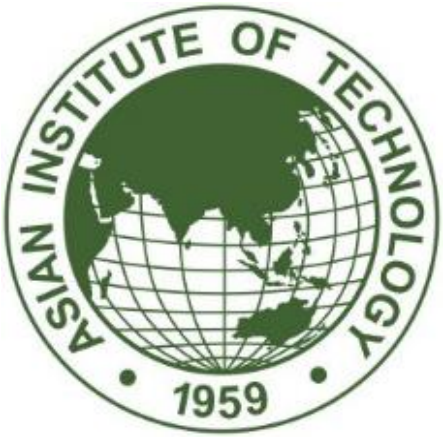
Asian Institute of Technology
School of Engineering and Technology
AT70.19: Software Development and Quality Improvement



"AIR QUALITY (PM2.5) MONITORING SYSTEM"
(IOT-BASED PLATFORM)

Big Data Tool with Apache Kylin

Submitted by:

Smrity Baral (st121662)

Shubhangini Gontia (st121473)

Suyogya Ratna Tamrakar (st121334)

Younten Tshering (st121775)

Submitted to:

Dr. Apichon Witayangkurn

Submitted Date:

7th April, 2021

Hadoop and Apache Kylin

Apache Kylin is an open-source distributed analytics engine designed to provide a SQL interface and multi-dimensional analysis on Hadoop supporting extremely large datasets. It was originally developed by eBay and is now a project of the Apache.

In addition, it easily integrates with BI tools via ODBC driver, JDBC driver, and REST API. It was created by eBay in 2014, graduated to Top Level Project of Apache Software Foundation just one year later, in 2015 and won the Best Open-Source Big Data Tool in 2015 as well as in 2016.

Currently, it is being used by thousands of companies worldwide as their critical analytics application for Big Data. While other OLAP engines struggle with the data volume, Kylin enables query responses in the milliseconds. It provides sub-second level query latency over datasets scaling to petabytes. It gets its amazing speed by precomputing the various dimensional combinations and the measure aggregates via Hive queries and populating HBase with the results.

Apache Kylin™ lets you query billions of rows at sub-second latency in 3 steps.

1. Identify a Star/Snowflake Schema on Hadoop.
2. Build Cube from the identified tables.
3. Query using ANSI-SQL and get results in sub-second, via ODBC, JDBC or RESTful API.

Big Data Tool with Apache Kylin

Now, before moving on to Apache Kylin, let us start the discussion with Big Data, that led to the development of Hadoop and then to Apache Kylin.

IoT connects our physical device to the internet and makes it smarter. Nowadays, we have smart air conditioners, televisions etc. Our smart air conditioner constantly monitors our room temperature along with the outside temperature and accordingly decides what should be the temperature of the room. Now imagine how much data would be generated in a year by smart air conditioner installed in tens & thousands of houses. By this we can understand how IoT is contributing a major share to Big Data.

- **Why Big Data is a problem statement and how Hadoop solves it.**

There were three major challenges with Big Data:

1. **The first problem is storing the huge amount of data**. Storing huge data in a traditional system is not possible. The reason is obvious, the storage will be limited to one system and the data is increasing at a tremendous rate.

2. **The second problem is storing heterogeneous data**. The data is not only huge, but it is also present in various formats i.e. unstructured, semi-structured and structured. So, we need to make sure that we have a system to store different types of data that is generated from various sources.
3. **Finally, the third problem, which is the processing speed**. Now the time taken to process this huge amount of data is quite high as the data to be processed is too large.

To solve the storage issue and processing issue:

- Two core components were created in Hadoop — **HDFS(**Hadoop Distributed File System**) and YARN(**Yet Another Resource Negotiator**)**. HDFS solves the storage issue as it stores the data in a distributed fashion and is easily scalable. And YARN solves the processing issue by reducing the processing time drastically.

While setting up a Hadoop cluster, we have an option of choosing a lot of services as part of your Hadoop platform, but there are two services which are always mandatory for setting up Hadoop. One is HDFS (storage) and the other is YARN (processing). HDFS stands for Hadoop Distributed File System, which is a scalable storage unit of Hadoop whereas YARN is used to process the data i.e. stored in the HDFS in **a distributed and parallel fashion.**

