Automatic Voting Machine using Hadoop

Ms. Shireen Fatima
M. Tech Student
Department of Computer Science and Engineering
Goel Institute of Technology & Management, Lucknow, India

Mr. Shivam Shukla
Assistant Professor
Department of Computer Science and Engineering
Goel Institute of Technology & Management, Lucknow, India

Abstract

Voting style has been changed from the word counting papers to the electronically voting records. The system provides many advantages over the traditional voting system by reducing the voting process time and provides the performance in terms of more flexibility and accuracy. But there are some drawbacks also. The large volume of data takes a lot of time to process which affects the system performance. This huge amount of data can be stored, processed and analyzed in many ways but they require fast retrieval technique. Hadoop is considered as the best solution for handling big data which uses parallel computing techniques. Hadoop gives a complete administration apparatus to manage the huge data. That leads us to do this project using Hadoop and HIVE. The demand for Automatic Voting Machine is ever increasing and the system creates a huge amount of data. The results that are produced should be processed in an efficient way. Traditional data storage system and their data processing techniques are not really effective in handling Big Data. These data can take different forms like structured, unstructured and semi structured. The processing power of the machine is influence by the large data size. The system is fully automated and be able to handle extremely large volumes of data. The datasets are created using the application and are used to analyze through HIVE and OOZIE. In Automatic Voting Machine user enter his/her voter Id. Through voterid it will check in HIVE tables whether voter ID is valid or invalid. If he/she is invalid, script will exit or else again check whether the voter is coming for the first time or not. If voter is coming for second time then script will exit and if he is coming for first time then voter can select the candidate (party name) of his/her choice and cast the vote. The project contains the sqoop command. This is schedule in oozie to bring the data from mysql (RDBMS) to HIVE table. If the voting is completed, here it is assume that at 5 PM voting will be completed and voter will not be allowed to cast his/her vote. So whenever wrapper script will execute it will check whether it is 5 PM if not then voter can cast his vote or else result will be displayed in terms of counting of votes with respect to candidates. Even the system also gives the result in terms of winning percentage with great accuracy.

Keywords- Hadoop, Sqoop, Hive- Hql, Mapreduce, Oozie

I. Introduction

We live in the data age. It’s difficult to measure the total volume of data stored electronically but an IDC estimate put the size of the “digital universe” at 4.4 zettabytes in 2013 and it is forecasted a tenfold growth in data by 2020 to 44 zettabytes. (A zettabyte = one billion terabytes =one billion 1021 Gigabytes). This is more than one disk drive for every person in the world. This flood of data comes from many sources. A survey states that The New York Stock Exchange generates about 4−5 terabytes of data per day. Face book hosts more than 240 billion photos means generating 7 petabytes per month. Ancestry.com- the geology site stores around 10 petabytes of data. So there’s a large amount of data. Most of the data is locked up in the largest web properties (like search engines) More generally the digital streams that individuals are producing are growing apace. The volume of data being made is increased every year. Not only the data is increasing but also there is a great need to store, manage and retrieving the data. Many Organizations need the accessing of stored data. The growth of data is termed as Big Data. This is good that big data is here. But the bad thing is that we are struggling to store and analyze it. Big data is a term applied to data sets whose size is beyond the ability of other popular software tools to capture, manage, and process the data within a tolerable elapsed time. Big data can be defined in V concepts--Volume, Velocity, Value, and Variety.

