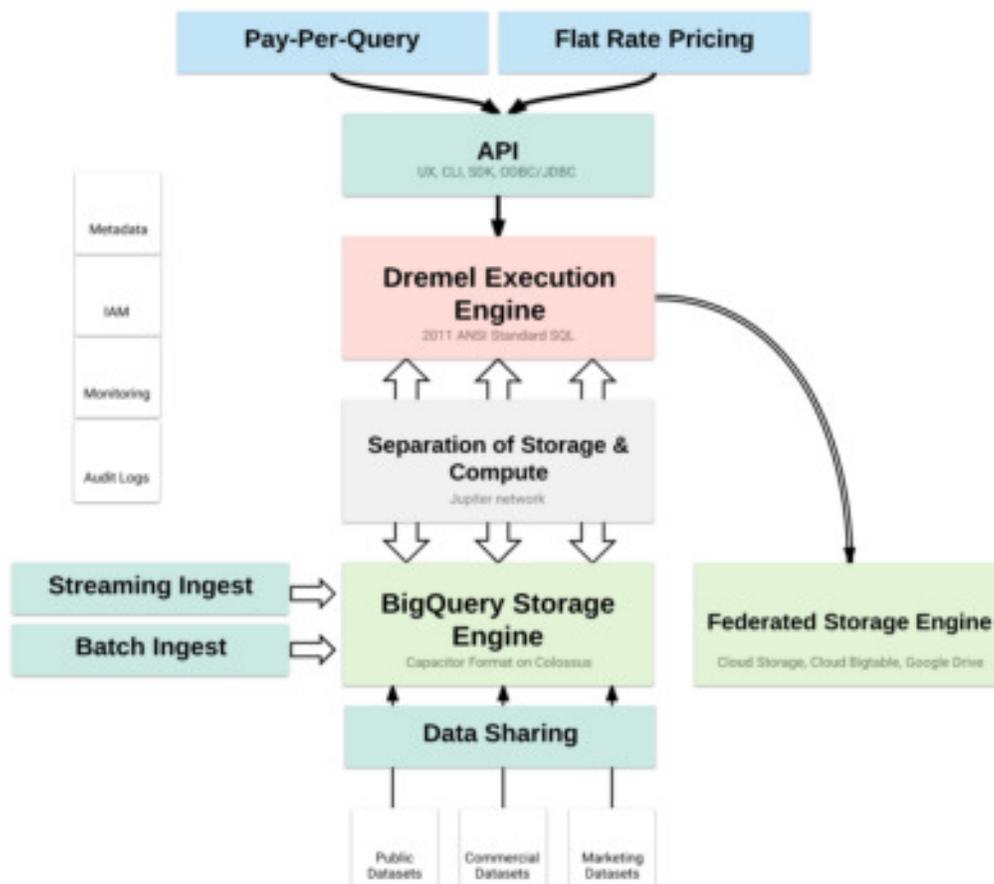


# **Migrating Teradata to BigQuery**

## **Activities and Stages**

- With increasing volume, variety and velocity of data, maintaining traditional data warehouses can be expensive and complex. Moreover, sorting, filtering the vital data for analytics becomes a mammoth task.
- Google BigQuery is a cloud enterprise data warehouse. It uses Google's advanced storage and the processing power of the Google's infrastructure to enable fast SQL queries and deliver scalable database solutions for clients. This serverless data warehouse enables real time data analysis. Migration of your complete database means moving the pivot of your business intelligence and entire data analytics to a new environment. The migration can be a lengthy and complex task. Transferring to BigQuery means going serverless and NoOps which enables clients to start using BigQuery for storage and processing of data without thinking of security of the warehouse, disk configuration and load balancing etc.
- Google BigQuery Components



# Migrating Teradata to Big Query is an attractive option because

- It enables super-fast analytics for petabytes of data
- It allows you to shift from CAPEX Model to OPEX model thereby reducing costs
- It doesn't require provisioning for storage and computing resources in advance
- It allows streaming ingestion of huge volumes of data
- Owing to column wise data store in BigQuery, high levels of compression of data is obtained
- BigQuery has integrations with various BI tools like Tableau, Looker and Microstrategy
- Elimination of traditional ETL process and tools



# PRE-MIGRATION

## PRE-MIGRATION CONSIDERATIONS

Three important functions that needs to be considered before migrating the Teradata database

### Functional differences owing to migration

While migrating, there may be differences in functionality. Big Query has different architecture from Teradata and may not be able to use the traditional star scheme. Not to worry!

