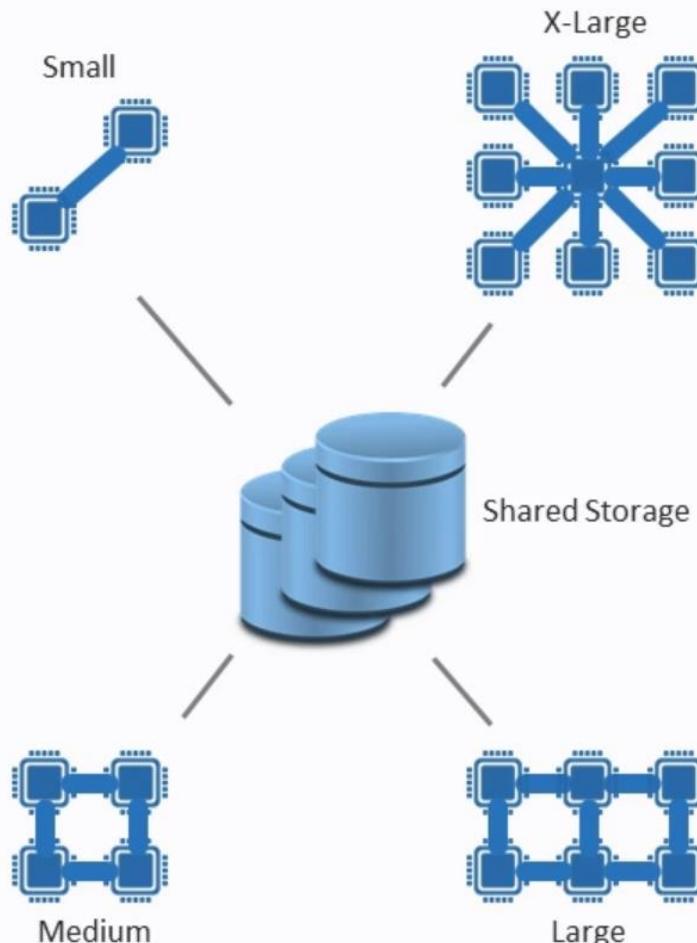


Shared Nothing

Storage & compute are decentralized
Performance scales with storage & compute increase
Generally storage increase must be accompanied by compute increase



SNOWFLAKE - HYBRID ARCHITECTURE



Shared Disk

Similar to shared disk architecture Snowflake uses a central data repository



MPP Compute Clusters

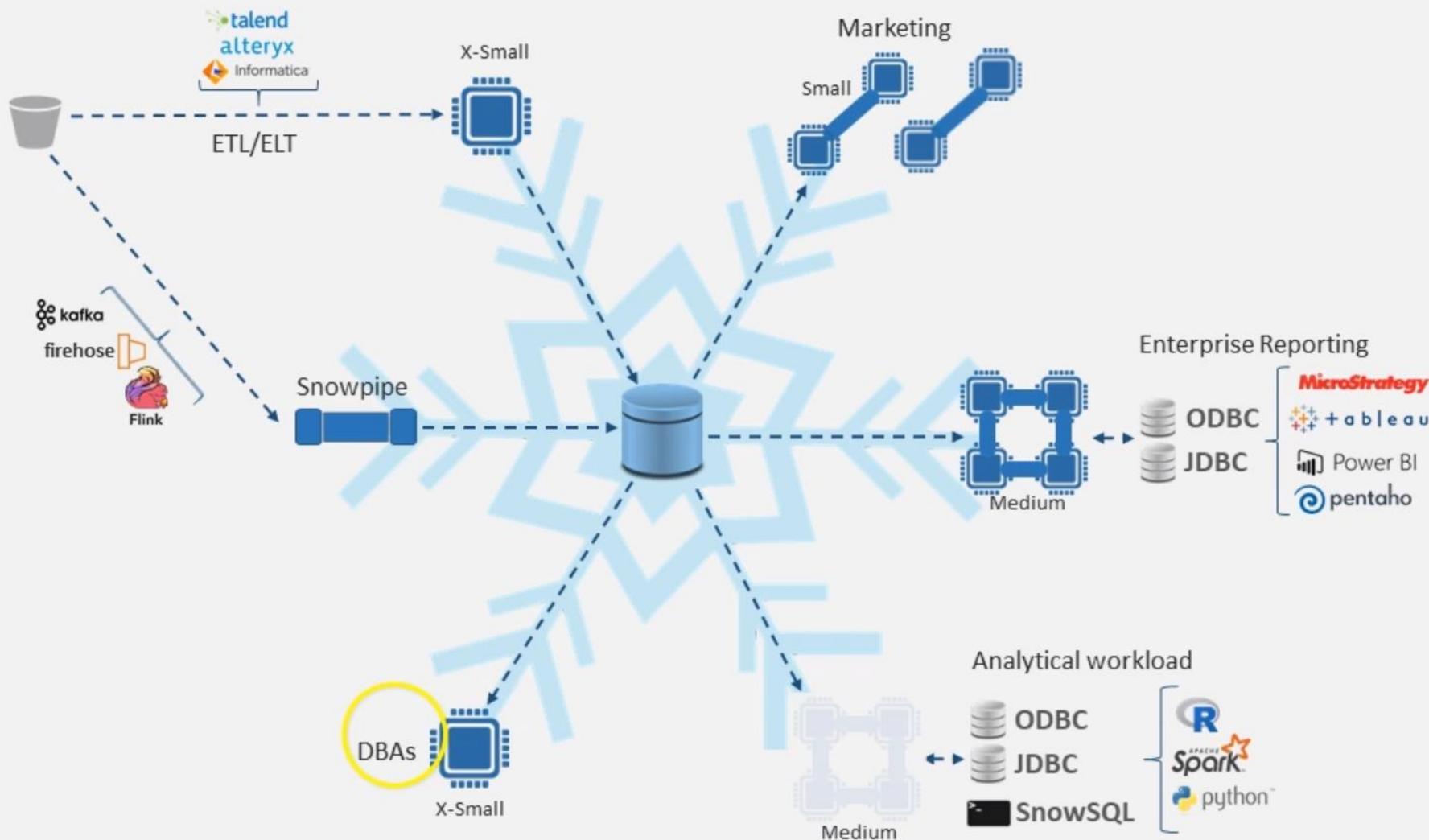
Similar to shared nothing architectures Snowflake uses massively parallel compute clusters where each node stores and processes its part of the data



Advantages

This hybrid approach provides the benefits of a simple single disk storage but still allows to scale out as and when needed

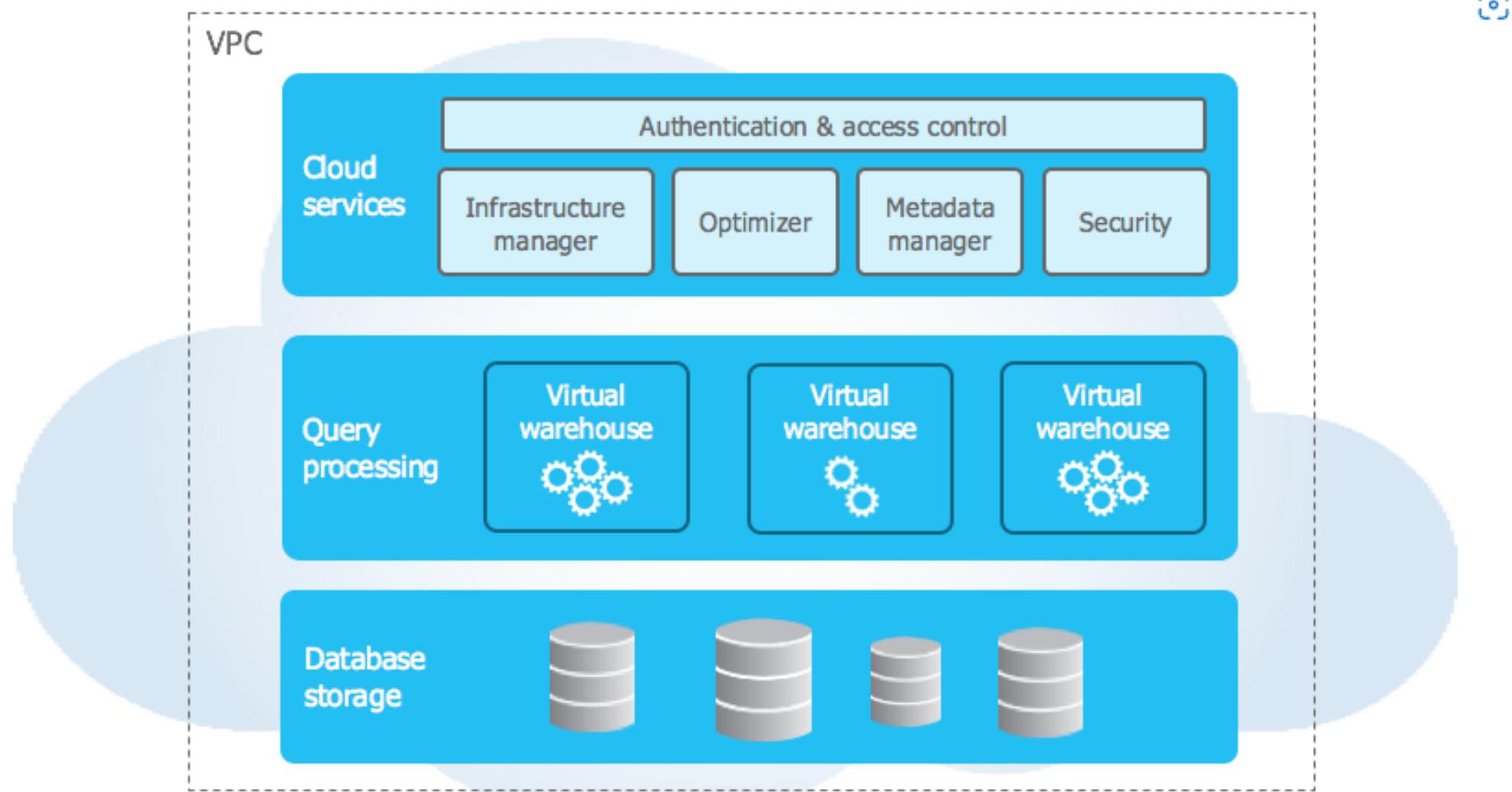
SNOWFLAKE - ARCHITECTURE



Data Platform as a Cloud Service

- Snowflake is a true SaaS offering.
 - There is no hardware (virtual or physical to select, install, configure or manage)
 - There is virtually no software to install, configure or manage
 - Ongoing maintenance, management, upgrades and tuning are handled by Snowflake
- Snowflake runs completely on cloud infrastructure. All components of Snowflake's service (other than optional command line clients, drivers, and connectors), run in public cloud infrastructures.
- Snowflake uses virtual compute instances for its compute needs and a storage service for persistent storage of data. Snowflake cannot be run on private cloud infrastructures (on-premises or hosted).
- Snowflake is not a packaged software offering that can be installed by a user. Snowflake manages all aspects of software installation and updates.

Snowflake Architecture

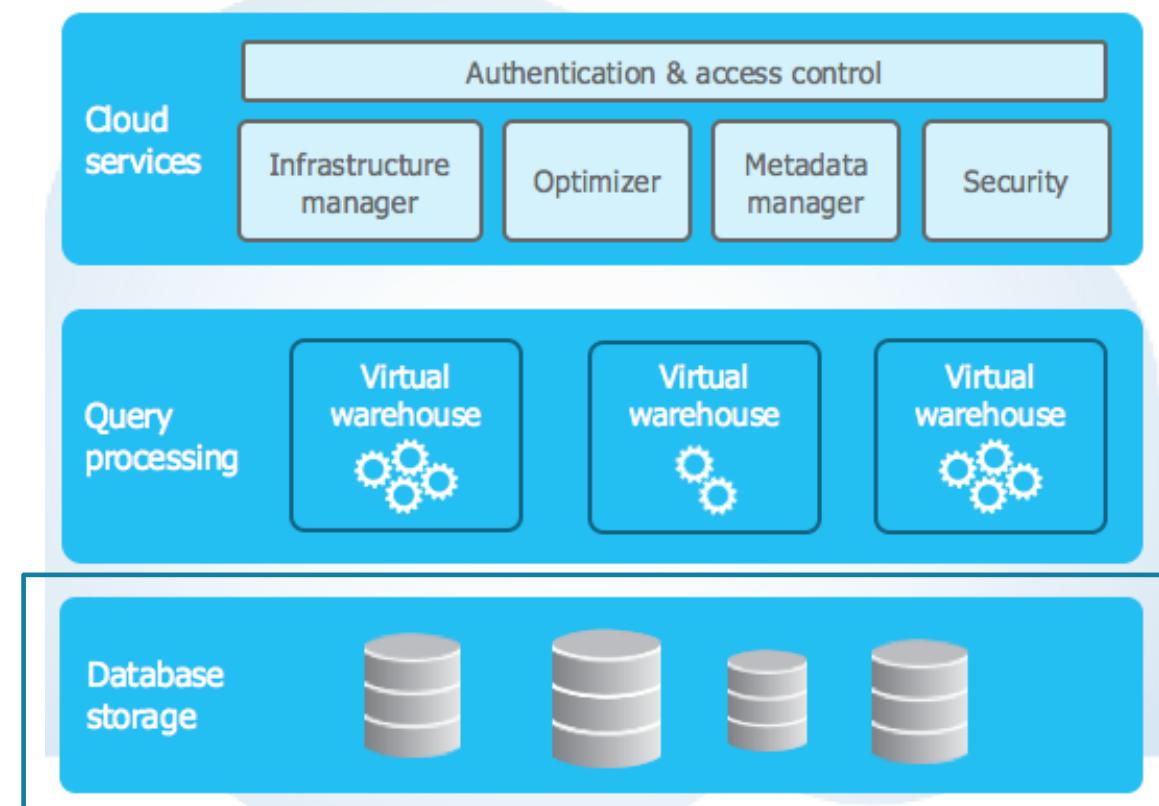


Snowflake Architecture

Database Storage

When data is loaded into Snowflake, Snowflake reorganizes that data into its internal optimized, compressed, columnar format. Snowflake stores this optimized data in cloud storage.

Snowflake manages all aspects of how this data is stored — the organization, file size, structure, compression, metadata, statistics, and other aspects of data storage are handled by Snowflake. The data objects stored by Snowflake are not directly visible nor accessible by customers; they are only accessible through SQL query operations run using Snowflake.



Snowflake Architecture

Database Storage

Snowflake's centralized database storage layer holds all data, including structured and semi-structured data. As data is loaded into Snowflake, it is optimally reorganized into a compressed, columnar format and stored and maintained in Snowflake databases. Each Snowflake database consists of one or more schemas, which are logical groupings of database objects such as tables and views.

Data stored in Snowflake databases is always compressed and encrypted. Snowflake takes care of managing every aspect of how the data is stored. Snowflake automatically organizes stored data into micro-partitions, an optimized, immutable, compressed columnar format which is encrypted using AES-256 encryption. Snowflake optimizes and compresses data to make metadata extraction and query processing easier and more efficient.

Snowflake's data storage layer is sometimes referred to as the remote disk layer. The underlying file system is implemented on Amazon, Microsoft, or Google Cloud. The specific provider used for data storage is the one you selected when you created your Snowflake account. Snowflake doesn't place limits on the amount of data you can store or the number of databases or database objects you can create. Snowflake tables can easily store petabytes of data. There is no effect on virtual warehouse size as the storage increases or decreases in a Snowflake account. The two are scaled independently from each other and from the cloud services layer.

Snowflake Architecture

Database Storage

There are two unique features in the storage layer architecture: Time Travel and zero-copy cloning.

Introduction to Zero-Copy Cloning

Zero-copy cloning offers the user a way to snapshot a Snowflake database, schema, or table along with its associated data. There is no additional storage charge until changes are made to the cloned object, because zero-copy data cloning is a metadata-only operation. For example, if you clone a database and then add a new table or delete some rows from a cloned table, at that point storage charges would be assessed. Most often, zero-copy clones will be used to support development and test environments.

Introduction to Time Travel

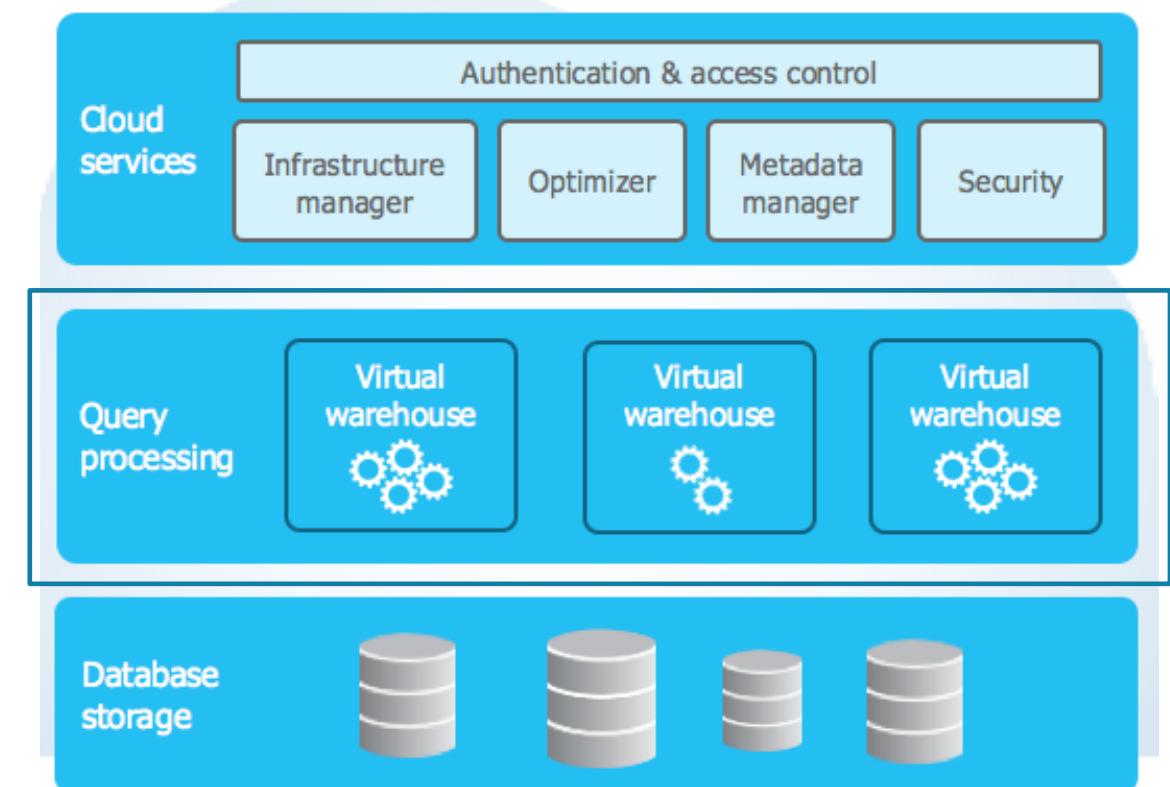
Time Travel allows you to restore a previous version of a database, table, or schema. This is an incredibly helpful feature that gives you an opportunity to fix previous edits that were done incorrectly or restore items deleted in error. With Time Travel, you can also back up data from different points in the past by combining the Time Travel feature with the clone feature, or you can perform a simple query of a database object that no longer exists. How far back you can go into the past depends on a few different factors.

Snowflake Architecture

Query Processing

Query execution is performed in the processing layer. Snowflake processes queries using “virtual warehouses”. Each virtual warehouse is an MPP compute cluster composed of multiple compute nodes allocated by Snowflake from a cloud provider.

Each virtual warehouse is an independent compute cluster that does not share compute resources with other virtual warehouses. As a result, each virtual warehouse has no impact on the performance of other virtual warehouses.

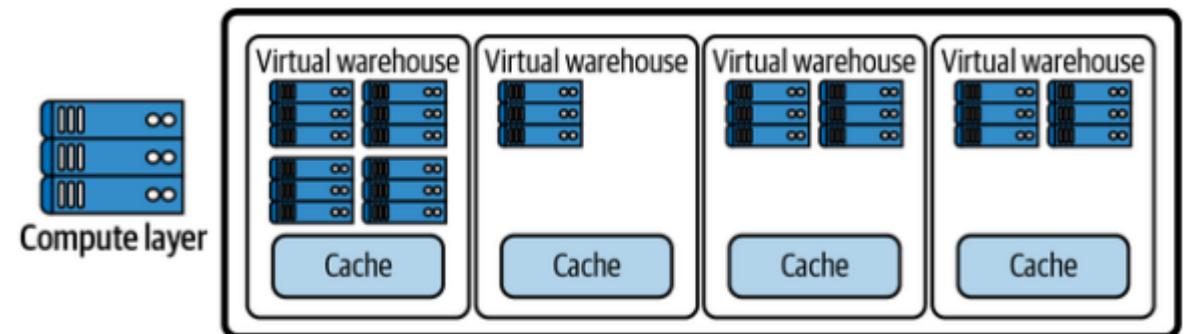


Snowflake Architecture

Query Processing

A Snowflake compute cluster, most often referred to simply as a virtual warehouse, is a dynamic cluster of compute resources consisting of CPU memory and temporary storage. Creating virtual warehouses in Snowflake makes use of the compute clusters—virtual machines in the cloud which are provisioned behind the scenes. Snowflake doesn't publish the exact server in use at any given time; it could change as the cloud providers modify their services. The Snowflake compute resources are created and deployed on demand to the Snowflake user, to whom the process is transparent.

A running virtual warehouse is required for most SQL queries and all DML operations, including loading and unloading data into tables, as well as updating rows in tables.



Snowflake Architecture

Query Processing

A virtual warehouse is always consuming credits when it is running in a session. However, Snowflake virtual warehouses can be started and stopped at any time, and can be resized at any time, even while running. Snowflake supports two different ways to scale warehouses. Virtual warehouses can be scaled up by resizing a warehouse and can be scaled out by adding clusters to a warehouse. It is possible to use one or both scaling methods at the same time.

A virtual warehouse is always consuming credits when it is running in a session. However, Snowflake virtual warehouses can be started and stopped at any time, and can be resized at any time, even while running. Snowflake supports two different ways to scale warehouses. Virtual warehouses can be scaled up by resizing a warehouse and can be scaled out by adding clusters to a warehouse. It is possible to use one or both scaling methods at the same time.

Unlike the Snowflake cloud services layer and the data storage layer, the Snowflake virtual warehouse layer is not a multitenant architecture. Snowflake predetermines the CPU, memory, and solid-state drive (SSD) configurations for each node in a virtual warehouse. While these definitions are subject to change, they are consistent in configuration across all three cloud providers.

Snowflake Architecture

Virtual Warehouse Size

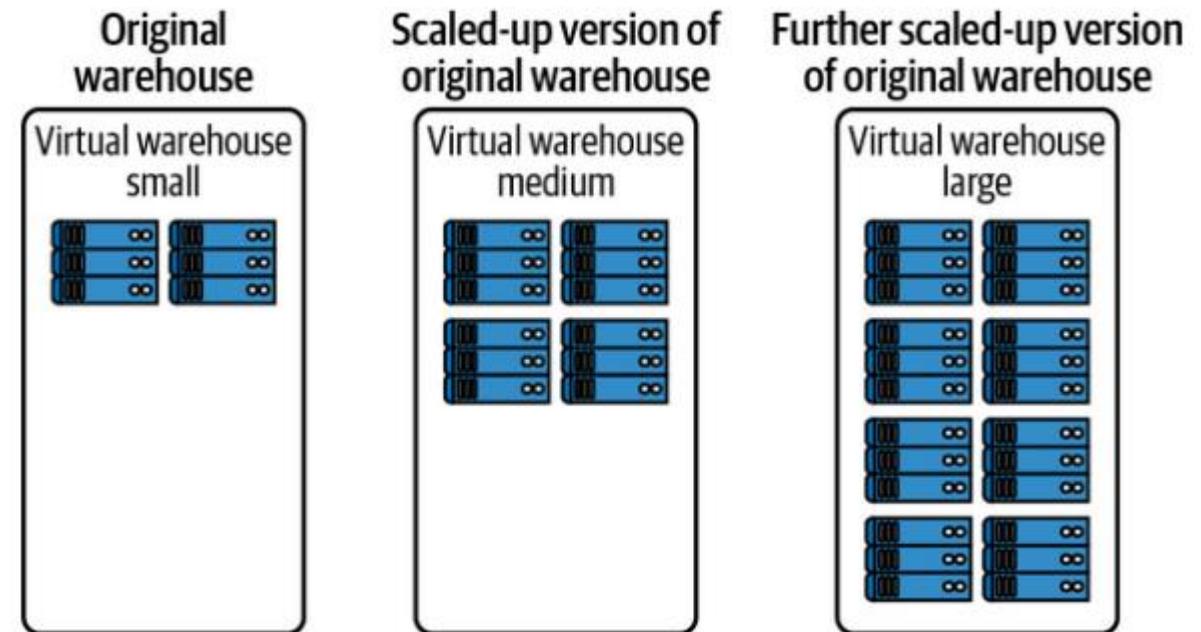
A Snowflake compute cluster is defined by its size, with size corresponding to the number of servers in the virtual warehouse cluster. For each virtual warehouse size increase, the number of compute resources on average doubles in capacity. Beyond 4X-Large, a different approach is used to determine the number of servers per cluster. However, the credits per hour do still increase by a factor of 2 for these extremely large virtual warehouses.