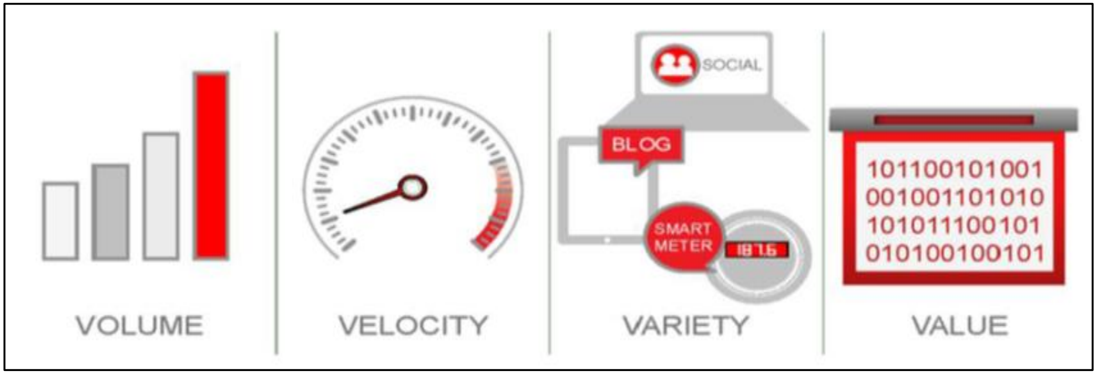


Fig. 1 Big Data In V Terms

Only Hadoop provides: a reliable, scalable platform for storage and analysis. Hadoop is affordable as it runs on commodity hardware and is open source. Even Hadoop isn't the first distributed system which store and analyze the data. It has some unique properties that stand it apart from the other systems.

II. CONCEPT: HADOOP

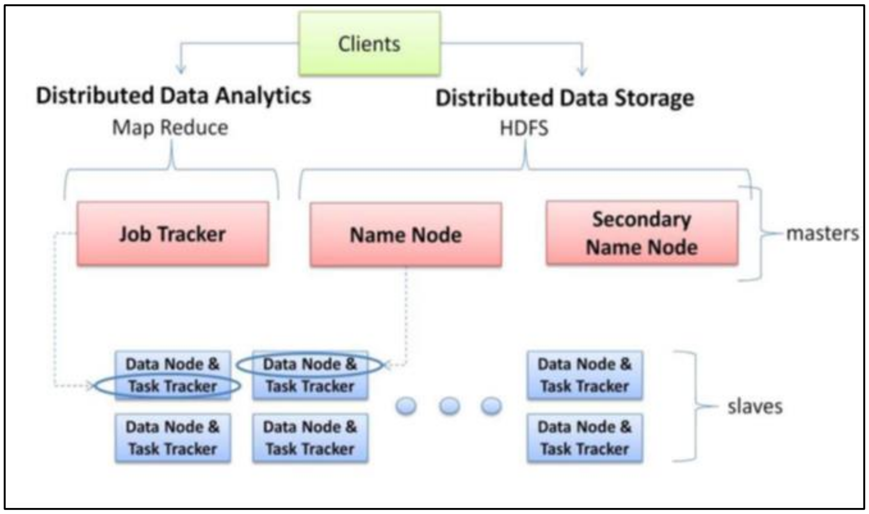


Fig. 2: Hadoop Architecture

The figure 2 shows the basic architecture of Hadoop framework. Hadoop works in cluster. Hadoop framework is divided in two parts-HDFS is used for storing the data set and Map Reduce is for processing. It has two phase Map and Reduce. HDFS stands for Hadoop Distributed File System. It consists of one namenode and one secondary namenode. There are some blocks in HDFS. An HDFS cluster works on master slave architecture. There are two types of nodes, which work in a master/slave pattern: name node (the master) and number of data nodes (slaves).

The name node organizes the file system namespace. It maintains the file system tree and the metadata for all the files and directories in the tree. This information is stored regularly on the local disk in the form of two files:- namespace image and edit log. The name node also knows the data nodes on which all the blocks for a given file are located. However it does not store block locations steadily as this information is reconstructed from data nodes when the system starts.

When A client accesses the file system on behalf of the user by communicating with the name node and data nodes, the client presents a file system interface similar to a Portable Operating System Interface (POSIX). So the user code does not have any need to know about the name node and data nodes which are in function Data nodes are the workhorses of the file system. When client access these, they store and retrieve blocks or the name node and they report back to the name node in regular interval with lists of blocks that they are storing. Data nodes cannot communicate directly to the client or to the other nodes.

A. Map Reduce Architecture

Map Reduce architecture is based on the Master - Slave architecture. Here Map-Reduce Master is termed as "Jobtracker" and Map-Reduce Slaves are termed as "Tasktrackers".