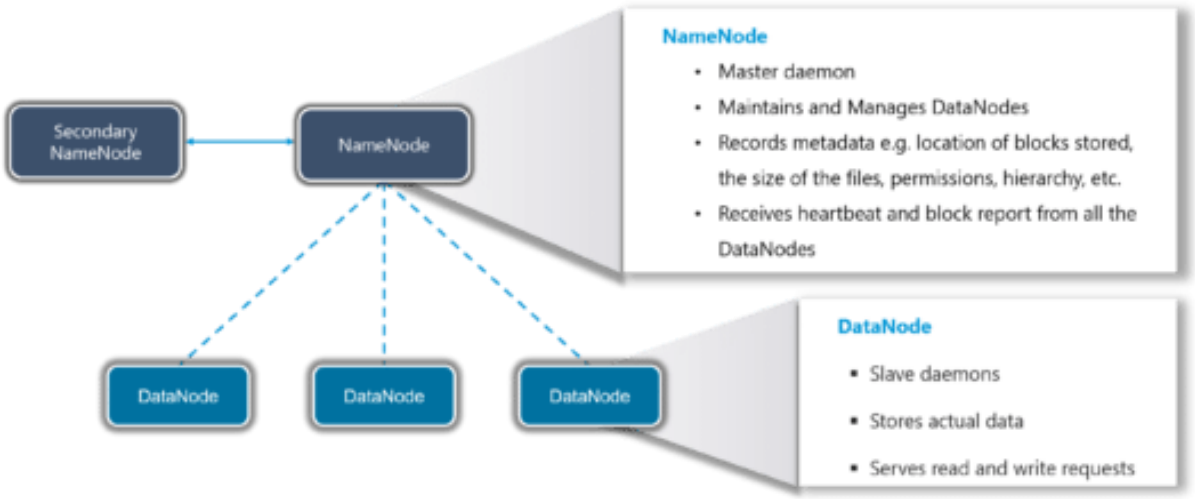


Figure1. HDFS

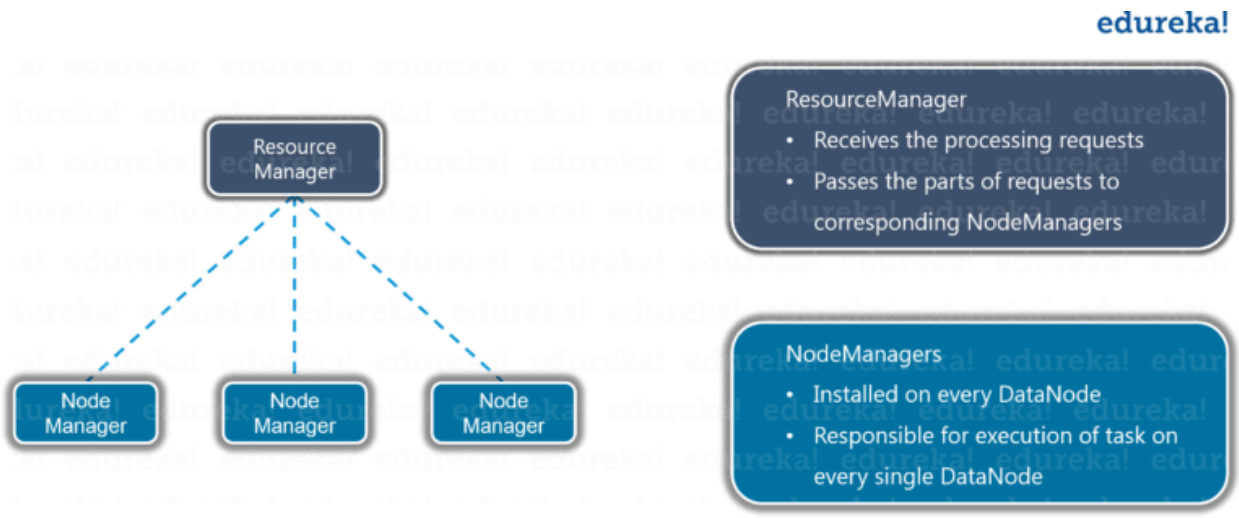


Figure 21. YARN

Data Warehouses have served very good purposes of Data Storage and Data Analytics. But with the exponential growth of data and with the potent intelligence lying in them, enterprises are pushing toward adopting Big Data Technologies.

While Hadoop provides a reliable, scalable and distributed computing power, its initial design was towards enabling batch processing of large data. But if you need to move our analysts to use Hadoop, to unleash the power of insights into large data, we need to enable:

1. Interactive querying capability
2. Reporting and dashboard development through BI Tools

Hadoop Eco-system:

The ecosystem around Hadoop is rapidly developing and adding on tools for various data needs. No single tool can provide both and hence we would have to go with a combination of tools.

