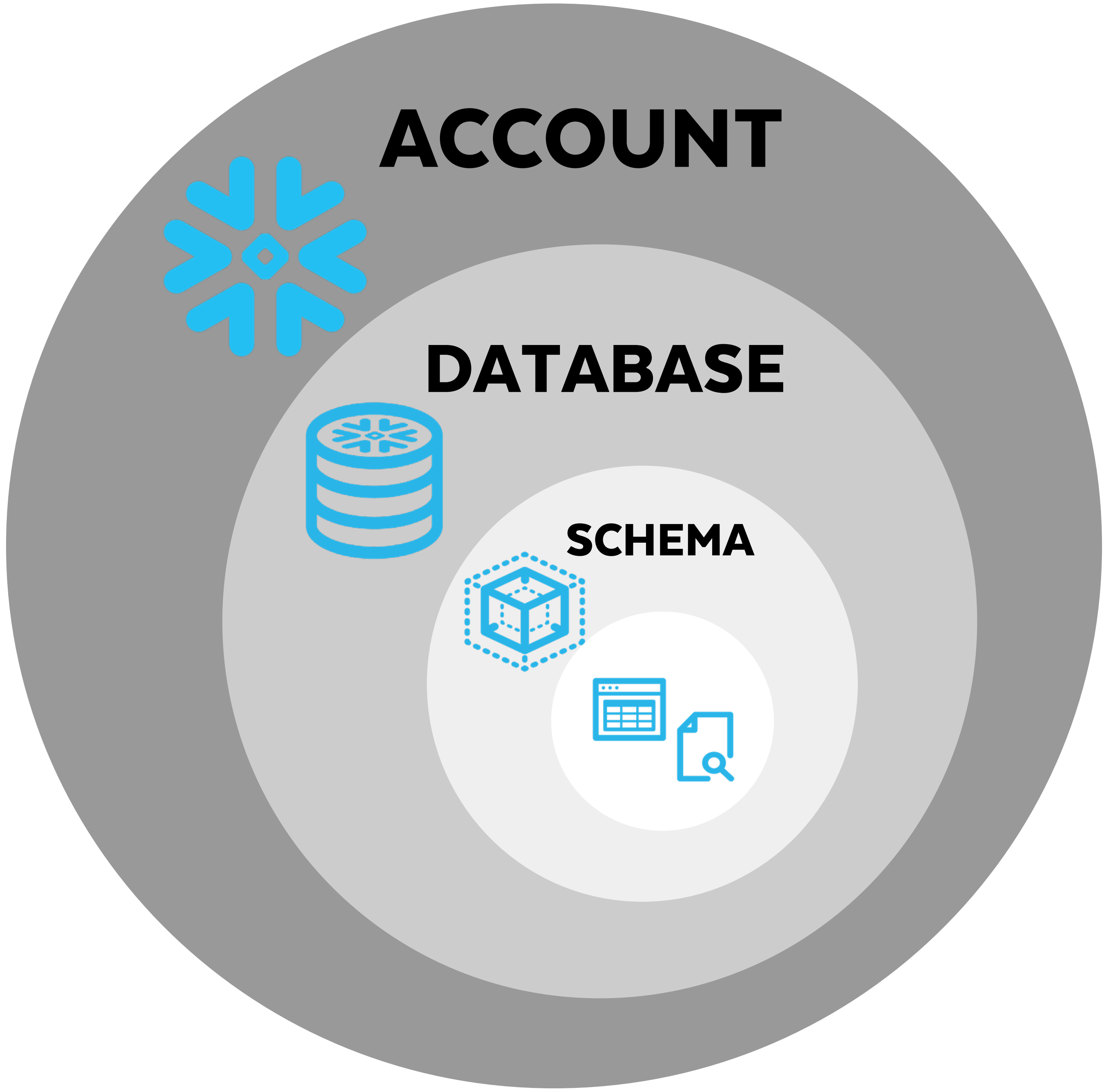
STORAGE HIERARCHY



Snowflake Accounts

- A customer can have as many accounts as they want
- Each account has its own URL – which uses the name of the account.
- Each account is deployed on a single cloud provider platform (AWS, Azure, GCP)
- Each account exists in a single geographic region
- Each account exists with a single Snowflake Edition (e.g. Standard, Premier, etc.)

Accounts Contain Databases

- Each database belongs to a single Snowflake account
- Databases can be replicated to other accounts, but they cannot SPAN multiple accounts

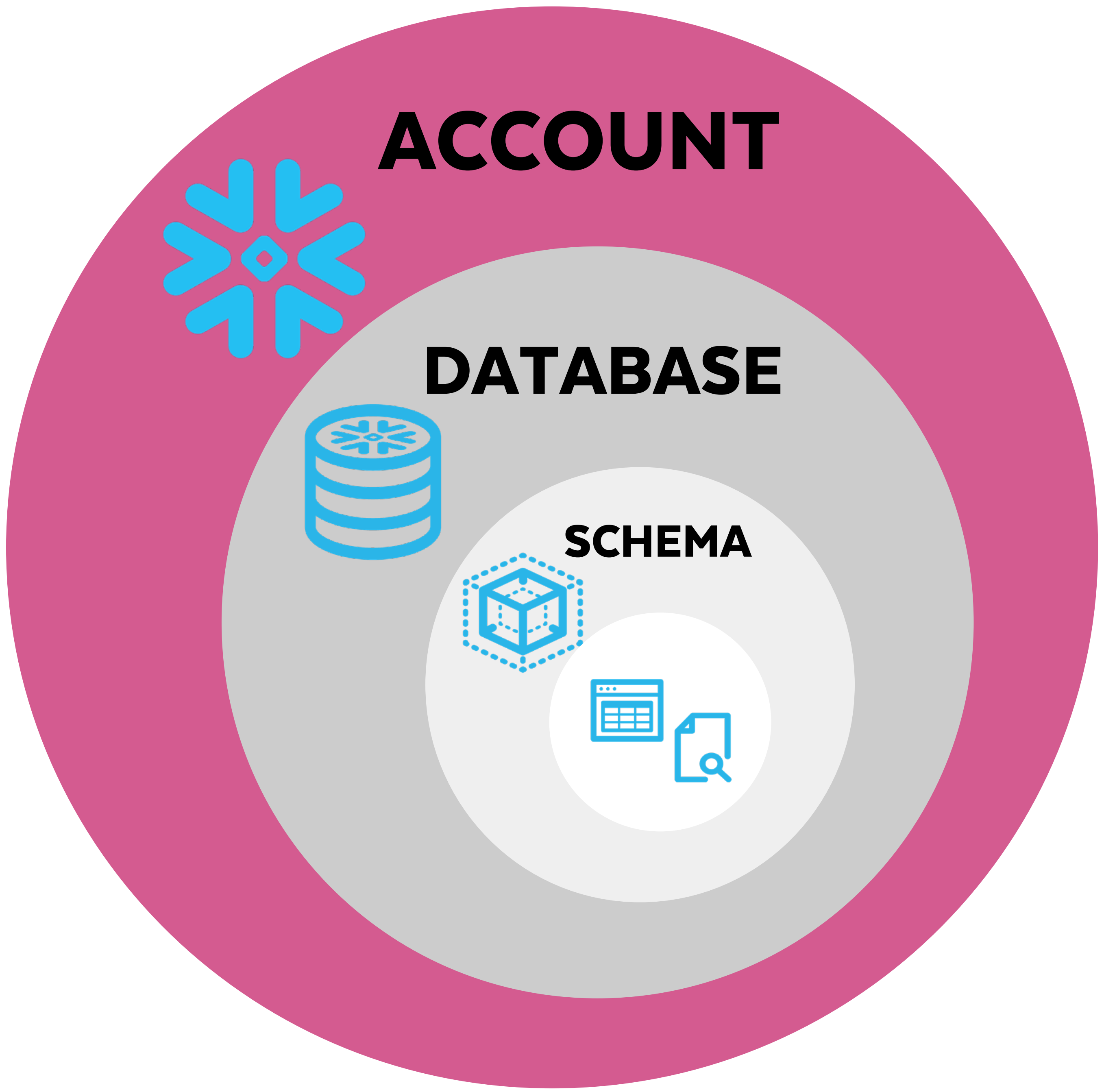
Databases Contain Schemas

- Each schema belongs to a single Snowflake database, in a single account
- Schemas can be replicated to other accounts or databases, but they cannot SPAN accounts or databases

Schemas Contain Other Objects

- Objects include tables, views, file formats, sequences, UDFs, & stored procs
- Objects belong to a single schema, in a single database, in a single account

STORAGE CONTAINER HIERARCHY



Snowflake Accounts

The account shown here is the account NS61174. Most trial accounts start with two letters followed by 5 numbers. Trial accounts can be converted to paid accounts.

