Practical Lesson: Data Engineering with Cloud SQL and CDC

Hint for Naming Resources

When naming resources, use the format `de-2023-[RESOURCE NAME]-[YOUR NAME]` where `**[YOUR NAME]**` is your account name and should be filled by you. Adjust the `**[RESOURCE NAME]**` accordingly based on the specific resource you're creating.

Exercise: Setting up and Configuring Cloud SQL with Change Data Capture (CDC)

In this exercise, you'll set up a Cloud SQL instance, import a database dump, and enable CDC to track changes in the database.

Prerequisites

- A Google Cloud Platform account.
- Basic understanding of SQL commands.

Step 1: Set Up Cloud SQL

1. **Set the active project**: `bash gcloud config set project [YOUR PROJECT ID]` This command sets your active project, replacing `[YOUR PROJECT ID]` with your actual GCP project ID.
2. **Upload `dump.sql` to Google Cloud Shell**: Ensure you have the `dump.sql` file available in your environment.
3. **Copy the SQL dump to your GCS bucket**: `bash gsutil cp dump.sql gs://[YOUR BUCKET]` This uses the `gsutil` utility to copy the `dump.sql` file to your specified Google Cloud Storage bucket.
4. **Create a new Cloud SQL instance**: `bash gcloud sql instances create postgres \ --database-version=POSTGRES_15 \ --cpu=2 \ --memory=8GB \ --region=europe-west6 \ --authorized-networks=0.0.0.0/0` This command creates a new Cloud SQL instance named 'postgres'. It specifies the database version, CPU, memory, and other configurations. The process might take 4 to 5 minutes.

5. **Set the password for the postgres user**: `bash gcloud sql users set-password postgres \ --instance=postgres \ --password=postgres` This sets the password for the default postgres user.
6. **Enable logical decoding**: `bash gcloud sql instances patch postgres --database-flags=cloudsql.logical_decoding=on` Logical decoding is a method to extract changes which were written to the database. This flag ensures it's enabled.
7. **Identify the ServiceAccount linked to the SQL instance**: `bash gcloud sql instances describe postgres | grep "account"` Note down the ServiceAccount name that appears. You'll need it for the next steps.
8. **Grant the ServiceAccount permissions to access the bucket**: `bash gsutil iam ch [YOUR SERVICE ACCOUNT]:objectViewer gs://[YOUR BUCKET]`
9. **Assign the required IAM role to the ServiceAccount**: `bash gcloud projects add-iam-policy-binding [YOUR PROJECT ID] --member=serviceAccount:[YOUR SERVICE ACCOUNT] --role=roles/storage.objectAdmin`
10. **Create a new database in the SQL instance**: `bash gcloud sql databases create adventureworks \ --instance=postgres`

Step 2: Connecting to the Database and Importing Data

1. **Open a second tab in the Cloud Shell**.
2. **Connect to the Cloud SQL instance**: `bash gcloud sql connect postgres --user=postgres`
3. Once connected, run the following SQL commands: `sql GRANT ALL PRIVILEGES ON DATABASE adventureworks TO postgres; \c adventureworks`
4. **Back in the first tab**, import the SQL dump into the new database: `bash gcloud sql import sql postgres gs://[YOUR BUCKET]/dump.sql \ --database=adventureworks`

Step 3: Setting Up CDC (Change Data Capture)

1. In the second tab, execute the following SQL commands to set up CDC: "'sql CREATE USER datastream WITH REPLICATION LOGIN PASSWORD 'datastream'; CREATE PUBLICATION psqlrepl FOR ALL TABLES;

   ALTER USER postgres WITH REPLICATION; ALTER USER datastream WITH REPLICATION;

SELECT PG_CREATE_LOGICAL_REPLICATION_SLOT(‘psqlreplslot’, ‘pgoutput’); “‘

2. **Grant privileges**: The following commands grant the necessary privileges to the ‘datastream’ user for various schemas in the database: “‘sql GRANT SELECT ON ALL TABLES IN SCHEMA person TO datastream; GRANT USAGE ON SCHEMA person TO datastream; ALTER DEFAULT PRIVILEGES IN SCHEMA person GRANT SELECT ON TABLES TO datastream;

   GRANT SELECT ON ALL TABLES IN SCHEMA production TO datastream; GRANT USAGE ON SCHEMA production TO datastream; ALTER DEFAULT PRIVILEGES IN SCHEMA production GRANT SELECT ON TABLES TO datastream;

   GRANT SELECT ON ALL TABLES IN SCHEMA sales TO datastream; GRANT USAGE ON SCHEMA sales TO datastream; ALTER DEFAULT PRIVILEGES IN SCHEMA sales GRANT SELECT ON TABLES TO datastream;

   GRANT SELECT ON ALL TABLES IN SCHEMA humanresources TO datastream; GRANT USAGE ON SCHEMA humanresources TO datastream; ALTER DEFAULT PRIVILEGES IN SCHEMA humanresources GRANT SELECT ON TABLES TO datastream;

   GRANT SELECT ON ALL TABLES IN SCHEMA purchasing TO datastream; GRANT USAGE ON SCHEMA purchasing TO datastream; ALTER DEFAULT PRIVILEGES IN SCHEMA purchasing GRANT SELECT ON TABLES TO datastream; “‘