Airline data analysis using Hive, Druid & Quicksight on AWS

Introduction to Big data

- Real time streaming data being captured at regular intervals of time from millions of IOT devices like sensors,clickstreams,logs from the device APIs and historical data from SQL databases.
- To store the huge volumes of data with high velocity and veracity ,we need an efficient scalable storage system which is distributed across different nodes either in local or in cloud.
- Here comes the Hadoop concept which can be classified into two groups -Storage and processing
- Storage will be done in HDFS and processing is done using Map reduce
- Initially the processing of this data used to be done in Java by writing mapreduce programs with controller services. Now a days lot of tools have been developed on top of map reduce such as Hive ,Presto,Druid etc which can be directly used by their built in SQL interfaces and are much faster due to inbuilt indexing of the data.
- Underlying storage will be HDFS for all kinds of big data scenarios and underlying processing is Map reduce in most of the cases.

Introduction to Data Pipeline

- It refers to a system for moving **data** from one system to another. The **data** may or may not be transformed, and it may be processed in real time (or streaming) instead of batches.
- Right from extracting or capturing data using various tools , storing raw data,cleaning, validating data, transforming data into query worthy format ,visualisation of KPIs including Orchestration of the above process is data pipeline.
- Scalability, reliability, security should be taken into consideration in each step of data pipeline building in big data environments.
- Transformed data can be used to derive KPIs and derive initial insights from the data for exploratory analysis
- Further analysis of data using ML algorithms can be used to derive predictions out of the data either historical or near real time .

Business Impact of building data pipelines

- Saves time and effort for actuaries,BI analysts
- Manual process of cleaning, validating of raw data takes on an average of 3-4 hours of time for each and every raw file received at a certain interval of time.
- Automating this simple process saves 3-4 hours for each run.
- Similarly automation of ETLs using Cron,Oozie, Airflow processes etc saves effort and time for Data engineers as well as Analysts.
- Only pipeline monitoring and maintenance needs to be taken care of once the pipeline is built.
- Grabbing the data from source to destination-sources being APIs, Google sheets ,Local systems ,SQL based databases etc. Destination systems can be HDFS for storage , Kafka,Flume,Storme,Logstash to grab real time streaming data

Few source and destination data systems

Source systems **Data capture** **Destination**

systems

- Streaming APIs Kafka HDFS followed
  by Hive
- SQL databases Flume Processing using
  Pyspark
- Google sheets Logstash HiveQL
- Local systems NiFi Elasticsearch

Storme Druid SQL

Python

Visualization tools

Tableau

PowerBI

AWS Quicksight

Kib

Overview

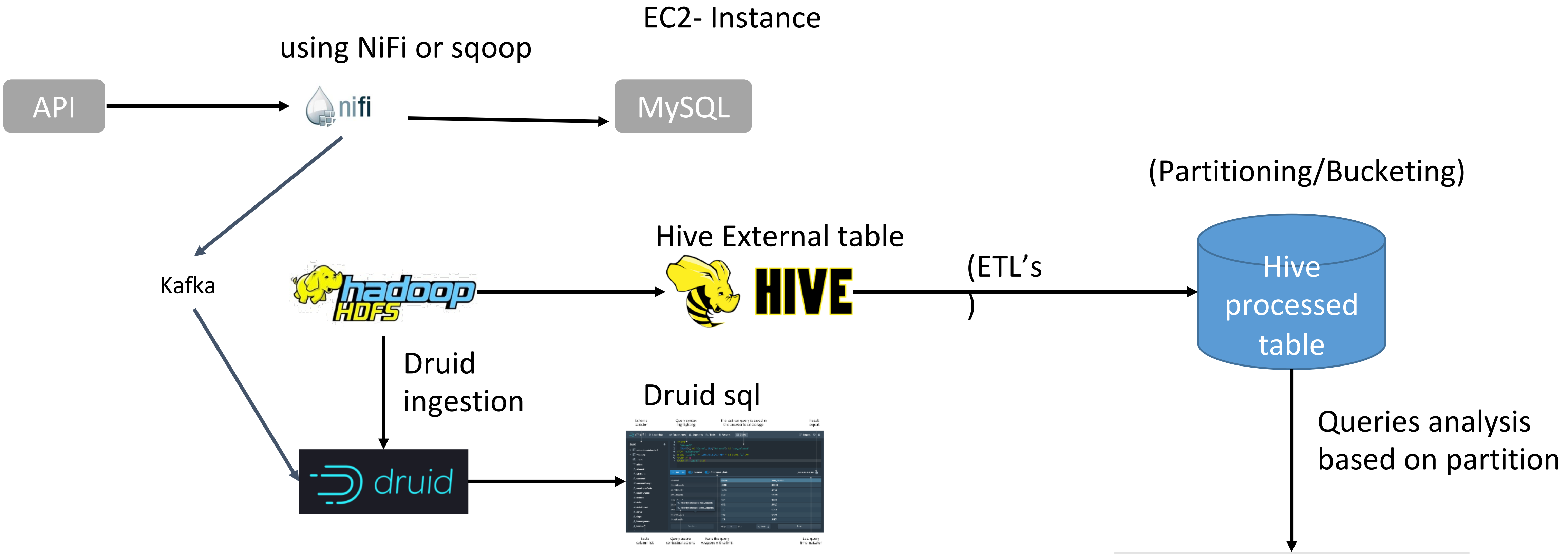
Process to gather streaming data from Airline API using NiFi & batch data using AWS redshift using Sqoop and build a data pipeline to analyse the data using Apache Hive and Druid and compare the performances ,to discuss the hive optimization techniques and visualise the data using AWS Quicksight