Because Snowflake utilizes per-second billing, it can often be cost-effective to run larger virtual warehouses because you are able to suspend virtual warehouses when they aren't being used. The exception is when you are running a lot of small or very basic queries on large virtual warehouse sizes. There won't likely be any benefit from adding the additional resources regardless of the number of concurrent queries.

It is recommended that you experiment with different types of queries and different virtual warehouse sizes to determine the best way to manage your virtual warehouses effectively and efficiently. The queries should be of a certain size and complexity that you would typically expect to complete within no more than 5 to 10 minutes. Additionally, it is recommended that you start small and increase in size as you experiment. It is easier to identify an undersized virtual warehouse than an underutilized one.

Snowflake Architecture

Larger virtual warehouses do not necessarily result in better performance for query processing or data loading. It is the query complexity, as part of query processing, that should be a consideration for choosing a virtual warehouse size because the time it takes for a server to execute a complex query will likely be greater than the time it takes to run a simple query. The amount of data to be loaded or unloaded can also greatly affect performance. Specifically, the number of files being loaded, and the size of each file are important factors for data loading performance. As such, you'll want to carefully consider your use case before selecting any virtual warehouse that is greater in size than Large. One exception to that general rule would be if you are bulk-loading hundreds or thousands of files concurrently.



Snowflake Architecture

Scaling out with Multicloud Virtual Warehouse to Maximize Concurrency

A multicloud virtual warehouse operates in much the same way as a single cluster virtual warehouse. The goal is to optimize the Snowflake system performance in terms of size and number of clusters. In the previous section, we learned that when there was a queuing problem due to either very longrunning SQL queries or a large data volume to be loaded or unloaded, scaling up could result in increased performance since the queries could run faster.

If a concurrency problem is due to many users, or connections, scaling up will not adequately address the problem. Instead, we'll need to scale out by adding clusters — example going from a Min Clusters value of 1 to a Max Clusters value of 3. Multicloud virtual warehouses can be set to automatically scale if the number of users and/or queries tends to fluctuate

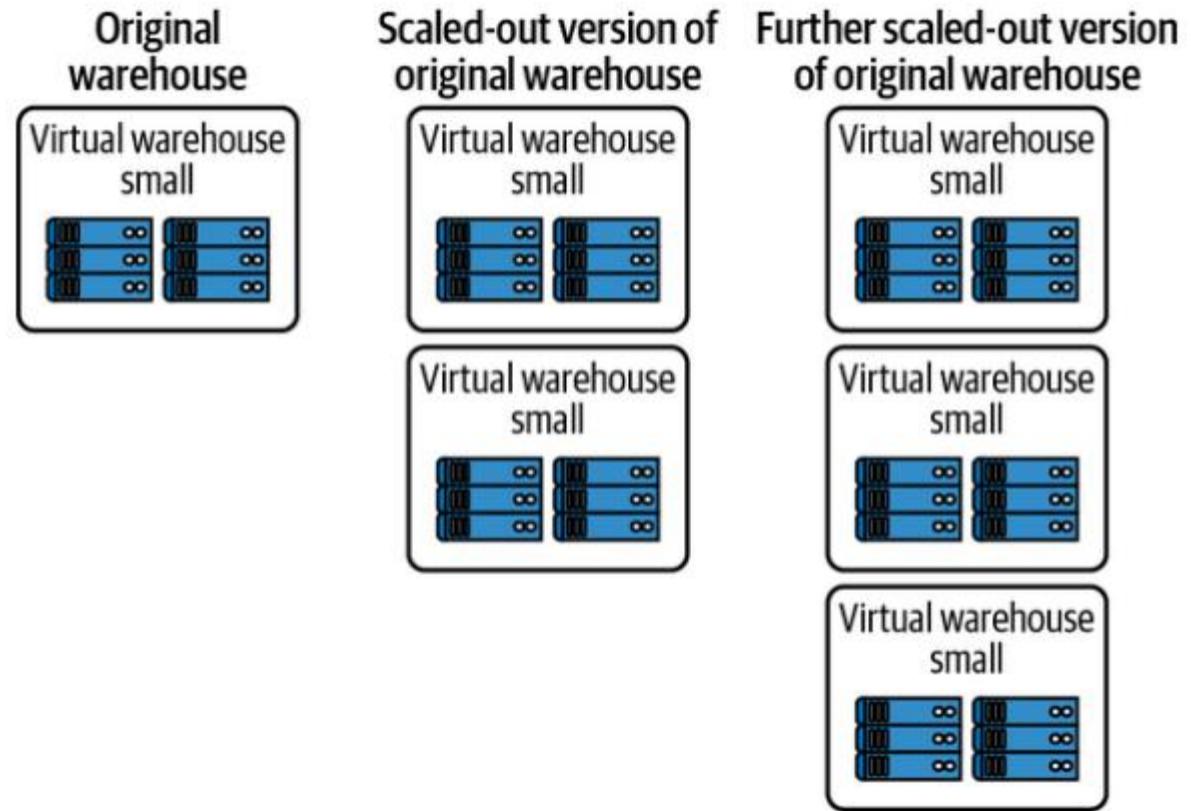
Unlike single-cluster virtual warehouses where sizing is a manual process, scaling in or out for multicloud virtual warehouses is an automated process. You'll just need to let Snowflake know how much you want the multicloud virtual warehouse to scale out.

The two modes that can be selected for a multicloud virtual warehouse are Auto-scale and Maximized. The Snowflake scaling policy, designed to help control the usage credits in the Auto-scale mode, can be set to Standard or Economy.

Snowflake Architecture

Whenever a multicluster virtual warehouse is configured with the scaling policy set to Standard, the first virtual warehouse immediately starts when a query is queued, or if the Snowflake system detects that there is one more query than the currently running clusters can execute. Each successive virtual warehouse starts 20 seconds after the prior virtual warehouse has started.

If a multicluster virtual warehouse is configured with the scaling policy set to Economy, a virtual warehouse starts only if the Snowflake system estimates the query load can keep the virtual warehouse busy for at least six minutes.

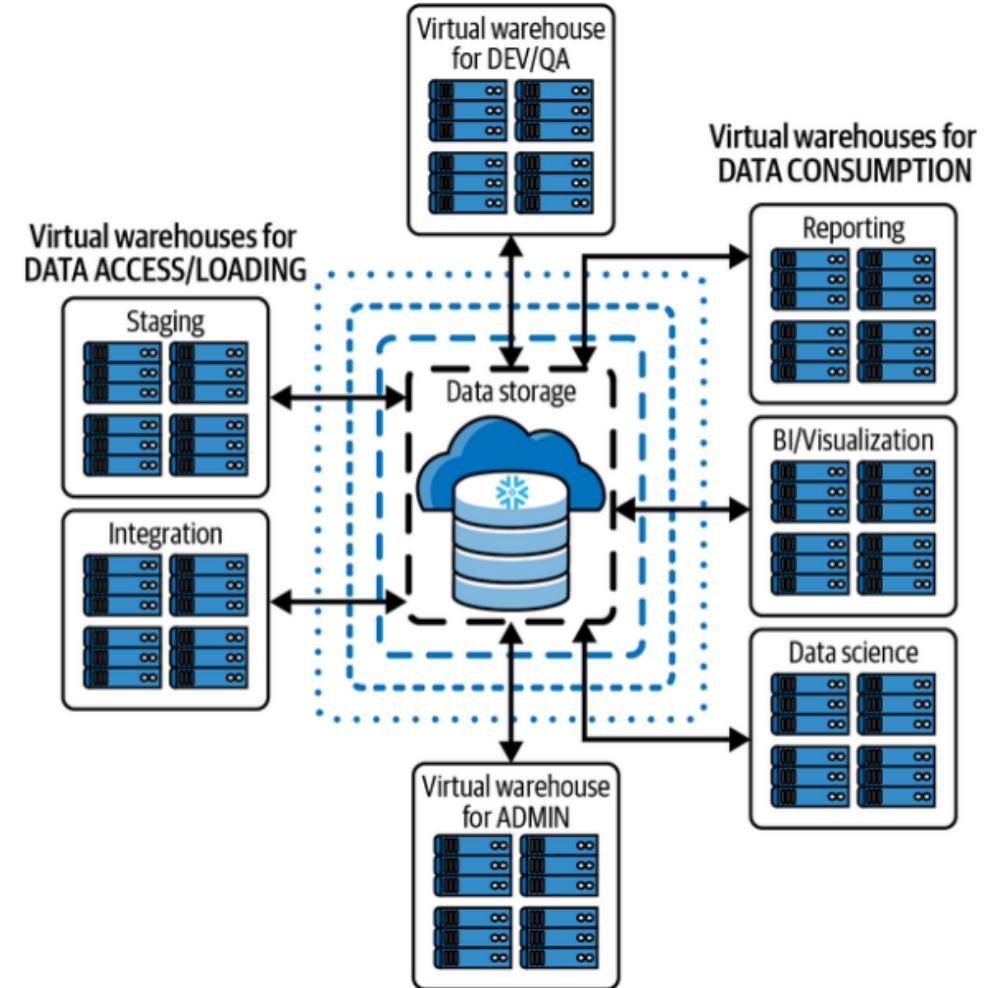
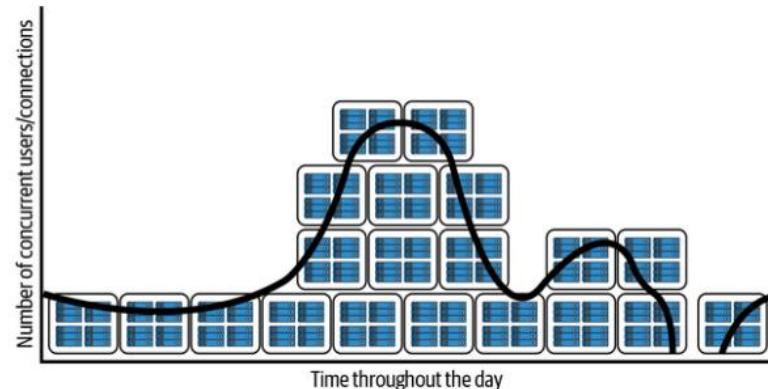


Snowflake Architecture

Separation of Workloads and Workload Management

Query processing tends to slow down when the workload reaches full capacity on traditional database systems. In contrast, Snowflake estimates the resources needed for each query, and as the workload approaches 100%, each new query is suspended in a queue until there are sufficient resources to execute them.

Several options exist to efficiently handle workloads of varying size. One way is to separate the workloads by assigning different virtual warehouses to different users or groups of users.

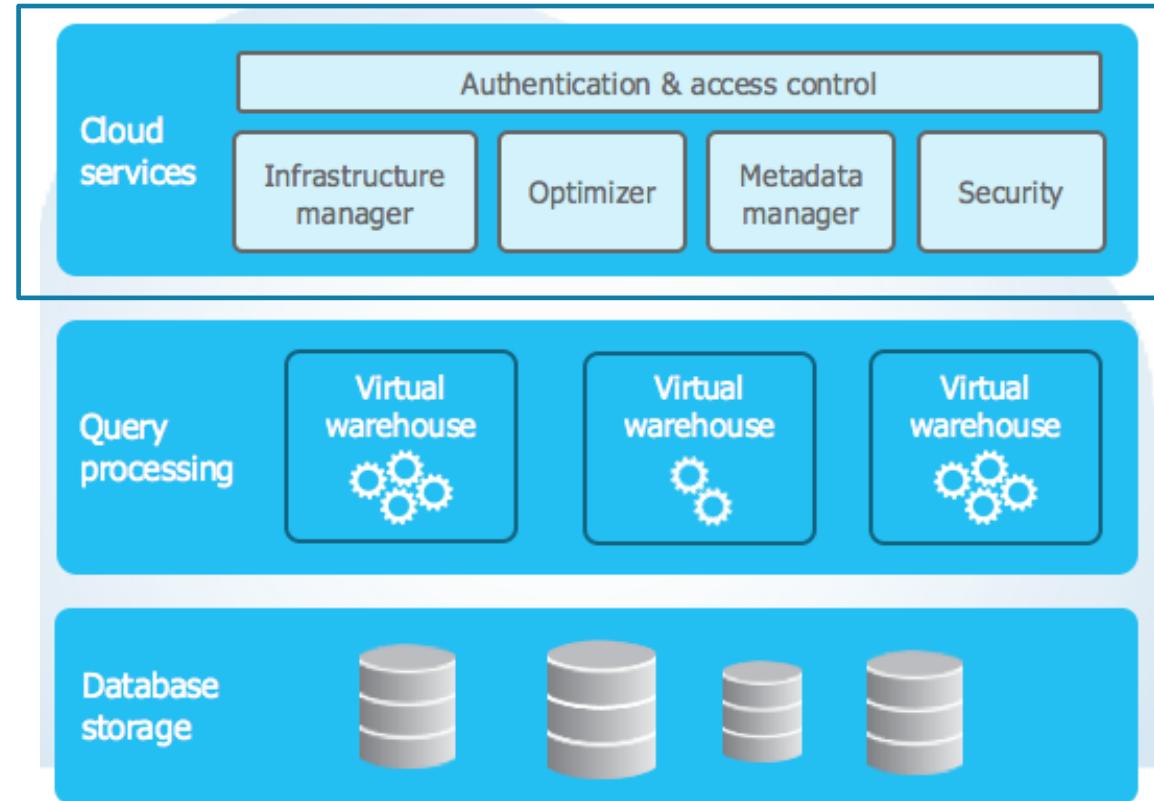


Snowflake Architecture

Cloud Services

The cloud services layer is a collection of services that coordinate activities across Snowflake. These services tie together all the different components of Snowflake in order to process user requests, from login to query dispatch. The cloud services layer also runs on compute instances provisioned by Snowflake from the cloud provider.

- Authentication
- Infrastructure management
- Metadata management
- Query parsing and optimization
- Access control



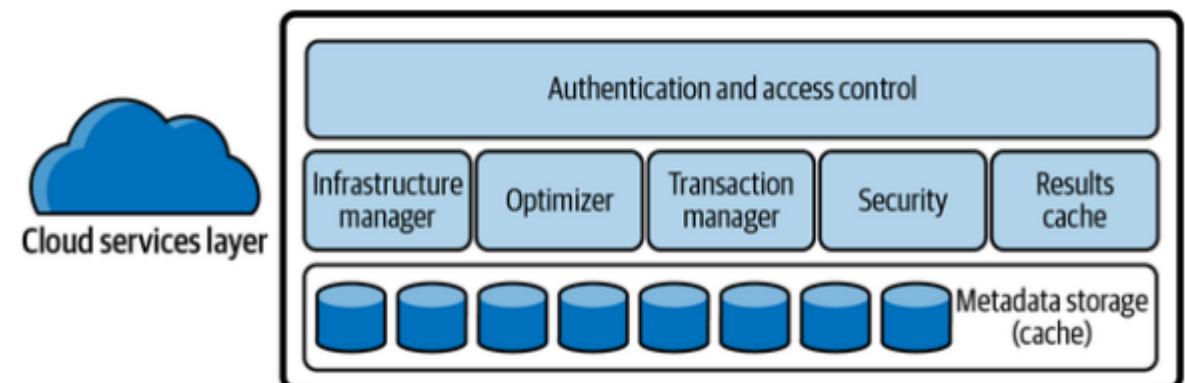
Snowflake Architecture

Cloud Services

Managing the Cloud Services Layer

The cloud services layer manages data security, including the security for data sharing. The Snowflake cloud services layer runs across multiple availability zones in each cloud provider region and holds the result cache, a cached copy of the executed query results. The metadata required for query optimization or data filtering are also stored in the cloud services layer.

Just like the other Snowflake layers, the cloud services layer will scale independently of the other layers. Scaling of the cloud services layer is an automated process that doesn't need to be directly manipulated by the Snowflake end user.

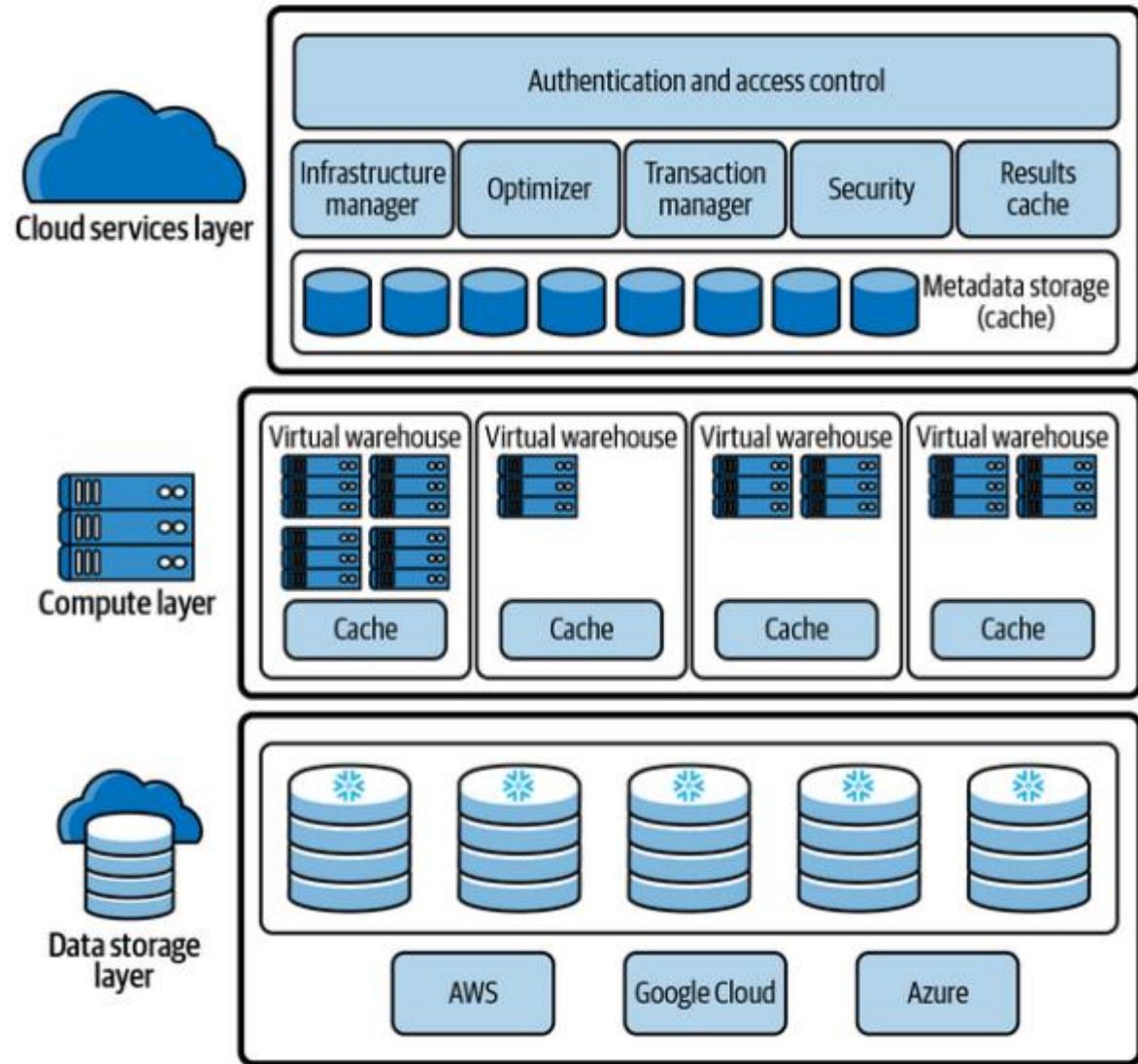


Snowflake Architecture

Cloud Services

The unique Snowflake design physically separates but logically integrates storage and compute along with providing services such as security and management.

The Snowflake hybrid-model architecture is composed of three layers, which are shown in below the cloud services layer, the compute layer, and the data storage layer. There are three caches in the architecture.



Snowflake Architecture

Snowflake Caching

When you submit a query, Snowflake checks to see whether that query has been previously run and, if so, whether the results are still cached. Snowflake will use the cached result set if it is still available rather than executing the query you just submitted. In addition to retrieving the previous query results from a cache, Snowflake supports other caching techniques.

There are three Snowflake caching types: the query result cache, the virtual warehouse cache, and the metadata cache.

Query Result Cache

The fastest way to retrieve data from Snowflake is by using the query result cache. The results of a Snowflake query are cached, or persisted, for 24 hours and then purged. This contrasts with how the virtual warehouse cache and metadata cache work. Neither of those two caches is purged based on a timeline. Even though the result cache only persists for 24 hours, the clock is reset each time the query is re-executed, up to a maximum of 31 days from the date and time when the query was first executed. After 31 days, or sooner if the underlying data changes, a new result is generated and cached when the query is submitted again.

The process for retrieving cached results is managed by GCS. However, once the size of the results exceeds a certain threshold, the results are stored in and retrieved from cloud storage.

Snowflake Architecture

Snowflake Caching

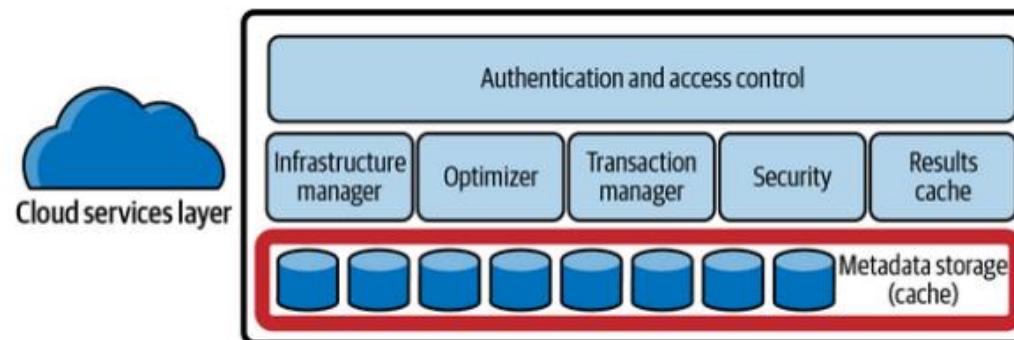
Metadata Cache

Snowflake collects and manages metadata about tables, micro-partitions, and even clustering. For tables, Snowflake stores row count, table size in bytes, file references, and table versions.

The Snowflake metadata repository includes table definitions and references to the micro-partition files for that table. The range of values in terms of MIN and MAX, the NULL count, and the number of distinct values are captured from micro-partitions and stored in Snowflake.

Snowflake also stores the total number of micro-partitions and the depth of overlapping micro-partitions to provide information about clustering.

The information stored in the metadata cache is used to build the query execution plan.



Snowflake Architecture

Snowflake Caching

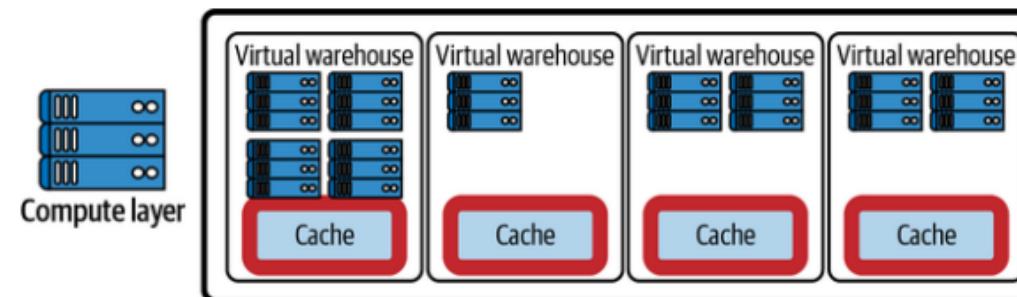
[Virtual Warehouse Local Disk Cache](#)

The traditional Snowflake data cache is specific to the virtual warehouse used to process the query. Running virtual warehouses use SSD storage to store the micro-partitions that are pulled from the centralized database storage layer when a query is processed.

The virtual warehouse data cache is limited in size and uses the LRU (Least Recently Used) algorithm.

Whenever a virtual warehouse receives a query to execute, that warehouse will scan the SSD cache first before accessing the Snowflake remote disk storage.

