# High level solution design for Google Analytic Like Backend System

## High level Architecture

A screen shot of a computer

Description automatically generated

To ensure this system’s high availability, it has to be hosted in a cloud architecture in diversified geo locations.

## Backend System Design

### Analytic Server and Engine

### System Diagram

Data Collection API

POST API Call

Analytic Engine

Messaging Service

Data Serving API

GET API Call

Shared Network Cache

### Details

This Service will be responsible for serving two APIs –

* POST API- This will collect customers Ad related data, authorize and validate them.

API Header: Authorization Details of the customer

POST Body: Information about served ad

* GET API – Analytics UI can call this API to get information about each customer’s ads. API Header: Authorization Details of the customer

Query Params: Information needed by the UI to create graphs

For, POST API calls- server will do the authorization check and pass the data to engine via another asynchronous POST API call. Analytic Engine is responsible to distribute the data in various buckets based on ad types and prepare as well as publish messages to asynchronous messaging system.

For GET API calls, analytics server will do authentication check and then do a GET API call to the shared network cache to retrieve data required data as per the query parameter combinations.

Purpose of this service is to so serve billions of POST as well as GET calls, making this service asynchronous will ensure freeing up server threads as soon as possible. Since DB read and write are always time consuming, we have to make sure that Analytic Server is delegating that task to another service asynchronously. During the GET call, fetching data from cache will make sure API latency to be minimal and data to be accurate all across the network.

### Analytics Daemon

### System Diagram

Data Retrieval Architecture

Master DATABASE

Messaging Service

Daemon

Data Serving Architecture

READ Only DATABASE

Shared Network Cache

Daemon

### Details

Analytics Daemon acts as an asynchronous messaging service consumer. It is responsible to write data to the master DB as well as fetch the data from DB and save it in the cache for faster retrieval of data by GET API.

Since, we are saving raw data in the database, this will make sure that data is re-processible in case of any error in the retrieving logic.

Also, this daemon will be responsible for refreshing the cache every few minutes, so that GET API call can serve as recent data as possible. Refreshing logic can be modified based on the weightage of the data.

* 1. **Reprocessing Engine**
     1. System Diagram

Data Reprocessing Architecture

Analytic Server

Reprocessing Engine

Shared Network Cache

DATABASE

* + 1. Details

A fall back mechanism to retrieve data from database, in case of unavailability of data from time series-based cache. This data can be saved in cache as well, for easy re-retrieval.