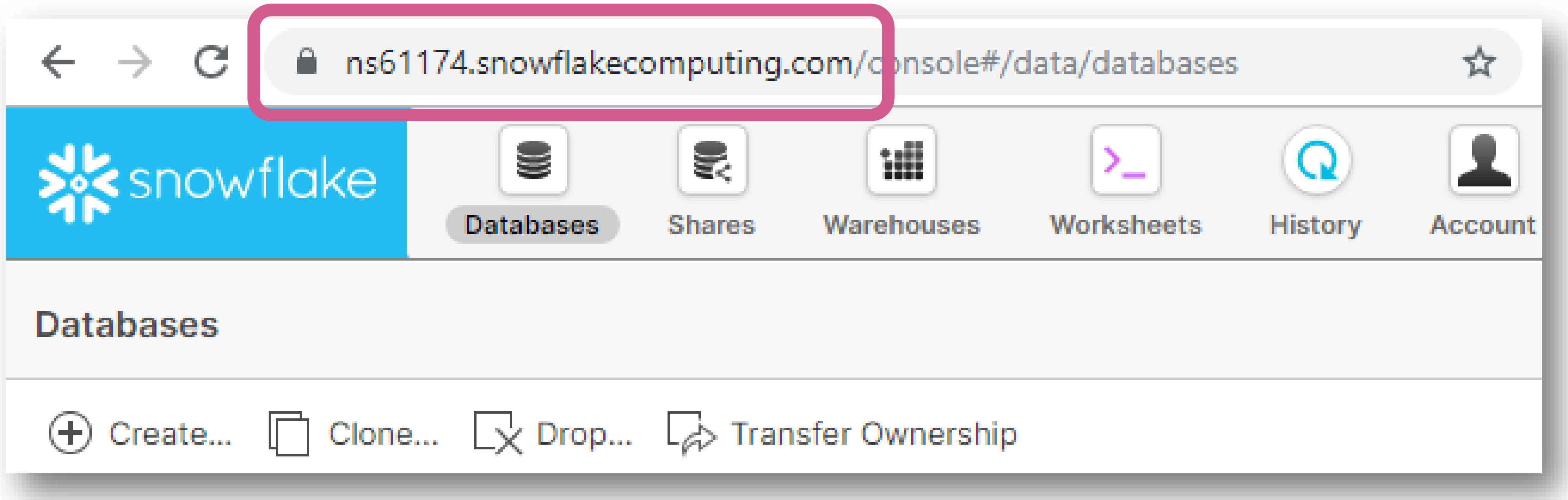
Customers can request a “vanity” address like:

https://robertsfishandchips.snowflakecomputing.com

When an account has just the account name, followed by “snowflakecomputing.com”, we know the cloud provider is **AWS** and the region is **us-west-2**. This is because this was the first cloud provider and region on which Snowflake deployed.

Later, when Snowflake expanded to other **regions** and **providers**, the region name was added to the account URL. For example, an address like:

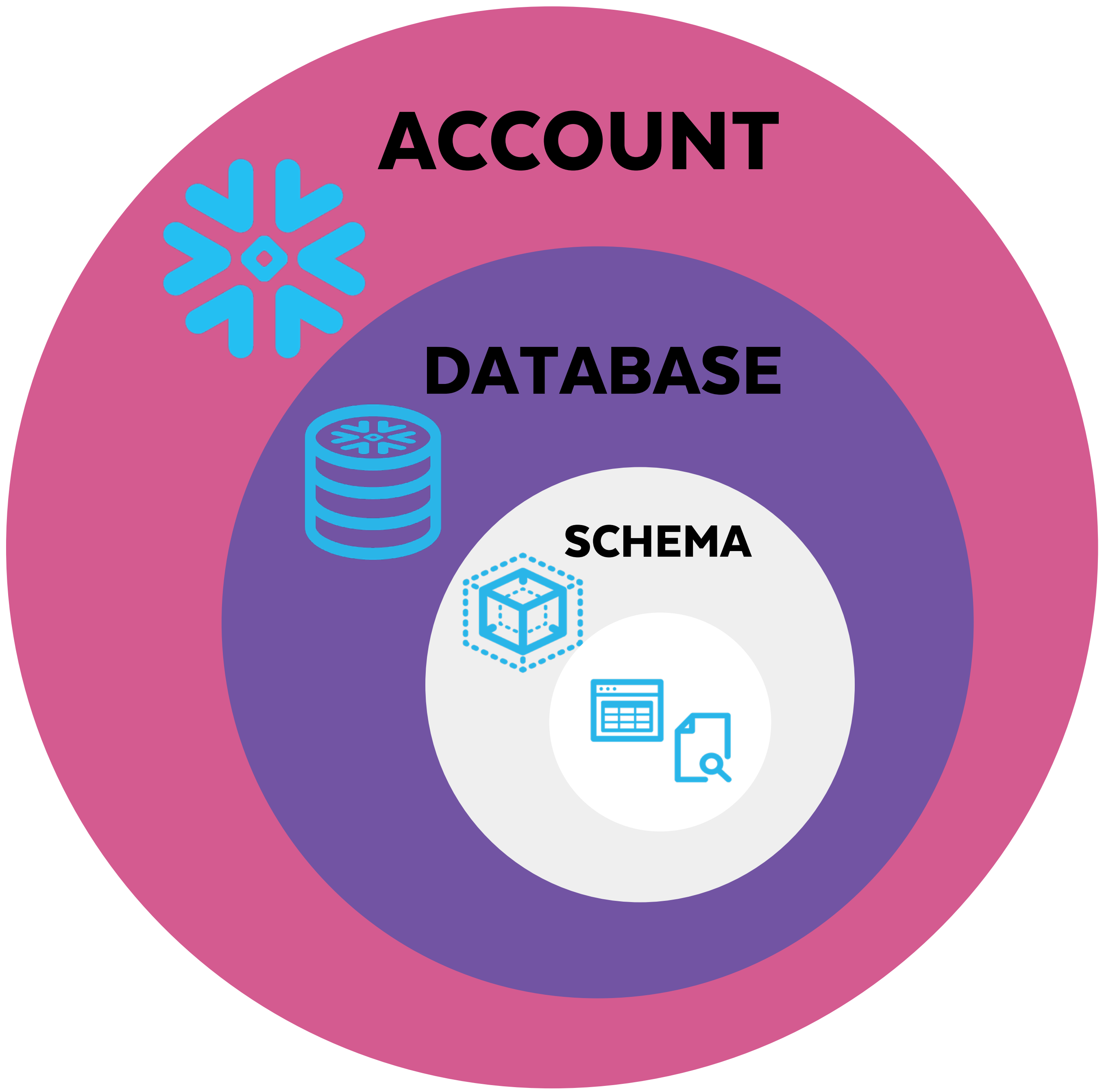
https://xy12345.east-us-2.azure.snowflakecomputing.com