- Extract streaming data from APIs at a certain interval of time using NiFi.
- Extract the batch data from RDS using sqoop and store into HDFS .
- Parse the data using Nifi which pushes the data into Kafka topic to send streaming data in turn directly to Druid.
- Extract the data from HDFS into hive and druid to analyse,optimize and compare performance of each of the tools .
- Perform ETLs on Hive table vs ETLs during druid data ingestion.
- Orchestrate the above process using Airflow
- Visualize the KPIs or metrics in AWS Quicksight.

System requirements

- Ec2 instance in AWS - t2.xLarge 16GB of RAM and Ubuntu 16.04 image on it , installation steps attached:
- https://docs.google.com/document/d/1sot7fCIgLZmldpcjodbQ0RZPRwN367-aOpa9d9UJS5M/edit?usp=sharing
- Good internet connectivity.
- Install required services required for Data architecture on Ubuntu machine installed above , installation steps attached:
- https://docs.google.com/document/d/1zPNDN8AUcZ6D4eKa_VFilCgcWwWXtMe2eZHnIbqEJrA/edit?usp=sharing
- Services included in documentation are :
- Hadoop
- Hive
- Zookeeper
- NiFi
- Kafka
- Druid
- AWS Quicksight

Airline data Analysis



EC2- Instance

using NiFi or sqoop

API

MySQL

Kafka

Hive External table

(ETL's)

(Partitioning/Bucketing)

Hive processed table

Druid ingestion

Druid sql

Queries analysis based on partition

- Compare the performance of Druid or HIVE.
- Discuss limitations and opportunities with Druid and Hive.
- Same query run on HIVE vs Druid

Visualization of data
Using Quicksight.

Apache NiFi

- Open source for automating and managing data flow between systems.
- Process distributed data and executes on the JVM on host OS.
- Web based UI for creating, monitoring & controlling data flows.
- Supports buffering of all Queued data.
- Processors connect to many source and destination systems.
- Easy to troubleshoot and flow optimisation.
- Role based authentication
- Build user defined processors and controller services.
- Easy of use- need not code much.

Apache Kafka

- Apache Kafka is an open-source stream-processing software which buffers the records as they occur without dat
- Uses I/O efficiently by batching and compressing records
- Supports streaming data and is scalable, durable and fault tolerant
- Uses file system for caching and storage
- Works on publish and subscribe mechanism
- Producers are source systems , consumers are destination systems
- Messages stored in topic
- Why Kafka over Apache Flume or Apache Storm?