Reading from SSD is faster than from the database storage layer but still requires the use of a running virtual warehouse



Snowflake Architecture

Connecting to Snowflake

Snowflake supports multiple ways of connecting to the service:

- A web-based user interface from which all aspects of managing and using Snowflake can be accessed.
- Command line clients (e.g., SnowSQL) which can also access all aspects of managing and using Snowflake.
- ODBC and JDBC drivers that can be used by other applications (e.g., Tableau) to connect to Snowflake.
- Native connectors (e.g., Python, Spark) that can be used to develop applications for connecting to Snowflake.
- Third-party connectors that can be used to connect applications such as ETL tools (e.g., Informatica) and BI tools (e.g., ThoughtSpot) to Snowflake.

Snowflake Architecture

Hands On Exercises



SNOWFLAKE PRICING

LET'S DIVE DEEP INTO HOW SNOWFLAKE PRICING WORKS





Different Editions

Standard

- Complete DWH
- Automatic Data Encryption
- Time Travel up to 1 day
- Disaster Recovery for 7 days beyond Time Travel
- Secure Data Sharing
- Premier Support 24/7

Enterprise

- All Standard features
- Multi-Cluster Warehouse
- Time Travel up to 90 days
- Materialized Views
- Search Optimization
- Column Level Security

Business Critical

- All Enterprise Features
- Additional Security features such as Data Encryption Everywhere
- Extended Support
- Database Failover and Disaster Recovery

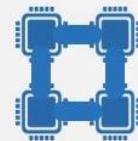
Virtual Private

- All Business-Critical features
- Dedicated Virtual Servers and Completely separate Snowflake Environment

Snowflake Pricing



Storage Cost



Compute Costs

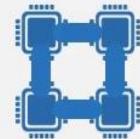
- Pay only for what you need
- Scalable amount of storage at affordable cloud price
- Pricing depending on the region
- Compute and Storage Costs decoupled

Snowflake Pricing



Storage Cost

- Monthly storage fees
- Based on average storage used per month
- Cost calculated after compression
- Cloud Providers



Compute Costs

- Charged for active warehouse per hour
- Depending on the size of the warehouse
- Billed by second (minimum of 1min)
- Charged in Snowflake Credit



Different Editions – Compute Pricing

Standard

Enterprise

Business Critical

Virtual Private

\$2 / Credit

- Platform : AWS
- Region : US East (Northern Virginia)

\$2.70 / Credit

- Platform : AWS
- Region : EU (Frankfurt)

\$3 / Credit

\$4 / Credit

Contact Snowflake

\$4 / Credit

\$5.40 / Credit

Contact Snowflake

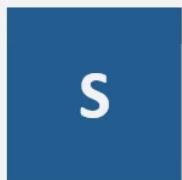
Snowflake Credit

VIRTUAL WAREHOUSE SIZE	XS	S	M	L	XL	2XL	3XL	4XL
NO NODES / SNOWFLAKE CREDITS PER HOUR	1	2	4	8	16	32	64	128

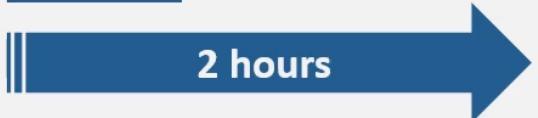
Snowflake Credit



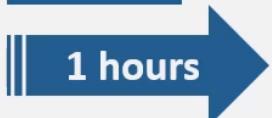
1 SNOW FLAKE CREDIT PER HOUR x 4 HOURS = **4 SNOWFLAKE CREDITS**



2 SNOW FLAKE CREDIT PER HOUR x 2 HOURS = **4 SNOWFLAKE CREDITS**



4 SNOW FLAKE CREDIT PER HOUR x 1 HOUR = **4 SNOWFLAKE CREDITS**





Different Editions – Storage Pricing

On Demand

Pay only for what you use

\$40 / TB

- Platform : AWS
- Region : US East (Northern Virginia)

\$45 / TB

- Platform : AWS
- Region : EU (Frankfurt)

Capacity Storage

Pay only for defined capacity upfront

\$23 / TB

\$24.50 / TB



Different Editions – Storage Pricing

On Demand

Capacity Storage

Use Case Scenario

We think we need 1TB of Storage

Scenario #1 We use only 100GB Storage



Pay only for what you use

\$40 / TB

$$0.1 \text{ TB} \times \$40 = \$4$$

Scenario #2 We use only 800GB Storage



$$0.8 \text{ TB} \times \$40 = \$32$$

- Platform : AWS
- Region : US East (Northern Virginia)

Pay only for defined capacity upfront

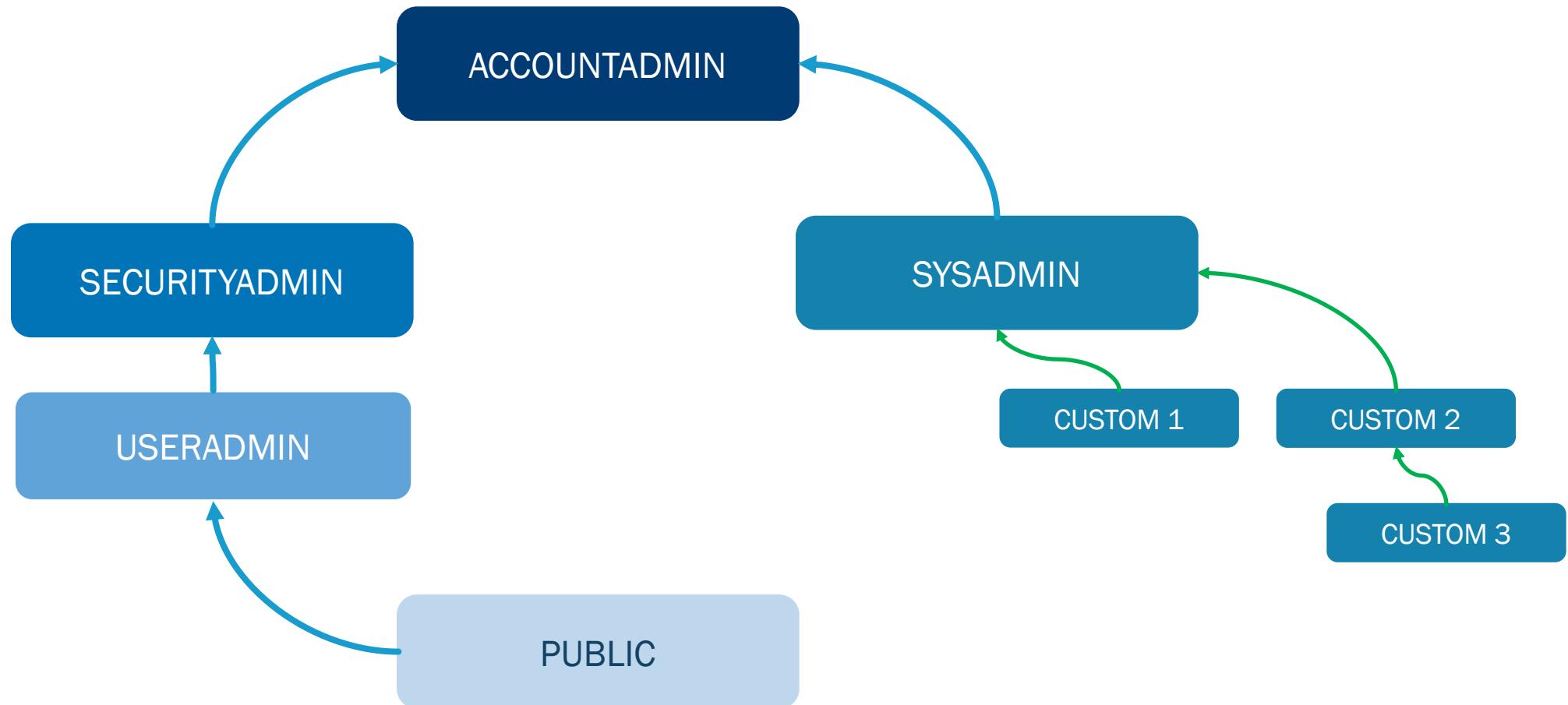
\$23 / TB

$$1 \text{ TB} \times \$23 = \$23$$

- If you are **Not Sure** about the usage start with On Demand and change to Capacity a little later.
- If you know the usage start with Capacity Storage



Snowflake Roles





Roles

ACCOUNTADMIN

- SYSADMIN and SECURITYADMIN
- TOP LEVEL role in the system
- Given only to a limited number of users

SECURITYADMIN

- USERADMIN role is granted to SECURITYADMIN
- Can manage users and roles
- Can manage any object grant globally

SYSADMIN

- Create warehouses and databases (and more objects)
- Recommended that all custom roles are assigned to SYSADMIN

USERADMIN

- Dedicated to user and role management only
- Can create users and roles

PUBLIC

- Automatically granted to every user
- Can create own objects like every other role (available to every other user/role)

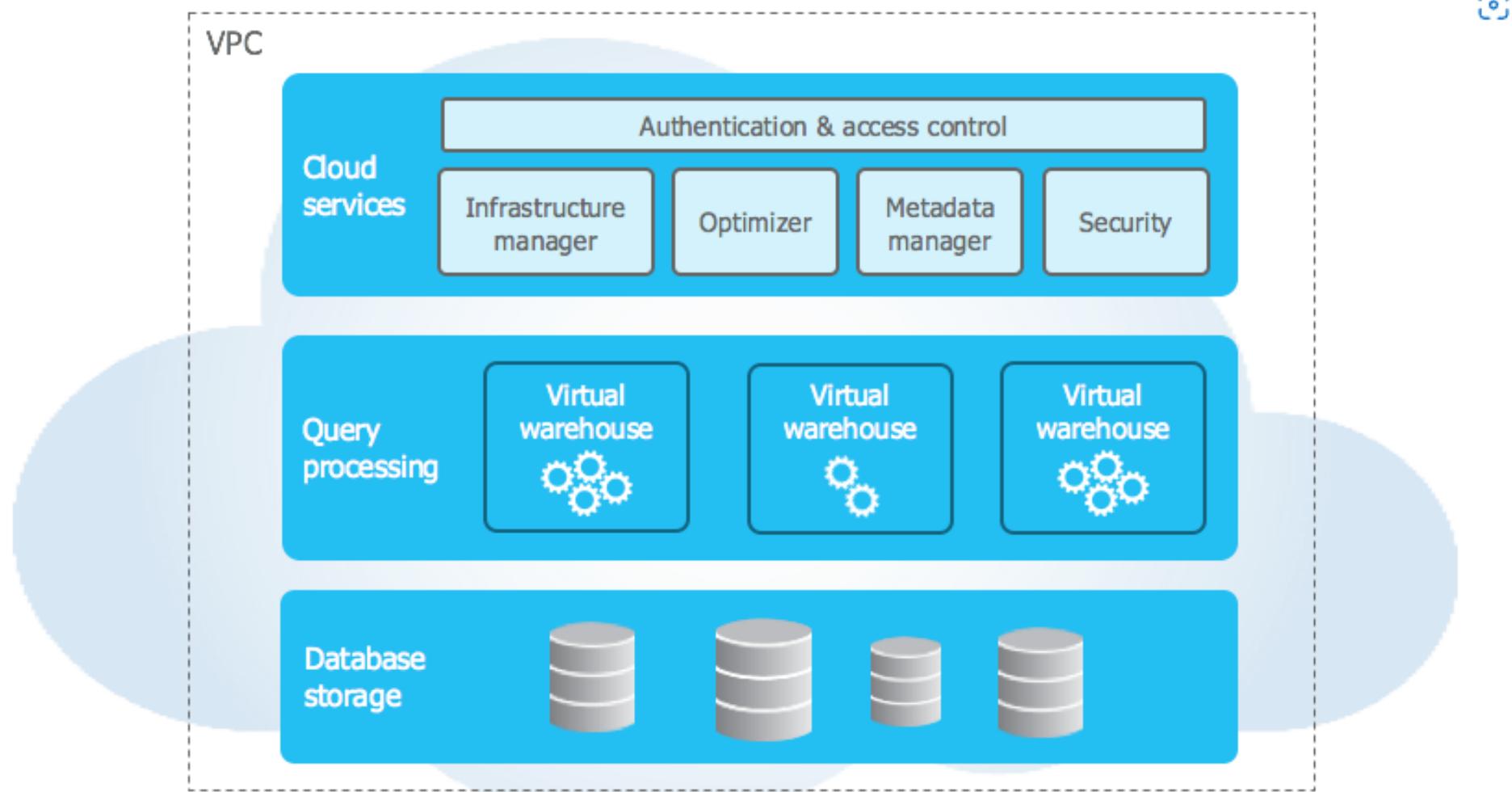


SNOWFLAKE VIRTUAL WAREHOUSES

LET'S DIVE DEEP INTO SNOWFLAKE VIRTUAL WAREHOUSES



Snowflake Architecture



Snowflake Virtual Warehouses

Warehouses are required for queries, as well as all DML operations, including loading data into tables. A warehouse is defined by its size, as well as the other properties that can be set to help control and automate warehouse activity.

Warehouses can be started and stopped at any time. They can also be resized at any time, even while running, to accommodate the need for more or less compute resources, based on the type of operations being performed by the warehouse.

Warehouse Size	Credits / Hour	Credits / Second	Notes
X-Small	1	0.0003	Default size for warehouses created using CREATE WAREHOUSE .
Small	2	0.0006	
Medium	4	0.0011	
Large	8	0.0022	
X-Large	16	0.0044	Default for warehouses created in the web interface.
2X-Large	32	0.0089	
3X-Large	64	0.0178	
4X-Large	128	0.0356	
5X-Large	256	0.0711	Preview feature.
6X-Large	512	0.1422	Preview feature.

Running Time	Credits (X-Small)	Credits (X-Large)	Credits (5X-Large)
0-60 seconds	0.017	0.267	4.268
61 seconds	0.017	0.271	4.336
2 minutes	0.033	0.533	8.532
10 minutes	0.167	2.667	42.668
1 hour	1.000	16.000	256.000

Impact on Credit Usage and Billing

There is a doubling of credit usage as you increase in size to the next larger warehouse size for each full hour that the warehouse runs; however, note that Snowflake utilizes per-second billing (with a 60-second minimum each time the warehouse starts) so warehouses are billed only for the credits they actually consume.

The total number of credits billed depends on how long the warehouse runs continuously. For comparison purposes, the following table shows the billing totals for three different size warehouses based on their running time (totals rounded to the nearest 1000th of a credit):

Impact on Data Loading

Increasing the size of a warehouse does **not** always improve data loading performance. Data loading performance is influenced more by the number of files being loaded (and the size of each file) than the size of the warehouse.

Running Time	Credits (X-Small)	Credits (X-Large)	Credits (5X-Large)
0-60 seconds	0.017	0.267	4.268
61 seconds	0.017	0.271	4.336
2 minutes	0.033	0.533	8.532
10 minutes	0.167	2.667	42.668
1 hour	1.000	16.000	256.000

Impact on Query Processing

The size of a warehouse can impact the amount of time required to execute queries submitted to the warehouse, particularly for larger, more complex queries.

In general, query performance scales with warehouse size because larger warehouses have more compute resources available to process queries.

If queries processed by a warehouse are running slowly, you can always resize the warehouse to provision more compute resources. The additional resources do not impact any queries that are already running, but once they are fully provisioned, they become available for use by any queries that are queued or newly submitted.

Auto-Suspension and Auto-Resumption

A warehouse can be set to automatically resume or suspend, based on activity:

- By default, auto-suspend is enabled. Snowflake automatically suspends the warehouse if it is inactive for the specified period of time.
- By default, auto-resume is enabled. Snowflake automatically resumes the warehouse when any statement that requires a warehouse is submitted, **and** the warehouse is the current warehouse for the session.

These properties can be used to simplify and automate your monitoring and usage of warehouses to match your workload. Auto-suspend ensures that you do not leave a warehouse running (and consuming credits) when there are no incoming queries. Similarly, auto-resume ensures that the warehouse starts up again as soon as it is needed.

Query Processing and Concurrency

The number of queries that a warehouse can concurrently process is determined by the size and complexity of each query.

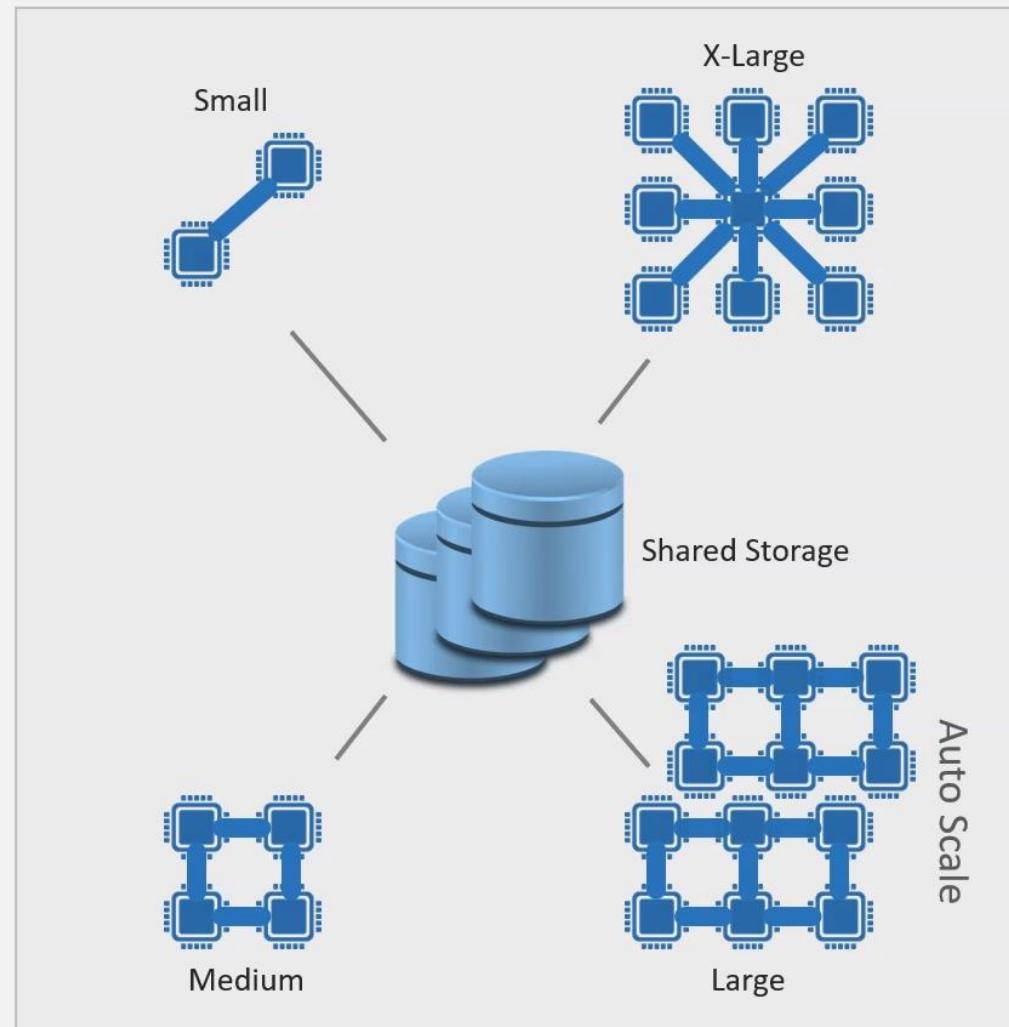
As queries are submitted, the warehouse calculates and reserves the compute resources needed to process each query.

If the warehouse does not have enough remaining resources to process a query, the query is queued, pending resources that become available as other running queries complete.

Snowflake provides some object-level parameters that can be set to help control query processing and concurrency:

- [STATEMENT_QUEUED_TIMEOUT_IN_SECONDS](#)
- [STATEMENT_TIMEOUT_IN_SECONDS](#)

SNOWFLAKE – SCALABILITY & VIRTUAL WAREHOUSE



Multi-Cluster Warehouses

Multi-cluster warehouses enable you to scale compute resources to manage your user and query concurrency needs as they change, such as during peak and off hours.

What is a Multi-Cluster Warehouse?

By default, a virtual warehouse consists of a single cluster of compute resources available to the warehouse for executing queries. As queries are submitted to a warehouse, the warehouse allocates resources to each query and begins executing the queries. If sufficient resources are not available to execute all the queries submitted to the warehouse, Snowflake queues the additional queries until the necessary resources become available.

With multi-cluster warehouses, Snowflake supports allocating, either statically or dynamically, additional clusters to make a larger pool of compute resources available. A multi-cluster warehouse is defined by specifying the following properties:

- Maximum number of clusters, greater than 1 (up to 10).
- Minimum number of clusters, equal to or less than the maximum (up to 10).

Maximized vs Auto-Scale

Maximized

This mode is enabled by specifying the **same** value for both maximum and minimum number of clusters (note that the specified value must be larger than 1). In this mode, when the warehouse is started, Snowflake starts all the clusters so that maximum resources are available while the warehouse is running.

This mode is effective for statically controlling the available compute resources, particularly if you have large numbers of concurrent user sessions and/or queries and the numbers do not fluctuate significantly.

Maximized vs Auto-Scale

Auto-Scale

This mode is enabled by specifying ***different*** values for maximum and minimum number of clusters. In this mode, Snowflake starts and stops clusters as needed to dynamically manage the load on the warehouse:

- As the number of concurrent user sessions and/or queries for the warehouse increases, and queries start to queue due to insufficient resources, Snowflake automatically starts additional clusters, up to the maximum number defined for the warehouse.
- Similarly, as the load on the warehouse decreases, Snowflake automatically shuts down clusters to reduce the number of running clusters and, correspondingly, the number of credits used by the warehouse.

To help control the usage of credits in Auto-scale mode, Snowflake provides a property, SCALING_POLICY, that determines the scaling policy to use when automatically starting or shutting down additional clusters.

When determining the maximum and minimum number of clusters to use for a multi-cluster warehouse, start with Auto-scale mode and start small (e.g. maximum = 2 or 3, minimum = 1). As you track how your warehouse load fluctuates over time, you can increase the maximum and minimum number of clusters until you determine the numbers that best support the upper and lower boundaries of your user/query concurrency.

Benefits of Multi-Cluster Warehouses

With a standard, single-cluster warehouse, if your user/query load increases to the point where you need more compute resources:

- You must either increase the size of the warehouse or start additional warehouses and explicitly redirect the additional users/queries to these warehouses.
- Then, when the resources are no longer needed, to conserve credits, you must manually downsize the larger warehouse or suspend the additional warehouses.

In contrast, a multi-cluster warehouse enables larger numbers of users to connect to the same size warehouse. In addition:

- In Auto-scale mode, a multi-cluster warehouse eliminates the need for resizing the warehouse or starting and stopping additional warehouses to handle fluctuating workloads. Snowflake automatically starts and stops additional clusters as needed.
- In Maximized mode, you can control the capacity of the multi-cluster warehouse by increasing or decreasing the number of clusters as needed.

Setting the Scaling Policy for Multi-Cluster Warehouse

To help control the credits consumed by a multi-cluster warehouse running in Auto-scale mode, Snowflake provides scaling policies, which are used to determine when to start or shut down a cluster.

The scaling policy for a multi-cluster warehouse only applies if it is running in Auto-scale mode. In Maximized mode, all clusters run concurrently so there is no need to start or shut down individual clusters.

Policy	Description	Warehouse Starts...	Warehouse Shuts Down...
Standard (default)	Prevents/minimizes queuing by favoring starting additional clusters over conserving credits.	<p>The first cluster starts immediately when either a query is queued or the system detects that there's one more query than the currently-running clusters can execute.</p> <p>Each successive cluster waits to start 20 seconds after the prior one has started. For example, if your warehouse is configured with 10 max clusters, it can take a full 200+ seconds to start all 10 clusters.</p>	After 2 to 3 consecutive successful checks (performed at 1 minute intervals), which determine whether the load on the least-loaded cluster could be redistributed to the other clusters without spinning up the cluster again.
Economy	Conserves credits by favoring keeping running clusters fully-loaded rather than starting additional clusters, which may result in queries being queued and taking longer to complete.	Only if the system estimates there's enough query load to keep the cluster busy for at least 6 minutes.	After 5 to 6 consecutive successful checks (performed at 1 minute intervals), which determine whether the load on the least-loaded cluster could be redistributed to the other clusters without spinning up the cluster again.

Classic Web Console – Creating a Warehouse

The screenshot shows the Snowflake Classic Web Console interface. At the top, there is a navigation bar with links for Databases, Shares, Data Marketplace, Warehouses (which is the active tab), Worksheets, History, and Account. To the right of the navigation bar are various icons for Partner Connect, Help, Notifications, and Snowsight, along with a user account dropdown for UCSC ACCOUNTADMIN.

