

Snowflake Data Loading: Overview

What is the stage in Snowflake?

In lay man terms **stage** are a data landing zone in Snowflake.

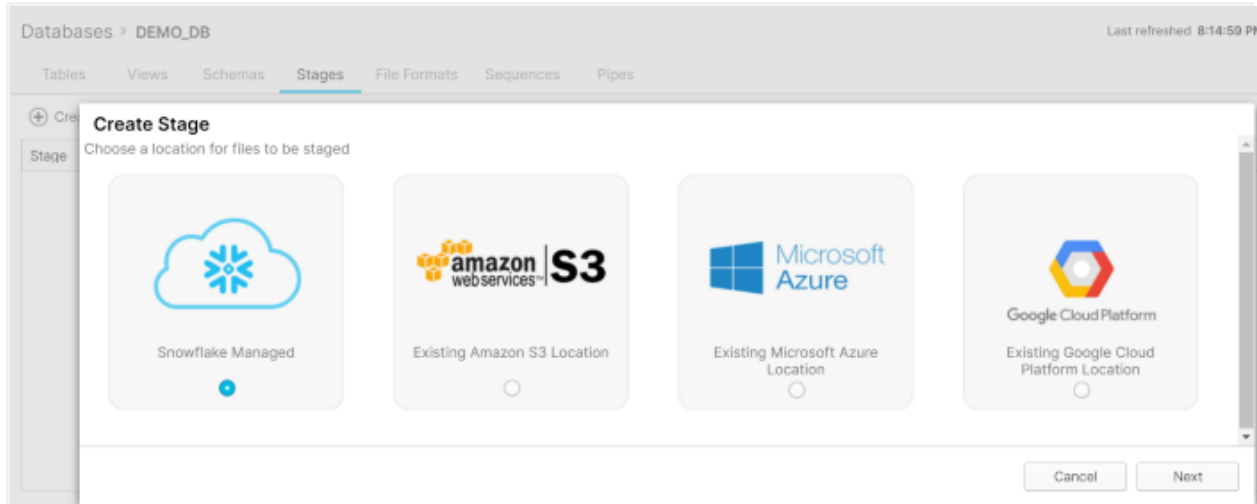
Is there different types of stage in Snowflake Account?

Yes, we have 4 types of stages.

- User stage
- Table stage
- Internal Named Stage
- External Named Stage

How to create a stage for data loading?

You can use Snowflake Web UI to create stage (Database>Stages>+Create) or you can go code way.



Code for Internal Name stage:

CREATE STAGE "DEMO_DB"."PUBLIC".TestStage;

Why we need 4 different types of stage? What is the purpose?

All stage has its own pro's and con's. Let's see how.

User Stage: You can't create user stage. 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 stages cannot be altered or dropped. To upload file you have to use put command and to download file you have to use get command from snowsql command line.

Table Stage: You can't create table stage. A table stage is available for each table created in Snowflake. This stage type is

designed to store files that are staged and managed by one or more users but only loaded into a single table. Table stages cannot be altered or dropped. Note that a table stage is not a separate database object; rather, it is an implicit stage tied to the table itself. To upload file you have to use put command and to download file you have to use get command from snowsql command line.

Internal Named Stage: You can create a named internal stage. A named internal stage is a database object created in a schema. This stage type can store files that are staged and managed by one or more users and loaded into one or more tables. Because named stages are database objects, the ability to create, modify, use, or drop them can be controlled using security access control privileges. To upload file you have to use put command and to download file you have to use get command from snowsql command line. You can't use Snowflake Web UI to Upload/Download any file to Internal Named Stage.

External Named Stage: External stage is just like named internal stage except this is created on top of cloud storage services rather than Snowflake internal storage. This object stores the URL to files in cloud storage. Snowflake supports AWS S3, Azure Blob & GCP bucket for external stage. To upload and download you have to use cloud native utility or any external utility which particular cloud supports. Snowflake proprietary put & get command will not work here.

What loading method do we have available in Snowflake?

Snowflake covers both bulk data loading scenario using “**COPY INTO**” command and incremental/continuous data load scenario using [Snowpipe](#). Copy Into is designed to load mass data while Snowpipe is designed to load small volume of data continuously.