BigQuery offer native support as extensions that enables querying nested or repeated data for standard SQL. It also allows for direct querying of data from Google Cloud data store or Google drives. It supports JSON, NL, CSV and AVRO files.

Federated data sources are helpful in such cases

- You can refine, load and clean the data in one go by data querying from a federated data source (a location outside BigQuery). Then you can write the cleaned data into BigQuery storage
- You can also couple small amounts of changing data with other tables. Owing to federated data source, the data is not required to be uploaded every time it is updated

# Usage and billing

Your usage and usage performance reflects on the productivity. It is important to determine whether the solution will allow you to scale with data volume and the complexities of queries solved. BigQuery employs the concept of slots to execute SQL queries. A slot is simply a virtual CPU and RAM or an unit of analytical computing. BigQuery automatically determines the number of slots for each query depending on the query size and its complexity. By default, 2000 slots are afforded for projects but if the consumption goes more than that, the allocation limit is automatically raised. But if your SLA demands more number of slots or reserved slots flat rate billing model would be better.

( Refer: [On-Demand pricing model and flat rate pricing model for slots](#) )

Integrating Google Stackdriver with BigQuery helps you monitor slot availability as well as allocations.

- Big Query services are billed on the amount of data stored and processed.
- Storage and processing are billed separately and there are pricing tiers for storage. This eliminates the need for planning for volume discounts
- The processing cost is based only on the data scanned. So , if we can identify the least number of columns required to produce results, we can reduce the processing costs.
- An extra charge is levied if API is used for streaming ingestion into BigQuery in real time.
- Various cost control tools like billing alerts or custom quotas can be used for mapping the resource usage cost per project

# Data Transformations for migration

Certain data transformations may be required for source data to gain full compatibility with target schema. For this you can create a views database of Teradata containing all views and macros needed. Create one view each table for migration. The views database enables transformation of data types so that they can be compatible with big query data schema. The data then needs to be loaded on temporary staging tables of Bigquery where it is parsed using javascript UDFs or its inbuilt functions. Then the data needs to be loaded into the target schema

## Data Ingestion

Knowing how to make the data ingestion work in the new environment is an essential part of Teradata migration to BigQuery. There are multiple options of loading data to BigQuery.

- Using the BigQuery API
- Loading from JSON/CSV and AVRO on GCS
- Using API Client libraries for streaming data into BigQuery
- Using Google Cloud dataflow

# Data Transfer Modes

There are three options for setting up data transfer from Terradata to Big Query

- Extraction Method: JDBC coupled with TPT (Teradata Parallel Transporter) or Fast Connect- In this mode a table is extracted to a specific location of local files into AVRO files. Extracted files are loaded on the storage bucket, and after transfer deleted from the local files
- Automatic Schema Conversion- The BigQuery data transfer services enable automatic schema detection and conversion of data during transfer
- On -Demand or Incremental Transfers- You have the option for snapshot transfer, recurring or periodic data transfer from Terradata database instance using BigQuery Transfer Service

# PRE-MIGRATION ACTIVITIES

- **PREPARE AND DISCOVER**– At this stage you and your stakeholders discover the use cases for migration of terradata to BigQuery and analyse the expected benefits from migration. The benefits can range anything from reducing the total cost of ownership, productivity gains, reducing data usage cost etc.

