

Snowflake Health Check Guide



| | |
|-----------------|--------------------------------|
| Product Version | 4.8.0 |
| Document Type | Health Check Preparation Guide |
| Authors | Snowflake Data source Team |
| Reviewer | Red Team & Architects |
| Approver | CTO |
| Total Pages | 10 |
| Document Status | Draft |

Table Of Contents

| | | |
|------------|---------------------|----------|
| 1.1 | Objectives | 2 |
| 1.2 | Architecture | 2 |

| | | |
|-----|---|------------------------------|
| 1.3 | Pre-requisite | 3 |
| 1.4 | Download Snowflake Metadata for health check. | 3 |
| 1.5 | Output Files | Error! Bookmark not defined. |

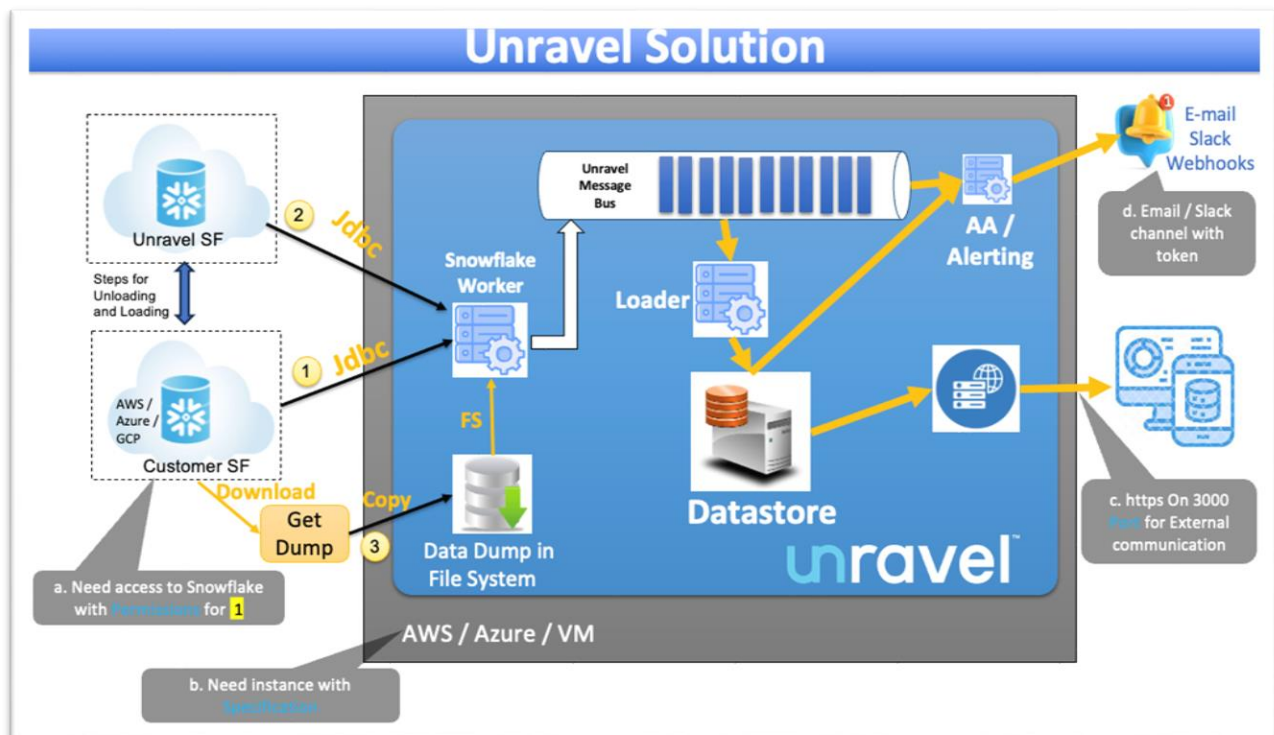
Document Version Record

| Date | Version # | Author | Remarks / Reason |
|-----------|-----------|----------|------------------|
| 02-May-23 | 1.0 | Dev Team | New Document |

1.1 Objectives

Health check download for snowflake unravel product.

1.2 Architecture



We are going to follow **2** Download from customer snowflake environment and upload to unravel snowflake environment.

1.3 Pre-requisite

A. Snowflake account Access

- a. Health check data upload from snowflake
 - i. User should have permission to create stage and file format.
 - ii. Snowsql 1.2.27 should be installed
 - iii. Required a database and schema to upload the new data.
 - iv. Required a user with create/delete table, insert data permission to the above created role.
 - v. Tested on Mac-OS(12.3), Windows 10, Ubuntu(16.04.7 LTS), CentOS-8

1.4 Upload Snowflake Metadata for health check.

The following scripts let you upload the account usage information to your Snowflake account.