The main content area is titled "Warehouses" and contains a sub-header: "Manage your warehouses from this page. To operate on your data, you need to create one or more warehouses." Below this, there is a toolbar with buttons for Create..., Configure..., Suspend..., Resume..., Drop..., and Transfer Ownership.

A table displays existing warehouses:

Status	Warehouse Name	Size	Clusters	Scaling Poli...
Started	COMPUTE_WH	X-Small	1 active (min: 1, ...)	Standard

A modal dialog box titled "Create Warehouse" is open in the center. It contains the following fields:

- Name *: A text input field.
- Size: A dropdown menu set to "X-Large (16 credits / hour)".
Subtext: "Learn more about virtual warehouse sizes [here](#)".
- Maximum Clusters: A dropdown menu set to "2".
Subtext: "Multi-cluster warehouses improve the query throughput for high concurrency workloads."
- Minimum Clusters: A dropdown menu set to "1".
Subtext: "The number of active clusters will vary between the specified minimum and maximum values, based on number of concurrent users/queries."
- Scaling Policy: A dropdown menu set to "Standard".
Subtext: "The policy used to automatically start up and shut down clusters."
- Auto Suspend: A dropdown menu set to "10 minutes".
Subtext: "The maximum idle time before the warehouse will be automatically suspended."
Checkboxes:
 - Auto Resume [?](#)
- Comment: A text input field.

At the bottom of the dialog are three buttons: "Show SQL", "Cancel", and "Finish".

Classic Web Console – Creating a Warehouse

Create Warehouse

Name * FinanceWH

Size Large (8 credits / hour)

Learn more about virtual warehouse sizes [here](#)

Maximum Clusters 2

Multi-cluster warehouses improve the query throughput for high concurrency workloads.

Minimum Clusters 1

The number of active clusters will vary between the specified minimum and maximum values, based on number of concurrent users/queries.

Scaling Policy Standard

The policy used to automatically start up and shut down clusters.

Auto Suspend 10 minutes

The maximum idle time before the warehouse will be automatically suspended.

Auto Resume [?](#)

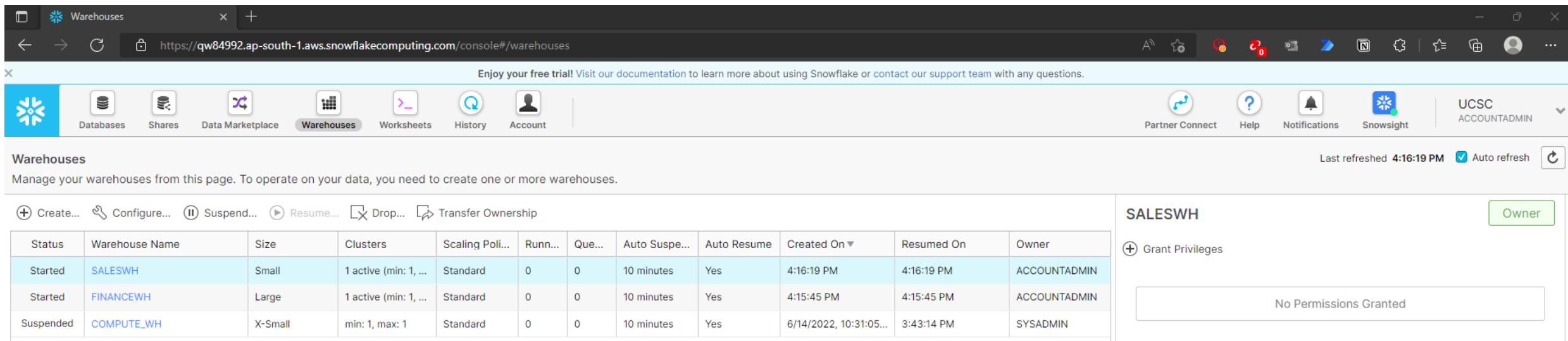
Comment

Show SQL

Cancel

Finish

Classic Web Console – Warehouses / Suspend / Resume



Warehouses

Manage your warehouses from this page. To operate on your data, you need to create one or more warehouses.

+ Create... Configure... Suspend... Resume... Drop... Transfer Ownership

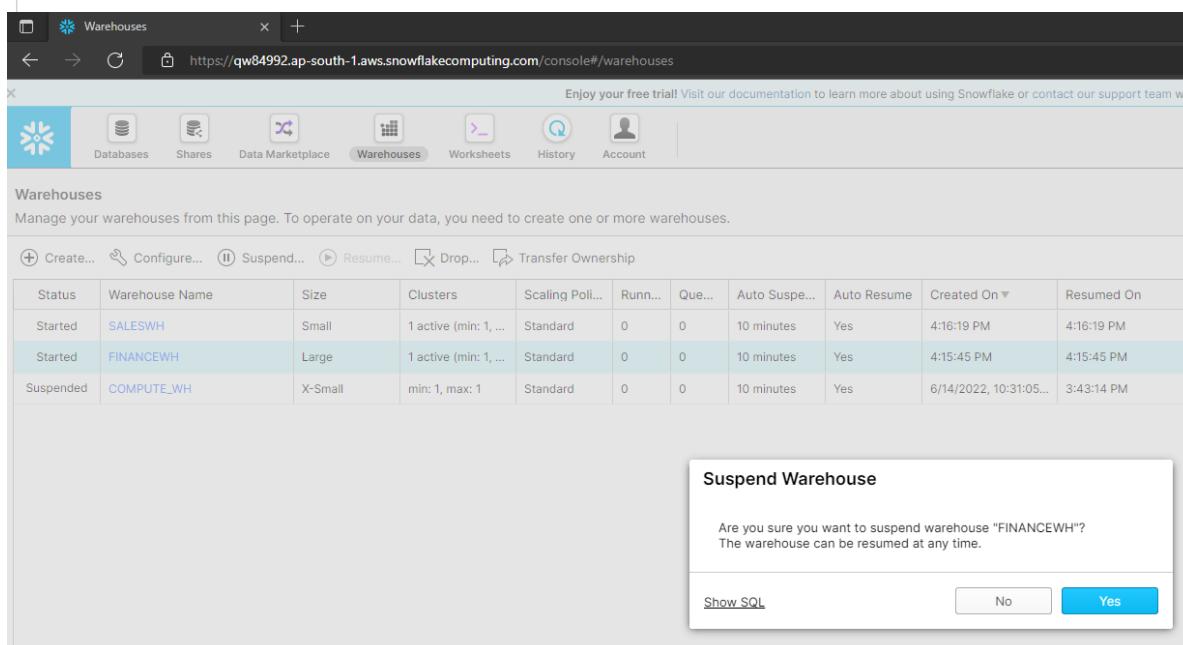
Status	Warehouse Name	Size	Clusters	Scaling Poli...	Runn...	Que...	Auto Suspe...	Auto Resume	Created On	Resumed On	Owner
Started	SALESWH	Small	1 active (min: 1, ...)	Standard	0	0	10 minutes	Yes	4:16:19 PM	4:16:19 PM	ACCOUNTADMIN
Started	FINANCEWH	Large	1 active (min: 1, ...)	Standard	0	0	10 minutes	Yes	4:15:45 PM	4:15:45 PM	ACCOUNTADMIN
Suspended	COMPUTE_WH	X-Small	min: 1, max: 1	Standard	0	0	10 minutes	Yes	6/14/2022, 10:31:05...	3:43:14 PM	SYSADMIN

SALESWH

Owner

+ Grant Privileges

No Permissions Granted



Warehouses

Manage your warehouses from this page. To operate on your data, you need to create one or more warehouses.

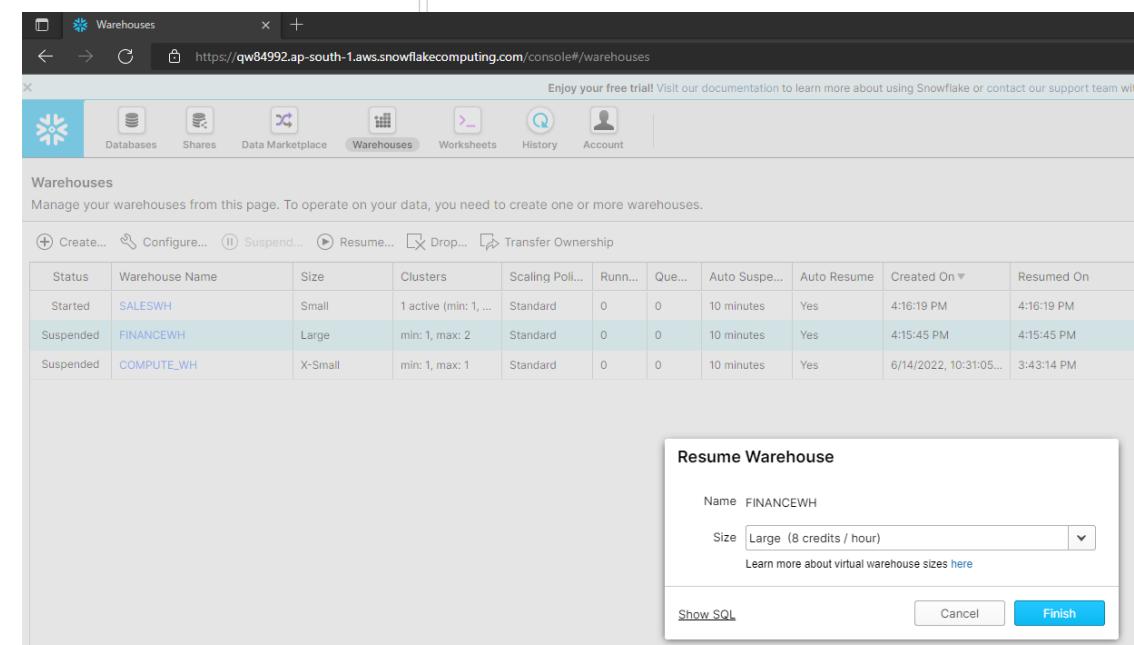
+ Create... Configure... Suspend... Resume... Drop... Transfer Ownership

Status	Warehouse Name	Size	Clusters	Scaling Poli...	Runn...	Que...	Auto Suspe...	Auto Resume	Created On	Resumed On
Started	SALESWH	Small	1 active (min: 1, ...)	Standard	0	0	10 minutes	Yes	4:16:19 PM	4:16:19 PM
Started	FINANCEWH	Large	1 active (min: 1, ...)	Standard	0	0	10 minutes	Yes	4:15:45 PM	4:15:45 PM
Suspended	COMPUTE_WH	X-Small	min: 1, max: 1	Standard	0	0	10 minutes	Yes	6/14/2022, 10:31:05...	3:43:14 PM

Suspend Warehouse

Are you sure you want to suspend warehouse "FINANCEWH"?
The warehouse can be resumed at any time.

Show SQL No Yes



Warehouses

Manage your warehouses from this page. To operate on your data, you need to create one or more warehouses.

+ Create... Configure... Suspend... Resume... Drop... Transfer Ownership

Status	Warehouse Name	Size	Clusters	Scaling Poli...	Runn...	Que...	Auto Suspe...	Auto Resume	Created On	Resumed On
Started	SALESWH	Small	1 active (min: 1, ...)	Standard	0	0	10 minutes	Yes	4:16:19 PM	4:16:19 PM
Suspended	FINANCEWH	Large	min: 1, max: 2	Standard	0	0	10 minutes	Yes	4:15:45 PM	4:15:45 PM
Suspended	COMPUTE_WH	X-Small	min: 1, max: 1	Standard	0	0	10 minutes	Yes	6/14/2022, 10:31:05...	3:43:14 PM

Resume Warehouse

Name FINANCEWH
Size Large (8 credits / hour)
Learn more about virtual warehouse sizes [here](#)

Show SQL Cancel Finish

Classic Web Console – Create Warehouse using SnowSQL

The screenshot shows the Snowflake Classic Web Console interface. At the top, there is a navigation bar with links for Databases, Shares, Data Marketplace, Warehouses, Worksheets (which is the active tab), History, and Account. On the right side of the header, there are icons for Partner Connect, Help, Notifications, and Snowsight, along with the account name 'UCSC ACCOUNTADMIN'. Below the header, a search bar says 'New Worksheet' and a 'Run' button is visible. The main workspace contains the following SQL code:

```
1 CREATE WAREHOUSE EngineeringWH
2 WITH WAREHOUSE_SIZE = 'MEDIUM'
3 WAREHOUSE_TYPE = 'STANDARD'
4 AUTO_SUSPEND = 600
5 AUTO_RESUME = TRUE
6 MIN_CLUSTER_COUNT = 1
7 MAX_CLUSTER_COUNT = 2
8 SCALING_POLICY = 'STANDARD';
```

Below the code, the 'Results' tab is selected, showing the output of the query. The results table has columns for Row and status. The single row shows the message: 'Warehouse ENGINEERINGWH successfully created.'

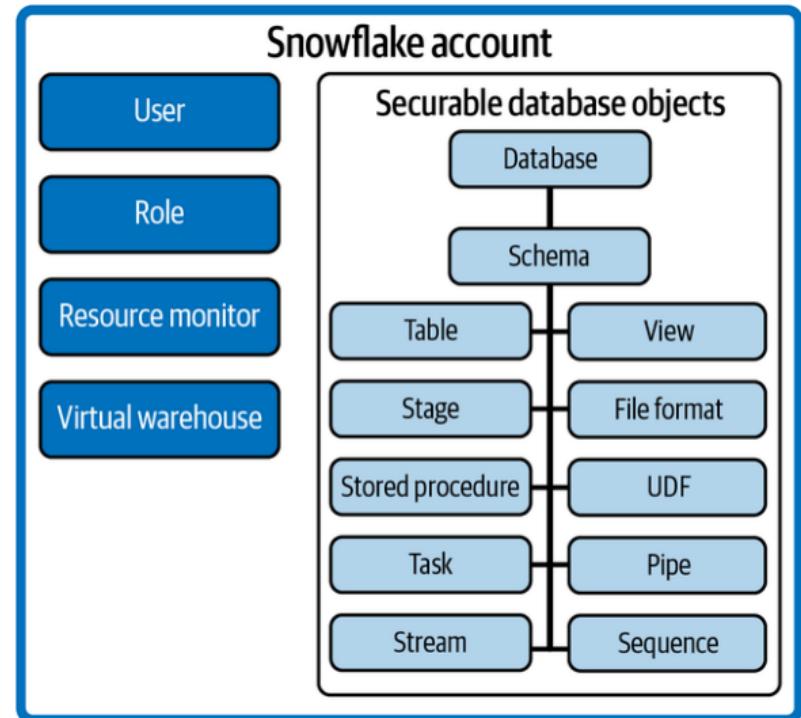
Creating & Managing Snowflake Securable Database Objects

Within Snowflake, all data is stored in database tables. Snowflake database tables are logically structured into collections of rows and columns.

A Snowflake securable object is an entity for which you grant access to specific roles. Roles, which have been granted access privileges, are assigned to users. A Snowflake user can be either an individual person or an application.

In a relational environment, database objects such as tables and views are maintained within databases. In Snowflake, the database logically groups the data while the schema organizes it. Together, the database and schema comprise the namespace.

We can create two main types of databases: permanent (persistent) and transient. At the time we create a database, the default will be a permanent database, if we don't specify which of the two types we want to create.

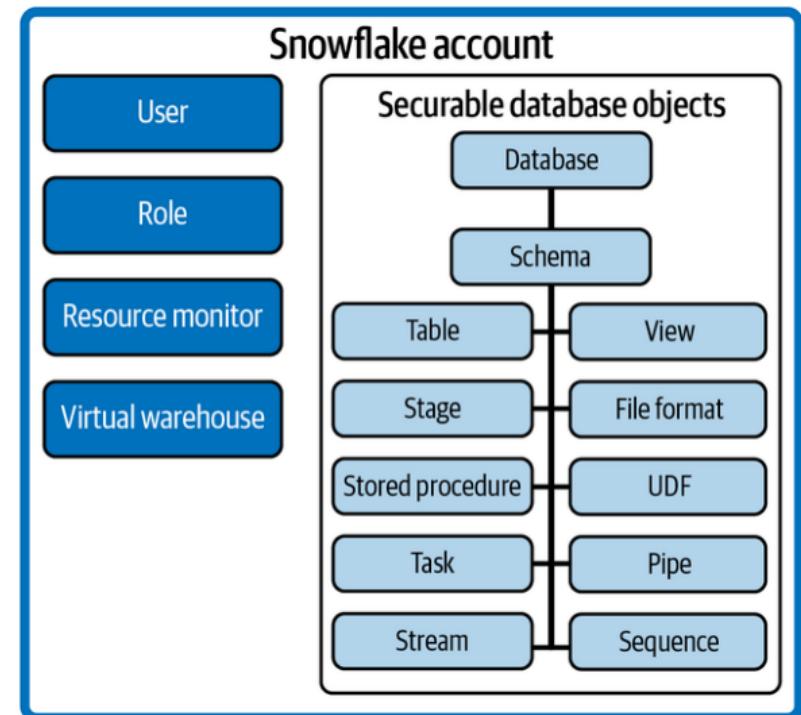


Creating & Managing Snowflake Securable Database Objects

Transient databases have a maximum one-day data retention period, aka Time Travel period, and do not have a fail-safe period.

The Snowflake Time Travel period is the time during which table data within the database can be queried at a historical point in time. This also enables databases and database objects to be cloned or undropped and historical data to be restored. The default Time Travel period is one day but can be up to 90 days for permanent databases; or a user could set the Time Travel period to zero days if no Time Travel period is desired. Note that the Enterprise Edition or higher is necessary to take advantage of the 90-day Time Travel period.

Snowflake's fail-safe data recovery service provides a seven-day period during which data from permanent databases and database objects may be recoverable by Snowflake. The fail-safe data recovery period is the seven-day period after the data retention period ends. Unlike Time Travel data, which is accessible by Snowflake users, fail-safe data is recoverable only by Snowflake employees.



Creating & Managing Snowflake Securable Database Objects

Data retention time (in days) is the same as the Time Travel number of days and specifies the number of days for which the underlying data is retained after deletion, and for which the CLONE and UNDROP commands can be performed on the database.

You can't change the data retention time in days for a transient database to something more than 1. Max you can set is 1.

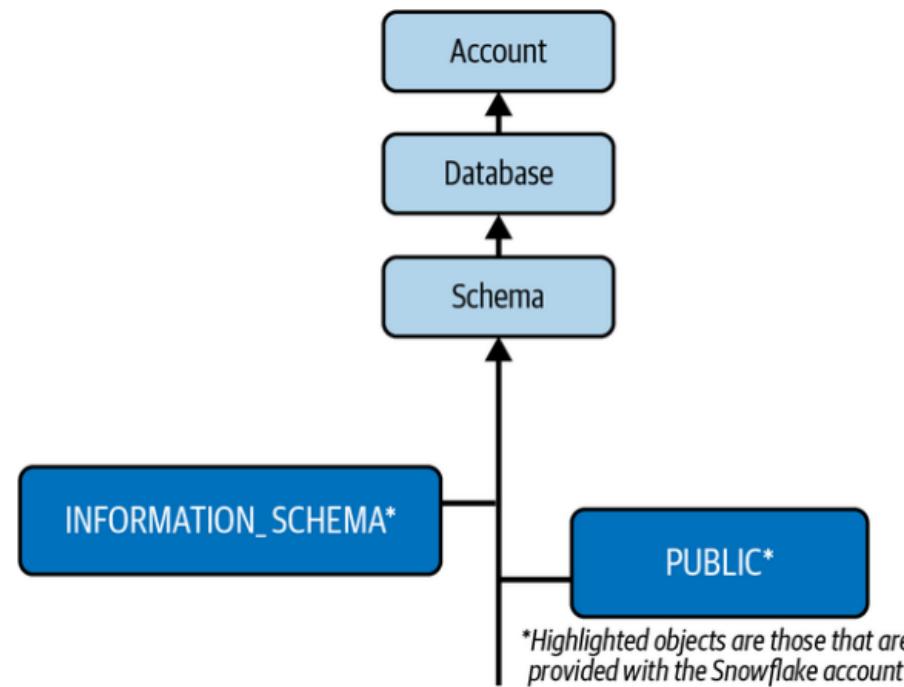
If we change the data retention time in days to 0 (zero) then we can't use CLONE or UNDROP features.

Snowflake uses a mixed approach when it comes to permanent databases but not transient databases. Permanent databases are not limited to the different types of objects that can be stored within them. For example, you can store transient tables within a permanent database, but you cannot store permanent tables within a transient database.

Creating & Managing Snowflake Securable Database Objects

The Information Schema and PUBLIC Schema (Default with any database)

Two database schemas, as shown in below, are included in every database that is created: INFORMATION_SCHEMA and PUBLIC. The PUBLIC schema is the default schema and can be used to create any other objects, whereas the INFORMATION_SCHEMA is a special schema for the system that contains views and table functions which provide access to the metadata for the database and account.



Creating & Managing Snowflake Securable Database Objects

Information Schema

INFORMATION_SCHEMA, also known as the Data Dictionary, includes metadata information about the objects within the database as well as account-level objects such as roles.

More than 20 system-defined views are included in every INFORMATION_SCHEMA. These views can be divided into two categories: account views and database views.

INFORMATION_SCHEMA account views include the following.

- APPLICABLE_ROLES
 - Displays one row for each role grant
- DATABASES
 - Displays one row for each database defined in your account
- ENABLED_ROLES
 - Displays one row for each currently enabled role in the session
- INFORMATION_SCHEMA_CATALOG_NAME
 - Displays the name of the database in which the INFORMATION_SCHEMA resides
- LOAD_HISTORY
 - Displays one row for each file loaded into tables using the COPY INTO <table> command, and returns history for all the data loaded in the past 14 days except for data loaded using SnowPIPE
- REPLICATION_DATABASES
 - Displays one row for each primary and secondary database in your organization

Creating & Managing Snowflake Securable Database Objects

Information Schema

INFORMATION_SCHEMA database views include the following views:

- COLUMNS
 - Displays one row for each column in the tables defined in the specified (or current) database
- EXTERNAL_TABLES
 - Displays one row for each external table in the specified (or current) database
- FILE_FORMATS
 - Displays one row for each file format defined in the specified (or current) database
- FUNCTIONS
 - Displays one row for each UDF or external function defined in the specified (or current) database
- OBJECT_PRIVILEGES
 - Displays one row for each access privilege granted for all objects defined in your account
- PIPES
 - Displays one row for each pipe defined in the specified (or current) database
- PROCEDURES
 - Displays one row for each stored procedure defined for the specified (or current) database
- REFERENTIAL_CONSTRAINTS
 - Displays one row for each referential integrity constraint defined in the specified (or current) database
- SCHEMATA
 - Displays one row for each schema in the specified (or current) database
- SEQUENCES
 - Displays one row for each sequence defined in the specified (or current) database
- STAGES
 - Displays one row for each stage defined in the specified (or current database)
- TABLE_CONSTRAINTS
 - Displays one row for each referential integrity constraint defined for the tables in the specified (or current) database
- TABLE_PRIVILEGES
 - Displays one row for each table privilege that has been granted to each role in the specified (or current) database

Creating & Managing Snowflake Securable Database Objects

Information Schema

- TABLE_STORAGE_METRICS
 - Displays table-level storage utilization information, includes table metadata, and displays the number of storage types billed for each table. Rows are maintained in this view until the corresponding tables are no longer billed for any storage, regardless of various states that the data in the tables may be in (i.e., active, Time Travel, fail-safe, or retained for clones).
- TABLES
 - Displays one row for each table and view in the specified (or current) database.
- USAGE_PRIVILEGES
 - Displays one row for each privilege defined for sequences in the specified (or current) database
- VIEWS
 - Displays one row for each view in the specified (or current) database.

We'll see in the query results that the INFORMATION_SCHEMA database views will return results specific to a database.

Creating & Managing Snowflake Securable Database Objects

ACCOUNT_USAGE Schema

The SNOWFLAKE database, viewable by the ACCOUNTADMIN by default, includes an ACCOUNT_USAGE schema that is very similar to the INFORMATION_SCHEMA, but with three differences:

- The SNOWFLAKE database ACCOUNT_USAGE schema includes records for dropped objects whereas the INFORMATION_SCHEMA does not.
- The ACCOUNT_USAGE schema has a longer retention time for historical usage data. Whereas the INFORMATION_SCHEMA has data available ranging from seven days to six months, the ACCOUNT_USAGE view retains historical data for one year.
- Most views in the INFORMATION_SCHEMA have no latency, but the latency time for ACCOUNT_USAGE could range from 45 minutes to three hours. Specifically, for the INFORMATION_SCHEMA, there may be a one- to two-hour delay in updating storage-related statistics for ACTIVE_BYTES, TIME_TRAVEL_BYTES, FAILSAFE_BYTES, and RETAINED_FOR_CLONE_BYTES.