The solution could be a combination of one Interactive Querying tool and one OLAP (online Analytical Processing) tool on Hadoop. **OLAP on Hadoop – Apache Kylin**

- **What is Kylin**

Kylin is an open source Distributed Analytical Engine that provides SQL interface and multidimensional analysis (OLAP) on Hadoop supporting extremely large datasets. Apache kylin pre-calculates OLAP cubes and store the cubes into a reliable and scalable datastore (HBase).

❖ **Why Kylin**

In most of the use cases in bigdata, we see the challenge is to get the result of the query within a sec. It takes lot of time to scan the database and to return the results. This is where the concept of OLAP in Hadoop emerged to combine the strength of OLAP and Hadoop and hence gives a significant improvement in query latency.

❖ **How it works?**

Below are the steps on how kylin fetches the data and saves the results.

- First, sync the input source table. In most of the cases, it reads data from Hive
- Next it runs map reduce / spark jobs (based on the engine you select) to pre-calculate and generate each level of cuboids with all possible combinations of dimensions and calculate all the metrics at different levels.
- Finally, it stores cube data(aggregated result) in HBase where the dimensions are rowkeys and measures are column family.

After aggregated data is ready, we can integrate with popular BI tools, such Tableau or Superset or connect with JDBC from our code. One more option is Kylin also provides the simple UI for quickly visit data.

