Snowflake-Azure POC Documentation for Automation:

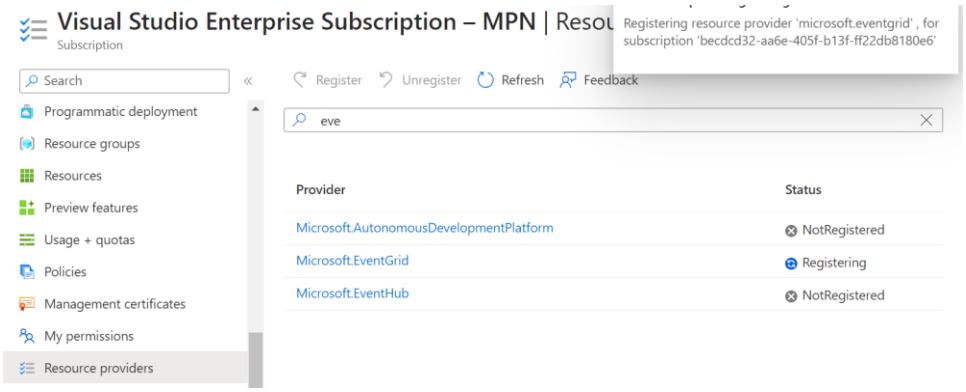
Grab the Tenent ID from Azure Active Directory:

- Tenent ID - 094d25ba-3306-4bc2-b789-6eb55f87b309

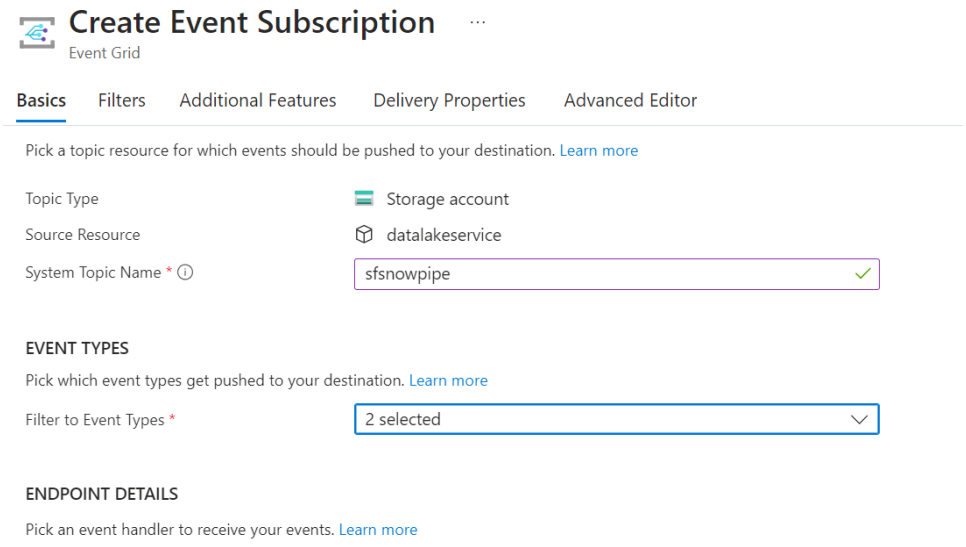
Create a new queue in Blob Account storage

- Queue - https://demodatagrp.queue.core.windows.net/snow-queue

Make sure to enable Event Grid:

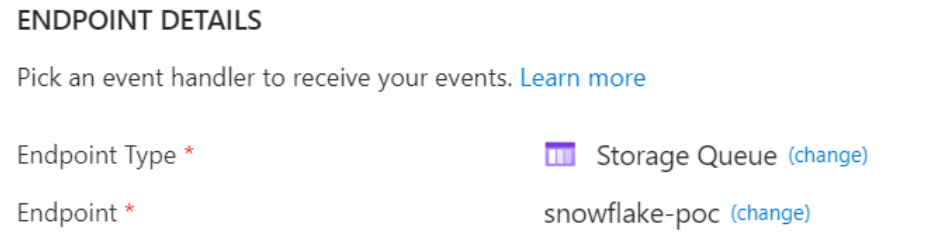


Then Create a new Event:

