**Snowflake Metadata Share Guide**



|  |  |
| --- | --- |
| Product Version | 4.8.0 |
| Document Type | Snowshare Guide |
| Authors | Snowflake Data source Team |
| Reviewer | Red Team & Architects |
| Approver | CTO |
| Total Pages | 10 |
| Document Status | Release |

Table Of Contents

* 1. Objectives 2
  2. Architecture 2
  3. Pre-requisite 3
  4. Share Snowflake Metadata with Snowflake secure share. 3
     1. Execute below statement to create the procedure and necessary functions. 4
     2. One time execution 14
     3. To continue execution on certain interval this is created as tasks. 14
  5. Share the Transient tables to unravel account. 15
  6. Receive the data in recipient account (sql mode). 16
  7. Share the Transient tables to unravel account (UI Mode). 16
  8. Receive the data in recipient account ui mode. 19
  9. Configure recipient account in unravel. 21

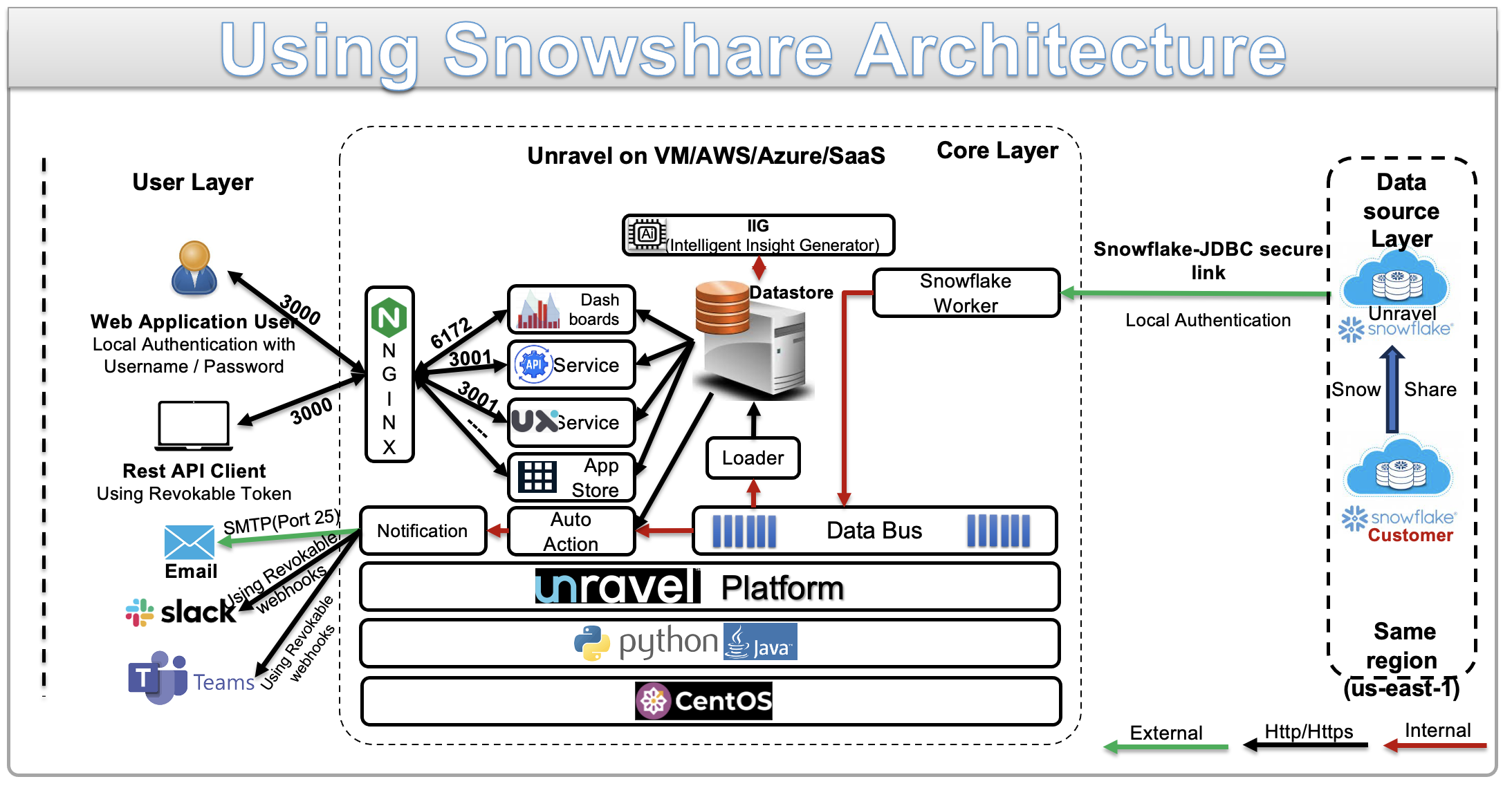
Document Version Record

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

## Objectives

Health check download for snowflake unravel product.

## Architecture



## Pre-requisite

1. **Snowflake account Access through snowshare**
   * 1. Create a unravel user in snowflake.
     2. Grant select for unravel user on schema SNOWFLAKE.ACCOUNT\_USAGE.
     3. Create a database unraveldb and a schema unravelschema to create a stored procedure required for metadata collection from account\_usage, information\_schema & Profile and store in local schema.
     4. Execute procedure to create tables and serverless task to populate the metadata from the account metadata.
     5. Need to have Snowshare access to share the local schema to unravel snowflake account , Execute the snowshare script to give access to ”Unravel Snowflake account”.
     6. Based on the latency required the serverless/warehouse task will cost.