Jobtracker accepts Map Reduce jobs which is submitted by users. Jobtracker assigns Map and Reduce tasks to Tasktrackers. Furthermore it monitors task tasktracker status. It re-executes tasks if any failure. Map-Reduce Slaves i.e. Tasktrackers perform map and reduce tasks according to the instruction given by the Jobtracker. Its function is to handle the storage and transmission of intermediate output.

MR architecture is generic reusable framework which supports pluggable user code even pluggable FileSystem like DFS, KOSMIX, S3.Map Reduce Data Flow is shown in figure3.When a client submit the job it goes to Jobtracker. Jobtracker in the namenode assigns the task to the tasktracker in the datanode. The job file is divided into some chunk nodes. The data nodes cannot communicate directly. If there is a failure of any data node, it will be acknowledged to the namenode first and then the namenode assigns the task to another one. The task is divided in two processing logic mapper and Reducer. After this we get the final result through the namenode.

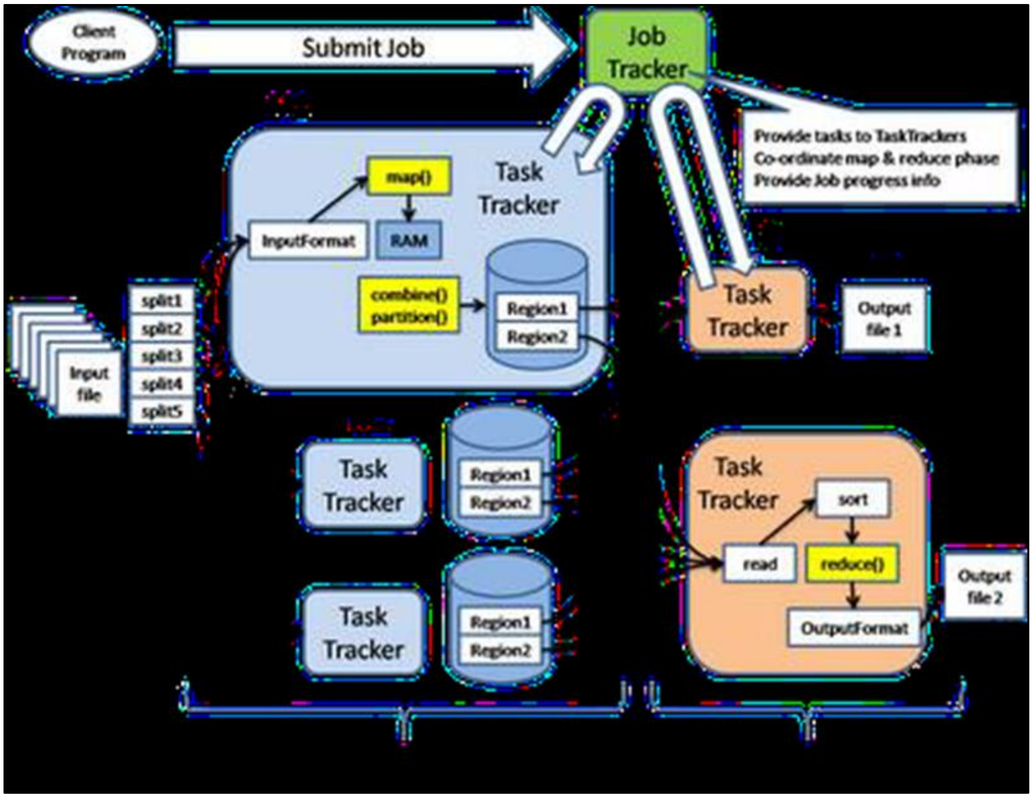


Fig. 3: Map Reduce Data Flow

B. Sqoop

Sqoop is an easy parallel database import/export tool. Sqoop can Insert data from RDBMS to HDFS and Export data from HDFS back into RDBMS.