Kafka Vs Flume

Kafka	Flume
Used as a bridge between source and destination systems	Mostly used to gather data into Hadoop from various sources
No data loss and more general purpose system can be used on any data	No data loss but mostly used for log based aggregations and log data
Multiple publishers and subscribers can share same topic	It's a tool to gather data into HDFS
Makes data available even after single point failure	Depends on the configuration

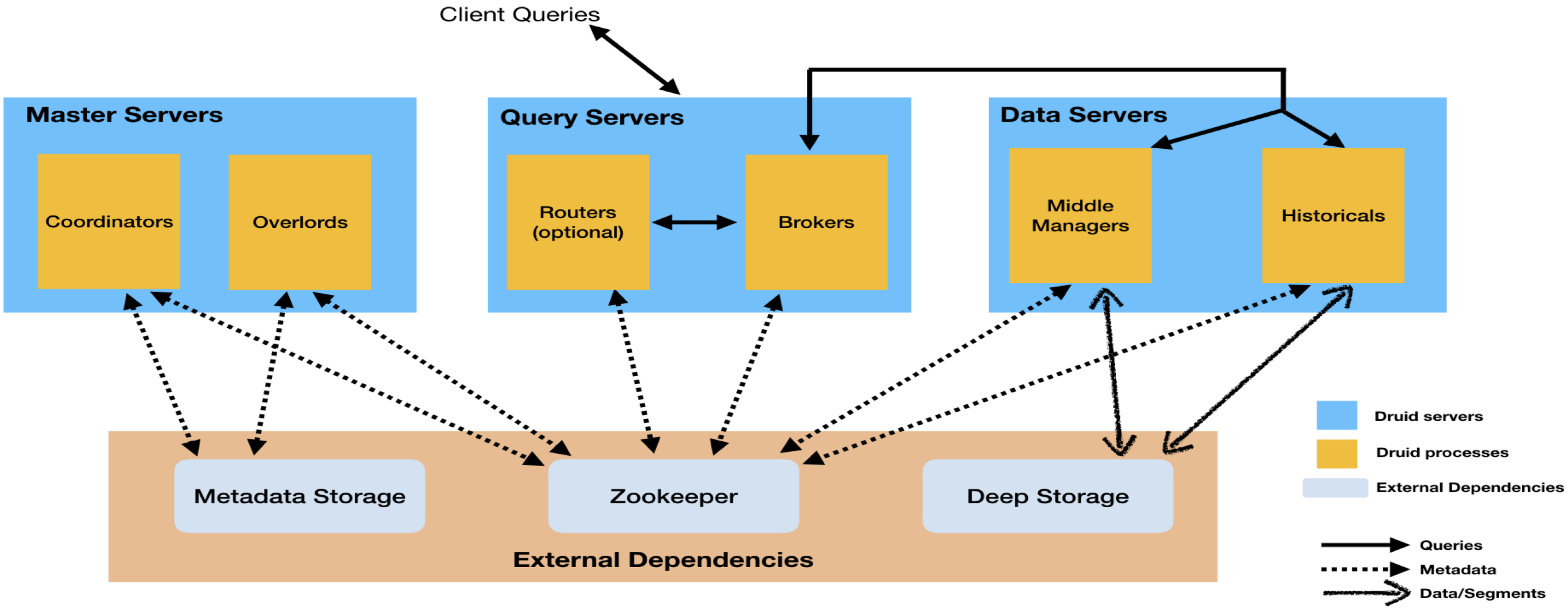
Hive optimization

1. Tez execution engine
2. partitioning
3. bucketing
4. Using suitable file format
5. Vectorization
6. Cost based optimisation
7. Using compression techniques

Challenges with Hive

- Long time to aggregate data in the warehouse using Hive or Presto at Query time.
- Once the velocity of the data increases , infrastructure to hold the data should be scalable and increased which results in more costs.
- But can be useful for complex calculations -So many architectures involve Hive for data computation and Presto/Druid for Querying the computed data.

Druid architecture



Hive Vs Presto Vs Druid

Hive	Presto	Druid
Data warehousing package for Hadoop	Distributed SQL engine which doesn’t store or hold any data	Druid is a column-oriented, open-source, distributed data store written in Java
Optimized for query throughput	Optimized for query latency	Optimized for very high query latency
All computations are performed as map reduce operations which takes more time	Computations are performed in-memory and doesn’t use map reduce	Architecture of separating data storage and caching temporarily makes Druid very fast querying engine.
Aggregations cannot be performed during data ingestion.	Aggregations are performed only during query time -but much faster than Hive	Ingest time aggregations are supported. Hence do not take much time to query aggregated data.