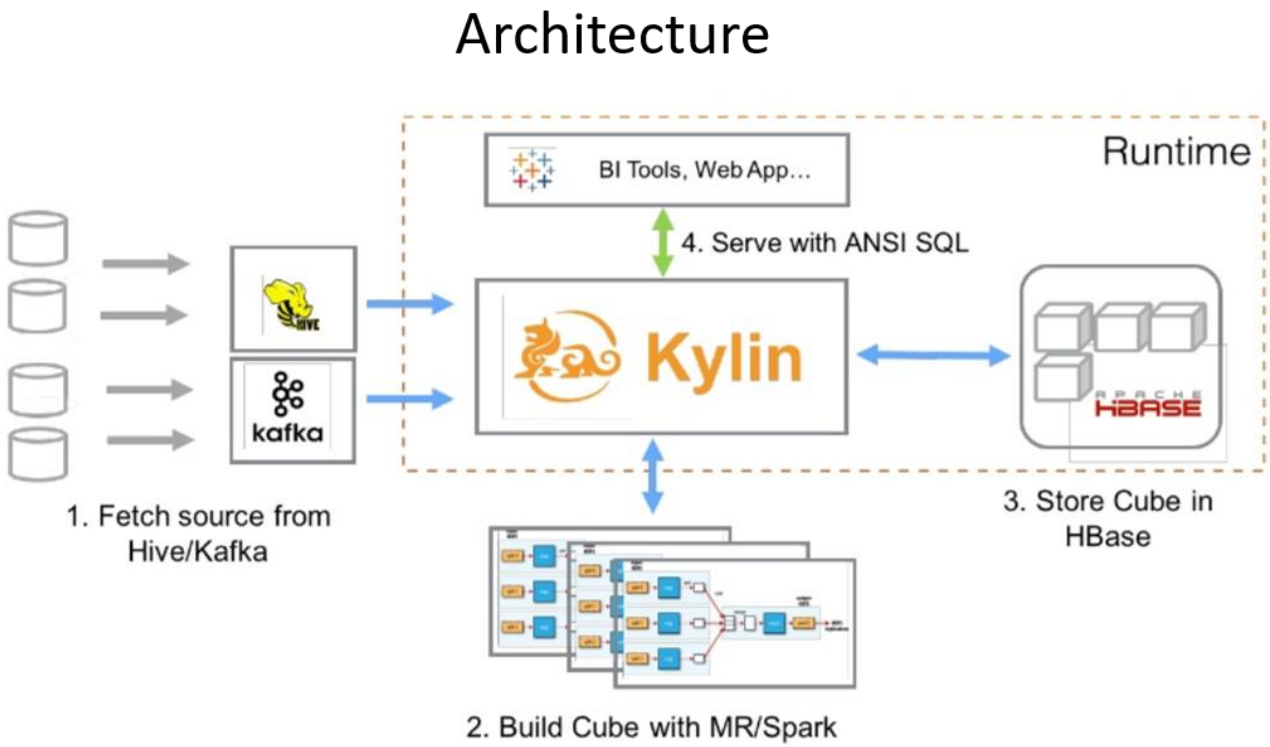


Figure 32. Apache Kylin – Architecture

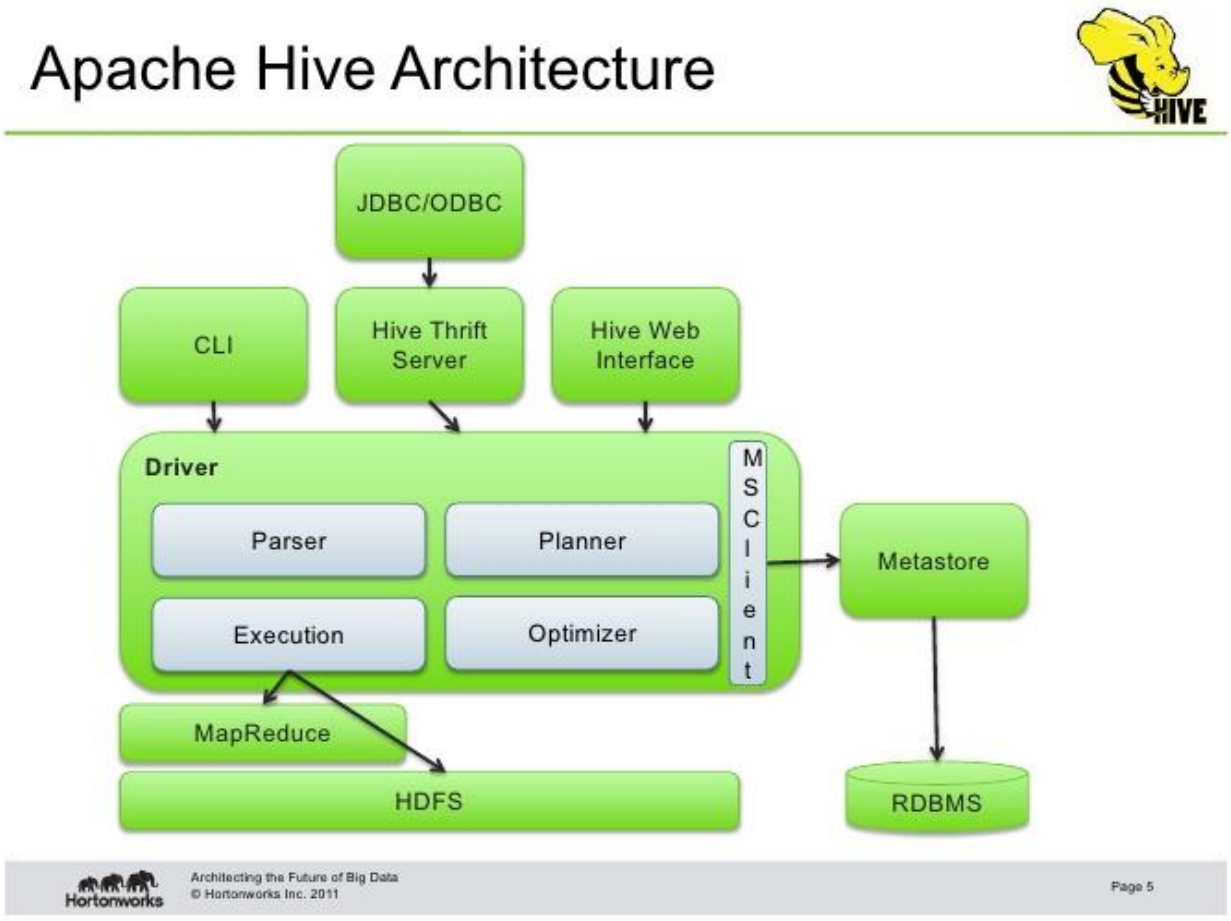
Hive

Apache Hive is a data warehouse system built on top of Hadoop or in Hadoop Ecosystem and is used for analyzing structured and semi-structured data. Basically, it provides a mechanism to project structure onto the data and perform queries written in HQL (Hive Query Language) that are similar to SQL statements. Internally, these queries or HQL gets converted to map reduce jobs by the Hive compiler.

Hive is not only a savior for people from a non-programming background, but it also reduces the work of programmers who spend long hours writing MapReduce programs.

