Alternative Cloud-Native Data Platform Architecture on Google Cloud Platform (GCP)

This guide details the implementation of a modern cloud-native data platform architecture on **Google Cloud Platform (GCP)**, leveraging services such as **BigQuery**, **Snowflake**, **Databricks**, and other GCP-native components to enable seamless data processing, storage, transformation, and analytics capabilities.

Table of Contents

1. Core Architecture Overview
2. Recommended Architecture Components
3. Step-by-Step Implementation Guide
4. Benefits of the Architecture

---

1. Core Architecture Overview

This architecture leverages GCP's fully managed services to:

- Enable **real-time data processing** and **advanced analytics**.
- Minimize operational overhead by utilizing managed cloud services.
- Incorporate **Snowflake** as the data warehouse and **Databricks** for data engineering and machine learning, allowing flexibility for both structured and unstructured data.

---

2. Recommended Architecture Components

2.1 OLTP Database (Transactional Layer)

- **Cloud SQL**: Use Google Cloud SQL to handle transactional workloads, offering scalability and high availability.

2.2 NoSQL Database (Catalog Data)

- **Firestore**: Store catalog and metadata in Firestore, which supports serverless, scalable, and low-latency operations.

2.3 Streaming and Real-Time Data Ingestion

- **Pub/Sub**: Capture and process real-time streaming data with Google Cloud Pub/Sub.

2.4 Data Lake Storage

- **Cloud Storage**: Use Google Cloud Storage as the data lake to store raw and processed data, accessible by both Snowflake and Databricks.

2.5 Data Warehouse

- **Snowflake on GCP**: Use Snowflake for scalable data warehousing, optimized for analytical queries with seamless integration to GCP services.

2.6 Data Processing and Machine Learning

- **Databricks on GCP**: Use Databricks for scalable data processing, advanced analytics, and machine learning. Leverage Delta Lake for lakehouse capabilities.

2.7 Data Orchestration

- **Cloud Composer**: Use Cloud Composer (Apache Airflow on GCP) for orchestrating ETL processes and data workflows.

2.8 Business Intelligence

- **Looker**: Use Looker for business intelligence and reporting, directly connecting it to Snowflake and Databricks for real-time analytics.

---

3. Step-by-Step Implementation Guide

Module 1: Setting Up the Transactional Database and Real-Time Ingestion

1. **Set Up Google Cloud SQL for OLTP**:
    - Create a **Cloud SQL** instance to manage transactional data.
    - Configure the instance with necessary network and access settings.
2. **Enable Change Data Capture (CDC)**:
    - Use **Dataflow** to capture and stream real-time changes from Cloud SQL to Pub/Sub.
3. **Set Up Google Cloud Pub/Sub for Streaming**:
    - Create **Pub/Sub topics** and subscriptions to ingest and process real-time streaming data from Cloud SQL.

Module 2: NoSQL Catalog Database Setup

1. **Set Up Firestore for Metadata Storage**:
    - Use **Firestore** as a serverless NoSQL database to store product catalog and metadata.
    - Configure Firestore with appropriate security and indexing options.
2. **Sync Firestore with the Data Lake**:
    - Use **Dataflow** or **Firestore export** to periodically sync data from Firestore to **Cloud Storage** for historical analysis and integration with the data lake.

Module 3: Data Lake and Data Warehouse Configuration

1. **Create Cloud Storage Buckets for the Data Lake**:
    - Set up **Google Cloud Storage** buckets to organize and store raw, processed, and curated data.
2. **Set Up Snowflake on GCP**:

- Provision a **Snowflake** account on GCP and connect it to **Cloud Storage** for seamless data access.
- Use **Snowflake's External Tables** feature to directly query data in Cloud Storage without loading it into Snowflake.

3. **Establish Lakehouse Capabilities with Delta Lake**:
    - Use **Databricks Delta Lake** on GCP to provide a unified data layer with ACID-compliant transactions.

Module 4: Data Transformation and Processing

1. **Data Ingestion and Transformation with Databricks**:
    - Set up **Databricks on GCP** for scalable data engineering and machine learning workflows.
    - Use **Delta Lake** to transform raw data into structured formats and store it back in Cloud Storage.
2. **Process and Enrich Data in Snowflake**:
    - Use **Snowflake** for transformations on structured data, leveraging its SQL capabilities.
3. **Set Up Batch ETL Pipelines with Dataflow**:
    - Use **Dataflow** for batch ETL processes, including data cleansing and joining datasets, before loading them into Snowflake.

Module 5: Orchestration and Workflow Management

1. **Orchestrate ETL Pipelines with Cloud Composer**:
    - Use **Cloud Composer** to automate and schedule data workflows across GCP services.
    - Set up DAGs (Directed Acyclic Graphs) to connect services like Cloud SQL, Firestore, Pub/Sub, and Cloud Storage.
2. **Use Databricks Notebooks for Real-Time Processing**:
    - Develop **Databricks Notebooks** for real-time transformations and analytics on event-driven data from Pub/Sub.

Module 6: Machine Learning Model Deployment

1. **Set Up MLflow for Model Experimentation and Management**:
    - Use **MLflow** within Databricks to track and manage machine learning models.
2. **Deploy Models with AI Platform**:
    - Register and deploy models to **Google Cloud AI Platform** for inference.
    - Connect these services to Snowflake and Looker for real-time inference and analytics.

Module 7: Business Intelligence and Visualization

1. **Create Looker Dashboards for Reporting**:
    - Connect **Looker** to Snowflake for interactive data visualizations and reports.
    - Set up data models in Looker to enable efficient and performant reporting on Snowflake data.

2. **Enable Real-Time Reporting with Looker**:
    - Use Looker's **Explore** functionality to allow real-time analysis on data in Snowflake.

---

4. Benefits of the Architecture

- **Scalability and Flexibility**: GCP's managed services dynamically scale with workloads, allowing cost optimization for varying demand.
- **Real-Time Processing Capabilities**: Pub/Sub and Databricks Delta Lake support real-time data ingestion and transformation for near real-time analytics.
- **Operational Efficiency**: Managed services like Cloud Composer, Dataflow, and Databricks reduce the need for infrastructure management, allowing teams to focus on analytics and insights.
- **Unified Lakehouse Architecture**: Using Delta Lake and Snowflake provides a unified data layer that supports both structured and unstructured data for comprehensive analytics.
- **Advanced ML and BI Capabilities**: Combining Databricks, AI Platform, and Looker enables a robust platform for advanced machine learning and visualization, enhancing data-driven decision-making.

This architecture offers a scalable, flexible, and robust data platform leveraging the best of GCP, Snowflake, and Databricks for end-to-end data analytics, transformation, and machine learning workflows.