Build an End-to-End Data Pipeline with AWS and Big Query
Advanced Project Series (5 Projects)

Main Prerequisites: Basic Python or Node.js (Working with strings, Working with lists, Using loops, Using functions, Understanding why the codes works (or doesn't) Basic knowledge of SQL (Understanding SELECT * FROM WHERE GROUP BY; Understanding of CTEs and temporary tables), Basic statistics and regression analysis.

Key Skills learned: Serverless architecture, Google Cloud functions, AWS Lambda functions, ETL/ELT data processing, Data definition and data manipulation with BigQuery Standard SQL, BigQuery ML linear modeling and logistic regression for churn prediction, Data Visualisation with Data Studio, Architecture as code.

Author: Mike Shakhomirov

1. Tell us about yourself

Mike Shakhomirov

Lead Data Engineer at The World's Online Festival. MBA. Google Cloud Certified Professional Data Engineer. MIT diploma in Big Data and Social Analytics.

https://www.linkedin.com/in/mshakhomirov/

Passionate and digitally focussed individual with an abundance of drive and enthusiasm, loving the challenges the full mix of digital marketing can offer.

I am an official writer for such publications as Towards Data Science and The Startup with more than 20 published articles on various topics. I write about Data Engineering, Machine learning and AI in Digital Marketing.

Read more: https://www.medium.com/@mshakhomirov/

I established over fifteen years of experience in data analytics, corporate banking risk units and digital marketing and obtained what I believe to be considerable expertise in risk management, data engineering and mathematical modelling, statistical analysis, business administration and marketing.

After having completed my MBA at Newcastle I worked with numerous start up companies in the UK in building BI systems, designing data pipelines, machine learning models and implementing genetic algorithms. These roles allowed me to apply combined marketing and data handling skills and I feel that a career in data driven marketing or computer science and artificial intelligence is definitely something I would like to pursue. These are engaging and interesting fields that allow the practical application of data science in an extremely rewarding way, as well as room for continuing professional development, innovation, implementing novel ideas and contributing to a constantly evolving field.

2. Give a brief description of each project

The idea of this Series of projects is that they teach what can be done with the data warehouse itself as a central part of the architecture diagram. Architecture is built around BigQuery as a main data storage.

Projects explain how to:

1. create data extraction pipelines,

2. data cleansing, aggregation and enrichment pipelines, 3. how to visualise

your data and build the reports (BI pipelines),

4. how to build and productionalise ML models.

However, each project can be completed separately disregarding any other projects in the series. I think it's an advantage.

The series itself provides a real-life example where architecture is hybrid (both AWS and GCP platforms are used) and explains how two programming language (Python and Node.js) can be used to alter it when each LP explains only one possible way of achieving the objectives.

In this tutorial you will learn how to create all layers (image below) of your modern data stack which is a combination of the best practices and right tools to develop a maintainable and flexible data intrastate that is easy to work with and allows a wide variety of users access to data. And it scales as your data grows!

Modern data stack tools (not a complete list of course):

* Ingestion: **Fivetran, Stitch**

* Warehousing: Snowflake, Bigquery, Redshift

* Transformation: dbt, Dataflow, APIs.