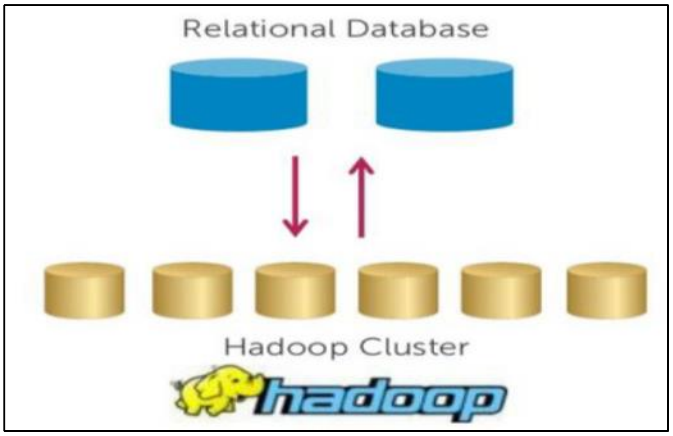


Fig. 4: Importing/Exporting Data through Sqoop

C. Hive

Although Map Reduce is very powerful, it can also be complex to master. Map Reduce is having Java codes which are not very easy for all IT experts. Even several organizations have business or data analysts who are good at writing SQL queries, but not at writing Java code. So for them it becomes difficult to handle it with Java codes. But several organizations have programmers who have skill in code writing in scripting languages. Thus Hive and Pig are two sub projects which evolved separately to help such people analyze huge amounts of data via Map Reduce.

Hive was initially developed at Facebook. HIVE is an analytical language which is first developed by Face book. It is an SQL-like interface to Hadoop. It is a Data Warehouse infrastructure that provides data summarization and ad hoc querying on top of Hadoop. Hive uses Map Reduce for the execution. It uses HDFS (Hadoop File Distribution System). It is the HIVE Query Language (HqL). It performs all the basic-SQL operation like: Select, From, Join, Group-By. Besides that through HIVE one can perform Equi-Join, Muti-TableInsert, Multi-Group-By and also Batch query.

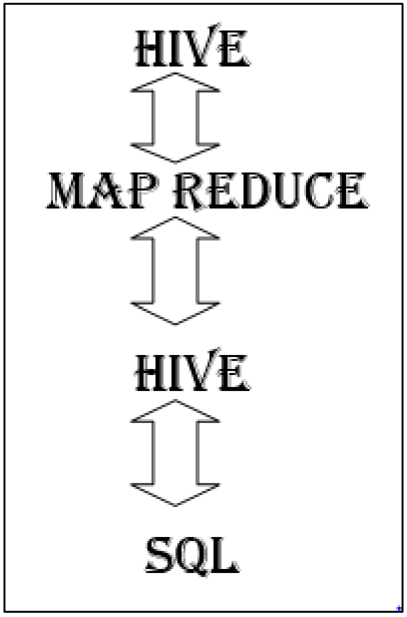


Fig. 5: HIVE

Conversion of Query to Map Reduce is done by HIVE compiler. Whatever table we store in HIVE stored in Metadata. Internally Map Reduce execution is done in HIVE.

III. PLAN, ANALYZE AND DESIGN

A. Low Level Diagram (LLD)

Here in the fig 6 low level overview of my project is shown through the general flow chart. In this flow chart, all the probable requirements are clearly described to design the AUTOMATIC VOTING MACHINE. First of all the system check the time validity after that it allow to enter the voter ID. Then it check the validity of voter ID. If the valid voter ID then it further check for the entry, if it is first time then allow to cast vote and exit. After 5 pm the system gives the result in terms of counting of vote with respect of candidates.