Creating & Managing Snowflake Securable Database Objects

Schema Object hierarchy

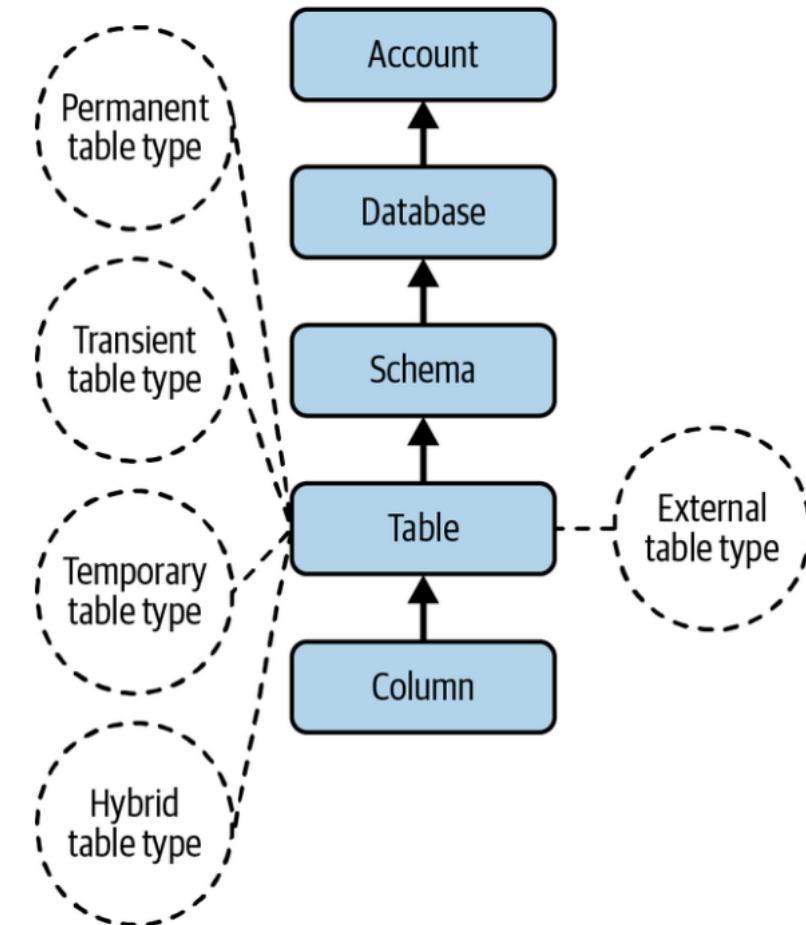
Many objects exist within a Snowflake schema object, including tables, views, stages, policies, stored procedures, UDFs, and more.

Introduction to Snowflake Tables

As previously mentioned, all Snowflake data is stored in tables. In addition to permanent and transient tables, it is also possible to create hybrid, temporary, and external tables, as shown in Figure below. Snowflake hybrid tables support the new Unistore workload.

Snowflake temporary tables only exist within the session in which they were created and are frequently used for storing transitory data such as ETL data.

Snowflake external tables give you the ability to directly process or query your data that exists elsewhere without ingesting it into Snowflake, including data that lives in a data lake.



Creating & Managing Snowflake Securable Database Objects

Transient tables are unique to Snowflake and have characteristics of both permanent and temporary tables. Transient tables are designed for transitory data that needs to be maintained beyond a session but doesn't need the same level of data recovery as permanent tables. As a result, the data storage costs for a transient table would be less than for a permanent table. One of the biggest differences between transient tables and permanent tables is that the fail-safe service is not provided for transient tables.

It isn't possible to change a permanent table to a transient table by using the ALTER command, because the TRANSIENT property is set at table creation time. Likewise, a transient table cannot be converted to a permanent table. If you would like to make a change to a transient or permanent table type, you'll need to create a new table, use the COPY GRANTS clause, and then copy the data. Using the COPY GRANTS clause will ensure that the table will inherit any explicit access privileges.

Characteristics	Permanent table	Transient table	Temporary table	External table
Persistence	Until explicitly dropped	Until explicitly dropped	Remainder of session	Until explicitly dropped
Time Travel retention (days)	0–90 ^a	0 or 1	0 or 1	0
Fail-safe period (days)	7	0	0	0
Cloning possible?	Yes	Yes	Yes	No
Create views possible?	Yes	Yes	Yes	Yes

^a In Enterprise Edition and above, 0–90 days. In Standard Edition, 0 or 1 day.

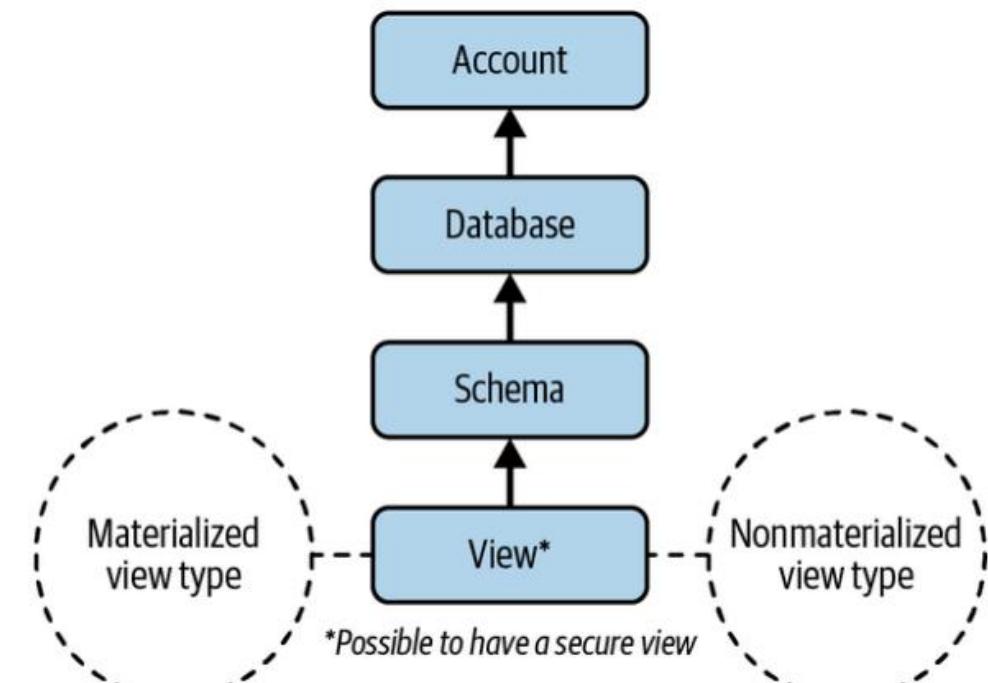
Creating & Managing Snowflake Securable Database Objects

Creating and Managing Views

Views are of two types: materialized and nonmaterialized. Whenever the term view is mentioned and the type is not specified, it is understood that it is a nonmaterialized view.

Materialized views are generally used to aggregate as well as filter data so that the results of resource-intensive operations can be stored in a materialized view for improved data performance. The performance improvement is especially good when that same query is used frequently.

Snowflake uses a background service to automatically update materialized views. As a result, data accessed through materialized views is always current, regardless of the amount of DML that has been performed on the base table. Snowflake will update the materialized view or use the up-to date portions of the materialized view and retrieve newer data from the base table if a query is run before the materialized view is up to date.



Creating & Managing Snowflake Securable Database Objects

As a rule, it is best to use a nonmaterialized view when the results of the view change frequently and the query isn't so complex and expensive to rerun. Regular views do incur compute costs but not storage costs. The compute cost to refresh the view and the storage cost will need to be weighed against the benefits of a materialized view when the results of a view change often.

Generally, it is beneficial to use a materialized view when the query consumes a lot of resources, as well as when the results of the view are used often, and the underlying table doesn't change frequently. Also, if a table needs to be clustered in multiple ways, a materialized view can be used with a cluster key that is different from the cluster key of the base table.

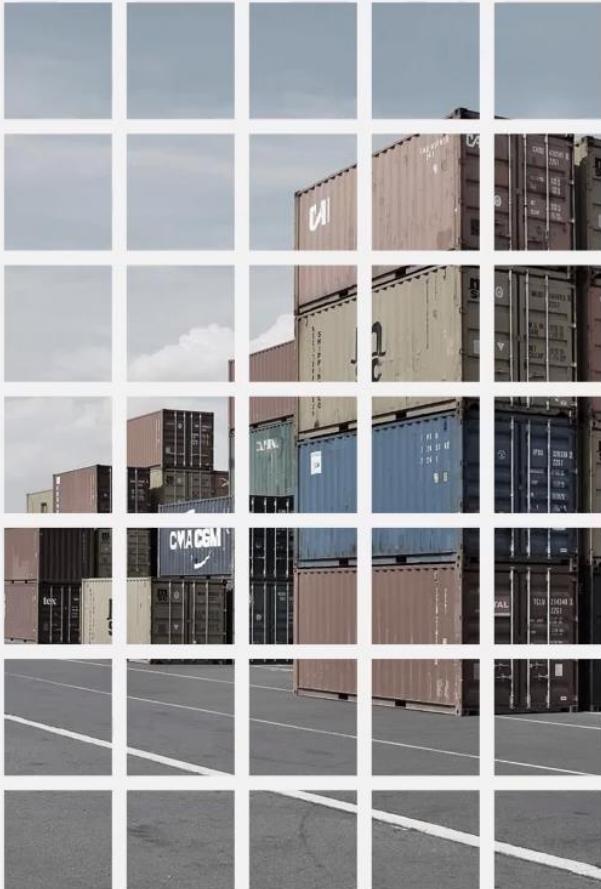


LOADING DATA INTO SNOWFLAKE

LET'S LOOK AT HOW WE CAN LOAD DATA INTO SNOWFLAKES



DATA LOADING OPTIONS



BULK LOAD

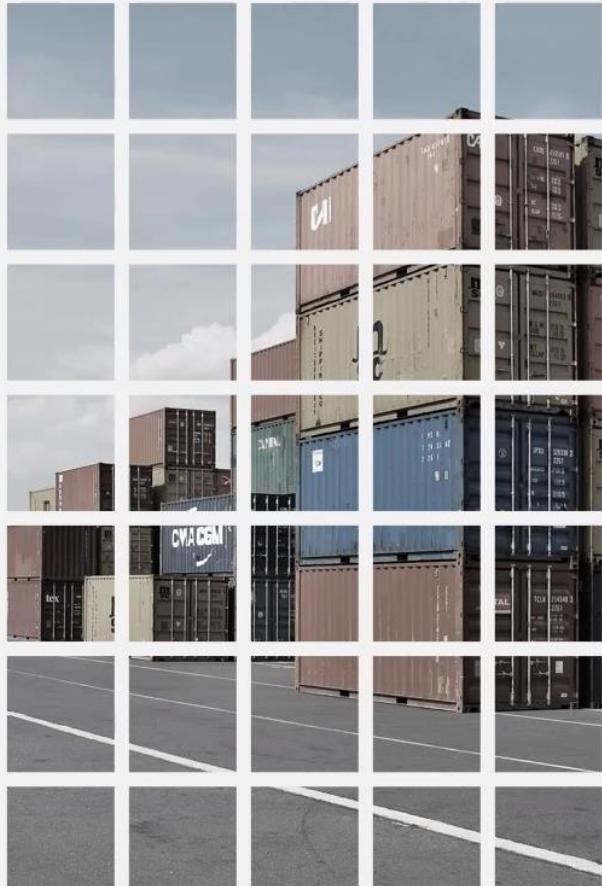
- Snowflake COPY command used for batch loading
- Batch Loading of data which is already available in Cloud Storage or an internal location
- COPY command uses Virtual Warehouse compute resources which need to be managed manually
- Allows basic Transformations such as re-ordering columns, excluding columns, data typing, truncating strings



CONTINUOUS LOAD

- Snowpipe used for loading streaming data
- Uses a serverless approach scaling up / down automatically
- Doesn't use the virtual warehouse compute resources

QUERY DATA WITHOUT LOADING DATA



EXTERNAL TABLES

- Its not always necessary to load data into Snowflake before you can access it
- In such cases you can use the external tables to access data externally
- This is useful if there is a lot of data externally but you want to query only a small subset of that data
- The external table performance and associate costs can be optimised by creating materialised views

LOADING BULK DATA FROM CLOUD & LOCAL STORAGE



Prepare your files

If required pre-process the files into a format that is optimal for loading



Stage the data

Make snowflake aware of the data. Internal or external staging area



Execute COPY command

COPY the data into the table



Managing regular loads

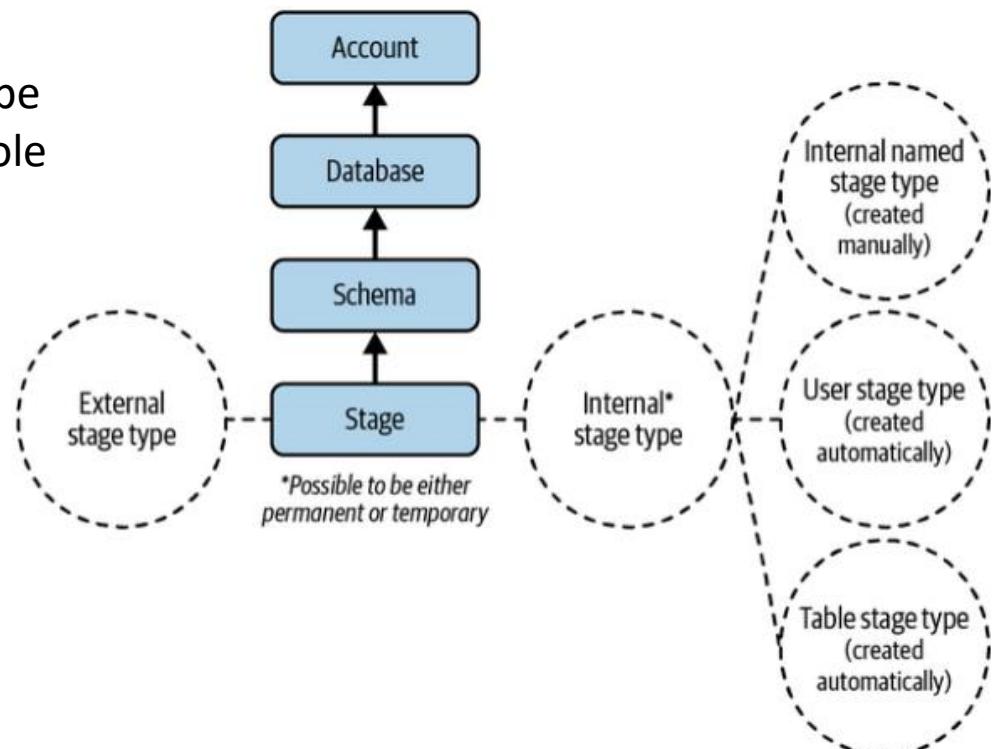
Organize files & schedule your loads

Introduction to Snowflake Stages: File format included

There are two types of Snowflake stages: internal and external. Stages are Snowflake objects that point to a storage location, either internal to Snowflake or on external cloud storage. Internal stage objects can be either named stages or a user or table stage. The temporary keyword can be used to create a session-based named stage object.

In most cases, the storage is permanent while the stage object, a pointer to the storage location, may be temporary or dropped at any time. Snowflake stages are often used as an intermediate step to load files to Snowflake tables or to unload data from Snowflake tables into files.

Internal named stages are database objects, which means they can be used not just by one user but by any user who has been granted a role with the appropriate privileges.



Snowflake Stages

Supported File Locations

Snowflake refers to the location of the data files in cloud storage as a “stage”.

The COPY INTO <table> command is used for both bulk and continuous data loads.

COPY command supports cloud storage accounts managed by your business entity (i.e. external stages) as well as cloud storage contained in your Snowflake Account (i.e. internal stages)



External Stages

Loading data from Cloud Storage Services

- Amazon S3
- Google Cloud Storage
- Microsoft Azure

You can't access data held in archival cloud storage classes that require restoration before it can be retrieved

A named external stage is a database object created in a schema.

The object stores the URL to files in the cloud storage, settings used to access cloud storage account, format of the files .

Upload (i.e. staged) files to your cloud storage account

Internal Stages

Snowflake maintains the following stages in your account.

- User
 - A user stage is allocated to each user for storing files. This stage type is designed to store files that are staged and managed by a single user but can be loaded into multiple tables. User tables can't be altered or dropped
- Table
 - Available for each table created in snowflake. Designed to store files that are staged and managed by one or more users but only loaded into single table
- Named
 - Is a database object created in a schema. This can store files that are staged and managed by one or more users and loaded into multiple tables. Can be modified, dropped using access control privileges.

PREPARE YOUR FILES



Delimited Text Files

- Fields should be delimited by a single character. E.g. pipe , comma, caret or tilde
- Rows delimited by a different character. Usual new line character is a common choice
- Number of columns in each row should be consistent
- If a field contains the delimiter character, the field should be enclosed in double quotes

Optimize file sizes

- To take advantage of parallelism optimal compress file size is 10MB to 100 MB
- so, split very large files into multiple chunks
- and, merge multiple very small file into a single file to achieve optimal sizes
- For continuous loads via Snowpipe the recommended sizing approach is different

Data Types

- Numeric data types shouldn't have embedded characters eg. 123,456 should be 123456
- Date Time data type should be consistent and according to a format e.g. 2019-09-23

STAGE THE DATA



Why

- Staging area is an intermediate, transient area used to process data for any of the extract, transform and load processes.
- The concept of 'Stage' is used to access temporary area, where data that is to be loaded can be accessed by Snowflake.
- A stage may be for cloud based data or could be for data you upload from your PC.

Staging Data for Cloud Storage

- The most common method for loading data into Snowflake is to stage the data in an S3 bucket or an Azure blob store.
- The S3 bucket or Azure blob store can also provide long term cheap storage for your raw data.

Staging Data on Internal Storage

Data can also be staged on your local file system before loading it into Snowflake

Commands

```
CREATE STAGE <stage_name> url='cloud_storage_url' credentials='<login_password>.'
```

EXECUTE COPY COMMAND



Bulk copy data

- Using the COPY INTO command is the common mechanism for loading data into Snowflake in a batch mode
- The COPY INTO command supports several options to specify the files to load
 - By specifying the path to load
 - By specifying the file names to be loaded (this is the method we have used so far in this course)
 - Using a pattern matching to load only files matching the pattern

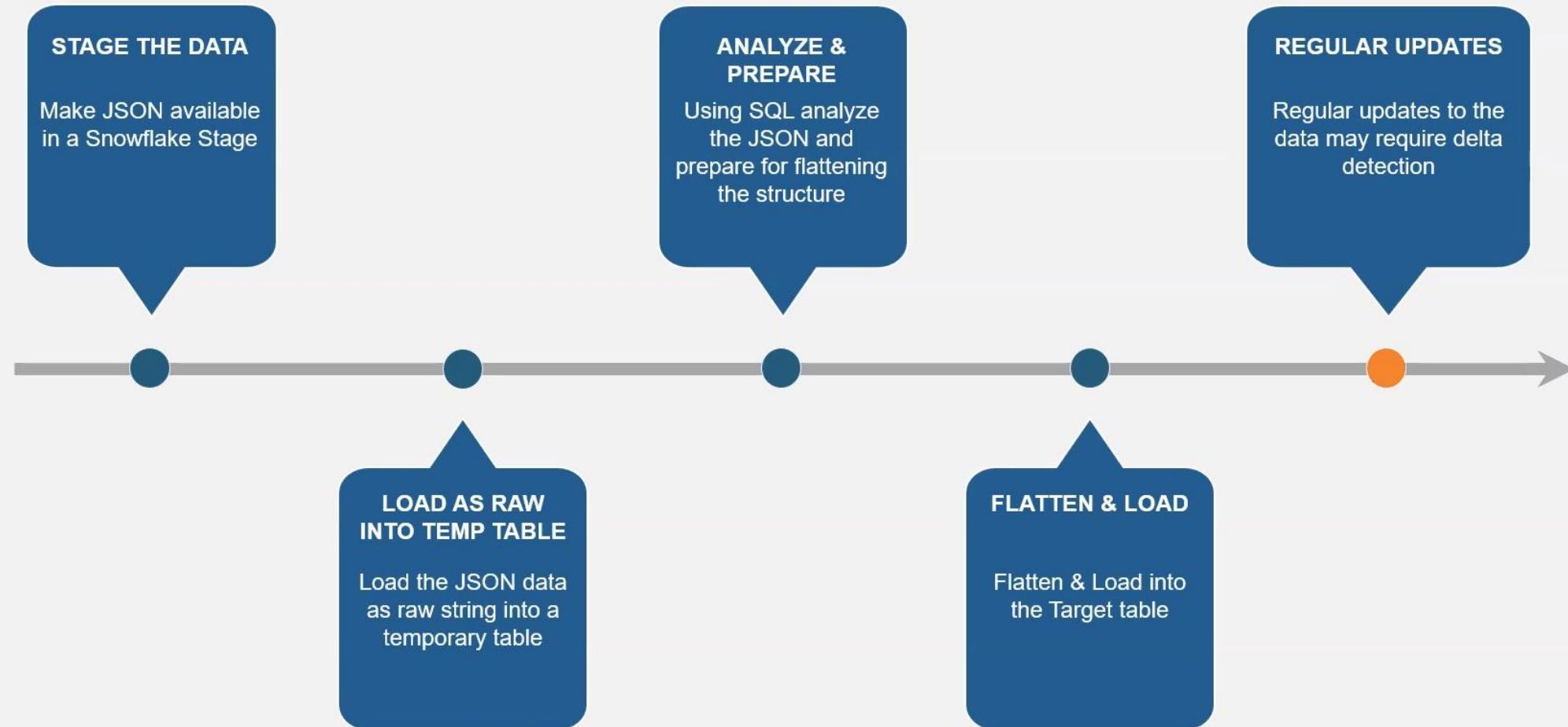
Commands

```
copy into <table_name>
  from @<stage_name>
  pattern='*.csv'
  file_format = (type = csv field_delimiter = '|' skip_header = 1);
```

Let's load our data into Snowflake

- Hands on activities

LOADING JSON DATA – VIA TEMPORARY TABLE



SNOWPIPE



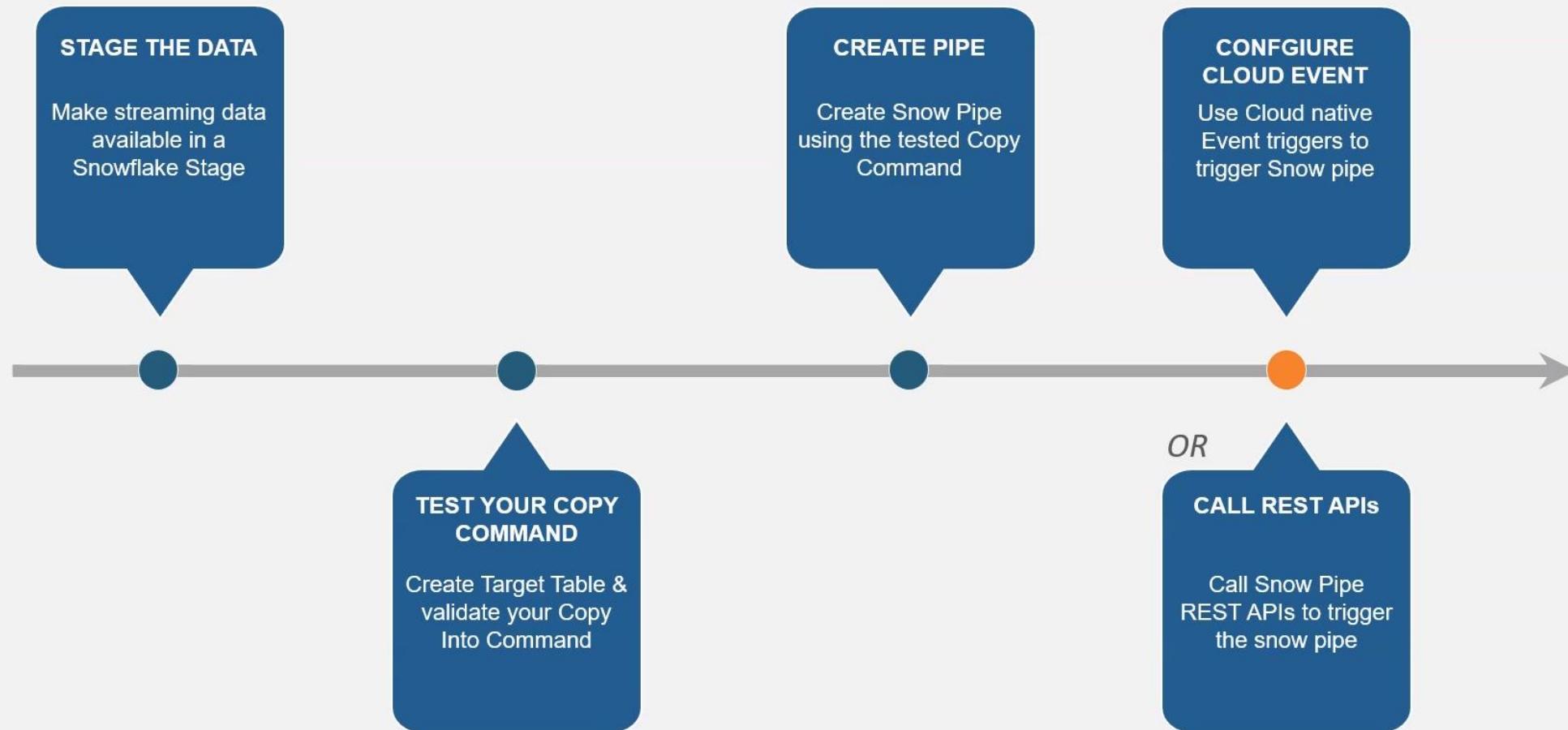
What is Snowpipe

- Snowpipe is a mechanism to enable loading of data as soon as it becomes available in a stage
- Using Snowpipe you can achieve micro-batched loading of data
- It is usually used where there is continuously arriving data like transactions or events and there is need to make that data available to business immediately
- Snowpipe uses server less architecture, so it will not use an virtual warehouse instance but has its own processing and is billed differently