ns61174.snowflakecomputing.com/console#/data/databases

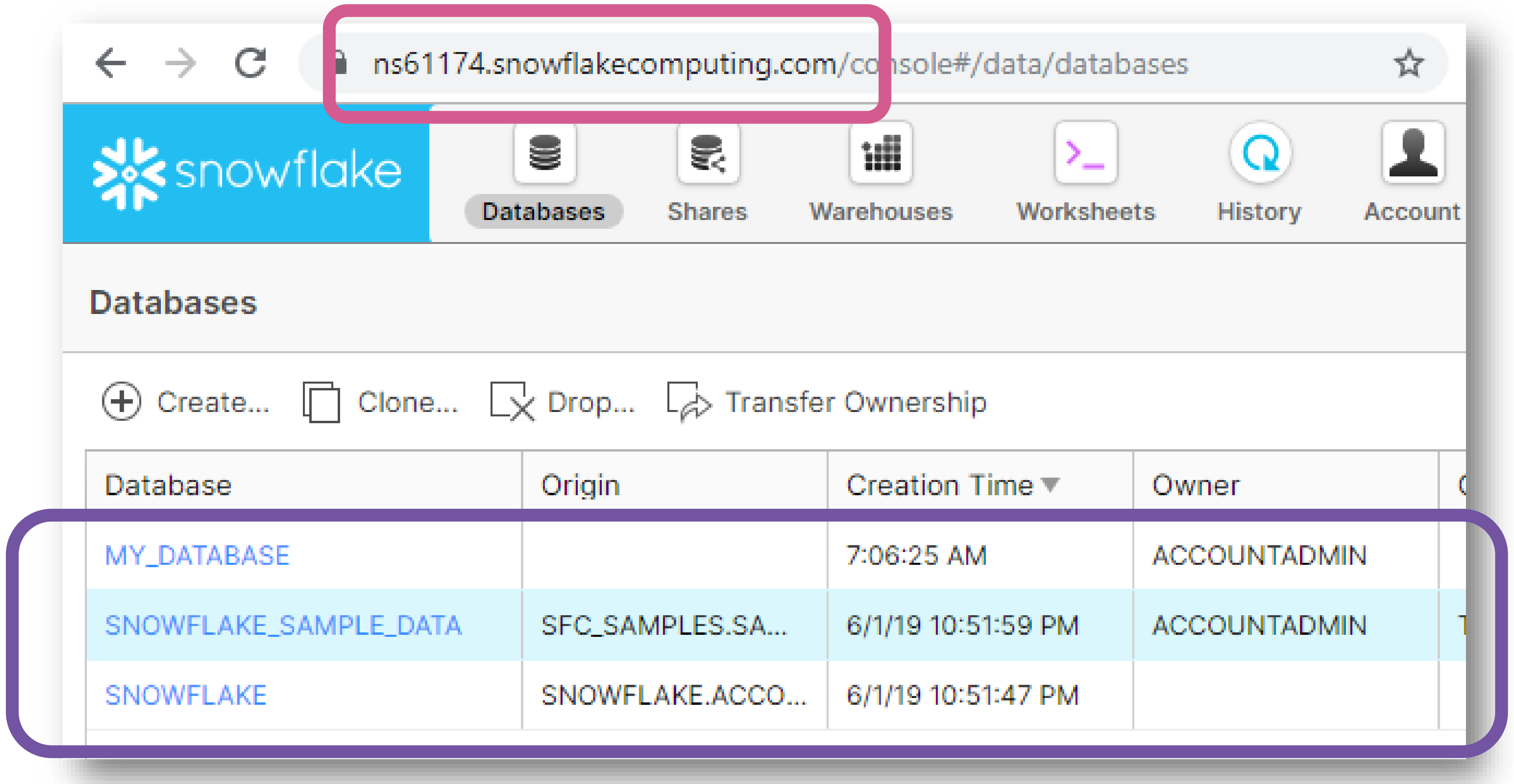
READ MORE ABOUT REGIONS HERE:
https://docs.snowflake.net/manuals/user-guide/intro-regions.html

STORAGE CONTAINER HIERARCHY



Accounts Contain Databases

- In this example, account NS61174 has three databases.
- The **SNOWFLAKE** database contains information about account usage. It is automatically added by Snowflake to each new account. It is sometimes called the "Account Usage Share" because it is shared by Snowflake so customers can check usage.
- The **SNOWFLAKE_SAMPLE_DATA** is also shared by Snowflake. This database allows for performance testing and tutorials. Customers are not charged for storing this data., because they are NOT storing it. The database is a shared database.