* BI: Looker, Mode, Periscope, Chartio, Metabase, Redash

Project 1: Extract data from an API

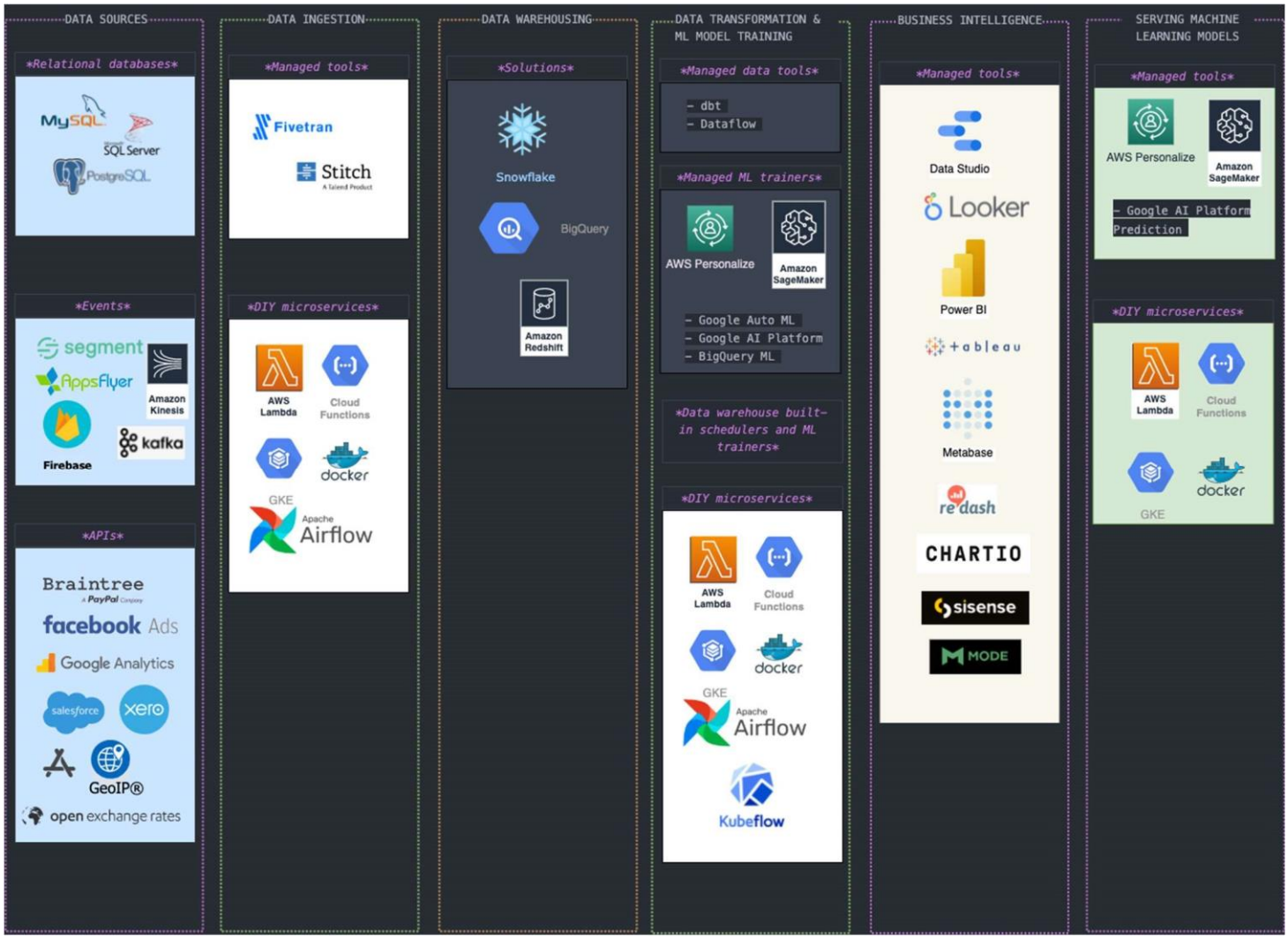
Project 2: Build an ingestion pipeline

Project 3: Build a transformation pipeline with Dataflow

Project 4: Create a BI report with Data Studio

Project 5: Create resources with Cloudformation

Modern data stack (image)



Project 2b. Building a Batch data pipeline from AWS S3 into BigQuery using Lambda Functions (Advanced).

This tutorial demonstrates how to stream and perform batch operations of new objects from a AWS S3 storage bucket into BigQuery by using AWS Lambda Functions. AWS Lambda is a serverless compute service that lets you run code without provisioning or managing servers, creating workload-aware cluster scaling logic, maintaining event integrations, or managing runtimes. With Lambda, you can run code for virtually any type of application or backend service - all with zero administration.

3. Who the project is for and what they will learn?

See Section 4. which addresses this question for each LP.

4. Projects' Outlines

Project 2b. Streaming and inserting batch data from AWS S3 into BigQuery using Lambda Functions. (Advanced)

(estimated time to complete project: 60-80 hours)

This project is for students who would like to choose a data analyst or developer career path, and data engineers who are tasked to build near real-time analytics on files added to Cloud Storage. The LP assumes you are familiar with Linux, Cloud Storage, Cloudformation and BigQuery.

In this tutorial you will learn how to load streaming and batch data into BigQuery from a different Cloud service provider (AWS). In LP 1 we created a basic data pipeline where files added to Google Cloud Storage bucket were automatically ingested into your BigQuery data warehouse. In reality things might be even more complex and you might be tasked to build a hybrid solution where files are stored in AWS S3 buckets and then uploaded into BigQuery.

Objectives

- Extract financial data from PayPal API (Data Source) ● Create an AWS S3 bucket to store your JSON files.
- Create a BigQuery dataset and table to batch upload your data in to.
- Configure an AWS Lambda Function to trigger whenever files are added to your bucket.
- Use Cloudformation to describe your data pipeline architecture ● Set up SNS topics.
- Configure additional functions to handle function output.
- Test your pipeline.
- Configure AWS Cloudwatch logs Monitoring to alert on any unexpected behaviors.
- Create a BI report (revenue reconciliation) in Google Data Studio The pipeline

  consists of the following steps:

1. JSON files are uploaded to the FILES_SOURCE AWS S3 bucket.
2. This event triggers the bq_batch_import AWS Lambda Function.

3. Data is parsed and inserted into BigQuery.
4. The ingestion status is logged into AWS DynamoDB and AWS Cloudwatch Logs.
5. A message is published in one of the following AWS SNS topics:
    - bq_batch_success_topic
    - bq_batch_error_topic
6. Depending on the results, Lambda Functions moves the JSON file from the FILES_SOURCE bucket to one of the following buckets:
    - FILES_ERROR
    - FILES_SUCCESS
7. Creating the first dataset and first table using the data and SQL query.
8. Creating a scheduled query
9. Creating a view on the data we have just loaded into BigQuery with DataFlow (or dbt)

Final deliverable:

By the end of this tutorial you will create a scalable data pipeline which consists of AWS Lambda function sourcing financial data (PayPal) to upload the files added to AWS S3 Storage bucket and an established process to monitor the loaded files and perform error handling using DynamoDB database and SNS topics.

Creating ELT (Extract - Load - Transform) data pipelines with DataFlow (or dbt). DataFlow is officially a part of Google Cloud.

Recommended read:

[1]
https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/Welcome.html

[2]

https://aws.amazon.com/quickstart/architecture/data-lake-foundation-with-awsservices/

[3]

https://docs.aws.amazon.com/serverless-applicationmodel/latest/developerguide/sam-resource-function.html

[4] https://cloud.google.com/bigquery/docs/reference/libraries

[5]

https://livebook.manning.com/book/designing-cloud-data-platforms/chapter-3/v-8/332

[6]

https://livebook.manning.com/book/google-cloud-platform-in-action/chapter-19/

1. Set extraction pipe (PayPal)
2. Set ingestion pipe (AWS S3 to BigQuery)
3. Set data transformation pipeline (Dataflow)
4. Set BI for revenue reconciliation (Data Studio)
5. Wrap it all up with Cloudformation (Software as a code).