- **prepare_schema.sql**: Create the required tables in the specified database/schema.
- **snowsql_upload_data.sql**: Uploads Snowflake data downloaded from customer environment to the specified database/schema.

Do the following to upload Snowflake metadata:

1. Download the download/upload scripts from this location:
<https://github.com/unraveldata-org/snowflake-data-loader/tree/main/script>
2. Using SnowSQL connect to a Snowflake account. This account must have access to creating **stage** and **file** format.
3. Execute the snowsql_upload_data.sql script with the required arguments to upload the metadata to Snowflake. Refer to the following list of arguments for more details:

Step-1:

Login to snowsight and execute the create table script - **prepare_schema.sql**.

Once the stored procedure got created, execute the stored procedure with the command:

```
call prepare_replication_schema(DbName, SchemaName);
```

For example:

```
call prepare_replication_schema ('testdb1', 'testschema1');
```

| Parameter | Description |
|------------|--|
| DbName | Name of the DB, under which customer data need to replicate. |
| SchemaName | Name of the schema, under which customer data need to replicate. |

After the script is executed, the database & the schema gets created.

Databases

Worksheets

Pinned (3)

 QUERY_HISTORY

 All Objects

...

 TESTDB

  TESTDB1

  INFORMATION_SCHEMA

  PUBLIC

  TESTSCHEMA1

 Tables

 ACCESS_HISTORY

 METERING_DAILY_...

 METERING_HISTORY

 QUERY_HISTORY

 QUERY_HISTORY_RT

 TABLES

 WAREHOUSES

 WAREHOUSE_EVE...

 WAREHOUSE_LOA...

 WAREHOUSE_MET...

 WAREHOUSE_PARA...

Step-2:

Execute the snowsql_upload_data.sql script with the required arguments to upload all the data downloaded in the above 3 steps. Refer to the following list of arguments for more details:

```
snowsql -f snowsql_upload_data.sql -d ${db} -s ${schema} -r ${role} -a ${account} -u ${user} -w ${warehouse} -o variable_substitution=true -o log_file=/opt/script/snowsql_upload_data.log --variable path=${path} --variable stage_name=${unravel_stage_upload} --variable file_format=${unravel_file_format_upload}
```

For example:

Windows :

```
snowsql -f "snowsql_upload_data.sql" -d testdb1 -s testschema1 -r accountadmin -a rtb81672.us-east-1 -u unraveluser -w UnravelData -o variable_substitution=true -o log_file=./snowsql_upload_data.log --variable path=c:\opt\unraveldata\ --variable stage_name=unravel_stage_upload --variable file_format=unravel_file_format_upload
```

During copy make sure double quotes are correctly transferred.

Mac / Linux :

```
snowsql -f snowsql_upload_data.sql -d testdb1 -s testschema1 -r accountadmin -a rtb81672.us-east-1 -u unraveluser -w UnravelData -o variable_substitution=true -o log_file=/opt/script/snowsql_upload_data.log --variable path=/opt/unraveldata/ --variable stage_name=unravel_stage_upload --variable file_format=unravel_file_format_upload
```

| Parameter | Description |
|-----------|---|
| -f | Specify the file name of the script that is executed. Either specify the fully qualified name or just filename from the current directly. For Windows: Please put the parameter value in double quotes (" ") |
| -d | Specify the target Snowflake Account database name. This database should have rights to create stage & file-format. |

| Parameter | Description |
|-----------------------------|--|
| -u | Specify the target Snowflake Account User name. This schema should have rights to create stage & file-format. |
| -s | Specify the target Snowflake Account Schema name. |
| -r | Target Snowflake Account Role |
| -a | Specify the target Snowflake Account name |
| -o variable_substitution | Enable the variable substitution switch in the script. Some variables are used in the script for which the values must be passed from the CLI. Set this to true . |
| -o log_file | Specify the name of log file that must be generated on command execution. |
| --variable path | Specify the local path to where the snowflake system metadata is downloaded or copied. |
| --variable stage_name | Specify the stage name, which is used to keep the temporary files for upload. |
| --variable file_format | Specify the file format name, which is used by the upload scripts. |
| -o | Provide the output related arguments to get the logs in the specified path and format. |

1.5 Video Guide

<https://github.com/unraveldata-org/snowflake-data-loader/blob/main/Resources/Snowflake%20health%20check%20upload%20guide.mov>