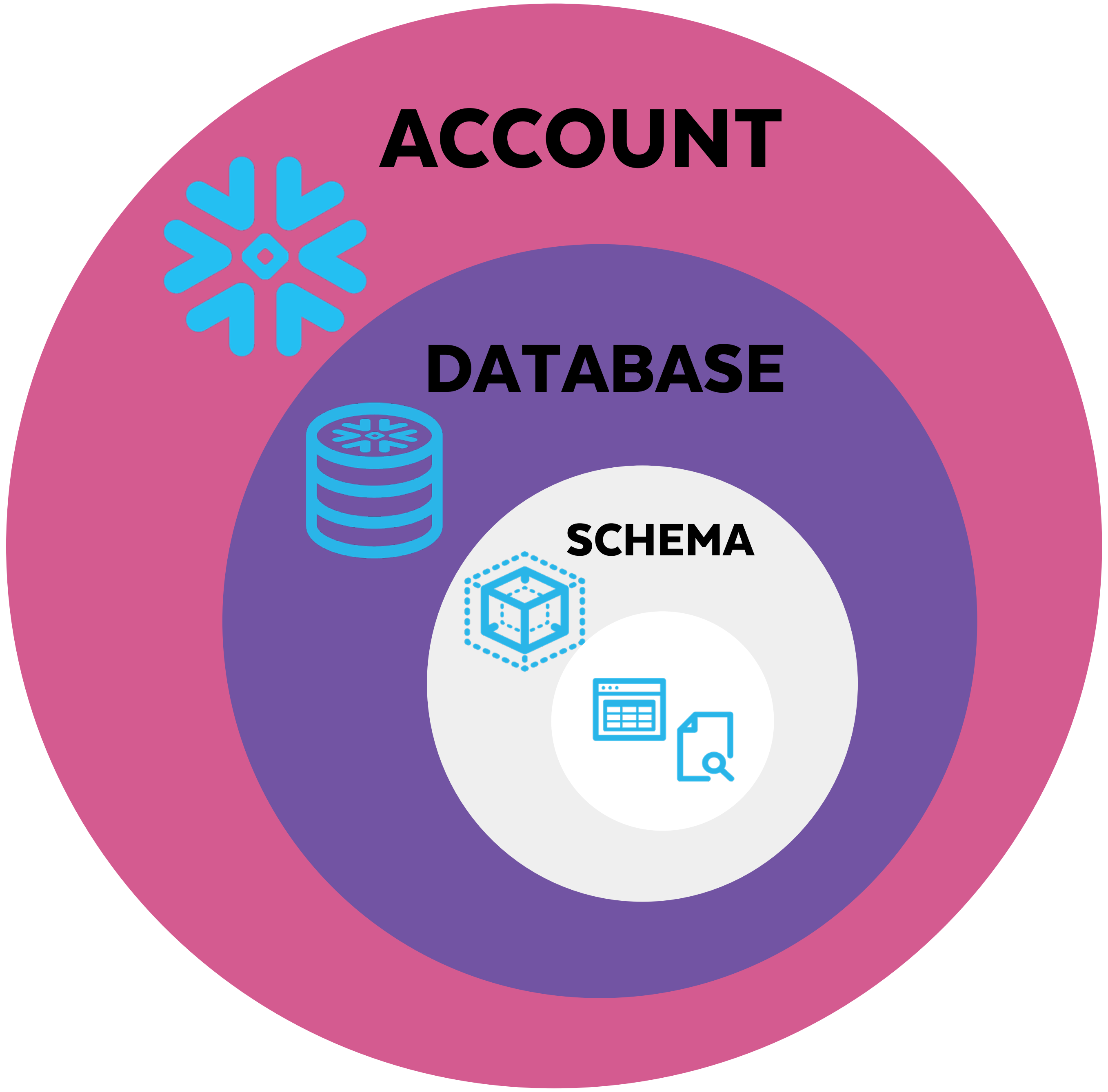


ns61174.snowflakecomputing.com/console#/data/databases

Databases

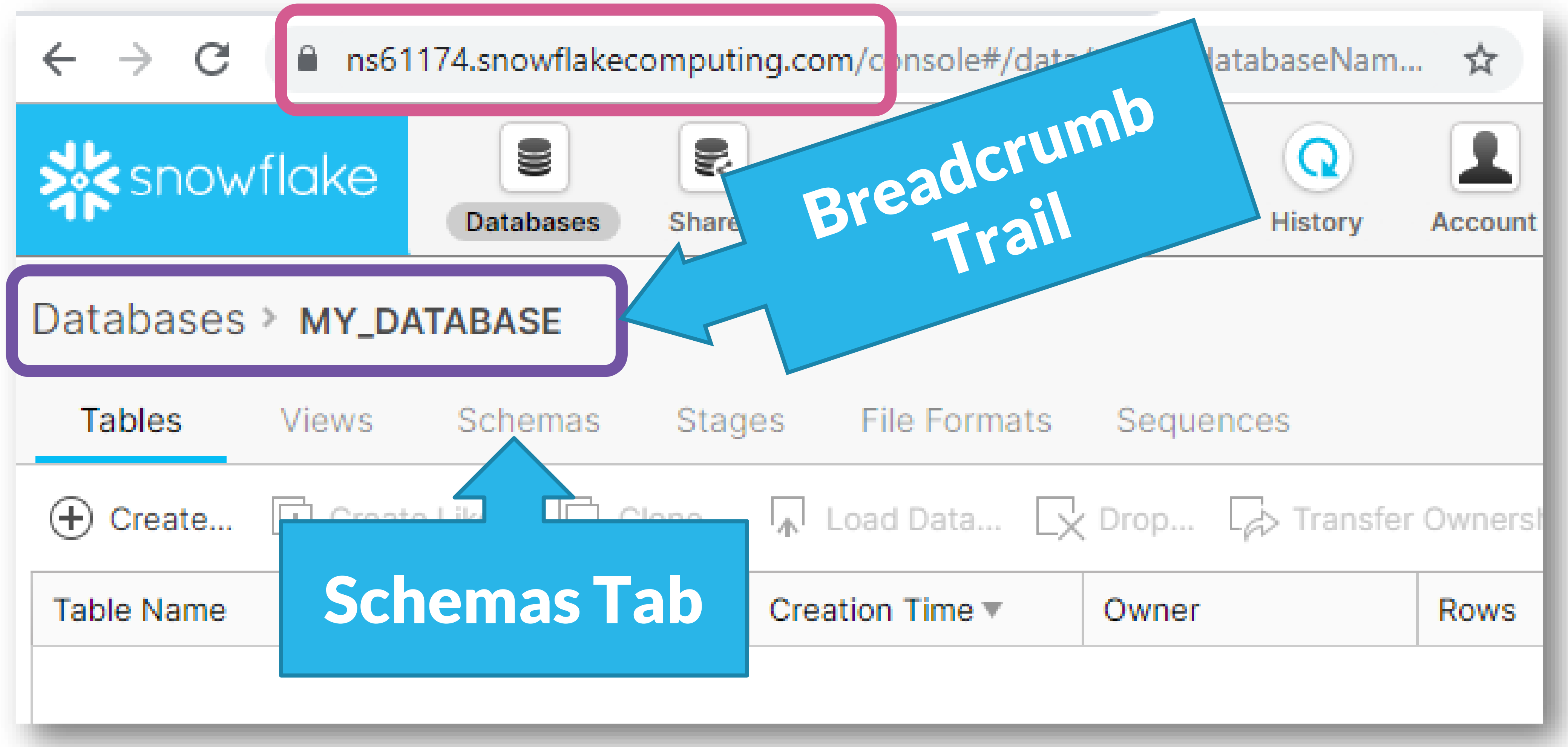
Database	Origin	Creation Time ▼	Owner
MY_DATABASE		7:06:25 AM	ACCOUNTADMIN
SNOWFLAKE_SAMPLE_DATA	SFC_SAMPLES.SA...	6/1/19 10:51:59 PM	ACCOUNTADMIN
SNOWFLAKE	SNOWFLAKE.ACCO...	6/1/19 10:51:47 PM	

STORAGE CONTAINER HIERARCHY

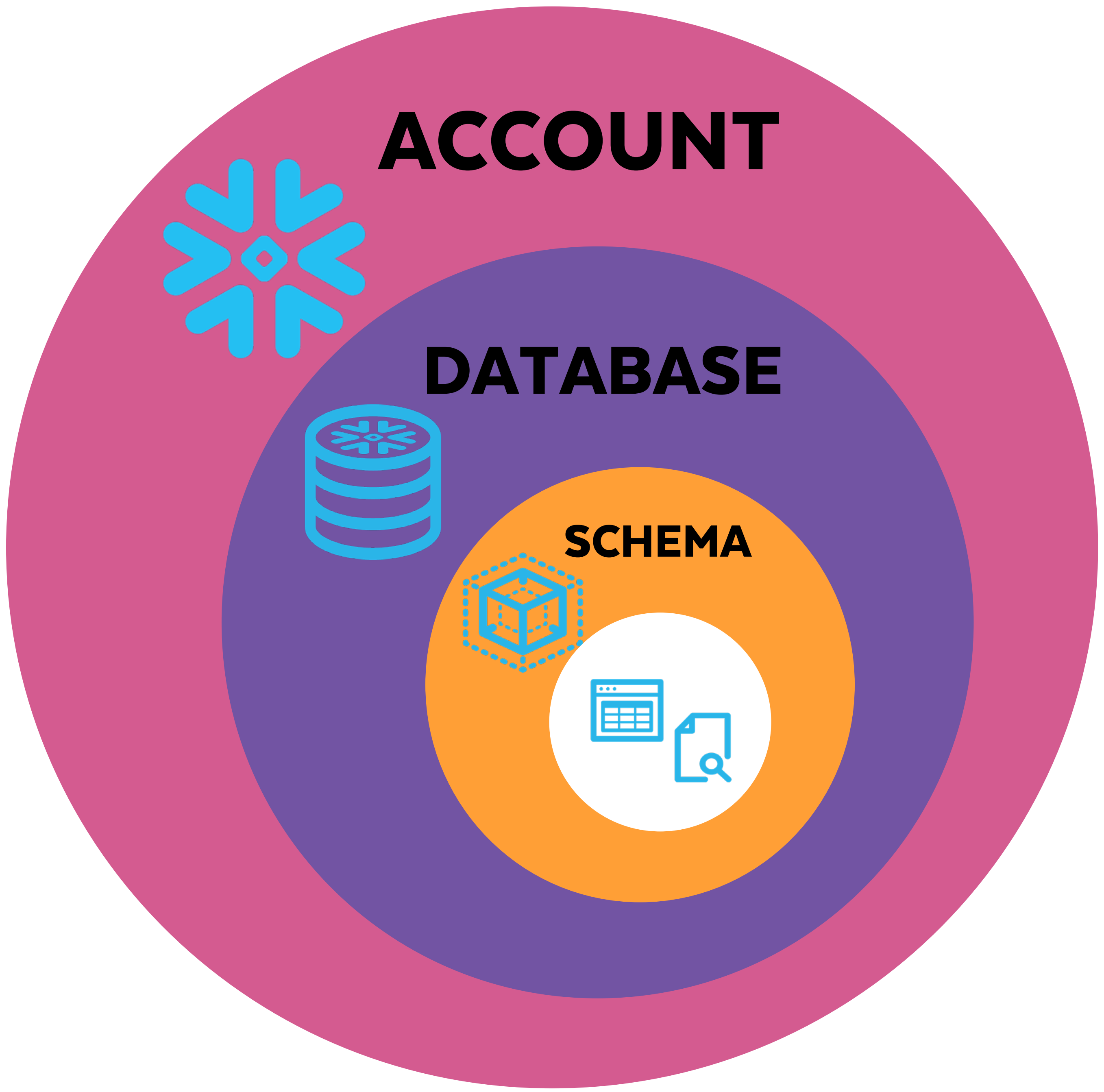


Accounts Contain Databases

When we drill into any database, the name of the database gets added to the breadcrumb trail and a set of tabs appears showing Schemas and other object types.