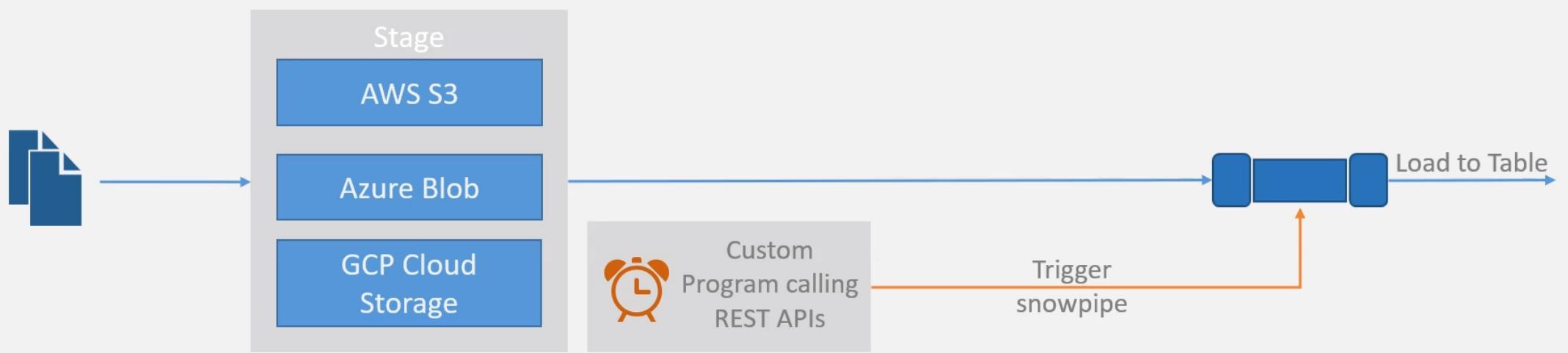
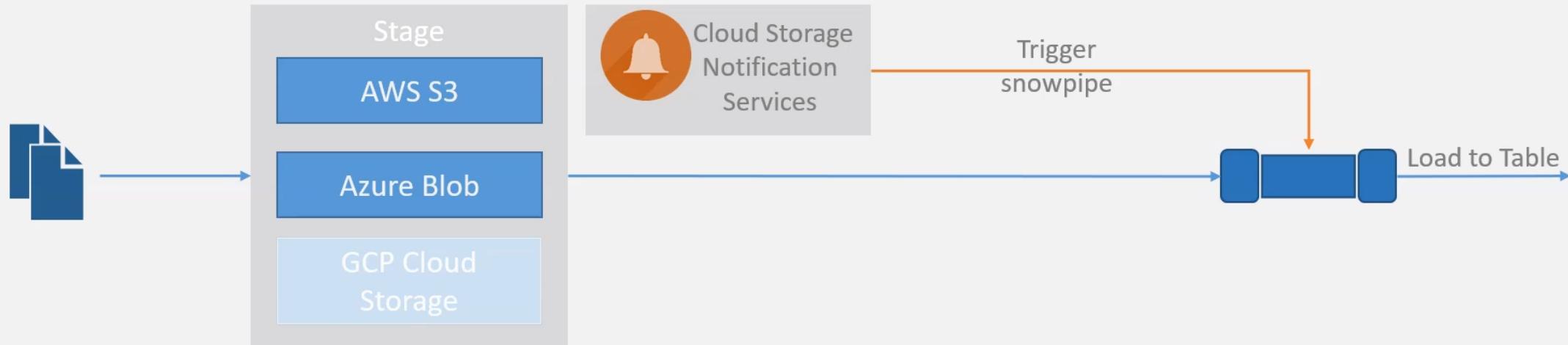
How does Snowpipe work

- Snowpipe definitions contain a COPY statement which is used by Snowflake to load the data
- The Snowpipe may be automatically or manually triggered to load data

LOADING DATA USING SNOWPIPE



LOADING STREAMING DATA via SNOWPIPE



Snowflake Architecture

snowsql installer download

<https://sfc-repo.snowflakecomputing.com/snowsql/bootstrap/index.html>

<https://sfc-repo.snowflakecomputing.com/index.html>

Hands On Exercises



PERFORMANCE OPTIMIZATION

LET'S LEARN WHAT & HOW TO OPTIMIZE SNOWFLAKES DATA WAREHOUSE PERFORMANCE

PERFORMANCE OPTIMISATION CONSIDERATIONS

1

Use dedicated Virtual Warehouses

Segregate the virtual warehouses for different workloads

2

Scale up for known large workloads

Scale the size of the virtual warehouse according to known patterns

3

Scale out for unknown & unexpected workloads

Auto spawn virtual warehouse based on workloads

4

Design to maximize cache usage

Caching is automatic, but usage can be maximized

5

Use cluster keys to partition large tables

For tables that have very large data use cluster keys to improve query performance

USE DEDICATED VIRTUAL WAREHOUSE



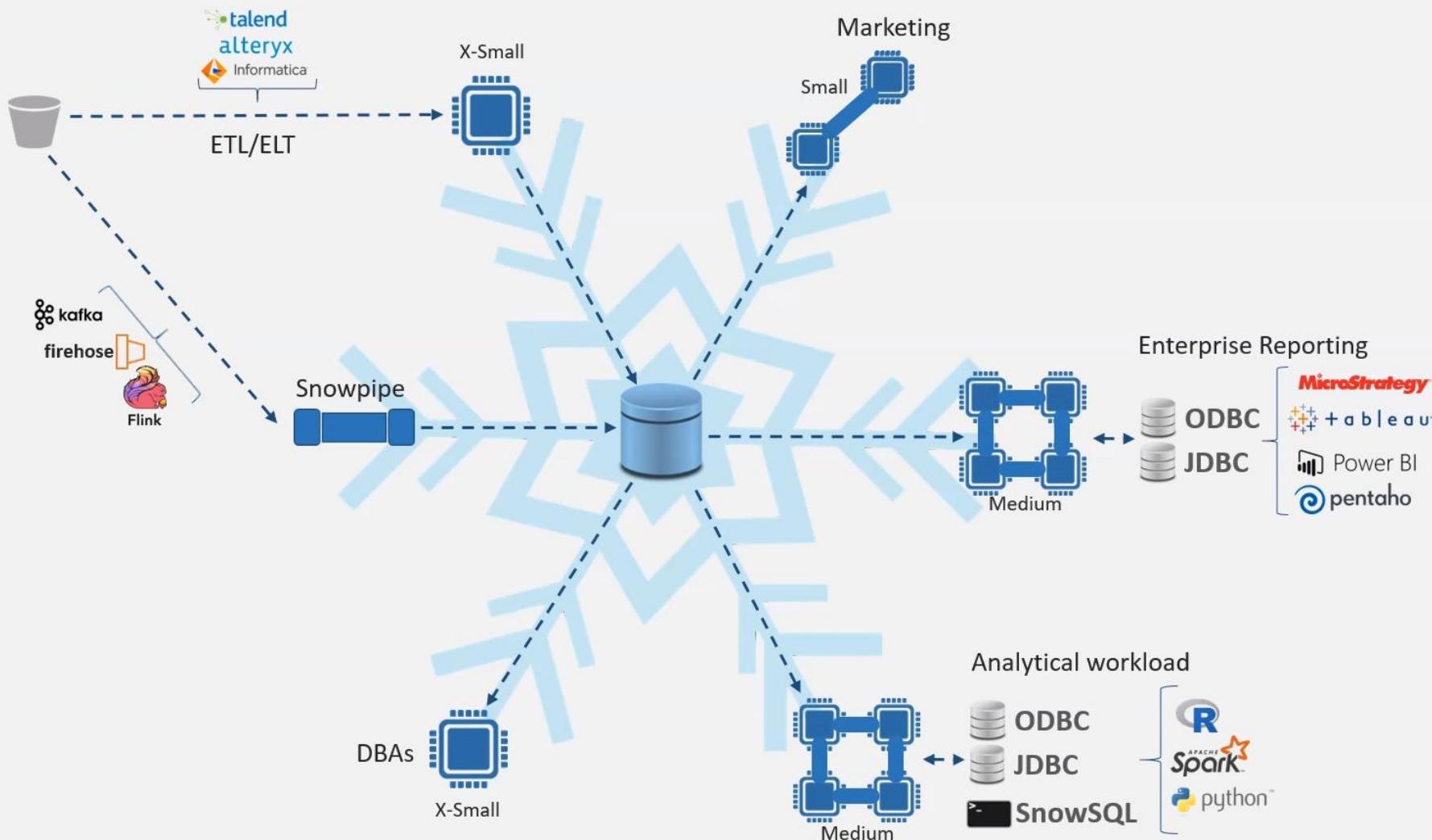
Segregate your workloads to different virtual warehouses

- Identify & categorize workloads executing on your snowflake implementation e.g. ETL, Data Science, Business Intelligence, Ad-hoc
- Identify & categorize your users executing those workloads
- Create dedicated virtual warehouse for each workload classification & attach to your users as a default virtual warehouse for their queries (don't go too fine grained)
- This classification of workloads & users will need to be refined every now and then, depending on how the workload patterns are changing in your Snowflake implementation

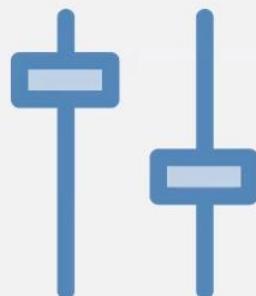
Let's demonstrate this concept

- Let's create dedicated virtual warehouse and allocate to users

USE DEDICATED VIRTUAL WAREHOUSE



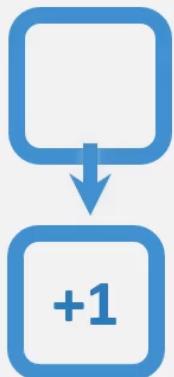
SCALE UP – INCREASE THE SIZE OF VIRTUAL WAREHOUSE



Increase / Decrease the size of the virtual warehouse as per demand

- You can increase the size of your virtual warehouse as a response to change in workloads
- This can be an ad-hoc event e.g. an urgent business requirements results in a rush of complex queries, so you can increase the size of the virtual temporarily to better service those queries
- This could also be as a response to a pattern repeatedly occurring e.g. your ETL may run throughout the day but has peak processing periods during 12:00 AM to 2:00M.
- You would normally scale up in scenarios where the query complexity has increased. If the number of users performing these queries increases exponentially then scale out may be a better option.

SCALE OUT



Configure Virtual warehouse to scale automatically

- Virtual warehouse can be configured to scale out automatically on periods of demand and scale back as the demand decreases. This is achieved by automatically spawning an exact replica of the virtual warehouse
- This is also referred to as Multi-cluster Warehouses. Multi-cluster Warehouses are only available to Snowflake users who subscribe to Snowflake Enterprise or higher license
- It requires setting a minimum and a maximum number of virtual warehouses. When autoscaling, new virtual warehouses up to the maximum number will be spawned and similarly when scaling back it will scale back to the minimum number
- Multi-cluster virtual warehouse have advantages compared to adding new virtual warehouses on demand, since if you opt for manually adding new virtual warehouse you will need to redirect users to new virtual warehouse

MAXIMISE CACHE USAGE

Caching is automatic

- Caching is automatic in Snowflake, so if you execute an identical query twice the chances are that you will be served the results from the cache. Results are cached for 24 hours (from the point of last re-use)
- Because the metadata management in Snowflake is quite strong, Snowflake knows when the underlying data has changed and therefore re-execute the query

Allocate virtual warehouse to run common queries on same warehouse

- To maximize cache usage ensure similar queries go to the same virtual warehouse.
- For example if your BI tool runs a count by region query many times during report refresh, having those queries run on same virtual warehouse will ensure cache utilization.

Let's demonstrate this concept

- Let's execute some queries to see how they get served through the cache



Table Clustering and Micro Partitions

There are two traditional partitioning methods. Horizontal Partitioning and Vertical Partitioning

Let's take below table as an example:

CID	Fname	Lname	Reward	Zip	Channel	Amount	Date
1	Arnold	Johnson	Gold	94015	Web	\$2,500	8/7/2022
2	Janice	Switzer	Bronze	76012	Retail	\$450	8/7/2022
3	Amy	Majors	Gold	76015	Web	\$1,220	8/7/2022
6	Marty	Anders	Silver	45032	Direct	\$3,315	8/7/2022
3	Amy	Majors	Gold	76012	Web	\$795	8/7/2022
7	Harold	Webb	Silver	94011	Web	\$490	8/8/2022
5	Nathan	Harris	Silver	76010	Retail	\$1,115	8/8/2022
4	Cathy	Amos	Silver	45615	Web	\$635	8/8/2022



Table Clustering and Micro Partitions

Horizontal Partitioning

Horizontal partition #1 (Last name A-J)

CID	Fname	Lname	Reward	Zip	Channel	Amount	Date
1	Arnold	Johnson	Gold	94015	Web	\$2,500	8/7/2022
6	Marty	Anders	Silver	45032	Direct	\$3,315	8/7/2022
5	Nathan	Harris	Silver	76010	Retail	\$1,115	8/8/2022
4	Cathy	Amos	Silver	45615	Web	\$635	8/8/2022

Horizontal partition #2 (Last name K-Z)

CID	Fname	Lname	Reward	Zip	Channel	Amount	Date
2	Janice	Switzer	Bronze	76012	Retail	\$450	8/7/2022
3	Amy	Majors	Gold	76015	Web	\$1,220	8/7/2022
3	Amy	Majors	Gold	76012	Web	\$795	8/7/2022
7	Harold	Webb	Silver	94011	Web	\$490	8/8/2022



Table Clustering and Micro Partitions

Vertical Partitioning

Vertical partition #1 (customer information)

CID	Fname	Lname	Reward	Zip
1	Arnold	Johnson	Gold	94015
2	Janice	Switzer	Bronze	76012
3	Amy	Majors	Gold	76015
6	Marty	Anders	Silver	45032
7	Harold	Webb	Silver	94011
5	Nathan	Harris	Silver	76010
4	Cathy	Amos	Silver	45615

Vertical partition #2 (transaction information)

CID	Channel	Amount	Date
1	Web	\$2,500	8/7/2022
2	Retail	\$450	8/7/2022
3	Web	\$1,220	8/7/2022
6	Direct	\$3,315	8/7/2022
3	Web	\$795	8/7/2022
7	Web	\$490	8/8/2022
5	Retail	\$1,115	8/8/2022
4	Web	\$635	8/8/2022



Table Clustering and Micro Partitions

Snowflake Micro Partitions

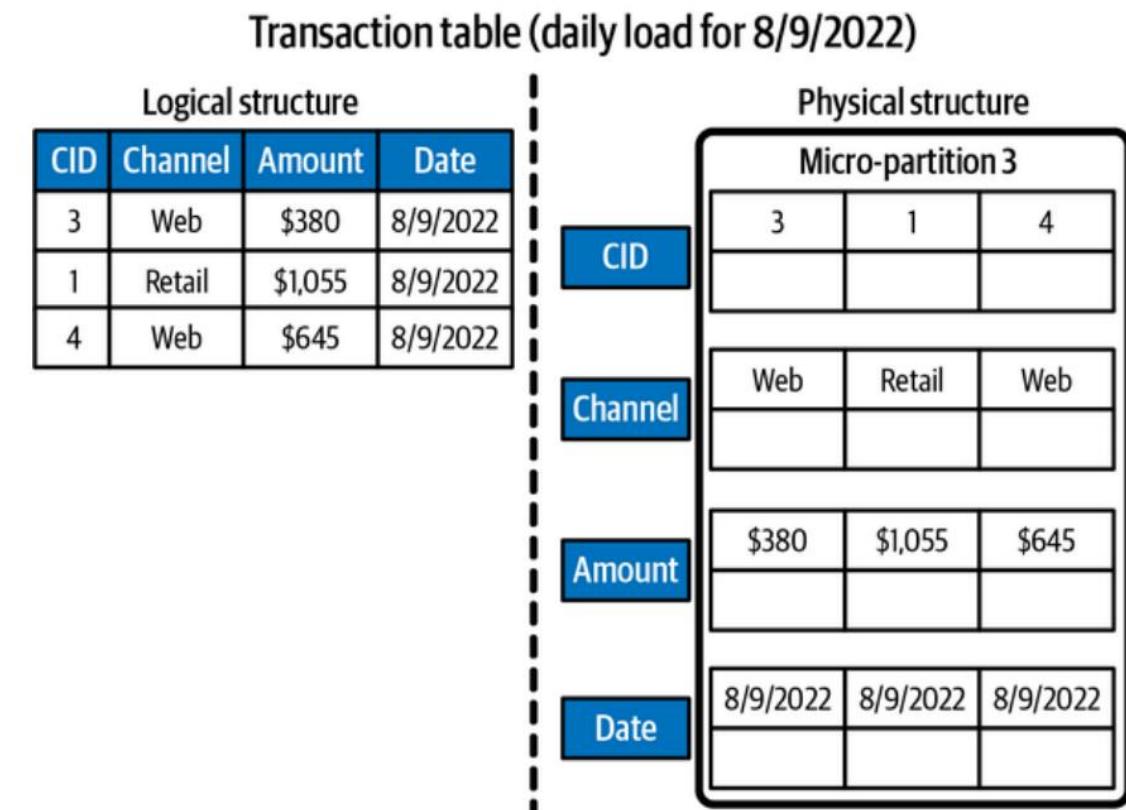
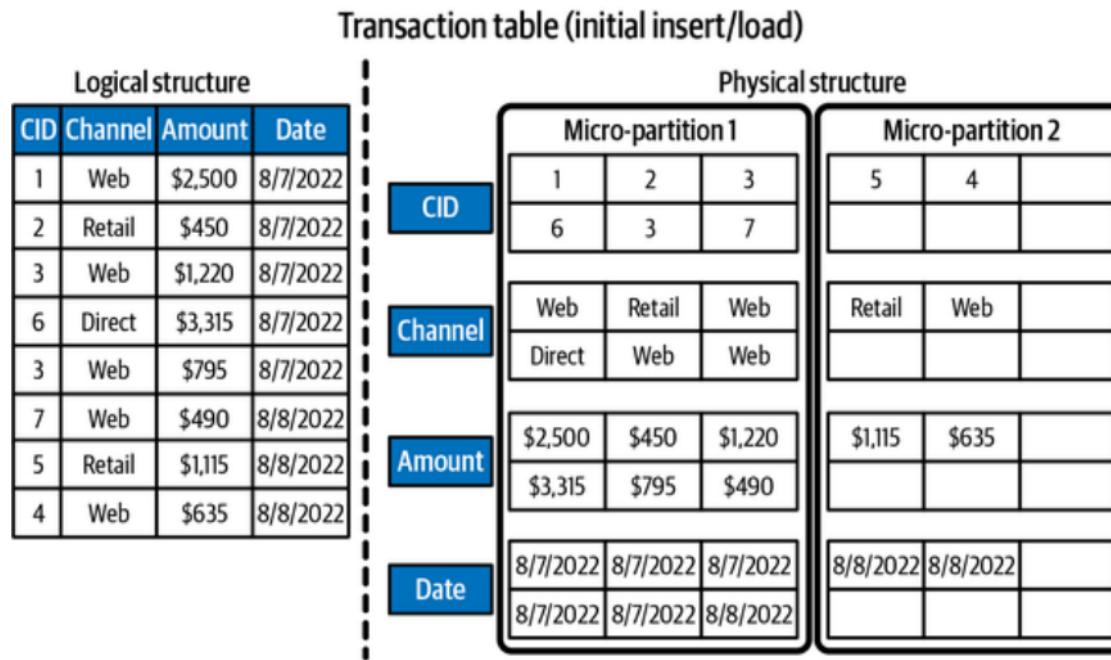
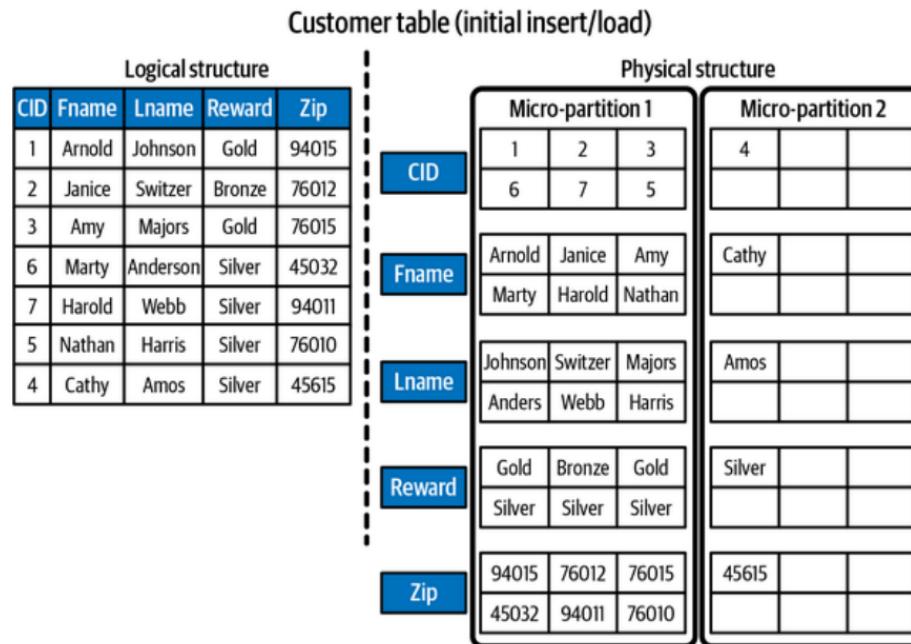




Table Clustering and Micro Partitions

Snowflake Micro Partitions



Customer table (updated zip code for Nathan Harris)