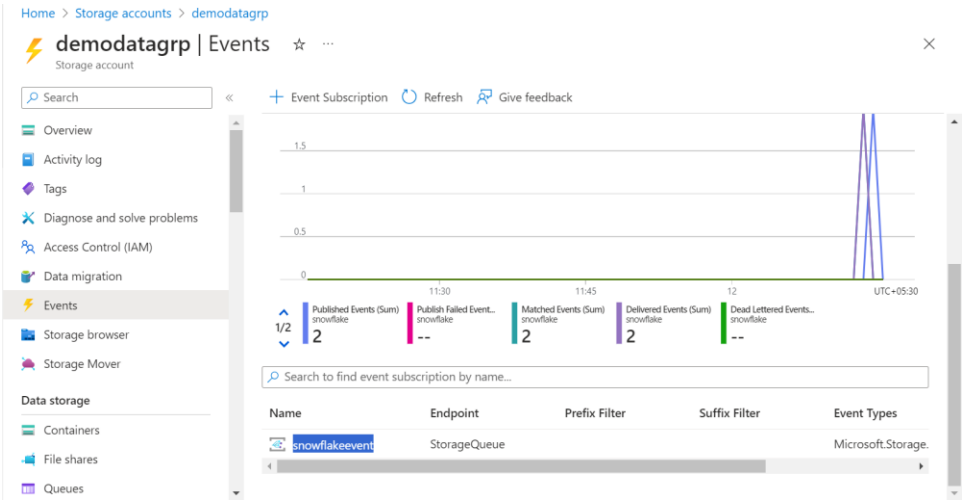


System Topic Name: snowflakeevent

Choose the queue we created:



Make Sure the Event is Created:



Navigate to Snowflake Account:

Create a Table in snowflake:

```
create or replace TABLE SNOWFLAKE_AZURE.PUBLIC.SNOWFLAKE_AZ (
ID NUMBER(38,0),
NAME VARCHAR(20),
MARKS NUMBER(38,0)
);
```

Create a New Integration in Snowflake:

```
create or replace notification integration myint_pipe_event
    enabled=TRUE
    type=QUEUE
    NOTIFICATION_PROVIDER=AZURE_STORAGE_QUEUE
```

AZURE_STORAGE_QUEUE_PRIMARY_URI='https://demodatagrp.queue.core.windows.net/snow-queue/'

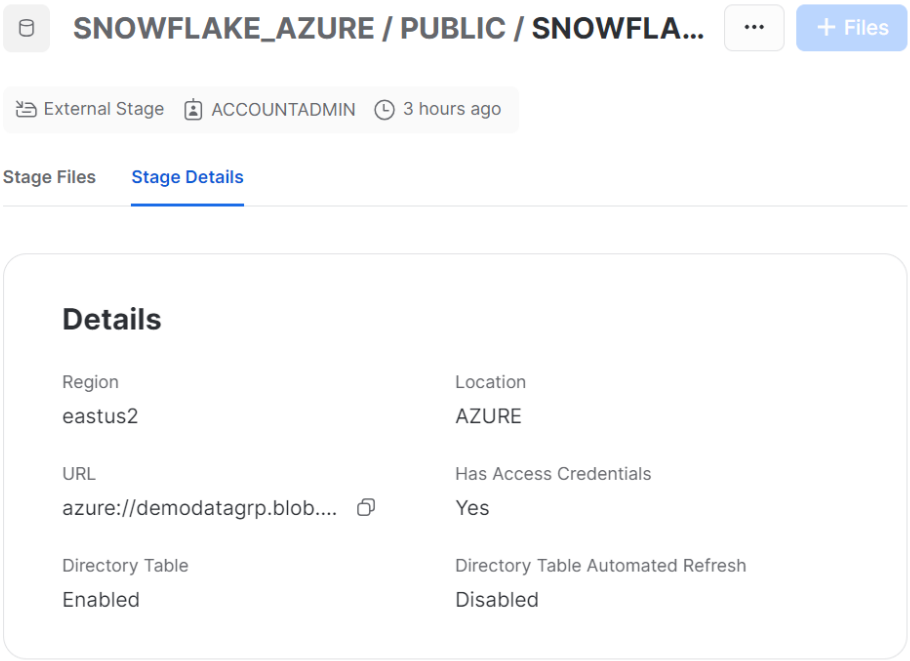
AZURE_TENANT_ID='094d25ba-3306-4bc2-b789-6eb55f87b309'

Describe the Integration We Created:

DESC NOTIFICATION INTEGRATION my_snowpipe_integration;

- Get the AZURE_CONSENT_URL from the output
  - (https://login.microsoftonline.com/094d25ba-3306-4bc2-b789-6eb55f87b309/oauth2/authorize?client_id=d8f5f476-256d-4d67-bbdb-0230ad7d9ca6&response_type=code)
- Get back to Azure
- Search for Enterprice Applications
- Search for SnowflakePAC and copy the Name
- Go to Azure Storage Account
- Go to IAM from Left side panel
- Navigate to Role Assignment
- Give a Storage Queue Contributor access to SnowflakePAC.
- Get a path that you post files from Data Lake Storage.
  - https://demodatagrp.blob.core.windows.net/snow-data-blob
- Create a new Shared Access Key

- Copy Blob service SAS URL from the generated key.
  - https://datalakeservice.blob.core.windows.net/?sv=2022-11-02&ss=bfqt&srt=co&sp=rwdlacupyx&se=2023-12-19T18:51:05Z&st=2023-12-19T10:51:05Z&spr=https,http&sig=W8nDef2526hGOEKjDSjzV9Ql%2FkcJHzH%2FU6FsyS6C9F4%3D
- Copy the Blob SAS Token
  - sp=racwdlmeop&st=2023-12-19T11:06:39Z&se=2023-12-19T19:06:39Z&spr=https&sv=2022-11-02&sr=c&sig=M4Hgiq7Zw3XqbFNNgGn74qVOIlfT%2FClnO3cxSz9xen4%3D
- Create a new STAGE using the above Credentials in snowflake.