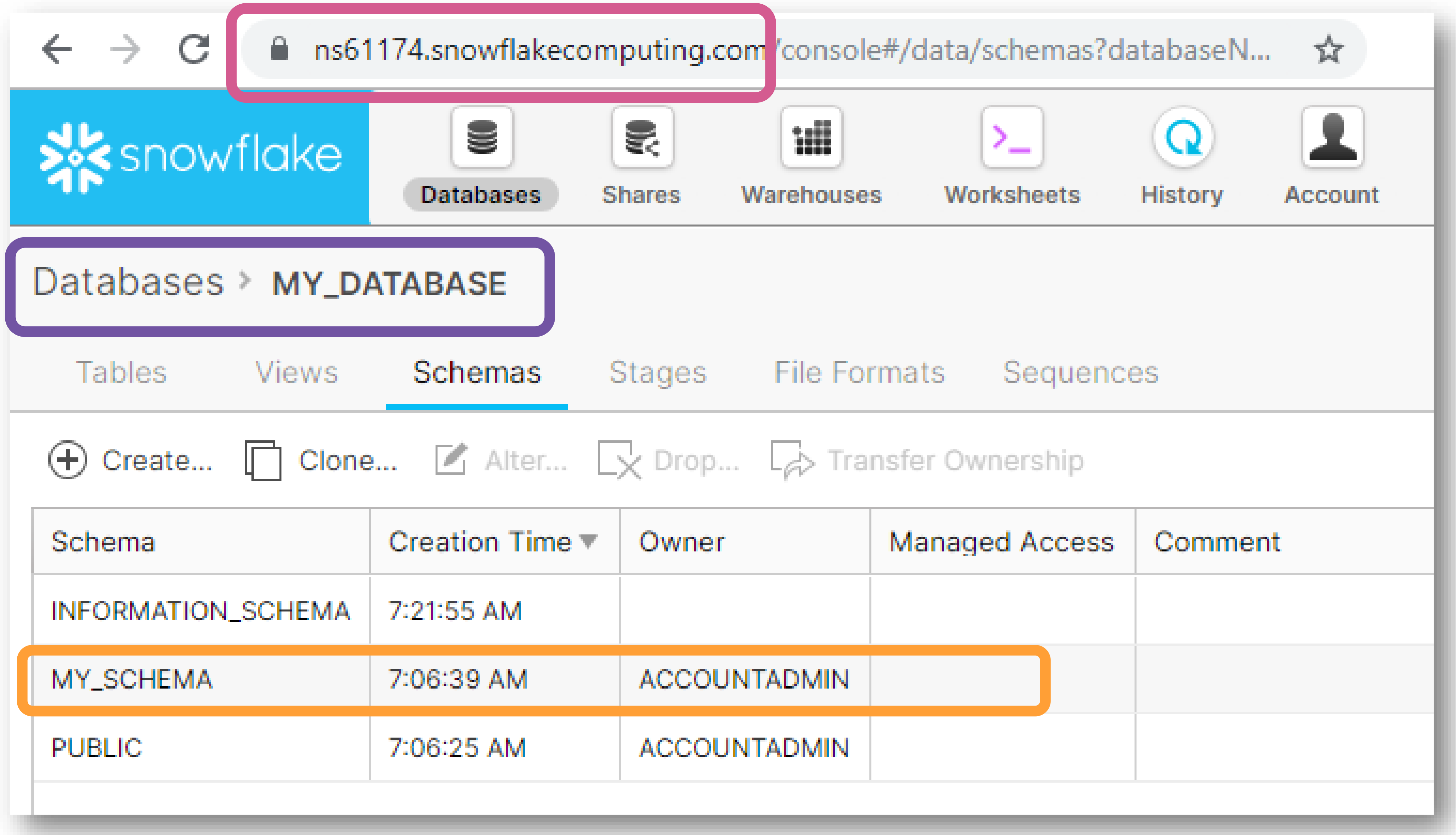
STORAGE CONTAINER HIERARCHY



Database.Schema = Namespace

Databases Contain Schemas

- The set of tabs below the breadcrumb trail makes it look as though Schemas are at the same level as Tables, Views, Stages, File Formats, and Sequence but this is not an accurate way to think of Schemas.
- Schemas are a logical grouping one level above the other objects listed.

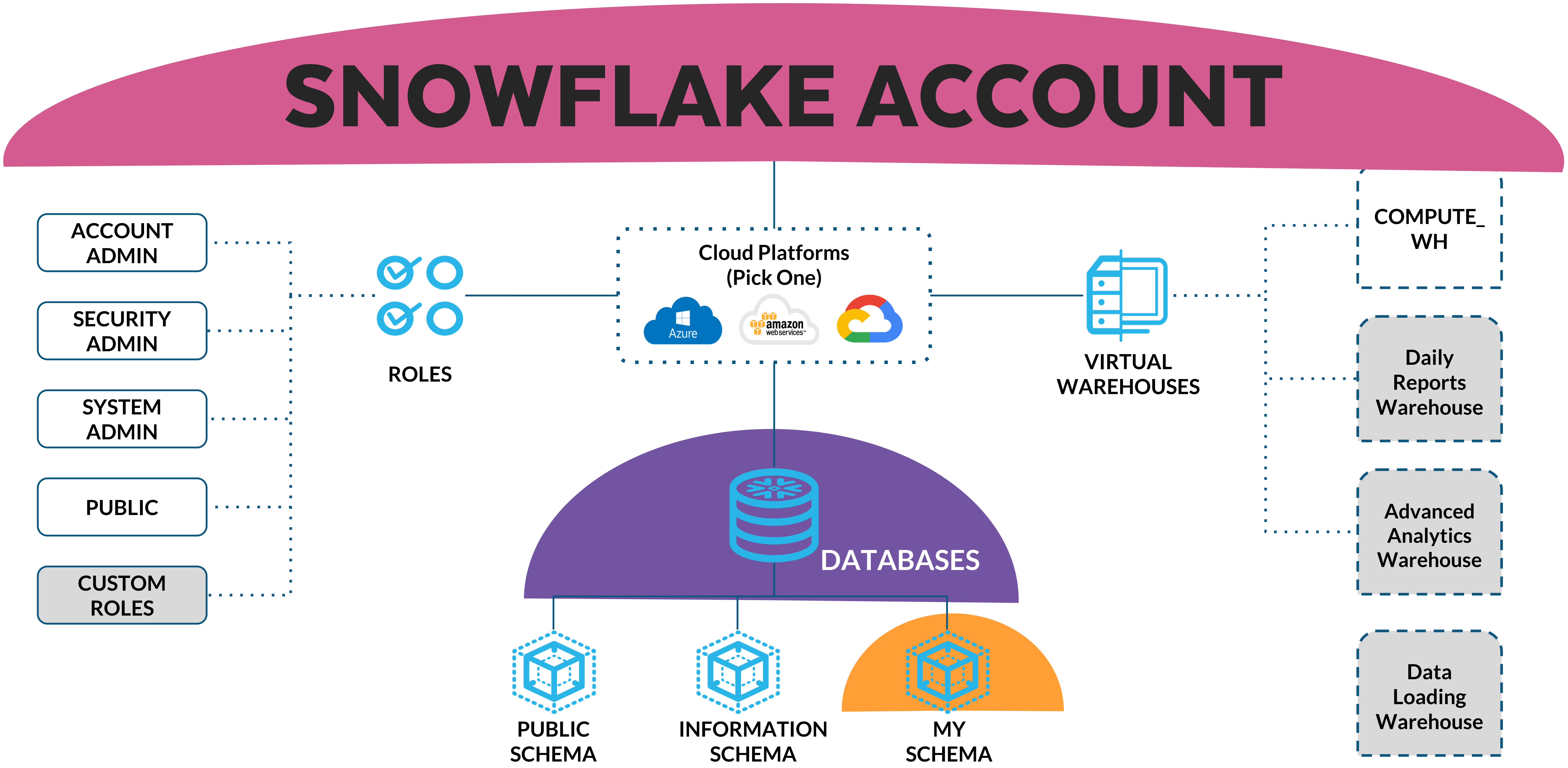


Databases › MY_DATABASE

Tables | Views | Schemas | Stages | File Formats | Sequences

Schema	Creation Time ▼	Owner	Managed Access	Comment
INFORMATION_SCHEMA	7:21:55 AM			
MY_SCHEMA	7:06:39 AM	ACCOUNTADMIN		
PUBLIC	7:06:25 AM	ACCOUNTADMIN		