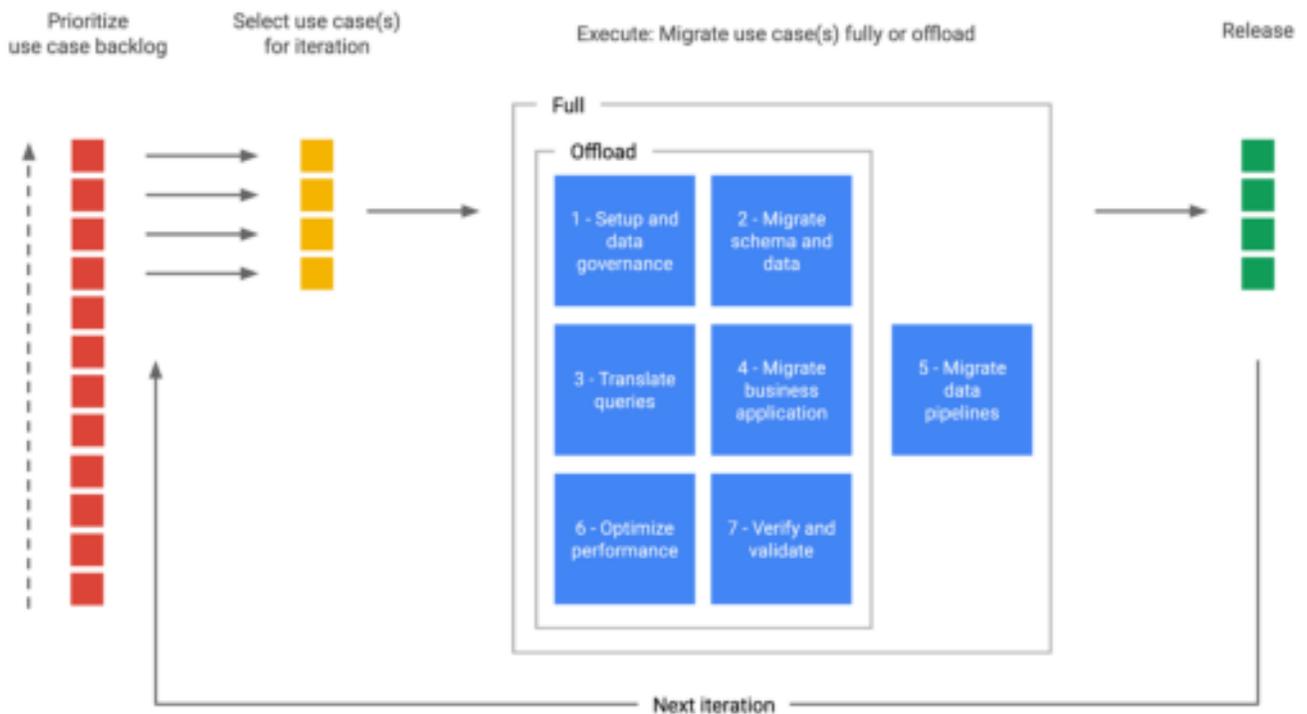
The stakeholders need to consider the following questions at this stage

- **ACCESS AND PLAN**– At this stage, you assess the state of your existent infrastructure, prioritize the use cases, design the proof of concept, choose your migration partner, do a cost estimation, set your internal metrics of success of data performance on BigQuery. Here the input from the discovery phase is taken into consideration for planning the migration

The planners need to consider the following questions at this stage

- How to ensure that the data loaded in the new environment is identical to premise data?
- How to manage disruptions while switching over to BigQuery?
- What should be the architecture of the BigQuery data pipeline?
- How to integrate reporting and BI tools with BigQuery?
- What would be the usage billing and how to monitor it?
- What would be the licensing and infrastructure requirements?

# MIGRATION PHASE



Here you need to

- Set the configurations
- Download the migration agent
- Enable the required APIs

## **For BigQuery data transfer service, you need to enable**

gcloud services enable bigquery-json.googleapis.com

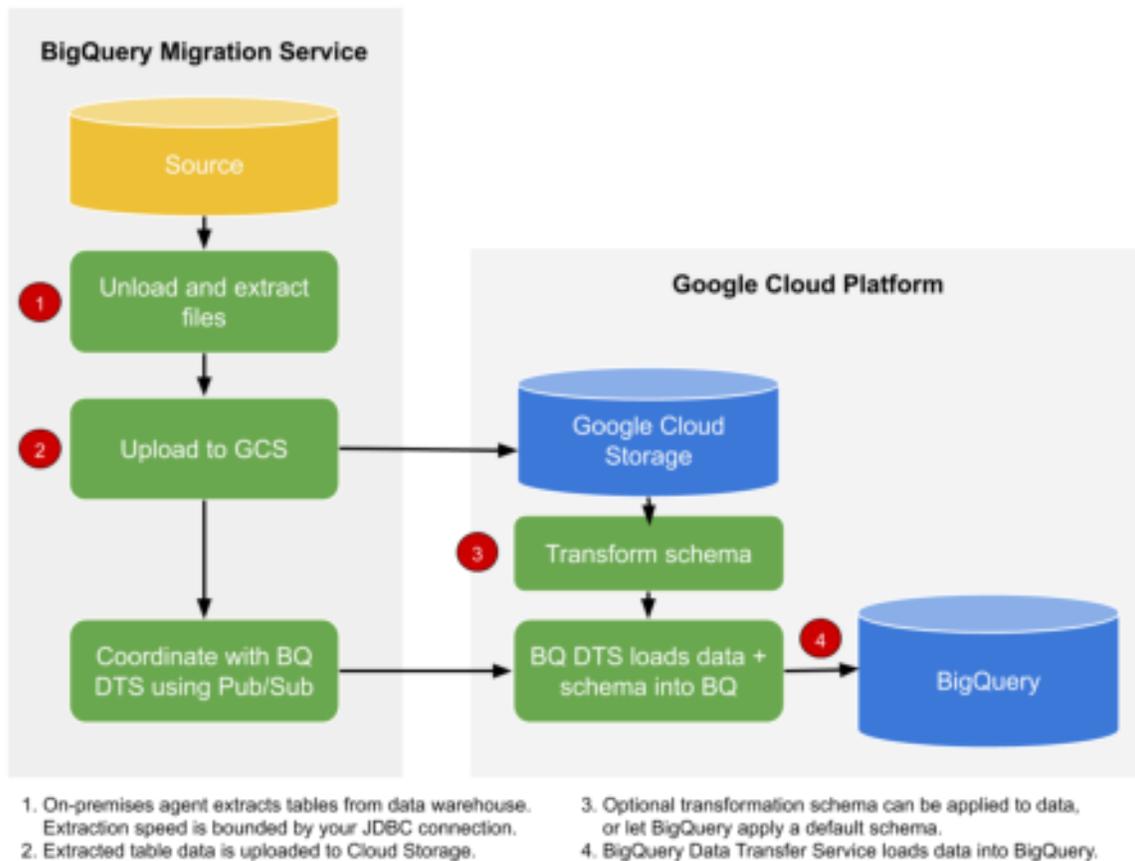
gcloud services enable storage-api.googleapis.com