-

Then Create a new SnowPIPE in snowflake:

create or replace PIPE mysnowpipe

AUTO_INGEST=TRUE

INTEGRATION=MYINT_PIPE_EVENT

AS

COPY INTO SNOWFLAKE_AZ FROM @SNOWFLAKE_FINAL

FILE_FORMAT=(TYPE='CSV');

After that create a Stream to implement CDC:

CREATE OR REPLACE STREAM snow_stream ON TABLE snowflake_az;

Create a Actual Table to insert the Data:

create or replace TABLE SNOWFLAKE_AZURE.PUBLIC.MYCSVTABLE (

ID NUMBER(38,0),

NAME VARCHAR(20),

MARK NUMBER(38,0)

);

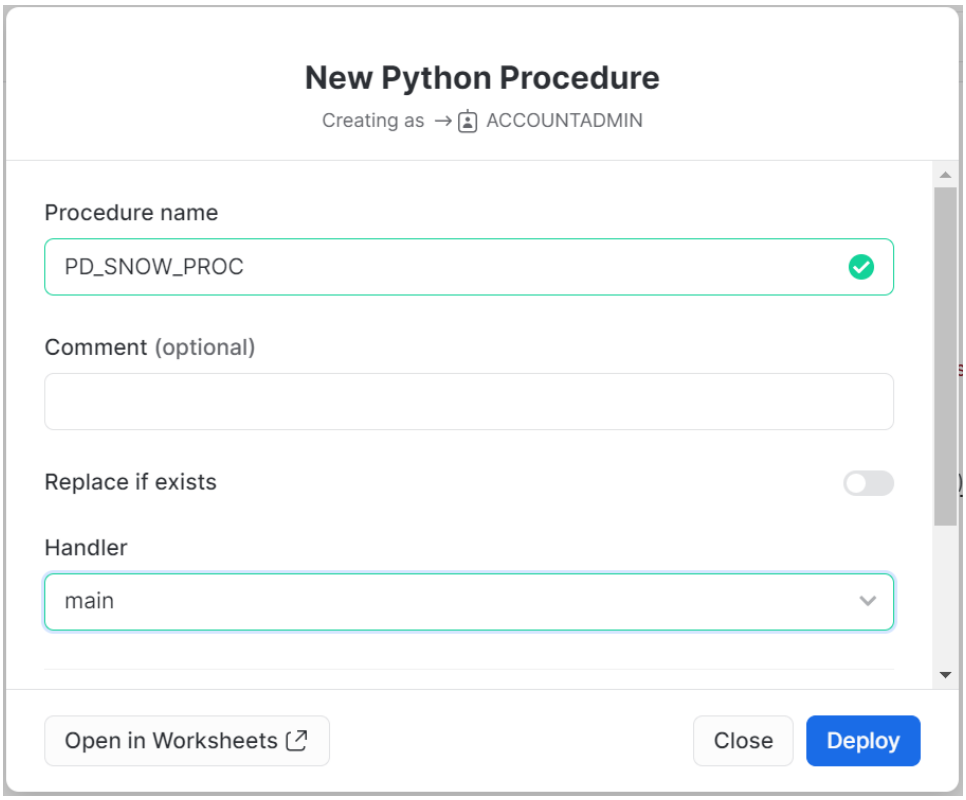
Implement Snowspark Code to Perform ETL Operation:

- Once the data was posted in Table, Access the table using snowpark.

```
from snowflake.snowpark.functions import col
import snowflake.snowpark as snowpark

def main(session: snowpark.Session):

    source_dataframe = session.sql('SELECT ID, Name, Marks FROM snow_stream')

    filtered_dataframe = source_dataframe.filter(col('MARKS') < 80)

    filtered_dataframe.write.mode('append').saveAsTable('mycsvtable')

    final_df = session.sql('truncate table snowflake_az')

    return final_df
```

-
- Here snow_stream is our Stream name which captures CDC from table where data loaded from Azure.

Deploy the Snowpark Code:



- Mycsvtable is a final table that we can store our increment data after Transformation.

Create a Task to Schedule the code:

CREATE OR REPLACE TASK DEMO_SNOW_TASK

WAREHOUSE = COMPUTE_WH

SCHEDULE = 'USING CRON */1 * * * * UTC'

WHEN

SYSTEM$STREAM_HAS_DATA(‘SNOW_STREAM’) = TRUE

AS

CALL PD_SNOW_PROC();

Resume the Task Initially:

alter task demo_snow_task resume;