SNOWFLAKE ACCOUNT



ACCOUNT ADMIN

SECURITY ADMIN

SYSTEM ADMIN

PUBLIC

CUSTOM ROLES

ROLES

Cloud Platforms (Pick One)

VIRTUAL WAREHOUSES

COMPUTE_ WH

Daily Reports Warehouse

Advanced Analytics Warehouse

Data Loading Warehouse

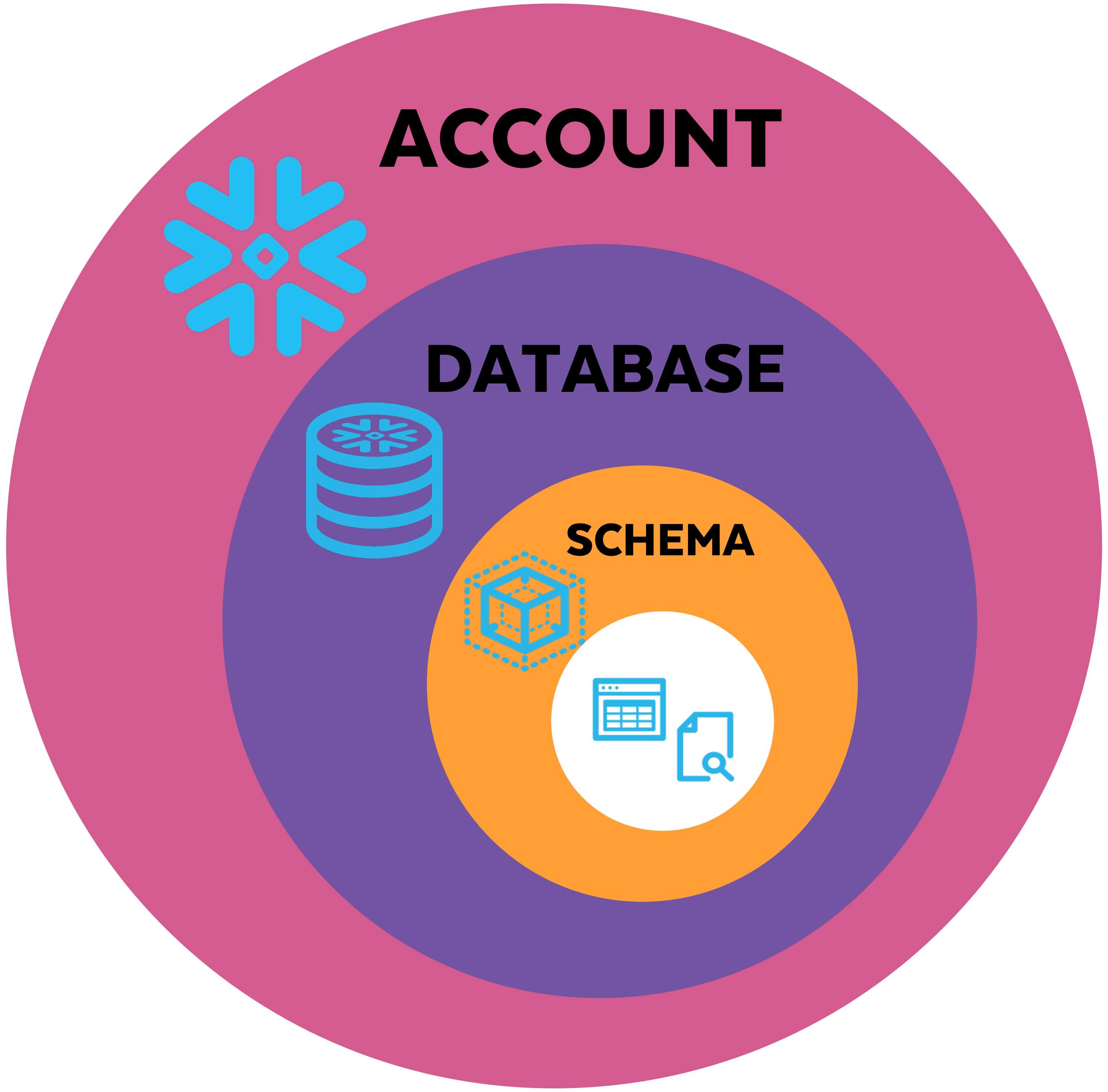
DATABASES

PUBLIC SCHEMA

INFORMATION SCHEMA

MY SCHEMA

STORAGE CONTAINER HIERARCHY



Tables, Views, Stages, File Formats, and Sequences are situated within Schemas

- The tables shown below are in different schemas.
- Snowflake allows you to bypass the schema during the drill-down process, making it seem as if tables and other objects are not within a schema, but they are.

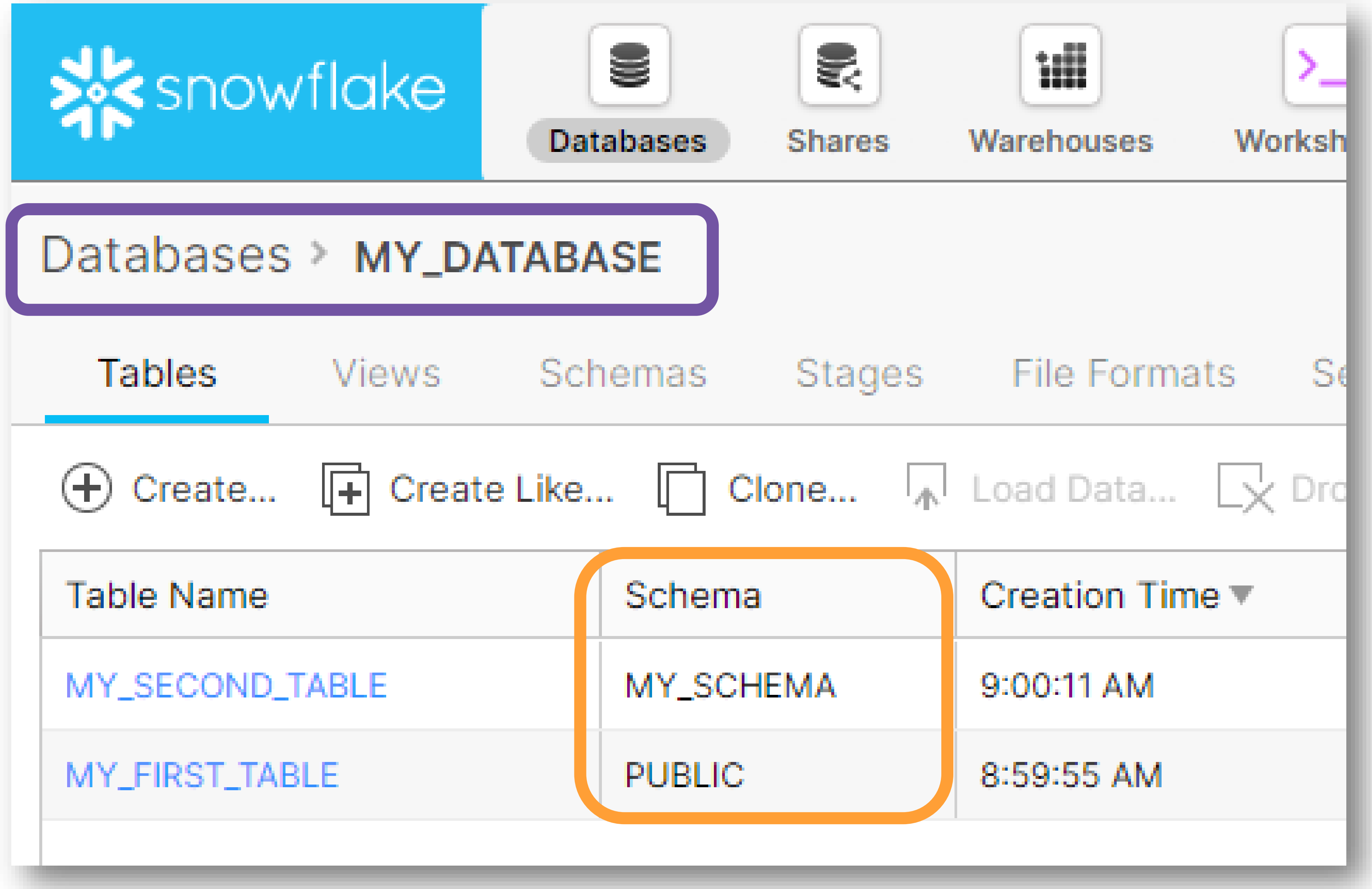


Table Name	Schema	Creation Time ▾
MY_SECOND_TABLE	MY_SCHEMA	9:00:11 AM
MY_FIRST_TABLE	PUBLIC	8:59:55 AM