How this service work’s under the hood?

Copy Into command uses user provided virtual warehouse
Snowpipe uses snowflake provided compute.

What all format Snowflake support while loading data?

Snowflake support multiple file format. See below:

Delimited files (CSV, TSV, etc.): Any valid delimiter is supported; default is comma (i.e. CSV).

JSON: JSON stands for JavaScript Object Notation. **JSON** is a lightweight format for storing and transporting data

Avro: Includes automatic detection and processing of staged Avro files that were compressed using Snappy.

ORC: Includes automatic detection and processing of staged ORC files that were compressed using Snappy or zlib.

Parquet: Includes automatic detection and processing of staged Parquet files that were compressed using Snappy.

XML: Supported as a preview feature.

Can we do transformation while loading data in Snowflake?

Yes, simple transformation is possible while loading data using either of loading methods. Options include:

- Column reordering
- Column omission
- Casts
- Truncating text strings that exceed the target column length

There is no requirement for your data files to have the same number and ordering of columns as your target table.

What if I don't want to load data but just wanted to query few data from my files?

Snowflake provide you the facility of External table to fulfill this requirement. External tables enable querying existing data stored in external cloud storage for analysis without first loading it into Snowflake. The source of truth for the data remains in the external cloud storage.

How about compression of files in stage ? Is it possible?

Yes, you can stage the file and depending upon whether it is compressed or uncompressed, it is taken care by snowflake. Below is a chart for more details.



Feature	Supported	Notes
Uncompressed files	gzip	When staging uncompressed files in a Snowflake stage, the files are automatically compressed using gzip, unless compression is explicitly disabled.
Already-compressed files	gzip	Snowflake can automatically detect any of these compression methods or you can explicitly specify the method that was used to compress the files.
	bzip2	
	deflate	
	raw_deflate	
	Brotli	Auto-detection is not yet supported for Brotli-compressed files; when staging or loading Brotli-compressed files, you must explicitly specify the compression method that was used.
	Zstandard	

(Source: Snowflake)

How secure is our data in Stage? Are they encrypted?

When staging unencrypted files in a Snowflake internal location, the files are automatically encrypted using 128-bit keys. 256-bit keys can be enabled (for stronger encryption); however, additional configuration is required. Files that are already encrypted can be loaded into Snowflake from external cloud

storage; the key used to encrypt the files must be provided to Snowflake.

Are there any guidelines or consideration we need to follow for data loading in Snowflake?

Yes, there is full documentation on the same. Please find it [here](#).

How does Snowflake overcome loading the same file twice ?

Snowflake maintains the detailed load metadata for each table into which data is loaded, including:

- Name of each file from which data was loaded
- File size
- ETag for the file
- Number of rows parsed in the file
- Timestamp of the last load for the file
- Information about any errors encountered in the file during loading

This load metadata expires after 64 days.

What if load metadata expires or anyone forcefully wanted to load the same file for some reason.?

To load files whose metadata has expired, set the **LOAD_UNCERTAIN_FILES** copy option to true. The copy

option references load metadata, if available, to avoid data duplication, but also attempts to load files with expired load metadata.

Alternatively, set the **FORCE** option to load all files, ignoring load metadata if it exists. Note that this option reloads files, potentially duplicating data in a table.

How actually we load data using COPY INTO command ?

Sample code below:

```
copy into dummy_data from @%dummy_data/data1/
```

```
pattern='.*dummy_data[0-9].csv';
```

```
copy into dummy_data from @%dummy_data/data1/
```

```
files=('dummyfile1.csv','dummyfile2.csv','dummyfile3.csv')
```

How does Snowpipe code looks like?

Sample code below:

```
CREATE OR REPLACE PIPE TEMPARATURE_PIPE
```

```
auto_ingest = true
```


integration = 'TEMPERATURE_NI'

as

copy into "IOTDB"."IOTDEMO"."TEMPERATURE"

(data)

from [@TEMPERATURE_STAGE](#)

file_format = (type = 'JSON');

For Data Loading Hand on please click [here](#).

To see more details on COPY INTO command you can refer Snowflake documentation [here](#).

To see more details on Snowpipe command you can refer Snowflake documentation [here](#).