



Google BigQuery Use Case

Use Case: Optimizing Data Analytics with Google BigQuery

GENERAL CHARACTERISTICS

Intent	To enable scalable, real-time data analytics for a client using Google BigQuery.
Scope	Implementation of a cloud-based data warehouse solution to handle large-scale datasets efficiently.
Level	System-level.
Client	Confidential (Retail Chain with Global Operations).
Last Update	03/12/2024
Status	Finalized.
Stage	Implementation and Analysis.

ACTORS

Primary Actor	Data Analyst.
Secondary Actors	Data Engineers, Business Intelligence Teams, Decision Makers.

PREREQUISITES

Static Preconditions	<ul style="list-style-type: none"> - Google Cloud Project set up with BigQuery API enabled. - Data sources identified and prepared for ingestion.
Dynamic Preconditions	<ul style="list-style-type: none"> - ETL pipelines ready for data extraction and transformation. - Access roles configured for secure data handling.
Assumptions	<ul style="list-style-type: none"> - Client has datasets exceeding terabytes in size. - Real-time analytics is critical for decision-making.

TRIGGERS

Trigger Event	The client faced delays and inefficiencies in processing large datasets, affecting business-critical decision-making.
---------------	---

EXPECTED OUTCOME

Success Postcondition	<ul style="list-style-type: none"> - Queries execute within seconds, even for terabyte-scale datasets. - Analytics dashboards update in near real-time.
-----------------------	---





Google BigQuery Use Case

Failed Postcondition	- Analytics delays persist, and insights are not actionable in time.
----------------------	--

OPERATIONS AND CONCEPTS

Operations	<ol style="list-style-type: none"> 1. Migrated the client's data to Google BigQuery. 2. Set up ETL pipelines using Cloud Dataflow for seamless data ingestion. 3. Designed partitioned and clustered tables to optimize query performance. 4. Implemented scheduled queries for automated reporting. 5. Integrated BigQuery with BI tools like Looker for real-time dashboards. 6. Configured access controls to ensure data security and compliance.
Concepts	<ul style="list-style-type: none"> - BigQuery: A serverless, highly scalable data warehouse. - Partitioning and Clustering: Techniques to improve query performance. - ETL Pipelines: Automate data transformation and ingestion processes.

MAIN SUCCESS SCENARIO

Step 1	Analyzed client requirements and identified inefficiencies in their existing analytics system.
Step 2	Migrated legacy data warehouse to Google BigQuery for better scalability.
Step 3	Designed efficient table schemas with partitioning and clustering for optimized queries.
Step 4	Set up ETL pipelines to ingest data from multiple sources in real-time.
Step 5	Integrated BI tools like Looker for creating interactive dashboards.
Step 6	Implemented access controls to ensure data security and compliance.
Step 7	Monitored system performance and optimized queries to maintain sub-second response times.