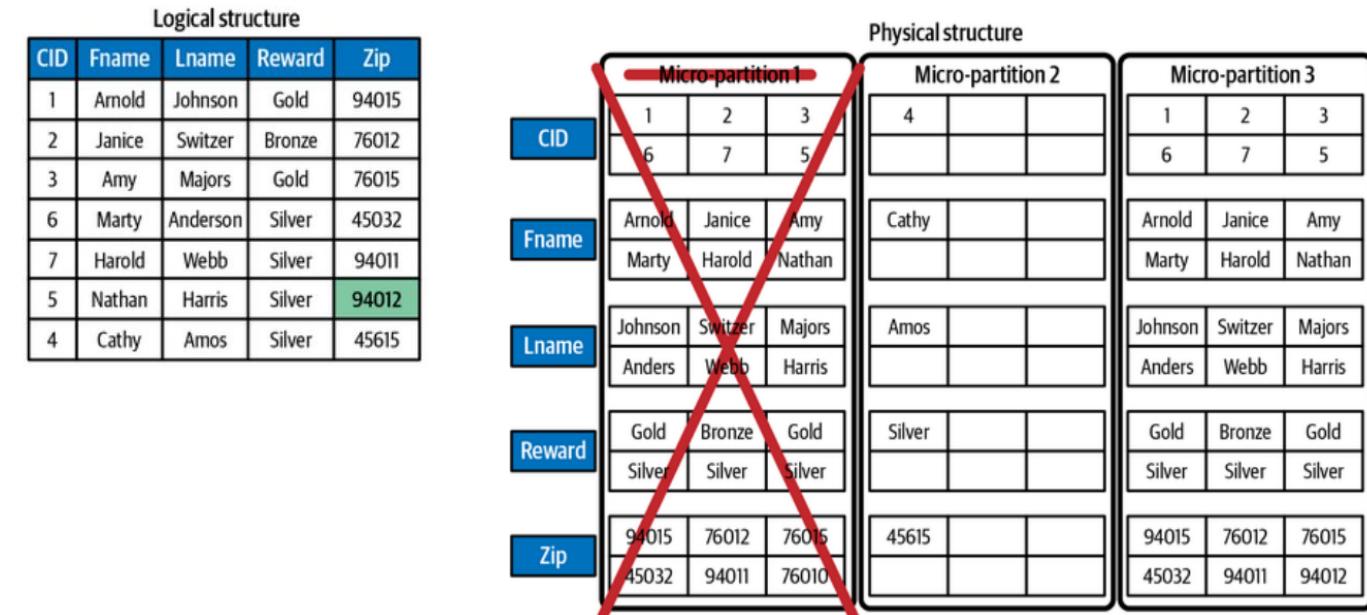




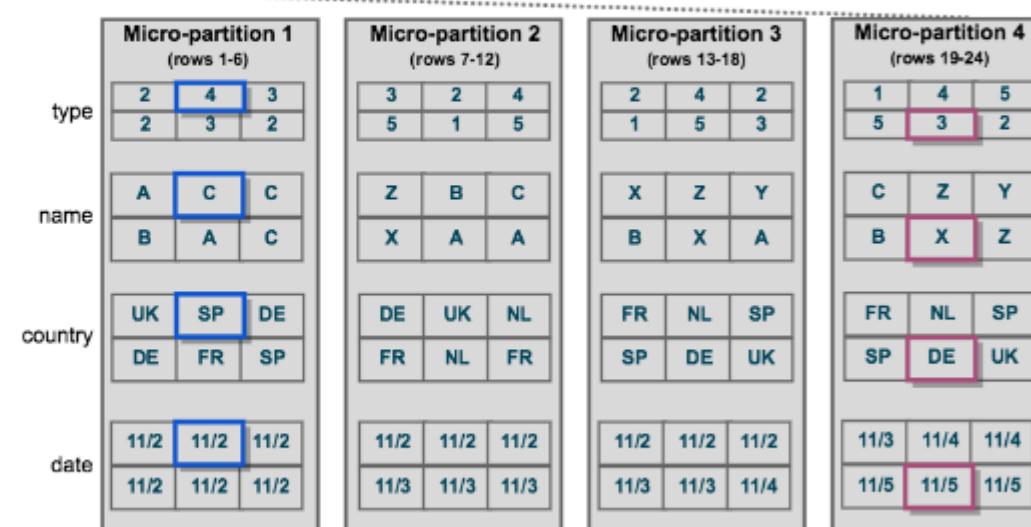
Table Clustering and Micro Partitions

Snowflake Table Clustering

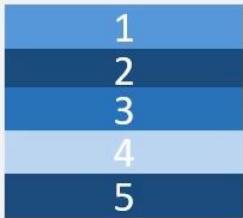
Logical Structure

type	name	country	date
2	A	UK	11/2
4	C	SP	11/2
3	C	DE	11/2
2	B	DE	11/2
3	A	FR	11/2
2	C	SP	11/2
3	Z	DE	11/2
2	B	UK	11/2
4	C	NL	11/2
5	X	FR	11/3
1	A	NL	11/3
5	A	FR	11/3
2	X	FR	11/2
4	Z	NL	11/2
2	Y	SP	11/2
1	B	SP	11/3
5	X	DE	11/3
3	A	UK	11/4
1	C	FR	11/3
4	Z	NL	11/4
5	Y	SP	11/4
5	B	SP	11/5
3	X	DE	11/5
2	Z	UK	11/5

Physical Structure



Partition your data – clustering keys



Very large tables can be partitioned to improve performance

- Snowflake maintains a table's micro partitions without requiring a DBA's involvement but ..
- For very large tables, over time the automatic partitioning scheme may not remain the most optimal
- You can custom cluster keys to partition the table according to your needs
- Normally you would use those columns in cluster keys which are frequently used in WHERE clauses or are frequently used in JOINS or ORDER BY statements
- There are other criterion as well on your choice of cluster keys for example cardinality of the cluster column
- Clustering is not for all tables, generally large tables in the multi terabyte size will benefit from clustering

Time taken to Insert the Data

The screenshot shows a user interface for managing data operations. At the top, there is a navigation bar with icons for Databases, Shares, Marketplace, Warehouses, Worksheets, History (which is selected), and Account.

The main area displays a history entry for an operation that took 3 minutes and 45 seconds, starting at 11:17:34 AM. The entry is categorized under 'Details'.

Key details from the history entry:

- Status: Success
- User: UCSC
- Warehouse: COMPUTE_WH
- Start Time: 11:17:34 AM
- End Time: 11:21:19 AM
- Total Duration: 3min 45s
- Scanned Bytes: 16KB
- Rows: 225M
- Query ID: 01a59ddb-0000-30fd-0003-a5560002702a
- Session ID: 1026213716406294

The 'SQL Text' section contains the following SQL code:

```
1 INSERT INTO COPY_DB.PUBLIC.ORDERS_CACHING
2 SELECT
3     t1.ORDER_ID
4     ,t1.AMOUNT
5     ,t1.PROFIT
6     ,t1.QUANTITY
7     ,t1.CATEGORY
8     ,t1.SUBCATEGORY
9     ,DATE(UNIFORM(1500000000,1700000000,(RANDOM())))
10    FROM ORDERS t1
11   CROSS JOIN (SELECT * FROM OUR_FIRST_DB.PUBLIC.ORDERS) t2
12   CROSS JOIN (SELECT TOP 100 * FROM OUR_FIRST_DB.PUBLIC.ORDERS) t3;
```

The 'Query Result' section shows the results of the query execution:

Results	
row#	number of rows inserted
1	225000000

SELECT Query without a WHERE Clause – Query Details

History > 11:31:32 AM for 21.0s Last refreshed 11:32:14 AM Auto refresh

[Details](#) [Profile](#)

Status	Success
User	UCSC
Warehouse	COMPUTE_WH
Start Time	11:31:32 AM
End Time	11:31:53 AM
Total Duration	21.0s
Scanned Bytes	1.1GB
Rows	225M
Query ID	01a59de9-0000-3100-0003-a5560002811a
Session ID	1026213716406294

SQL Text

```
1 | SELECT * FROM ORDERS_CACHING;
```

Query Result

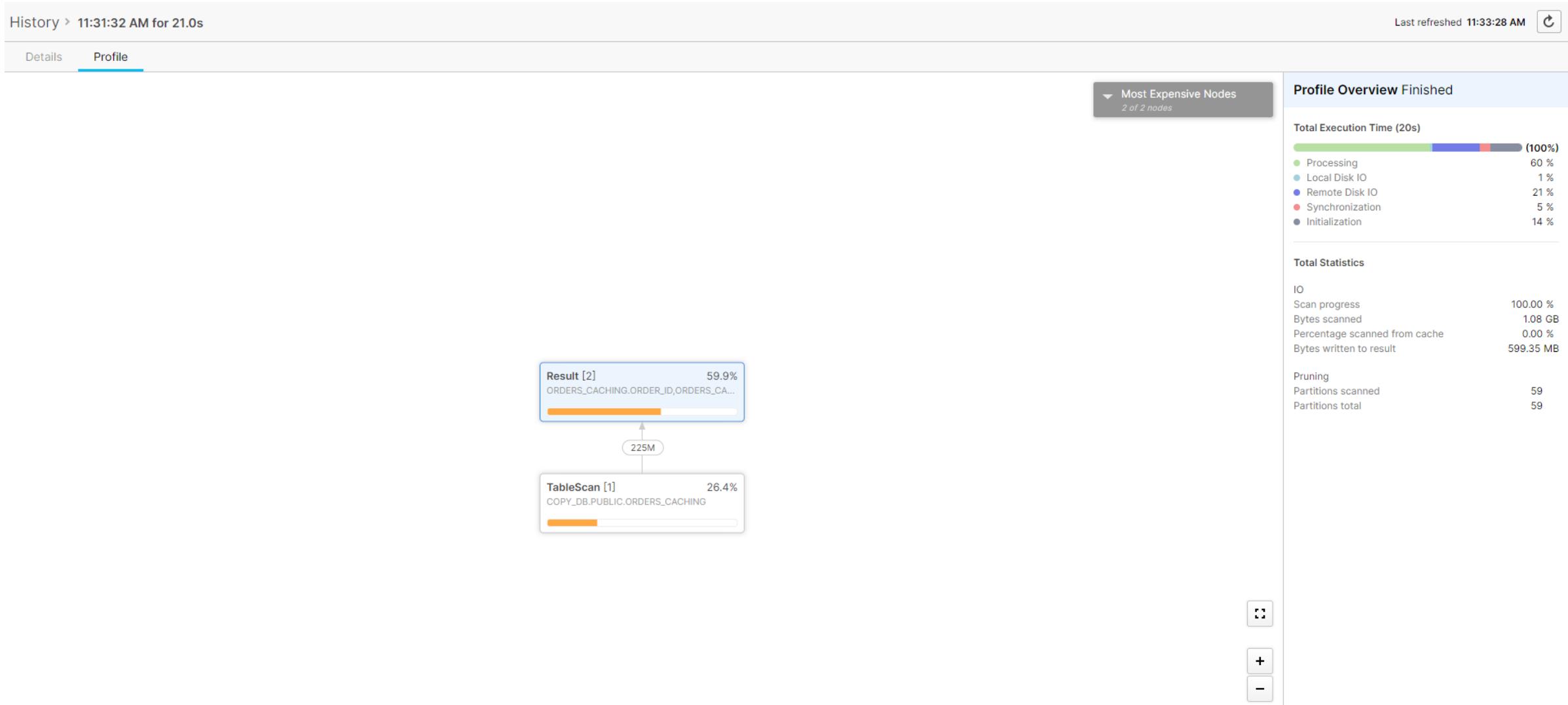
Results

row#	ORDER_ID	AMOUNT	PROFIT	QUANTITY	CATEGORY	SUBCATEGORY	DATE
1	B-25601	1275	-1148	7	Furniture	Bookcases	2020-09-15
2	B-25601	1275	-1148	7	Furniture	Bookcases	2021-11-10
3	B-25601	1275	-1148	7	Furniture	Bookcases	2017-07-23
4	B-25601	1275	-1148	7	Furniture	Bookcases	2023-01-04
5	B-25601	1275	-1148	7	Furniture	Bookcases	2023-10-26
6	B-25601	1275	-1148	7	Furniture	Bookcases	2021-08-15
7	B-25601	1275	-1148	7	Furniture	Bookcases	2022-08-26
8	B-25601	1275	-1148	7	Furniture	Bookcases	2023-09-01

225,000,000 rows produced (only 1,000,000 shown)

[Export Result](#)

SELECT Query without a WHERE Clause – Query Profile



SELECT Query with a WHERE Clause – Query Details

History > 11:39:31 AM for 1.0s Last refreshed 11:41:23 AM Auto refresh 

[Details](#) [Profile](#)

Status	Success
User	UCSC
Warehouse	COMPUTE_WH
Start Time	11:39:31 AM
End Time	11:39:32 AM
Total Duration	1.0s
Scanned Bytes	1.1GB
Rows	97.1K
Query ID	01a59df1-0000-30b9-0003-a5560002902e
Session ID	1026213716406294

SQL Text

```
1 | SELECT * FROM ORDERS_CACHING WHERE DATE = '2020-06-09';
```

[Select SQL](#) 

Query Result

Results

row#	ORDER_ID	AMOUNT	PROFIT	QUANTITY	CATEGORY	SUBCATEGORY	DATE
1	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
2	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
3	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
4	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
5	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
6	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
7	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
8	B-25678	27	-25	2	Clothing	Shirt	2020-06-09

97,056 rows produced [Export Result](#) 

SELECT Query with a WHERE Clause – Query Profile

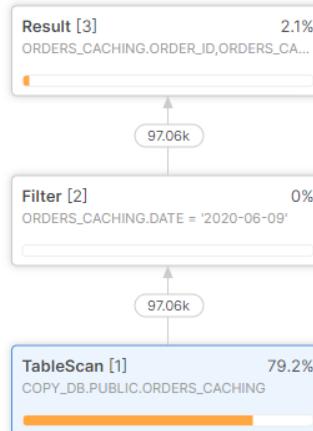
History > 11:39:31 AM for 1.0s

Last refreshed 11:40:16 AM

Details

Profile

▼ Most Expensive Nodes
2 of 3 nodes



Profile Overview Finished

Total Execution Time (835ms)



Total Statistics

IO

Scan progress	100.00 %
Bytes scanned	1.08 GB
Percentage scanned from cache	100.00 %
Bytes written to result	0.04 MB

Pruning

Partitions scanned	59
Partitions total	59



SELECT Query with a WHERE Clause on a Different WH – Query Details

History > 11:43:55 AM for 63ms

Last refreshed 11:45:49 AM Auto refresh 

Details Profile

Status Success
User UCSC
Warehouse DATABASE_WH
Start Time 11:43:55 AM
End Time 11:43:55 AM
Total Duration 63ms
Scanned Bytes 0
Rows 0
Query ID 01a59df5-0000-3112-0003-a5560002b012
Session ID 1026213716406294

SQL Text

```
1 | SELECT * FROM ORDERS_CACHING WHERE DATE = '2020-06-09';
```

Query Result

Results

row#	ORDER_ID	AMOUNT	PROFIT	QUANTITY	CATEGORY	SUBCATEGORY	DATE
1	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
2	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
3	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
4	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
5	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
6	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
7	B-25678	27	-25	2	Clothing	Shirt	2020-06-09
8	B-25678	27	-25	2	Clothing	Shirt	2020-06-09

97,056 rows produced

Export Result 

SELECT Query with a WHERE Clause on a Different WH – Query Profile

History > 11:43:55 AM for 63ms Last refreshed 11:44:09 AM

Details **Profile**

▼ Most Expensive Nodes
1 of 1 nodes

Profile Overview Finished

Total Execution Time (1ms) (100%)
● Other 100 %

QUERY RESULT REUSE [0] 100%
01a59df5-0000-3112-0003-a5560002b012

The screenshot shows a database query profile interface. At the top, it displays the history entry '11:43:55 AM for 63ms' and the last refresh time '11:44:09 AM'. Below this, there are two tabs: 'Details' and 'Profile', with 'Profile' being the active tab. A dropdown menu titled 'Most Expensive Nodes' indicates '1 of 1 nodes'. To the right, a 'Profile Overview' section shows the status as 'Finished' and the total execution time as '1ms' (100%), with the 'Other' category accounting for 100%. In the center, a box highlights 'QUERY RESULT REUSE [0]' at 100%, with the ID '01a59df5-0000-3112-0003-a5560002b012'. On the far right, there are three small icons: a square with a cross, a plus sign, and a minus sign.

SELECT Query with a WHERE Clause on a Different WH – Query Details

History > 11:47:57 AM for 3.9s Last refreshed 12:07:34 PM Auto refresh 

Details Profile

Status	Success
User	UCSC
Warehouse	DATABASE_WH
Start Time	11:47:57 AM
End Time	11:48:01 AM
Total Duration	3.9s
Scanned Bytes	1.1GB
Rows	97.6K
Query ID	01a59df9-0000-3100-0003-a5560002814e
Session ID	1026213716406294

SQL Text

```
1 | SELECT * FROM ORDERS_CACHING WHERE DATE = '2020-09-15';
```

Select SQL 

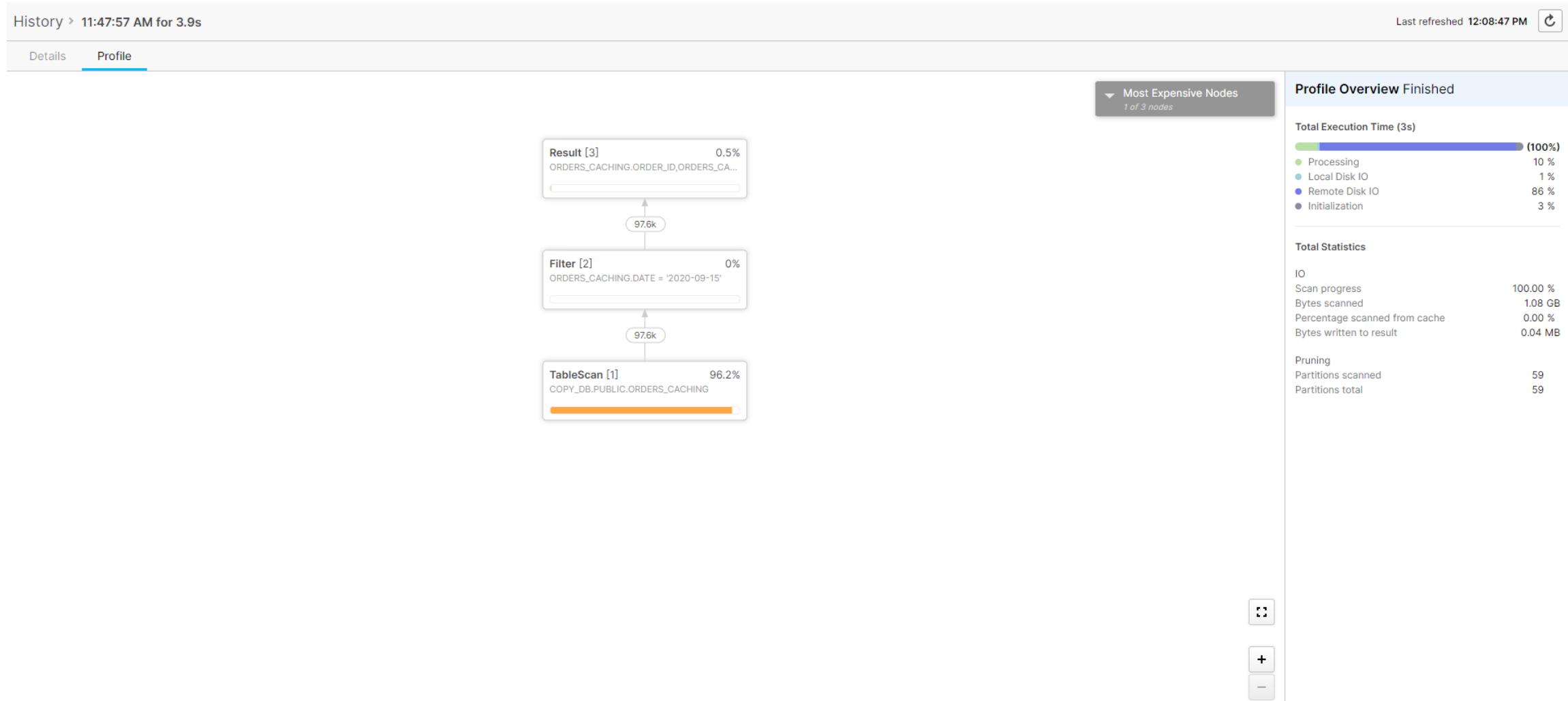
Query Result

Results

row#	ORDER_ID	AMOUNT	PROFIT	QUANTITY	CATEGORY	SUBCATEGORY	DATE
1	B-25608	856	385	6	Electronics	Printers	2020-09-15
2	B-25608	856	385	6	Electronics	Printers	2020-09-15
3	B-25608	856	385	6	Electronics	Printers	2020-09-15
4	B-25608	856	385	6	Electronics	Printers	2020-09-15
5	B-25608	856	385	6	Electronics	Printers	2020-09-15
6	B-25608	856	385	6	Electronics	Printers	2020-09-15
7	B-25608	856	385	6	Electronics	Printers	2020-09-15
8	B-25608	856	385	6	Electronics	Printers	2020-09-15

97,600 rows produced  Export Result 

SELECT Query with a WHERE Clause on a Different WH – Query Profile



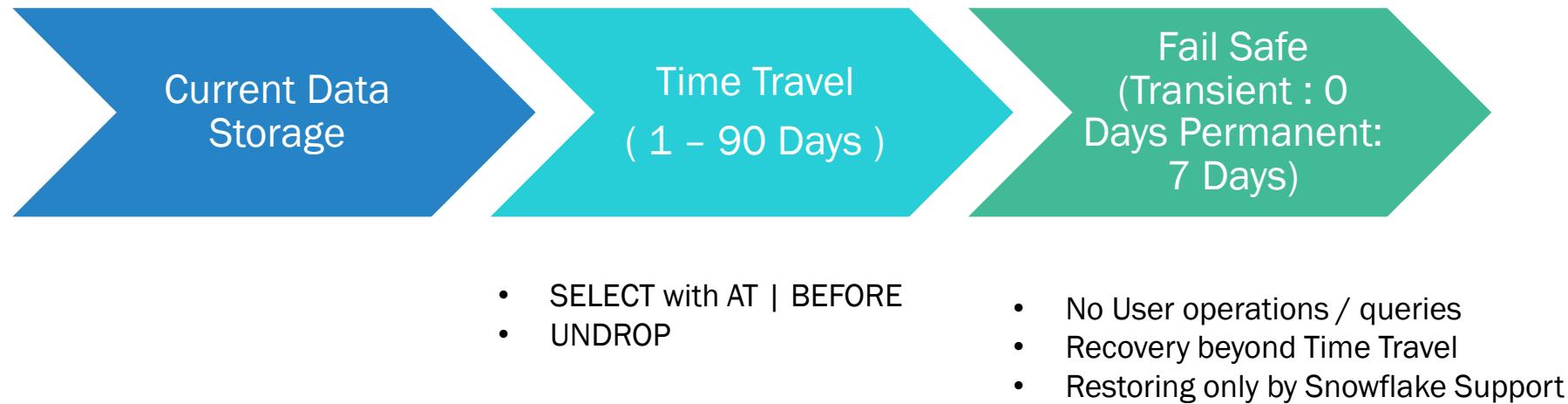


SNOWFLAKE DATA PROTECTION LIFECYCLE

LET'S LEARN HOW TO SNOWFLAKE PROTECTS YOUR DATA



Snowflake Data Protection Lifecycle





Different Table Types

Permanent

- CREATE TABLE
- Time Travel Retention Period
 - 0 – 90 days
- Fail Safe
- Available Until Dropped

Used for permanent data

Transient

- CREATE TRANSIENT TABLE
- Time Travel Retention Period
 - 0 – 1 day
- No Fail Safe
- Available Until Dropped

Used for data that does not need to be protected. Such as Development work.

Temporary

- CREATE TEMPORARY TABLE
- Time Travel Retention Period
 - 0 – 1 day
- No Fail Safe
- Available Only in Session

Used for non-permanent data. Such as temporary tables in ETL processes, development work.

These different types also exist for Database and Schemas



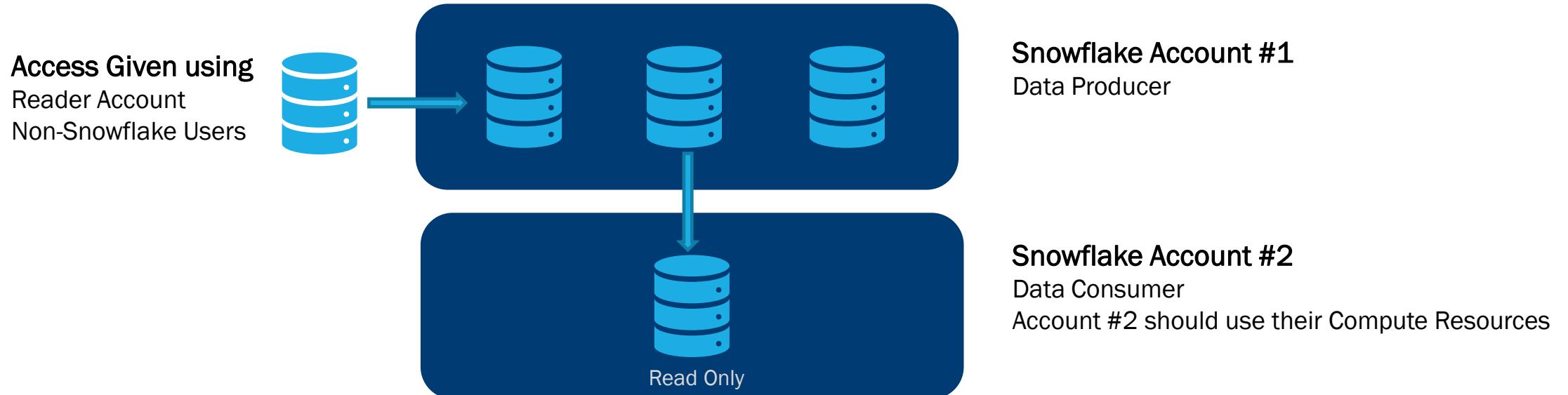
DATA SHARING

LET'S LEARN HOW TO SHARE DATA BETWEEN DIFFERENT CONSUMERS



Snowflake Data Sharing

- Usually, data sharing with traditional methods is a complicated process
- Snowflake data sharing allows data sharing without actual copy of the data as well as gives access to up-to-date information
- Shared data can be consumed by consumers using their own compute resources
- Non-snowflake users can also access the shared data through a reader account



Snowflake Data Sharing – Using Reader Account for Non-Snowflake Users

- Step #1: Create a New Reader Account
 - Independent instance with own url and own compute resources
- Step #2: Share the Data and tables
- Step #3: Create Users
- Step #4: Create Database inside the reader account using the share



ACCESS MANAGEMENT

LET'S LEARN HOW TO GOVERN ACCESS TO DATA IN SNOWFLAKES

Introduction to Access Management in Snowflake

Snowflake Approach to Access Control

- Discretionary Access Control (DAC) – Each object has an owner and can grant access to that object
- Role Based Access Control (RBAC) – Access is through roles. Roles are granted privileges and then the roles are granted to users & other roles

Key Concepts

- User – Human or machine user
- Role – An entity to which privileges are granted. Roles can then be granted to users & other roles, potentially creating a role hierarchy
- Privilege – Level of access to an object e.g. SELECT, DROP, CREATE etc..
- Securable Object – Objects on which access can be granted. E.g. tables, views etc. more examples shortly.