## Share Snowflake Metadata with Snowflake secure share.

## Execute below statement to create the procedure and necessary functions.

<https://github.com/unraveldata-org/snowflake-data-loader/blob/ba204fad9def3de7ded263a59d462f669be5346c/script/replication/replication_query.sql>

## One time execution for POV for 2 days

CALL CREATE\_TABLES('UNRAVEL\_SHARE','SCHEMA\_4823\_T');

CALL REPLICATE\_ACCOUNT\_USAGE('UNRAVEL\_SHARE','SCHEMA\_4823\_T',2);

CALL WAREHOUSE\_PROC('UNRAVEL\_SHARE','SCHEMA\_4823\_T');

CALL REPLICATE\_REALTIME\_QUERY('UNRAVEL\_SHARE','SCHEMA\_4823\_T',10);

CALL CREATE\_QUERY\_PROFILE(dbname => 'UNRAVEL\_SHARE',schemaname => 'SCHEMA\_4823\_T', cost => '1', days => '2');

## One time execution for health check for 7 days

CALL CREATE\_TABLES('UNRAVEL\_SHARE','SCHEMA\_4823\_T');

CALL REPLICATE\_ACCOUNT\_USAGE('UNRAVEL\_SHARE','SCHEMA\_4823\_T',7);

CALL WAREHOUSE\_PROC('UNRAVEL\_SHARE','SCHEMA\_4823\_T');

CALL REPLICATE\_REALTIME\_QUERY('UNRAVEL\_SHARE','SCHEMA\_4823\_T',10);

CALL CREATE\_QUERY\_PROFILE(dbname => 'UNRAVEL\_SHARE',schemaname => 'SCHEMA\_4823\_T', cost => '1', days => '7');

## To continue execution on certain interval this is created as tasks for POV not required for health check.

1. Based on requirement schedule time / lookbackdays /cost can be changed.

-- create account usage tables Task

CREATE OR REPLACE TASK replicate\_metadata

WAREHOUSE = UNRAVELDATA

SCHEDULE = '60 MINUTE'

AS

CALL REPLICATE\_ACCOUNT\_USAGE('UNRAVEL\_SHARE','SCHEMA\_4823\_T',2);

-- create warehouse replicate Task

CREATE OR REPLACE TASK createWarehouseTable

WAREHOUSE = UNRAVELDATA

SCHEDULE = '60 MINUTE'

AS

CALL warehouse\_proc('UNRAVEL\_SHARE','SCHEMA\_4823\_T');

-- create profile replicate task

CREATE OR REPLACE TASK createProfileTable

WAREHOUSE = UNRAVELDATA

SCHEDULE = '60 MINUTE'

AS

CALL create\_query\_profile(dbname => 'UNRAVEL\_SHARE',schemaname => 'SCHEMA\_4823\_T', cost => '1', days => '1');

-- create Task for replicating information schema query history

CREATE OR REPLACE TASK replicate\_realtime\_query

WAREHOUSE = UNRAVELDATA

SCHEDULE = '30 MINUTE'

AS

CALL REPLICATE\_REALTIME\_QUERY('UNRAVEL\_SHARE','SCHEMA\_4823\_T',10);

-- START ALL THE TASKS

ALTER TASK replicate\_metadata RESUME;

ALTER TASK createWarehouseTable RESUME;

ALTER TASK createProfileTable RESUME;

ALTER TASK replicate\_realtime\_query RESUME;

## Share the Transient tables to unravel account.

From SQL (Recommended): replace ${CUSTOMER\_NAME} with unique value to share

-- Share tables

Create share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

Grant Usage on database UNRAVEL\_SHARE to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

Grant Usage on schema SCHEMA\_4823\_T to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE WAREHOUSE\_METERING\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE WAREHOUSE\_EVENTS\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE WAREHOUSE\_LOAD\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE TABLES to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE METERING\_DAILY\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE METERING\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE DATABASE\_REPLICATION\_USAGE\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE REPLICATION\_GROUP\_USAGE\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE DATABASE\_STORAGE\_USAGE\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE STAGE\_STORAGE\_USAGE\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE SEARCH\_OPTIMIZATION\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE DATA\_TRANSFER\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE AUTOMATIC\_CLUSTERING\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE SNOWPIPE\_STREAMING\_FILE\_MIGRATION\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE TAG\_REFERENCES to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE QUERY\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE ACCESS\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE IS\_QUERY\_HISTORY to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE WAREHOUSE\_PARAMETERS to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE WAREHOUSES to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

GRANT SELECT ON TABLE QUERY\_PROFILE to share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

alter share ${CUSTOMER\_NAME}\_UNRAVEL\_SHARE add accounts = HFB47355;

## Receive the data in recipient account (Sql mode).

--Validate that the inbound share is available to the consumer account.

Show shares like '%UNRAVEL%';

Use role AccountAdmin;

create database ${CUSTOMER\_NAME}\_SHARE from share FWTTICE.PRIMARY\_PG.${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

## Share the Transient tables to unravel account (UI Mode).

Sharing through UI :

E.g. URL : <https://app.snowflake.com/fwttice/primary_pg/#/data/shared/outbound>

Replace your account in above URL instead of “fwttice”

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Now the tables are shared to **HFB47355 Account**

## Receive the data in recipient account ui mode.

Login as account admin.

A screenshot of a computer

Description automatically generated

Click here.

Select the role.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Now data is available in recipient account.

A screenshot of a computer

Description automatically generated

## Configure recipient account in unravel.

Show shares like '%UNRAVEL%';

Use role AccountAdmin;

create database ${CUSTOMER\_NAME}\_SHARE from share FWTTICE.PRIMARY\_PG.${CUSTOMER\_NAME}\_UNRAVEL\_SHARE;

Database: ${CUSTOMER\_NAME}\_SHARE

Other configurations are same.