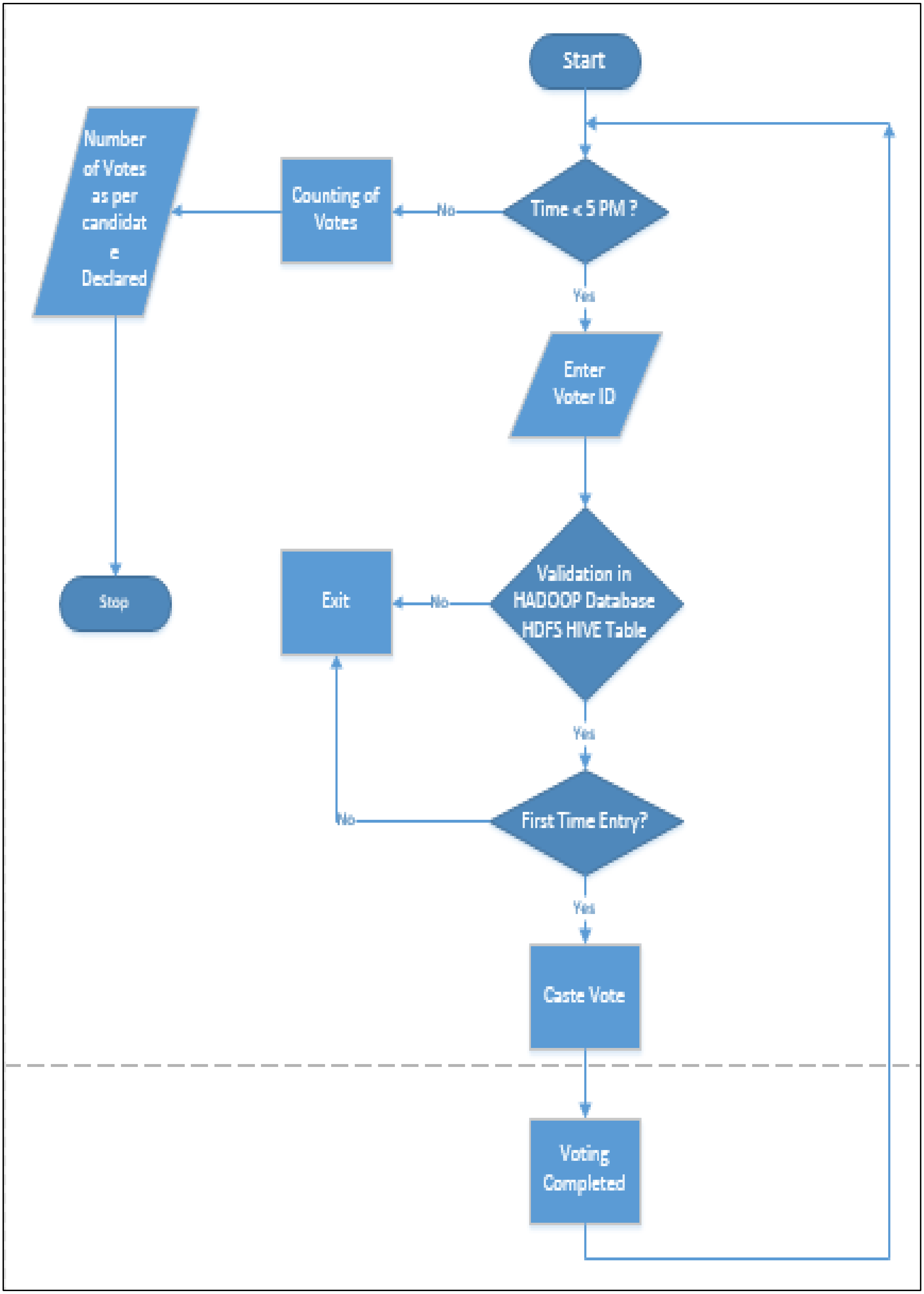


Fig. 6: Low level Overview

B. HLD (High Level Diagram)

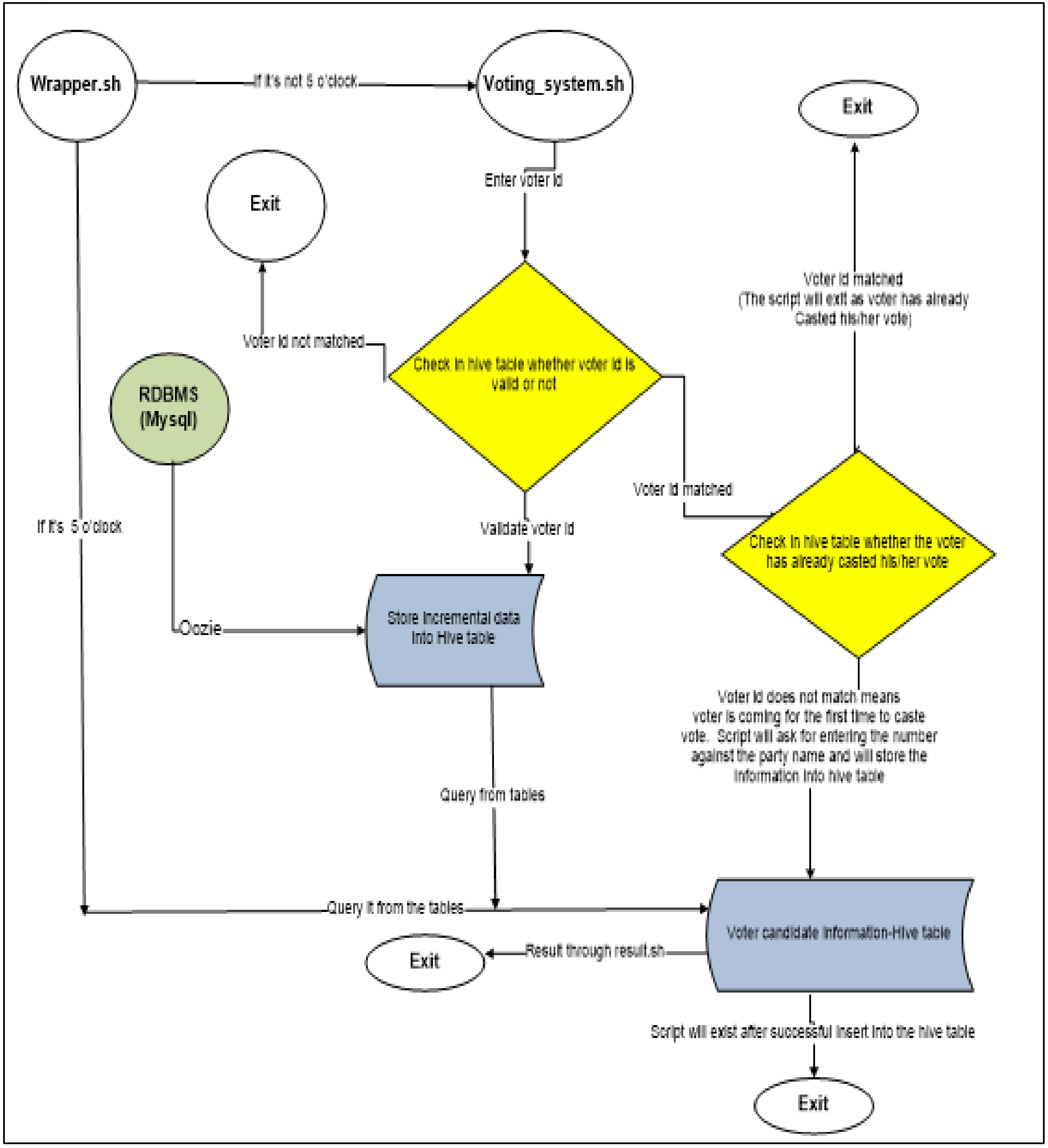

Wrapper.sh
If it's not 5 o'clock
Voting_system.sh
Exit
Enter voter Id
Exit
Voter Id matched
(The script will exit as voter has already Casted his/her vote)
Voter Id not matched
Check in hive table whether voter id is valid or not
RDBMS (Mysql)
Voter Id matched
If it's 5 o'clock
Validate voter id
Check in hive table whether the voter has already casted his/her vote
Oozie
Store Incremental data into Hive table
Voter Id does not match means voter is coming for the first time to caste vote. Script will ask for entering the number against the party name and will store the information into hive table
Query from tables
Query it from the tables
Voter candidate information-Hive table
Exit
Result through result.sh
Script will exist after successful insert into the hive table
Exit