Key Concepts



User

Securable Objects

Privileges

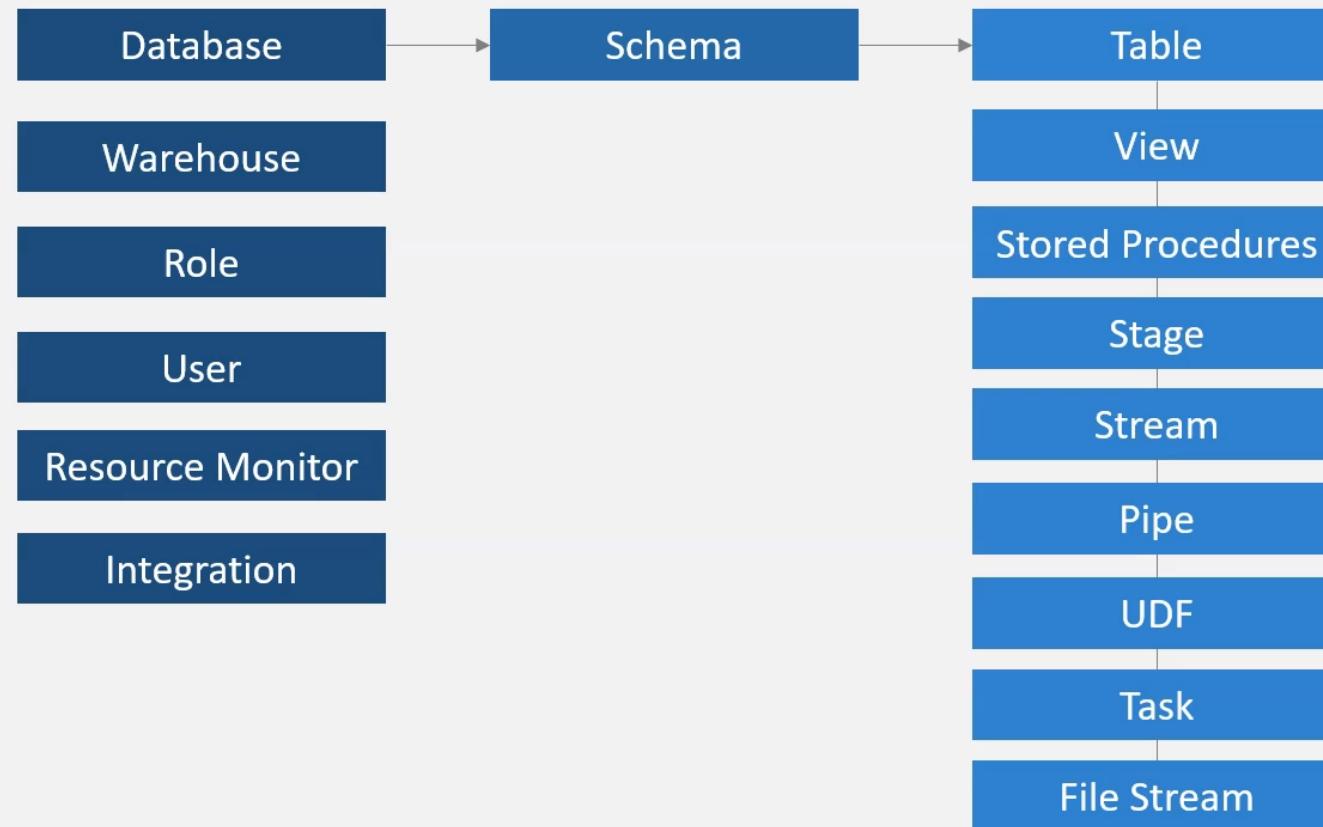
Roles

Users

Key Concepts



Objects on which Security can be applied (Securable)



Out of the box roles

ACCOUNT ADMIN

- Top level role, it is the most powerful role & is the account level administrator
- Access to this role should be restricted to only a very few users
- This role automatically encapsulates the SECURITY_ADMIN & SYS_ADMIN roles

SECURITY ADMIN

- Role that is used to create, monitor & manage users & roles.

SYS ADMIN

- Role which has rights to create databases, tables , warehouses and other objects.

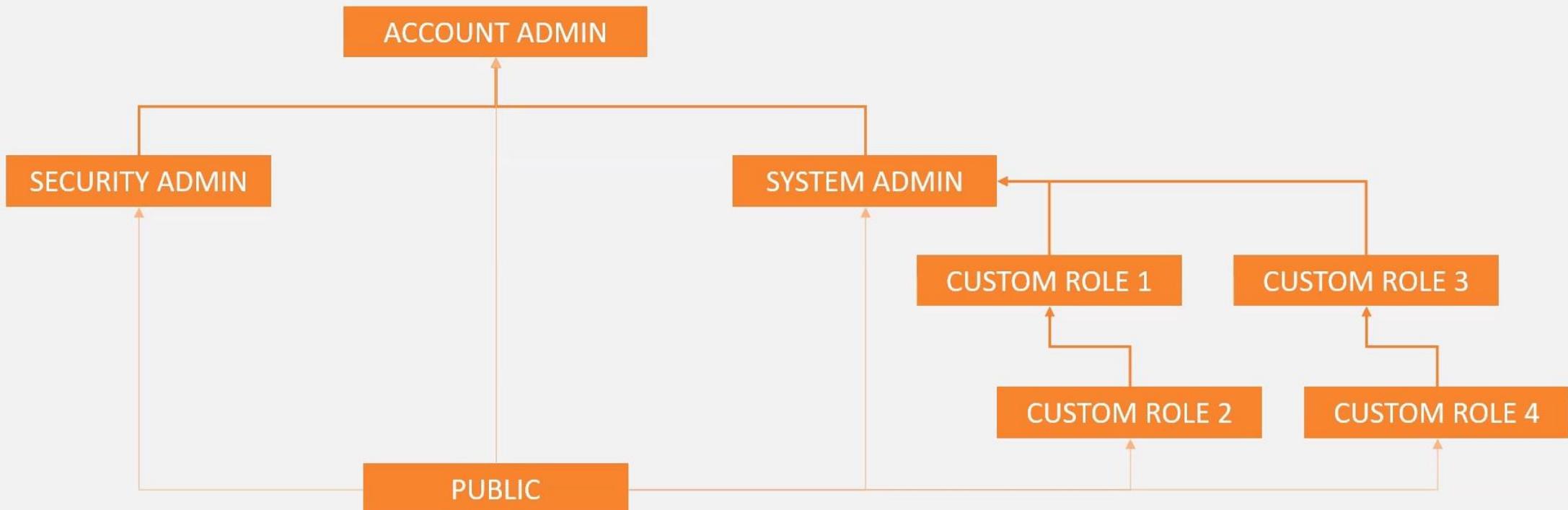
PUBLIC

- Role that is automatically granted to every user in your Snowflake system

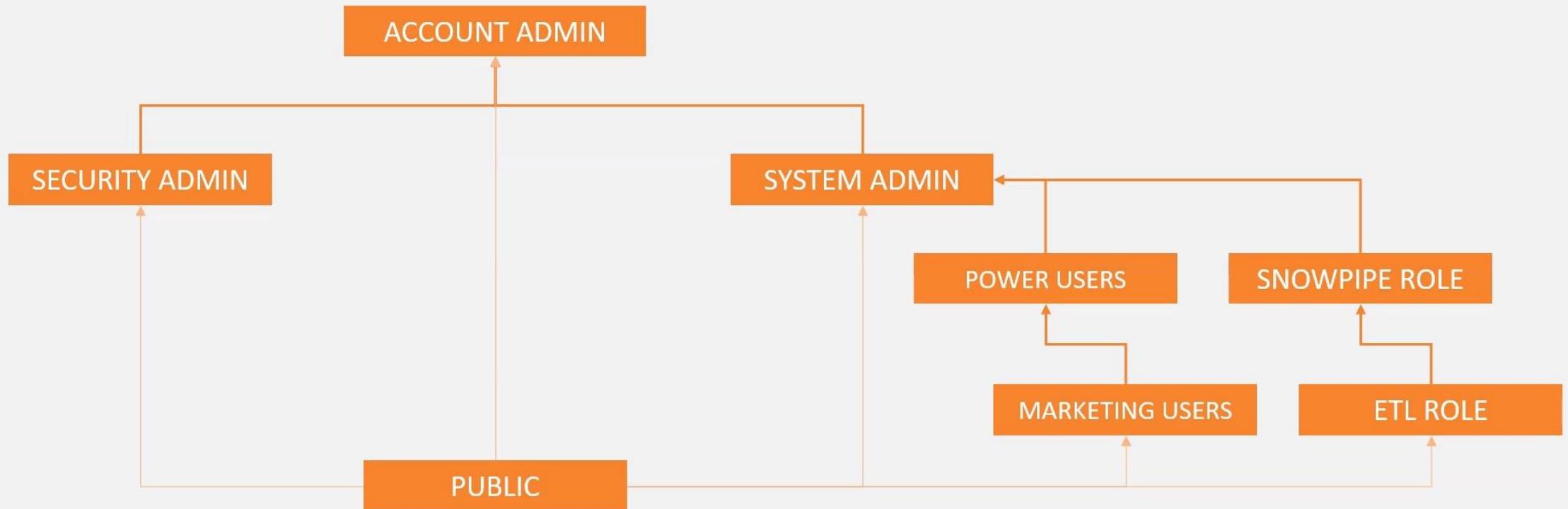
CUSTOM ROLES

- New roles that can be defined by the security admin.
- New customer role are not automatically granted and must be granted explicitly

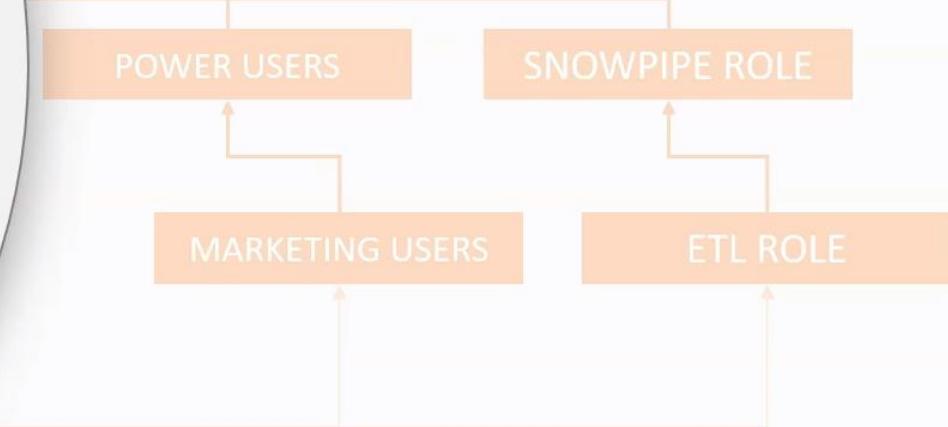
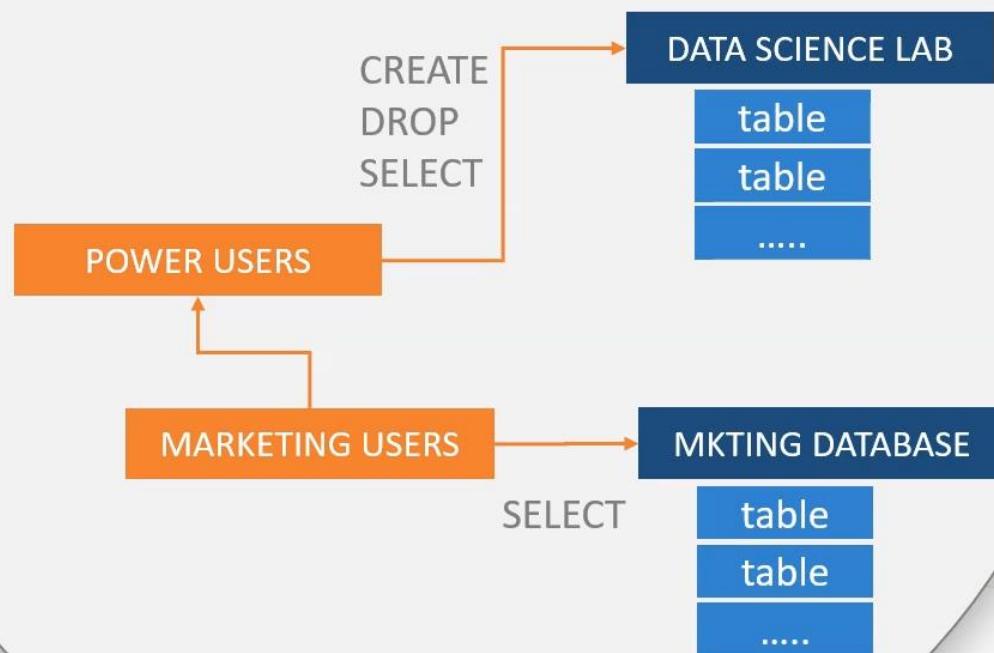
Out of the box roles hierarchy + custom roles



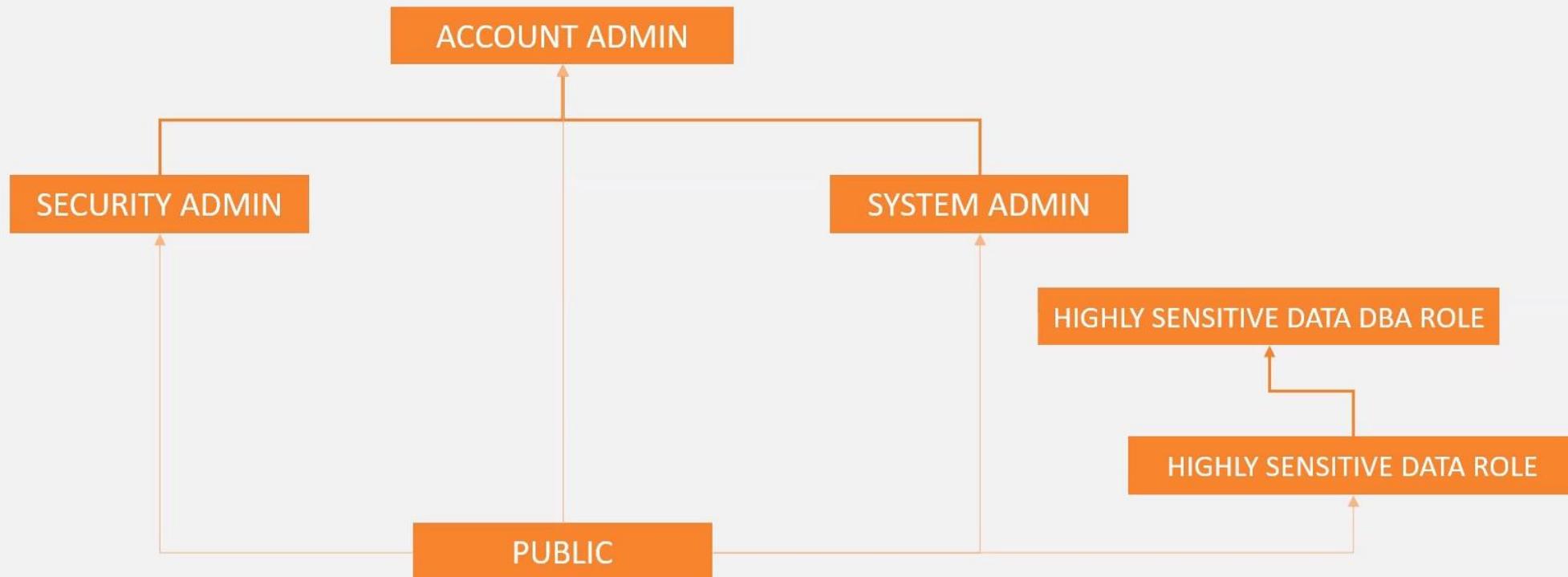
Out of the box roles hierarchy + custom roles



Out of the hierarchy



Out of the box roles hierarchy + custom roles

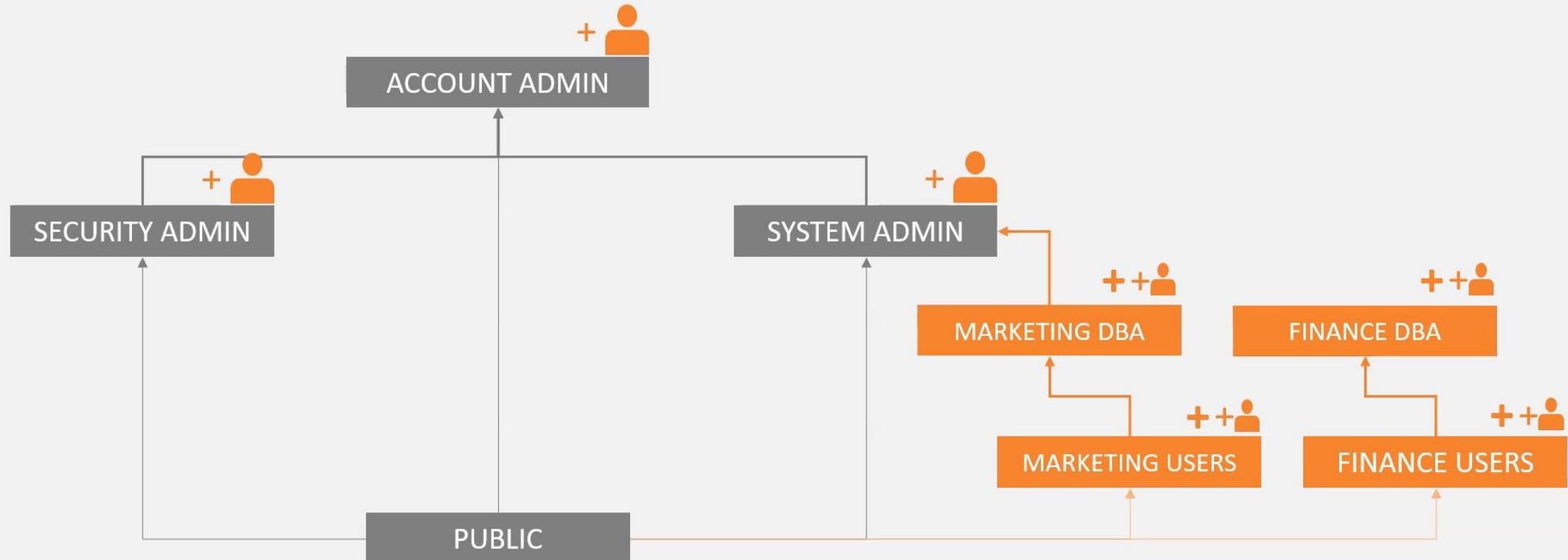


ACCOUNTADMIN role

- ▶ Most powerful role in the system
- ▶ Users with ACCOUNTADMIN role can
 - View and manage all objects
 - View billing and credit data
 - Stop any SQL statements
 - Perform other account level activities like creating reader accounts
- ▶ The first user on your Snowflake system automatically gets the ACCOUNT_ADMIN role
- ▶ Control Assignment of ACCOUNT_ADMIN role
- ▶ Assign the role to at least two people in the organization
- ▶ Users who have this role should use 2 factor authentication
- ▶ Don't create objects using ACCOUNTADMIN role unless you HAVE to

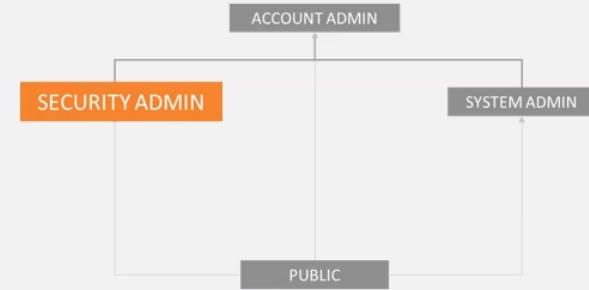


Roles Hands On - Outline

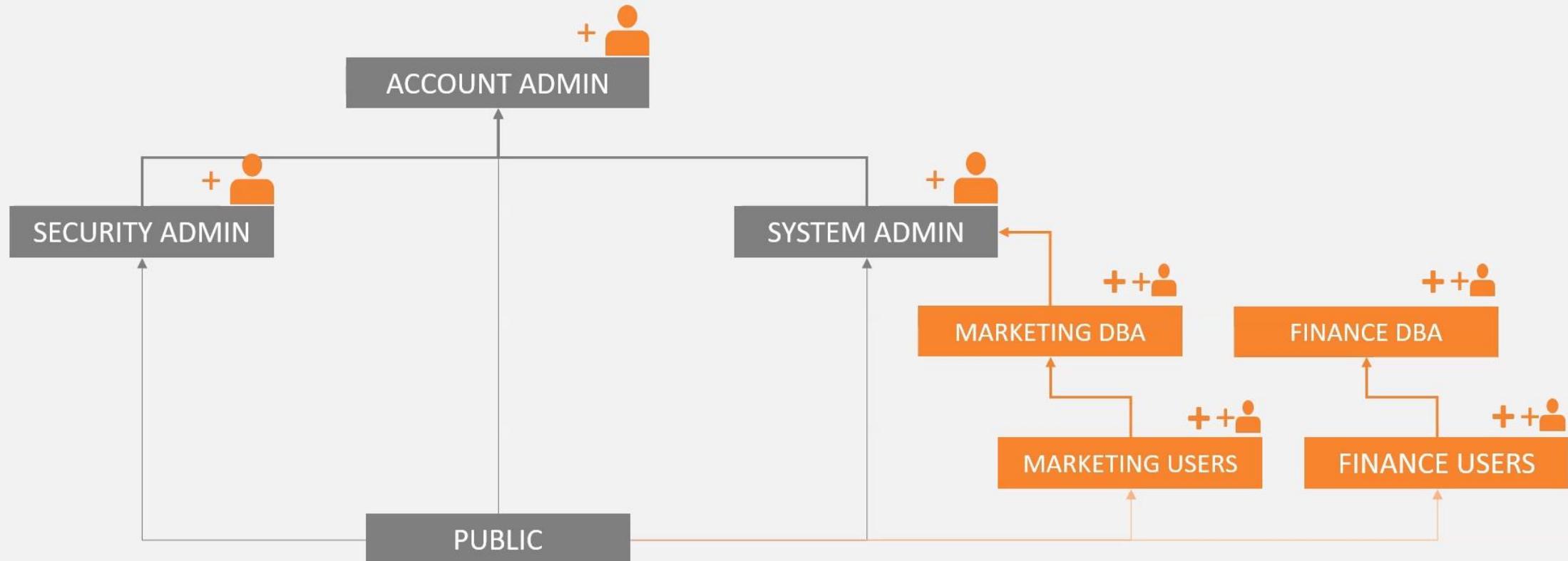


SECURITYADMIN role

- ▶ Users with SECURITYADMIN role can
 - Create and Manage Users
 - Create and Manage Roles
 - Assign privileges to the Roles
 - Grant, Revoke any privilege
 - Modify, monitor Sessions



Roles Hands On - Outline



SYSADMIN role

► Users with SYSADMIN role can

- Create & Manage Warehouse
- Create and Manage Databases, Tables & Other Objects



► Maintain the Role Hierarchy and ensure all custom role is directly or indirectly given to SYSADMIN

- Otherwise the SYSADMIN role will not be able to administer all the objects
- If you explicitly want to limit privilege to only the owning user then don't assign to SYSADMIN

PUBLIC role

- ▶ A role automatically given to all users
- ▶ The least privileged role as every one has this access
- ▶ The role can own objects
- ▶ Objects that are created using PUBLIC role are automatically available to every one in the system





Thank You!