SCHEMA OBJECTS

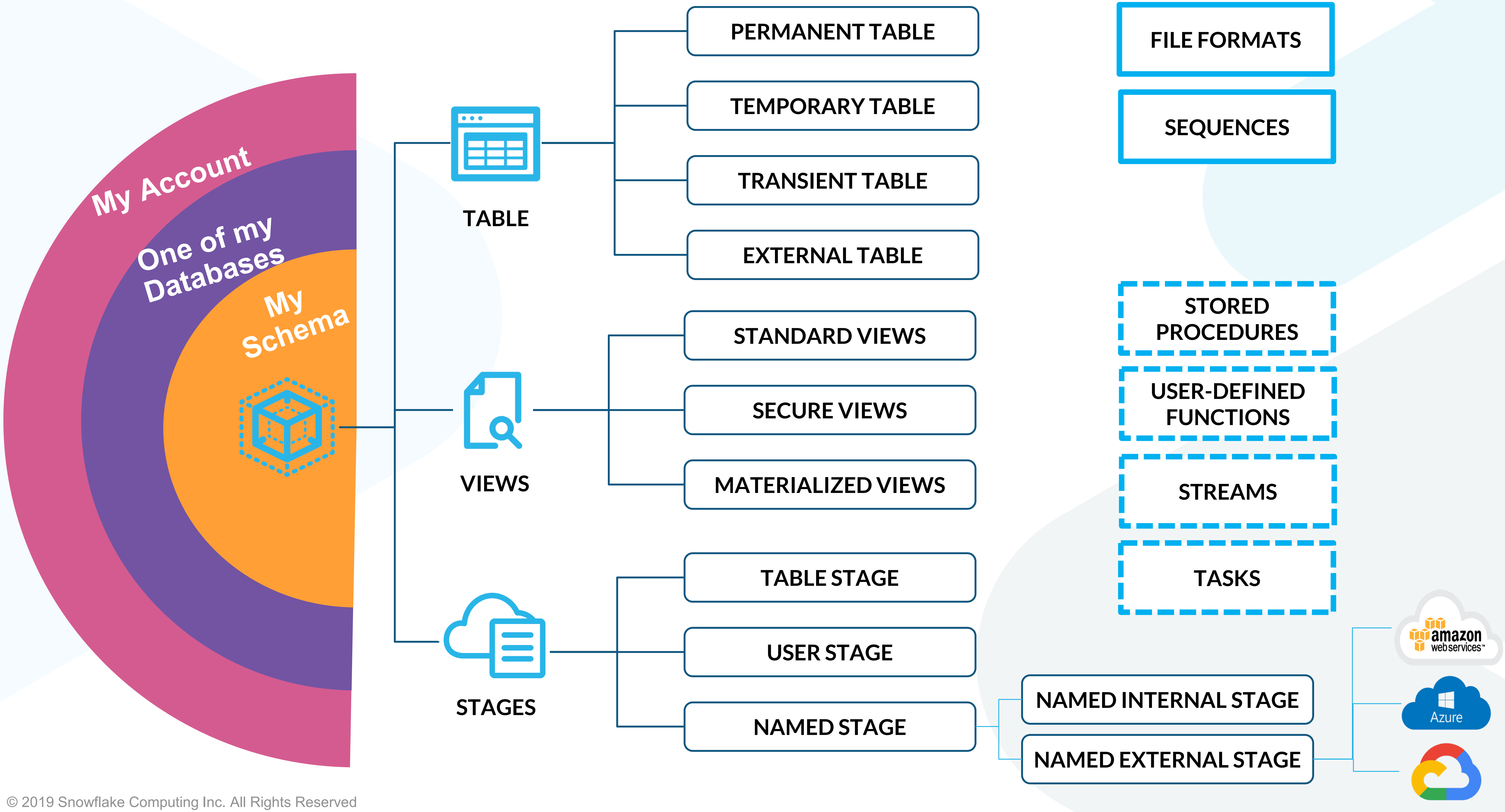


TABLE TYPES

PERMANENT

- Standard, default table type.
- Most commonly used.
- Default Time-Travel
- 7 Day Fail-Safe – non-configurable

TEMPORARY

- Lasts single session
- Can be used for moving data from one permanent to another with some transformation within the table
- No costs associated with failsafe storage

TRANSIENT

- No costs associated with failsafe storage
- Usually less costly than permanent tables.
- Time travel up to one day is available

EXTERNAL

- Stored in External Stages
- Read Only
- DML SQL works: Selects, Aggregations, Joins
- Can use in a View
- No cloning, time travel or fail-safe

	PERMANENT	TEMPORARY	TRANSIENT	EXTERNAL
Time-Travel	✓	✓	✓	X
Fail-Safe	✓	X	X	X

VIEW TYPES

Standard/Regular View

- Default View type
- Most commonly used
- Underlying DDL available to any role with access to the view

Secure View

- The underlying query definition and details are only visible to authorized users
- Snowflake query optimizer behaves differently when running the SQL contained in a secure view so that it doesn't accidently reveal things about the underlying data
- May not run as fast as a standard view, because the optimizer isn't used the same way

Materialized View

- It has "view" in the name, but it behaves more like a table
- Results are stored just as a table would be
- In Snowflake, these are auto-refreshed
- Can provide cost savings but are inflexible and somewhat limited in use

STAGE TYPES

Table Stage
@%[TABLE_NAME]

User Stage
@~[LOGIN]

- Automatically defined – No set up needed
- Not viewable on Stages tab of WebUI
- You must specify certain file format-like parameters as part of the command during each COPY INTO event

Named Internal Stage
@[STAGE_NAME]

Named External Stage
@[STAGE_NAME]