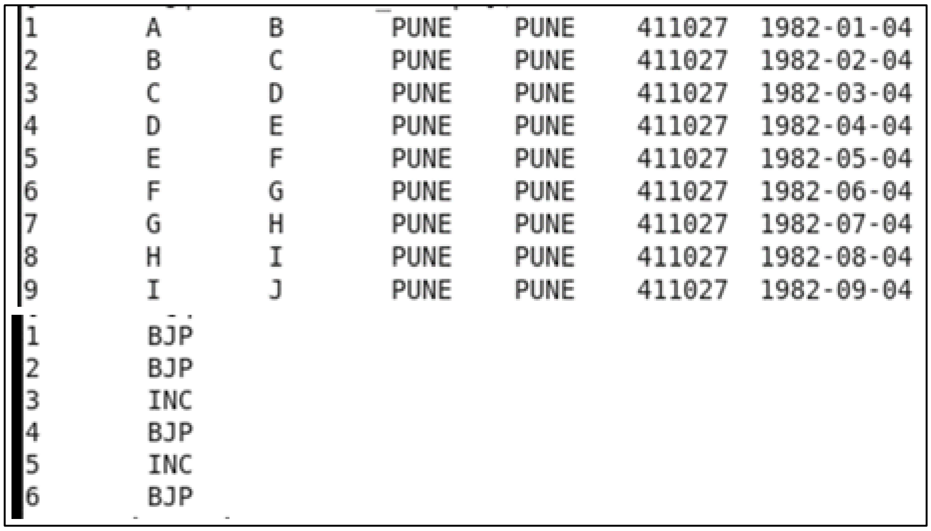
Fig. 7: High Level Overview of project

As shown in fig 7, first voting system script come in existence where voter id validation will be done through hive table. Incremental data will be stored in HIVE table from RDBMS through sqoop. Oozie is work scheduler. If voter comes for second time, the script will exit. Validated voter can caste vote and choose the candidate. Exact 5 pm wrapper script comes in existence that doesn't allow any casting of vote. Result script gives the result.Voting_system.sh is used to let user enter his/her voterid. Through voterid it will check in hive tables whether voter is valid voter or invalid voter. If he/she is invalid script will exit or else again check whether voter is coming for the first time or not. If voter is coming for second time then scirpt will exit and if he is coming for first time then voter can select the candidate (party name) of his/her choice and caste the vote. Wrapper script will execute first, if the voting is completed, here it is assume that at 5 PM voting will be completed and voter will not be allowed to caste his/her vote. So whenever wrapper script will execute it will check whether it is 5 PM if not then voter can cast his vote (voting_system.sh will execute) or else result will be displayed (result.sh will execute).

IV. Results

This project Automatic Voting Machine using Hadoop is made in Hadoop environment. The desired result is being shown below.