gcloud services enable storage-component.googleapis.com

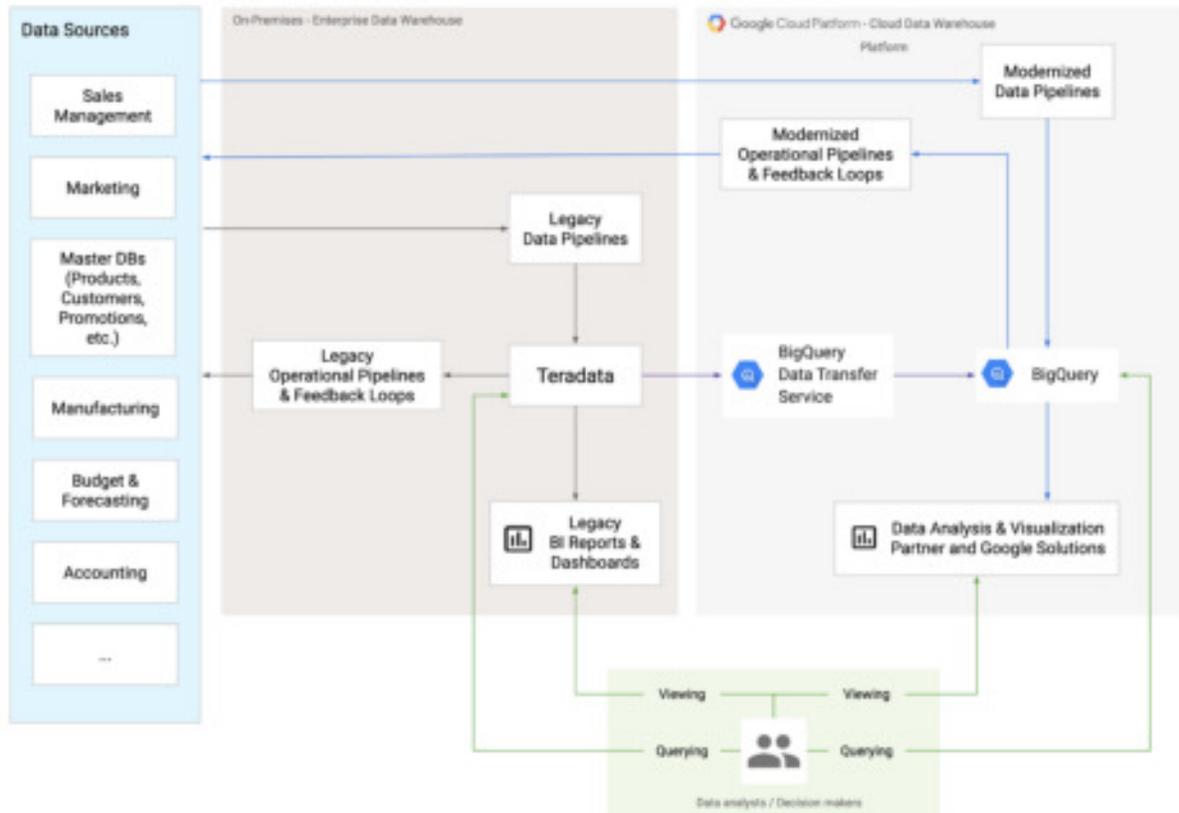
gcloud services enable pubsub.googleapis.com