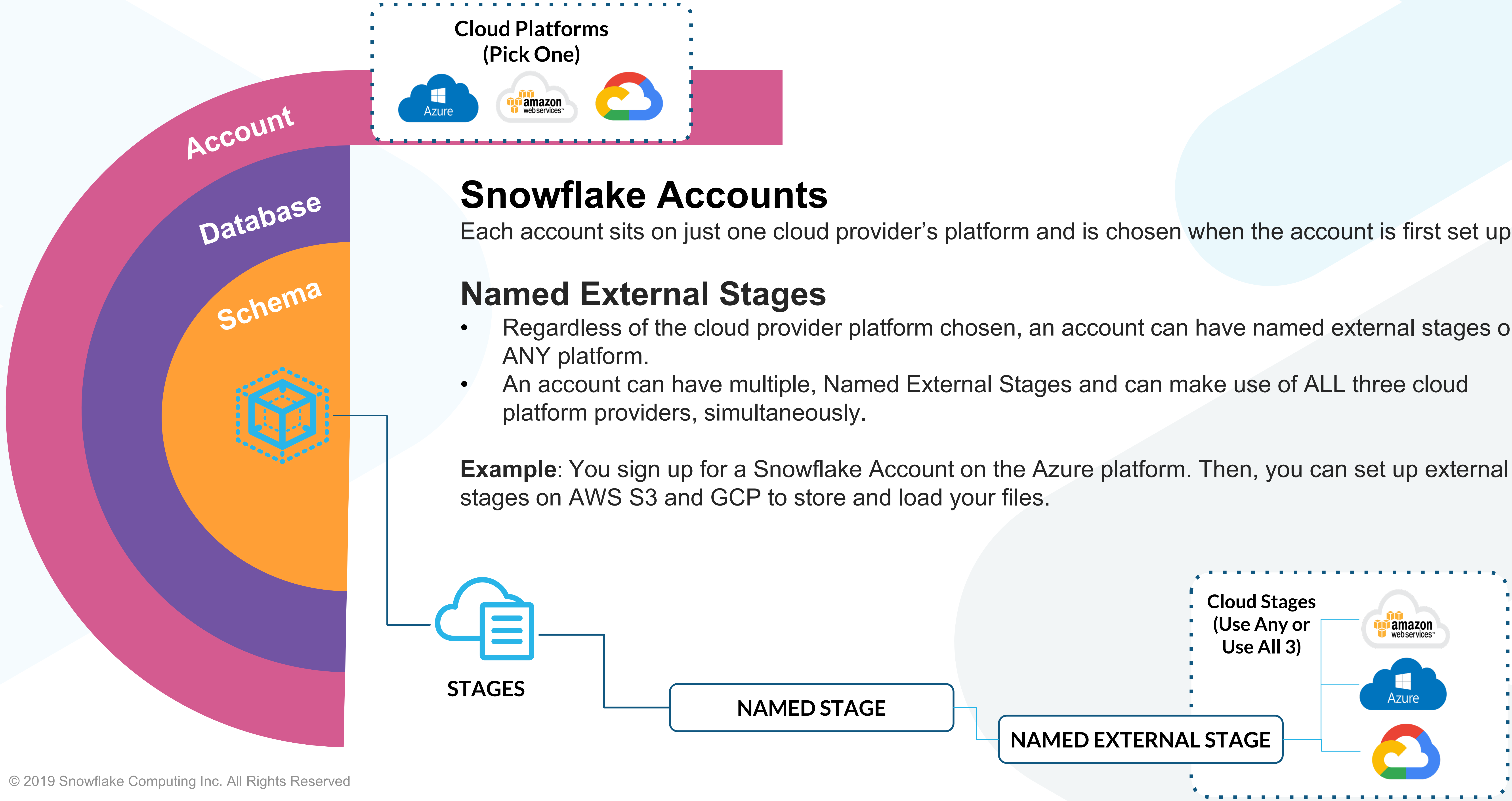
- Definable via the WebUI
- More flexible than Table and User Stages
- Viewable on Stages tab of Database Area of WebUI
- Have embedded default File Format definitions or you can specify named File Formats for use

GETTING FILES INTO AND OUT OF STAGES

- Use the PUT command to load files into any type of stage*
- Use the COPY INTO command to move files from any stage into a table
- Use the COPY INTO command to unload files from tables and put them into any kind of stage.

* The PUT command cannot be run from WebUI Worksheet. The PUT command can be run from SnowSQL, the Snowflake CLI client. The PUT command can also be run via other programmatic interfaces. You can also load files into a stage by going to the AWS, GCP, or Azure console and dragging them into storage or uploading them via the tools offered by the cloud platform providers.

PLATFORM VS. STAGES



Snowflake Accounts

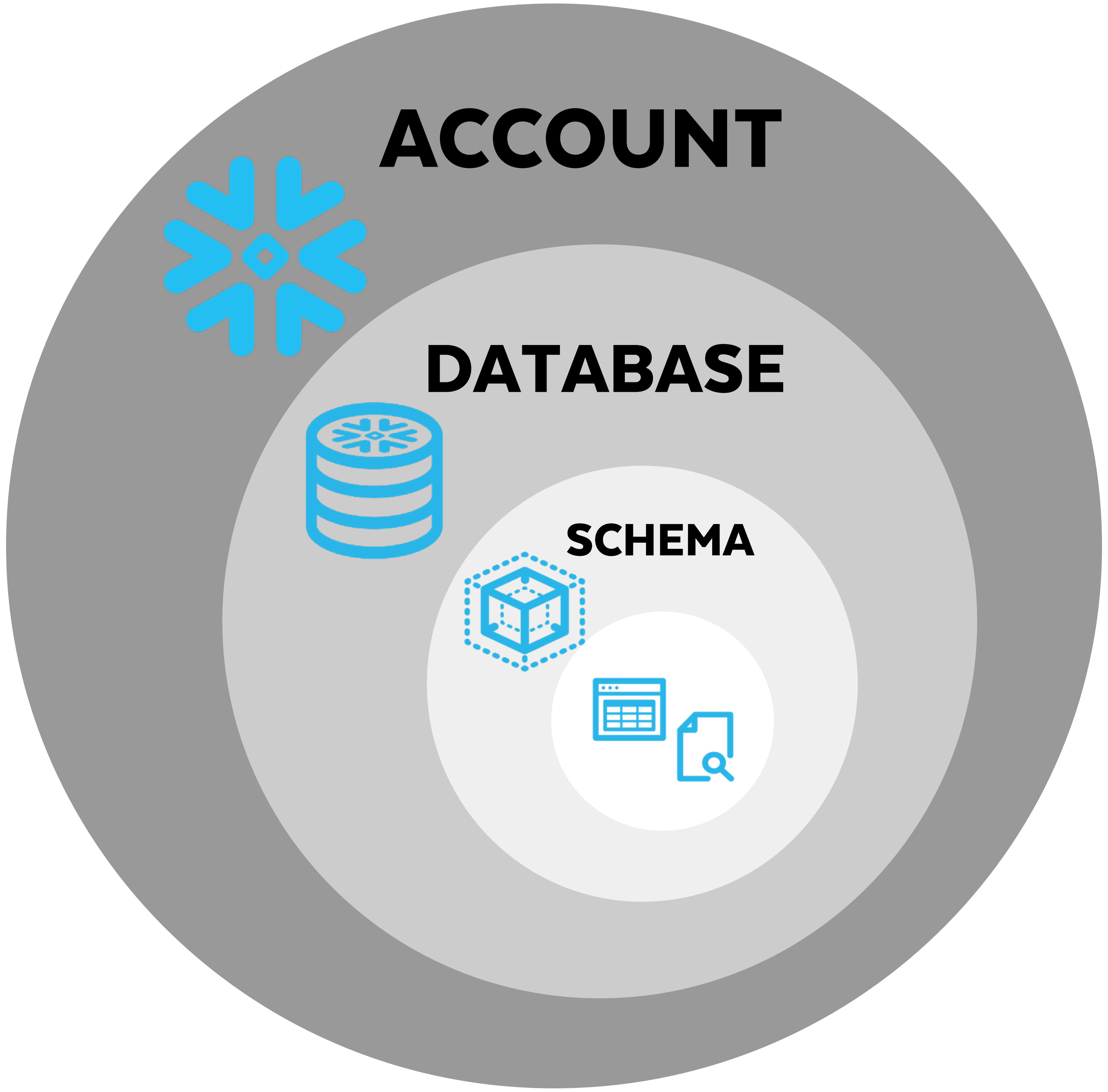
Each account sits on just one cloud provider’s platform and is chosen when the account is first set up.

Named External Stages

- Regardless of the cloud provider platform chosen, an account can have named external stages on ANY platform.
- An account can have multiple, Named External Stages and can make use of ALL three cloud platform providers, simultaneously.

Example: You sign up for a Snowflake Account on the Azure platform. Then, you can set up external stages on AWS S3 and GCP to store and load your files.

STORAGE HIERARCHY



Further Reading:

DATABASES AND SCHEMAS
https://docs.snowflake.net/manuals/sql-reference/ddl-database.html#database-schema-share-ddl

TABLES
https://docs.snowflake.net/manuals/user-guide/tables-temp-transient.html#comparison-of-table-types

VIEWS
https://docs.snowflake.net/manuals/user-guide/views-introduction.html

STAGES
https://docs.snowflake.net/manuals/user-guide/data-load-local-file-system-create-stage.html#types-of-stages