Apache Hive supports Data Definition Language (DDL) and Data Manipulation Language (DML)

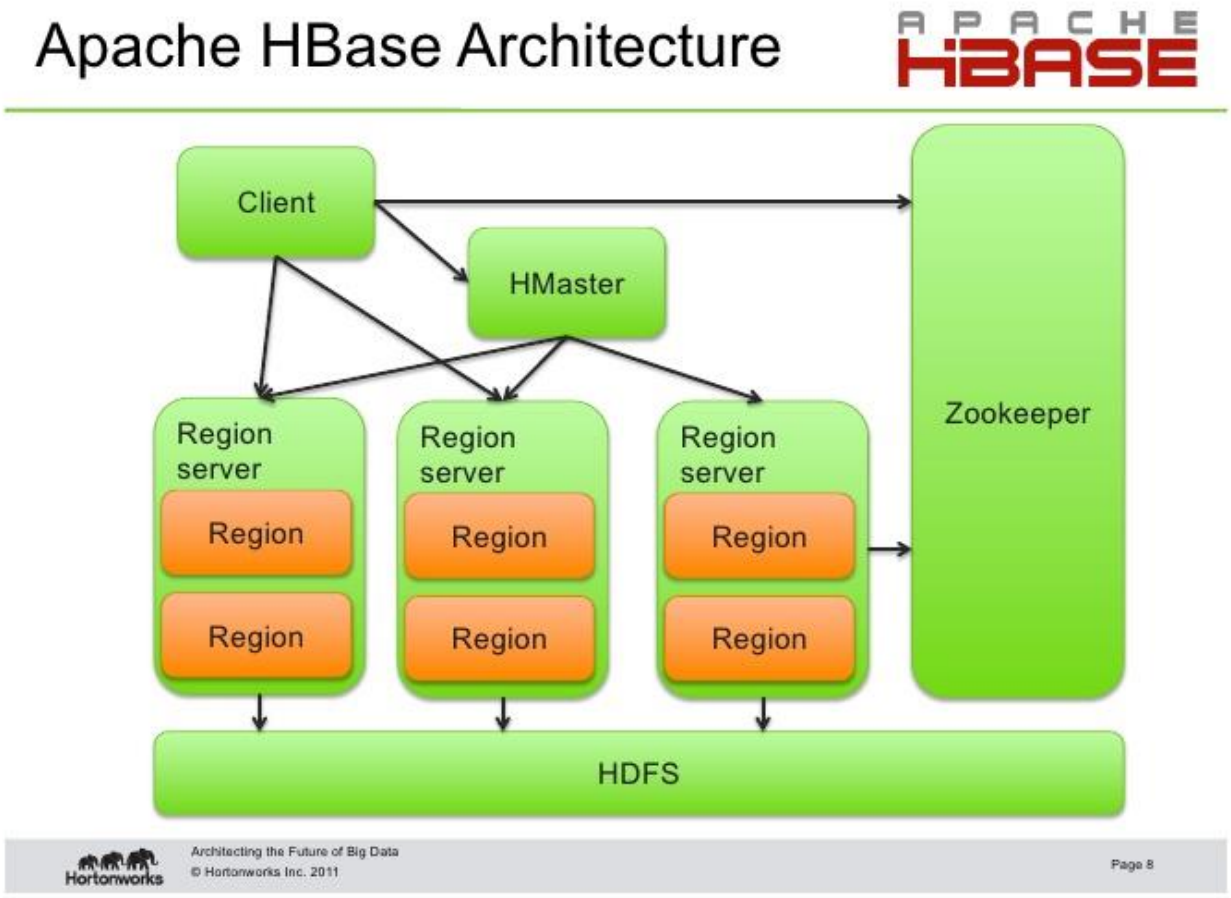
SQL + Hadoop MapReduce = HiveQL



HBase

HBase is an open-source, multidimensional, distributed, scalable, and a NoSQL database written in Java. HBase runs on top of HDFS (Hadoop Distributed File System) and provides Bigtable like capabilities to Hadoop.

HBase supports random read and writes while HDFS supports WORM (Write once Read Many or Multiple times).



- **Where we can use HBase?**

We should use HBase where we have large data sets (millions or billions of rows and columns) and we require fast, random, and real-time, read, and write access over the data.

OLAP cube definition

First the user must identify the tables and the relationship between them over the data model stored in the source system chosen. Secondly, to reduce the effects of the dimensionality curse, it is very important to determine the optimal type for each dimension column. Finally, for those dimensions defined as “Normal”, we must create one or more aggregation groups and apply to them the possible optimizations allowed.

Cubes:

For designing the OLAP cube, Apache Kylin provides a Web UI with step-based design. To define the OLAP cube, Kylin can use as data source (i), a Data Warehouse (DW) stored on Hive for batch applications, or (ii), a Kafka topic for real-time OLAP applications.

Then the user has to define a cube design including facts, dimensions and their mapping with the data sources along with other parameters, such as dimensions optimizations, algorithm selection or the cube engine used. Using this metadata, the cube building process takes advantage of the distributed processing of Map Reduce or Spark. As a result of this process, an OLAP cube with the aggregated data are generated and stored on Hbase.