gcloud services enable bigquerydatatransfer.googleapis.com

- Create the big query datasets,
- Fulfill on-premise requirements (local machine requirements, Teradata connection details)
- Create a service account for Big Query data transfer service
- Bid the service account to storage admin/ Big Query admin/Pub-Sub admin role
- Accept licensing for downloading the Teradata Express VM image, Teradata tools, and JDBC drivers from Teradata's support site.
- Select the modes of data transfer
- Select the appropriate tool for transfer
- Prioritize the data sets, applications and others for migration
- Configure Google Cloud Environment
- Initialize and run the migration agent
- Monitor the Progress of migration



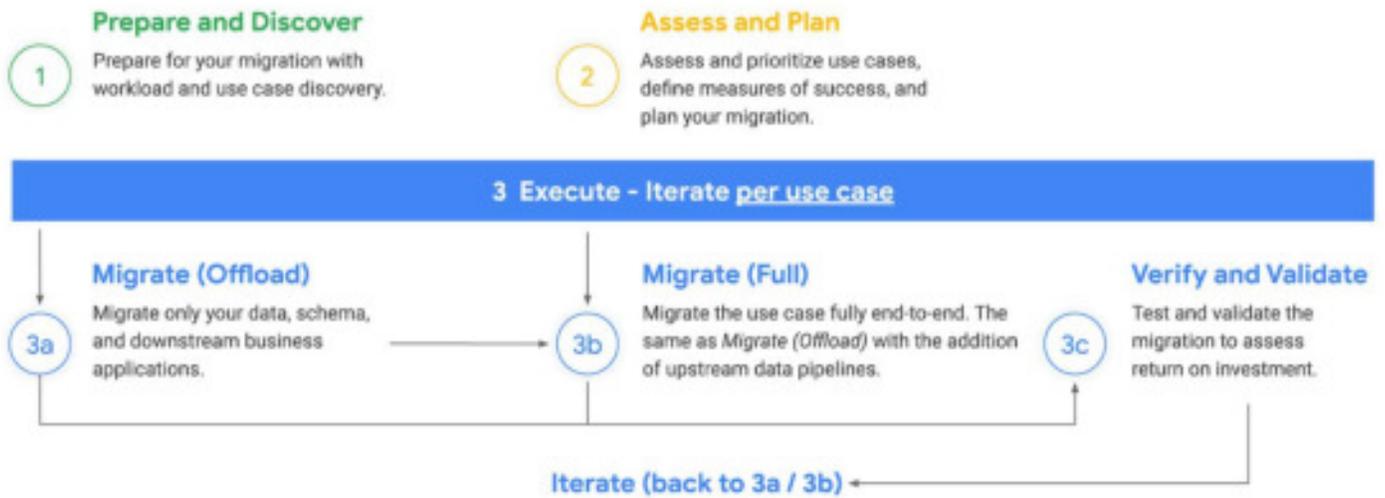
To stop disruption in workflow during the migration, an architecture where both your data warehouse and BigQuery can be used simultaneously, where both can ingest data, provide access to users and run the applications has been shown below.



## POST MIGRATION

Testing and validating data so as to ensure consistency of data and to ensure that the applications are providing expected performance.

# The complete framework of migration



## About us

Ismile Technologies is a global technology services company that helps businesses compete by adopting disruptive technologies such as advanced analytics, big data, cloud, databases, DevOps, and infrastructure management to advance innovation and increase agility. Specializing in designing, implementing, and managing systems that directly contribute to revenue growth and business success, Ismile Technologies's highly skilled technical teams work as an integrated extension of our clients' organizations to deliver solutions that enable the strategic use of data, accelerate software delivery, and ensure reliable, scalable IT systems.

Connect With Us  
service@iSmileTechnologies.com  
(732) 347-6245



<https://www.facebook.com/ismileTechnologies>

<https://www.instagram.com/ismiletechno/>

<https://www.linkedin.com/company/ismiletechnologies/>

<https://www.youtube.com/channel/UCAIYEyaJeOeCk-wVBDY4-0w>