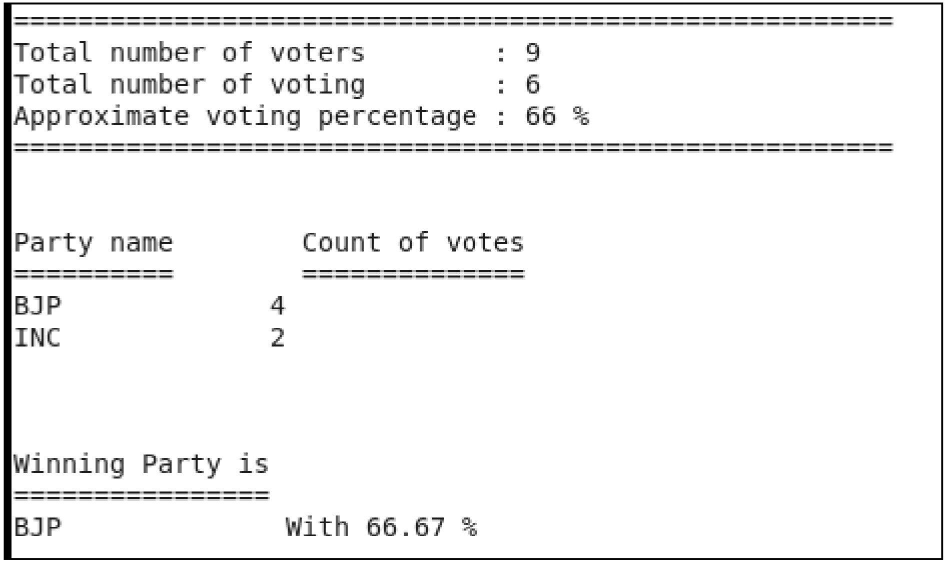
This will give the final result in terms of counting. It will show the result of the election for particular voting booth. For example database of 9 voters has been created for testing. The result will be:
Total voter: 9
Total voter who has caste their vote: 6
It will show result as below.



```
1       A       B       PUNE    PUNE    411027  1982-01-04
2       B       C       PUNE    PUNE    411027  1982-02-04
3       C       D       PUNE    PUNE    411027  1982-03-04
4       D       E       PUNE    PUNE    411027  1982-04-04
5       E       F       PUNE    PUNE    411027  1982-05-04
6       F       G       PUNE    PUNE    411027  1982-06-04
7       G       H       PUNE    PUNE    411027  1982-07-04
8       H       I       PUNE    PUNE    411027  1982-08-04
9       I       J       PUNE    PUNE    411027  1982-09-04
1       BJP
2       BJP
3       INC
4       BJP
5       INC
6       BJP
```

Fig. 8: Total vote done

Result is declared in terms of counting of votes with respect to candidates and percentage.



```
==================================================
Total number of voters        : 9
Total number of voting        : 6
Approximate voting percentage : 66 %
==================================================


Party name        Count of votes
==========        ==============
BJP             4
INC             2



Winning Party is
================
BJP               With 66.67 %
```

Fig. 9: showing final result

V. Conclusions

The project Automatic Voting Machine using Hadoop is successfully completed. This system is validating the voter ID efficiently. HIVE is dealing with real time query and gives the result accurately. Even we have the previous voting machines based on microcontroller but this project is giving more efficient and accurate result as traditional one are not great enough in quick counting process. This system gives the counting with the great accuracy in a very less timing. The system was given some electronic data and it gives the accurate result successfully.

References

[1] R. K. Nadesh, K. Arivuselvan, and Srinivasan Pathanjali, A Quantitative Review on Introducing the Election Process with Cloud Based Electronic Voting and Measuring the Performance using Map Reduce, Indian Journal of Science and Technology, Vol 9(39), DOI: 10.17485/ijst/2016/v9i39/85585, October 2016

[2] Apache Hadoop – Wikipedia
https://en.wikipedia.org/wiki/Apache_Hadoop, http://en.wikipedia.org/wiki/Big_data

[3] http://www.cnet.com/news/facebook-processes-more-than-500-tb-ofdata-daily/

[4] http://en.wikipedia.org/wiki/Apache_Hadoops

[5] Zhouwei, Pierre Guillaume and Chi-Hung Chi. Cloud TPS: Scalable Transactions for web applications in the cloud. IEEE transactions of scalable computing. 2012 Dec; 5(04).

[6] Megiba Jasmine R and Nishiba GM. Public Cloud secure group sharing and accessing in cloud computing. Indian Journal of Science and Technology. 2015 July; 8(15).

[7] Bhosale Poonam, Vethaka Priyanaka, Thorat Lata, Archana Lomte. Identity Access Management using Multitier Cloud Infrastructure for secure online voting system. IJMRD 2015 March; 2(4)

[8] Shymala K, Sunitha Rani T. An analysis on efficient resource allocation mechanism in cloud computing. Indian Journal of Science and Technology. 2015 May; 8(9).

[9] Rama Satish KV, Kavya NP. Big Data Processing with harnessing Hadoop-MapReduce for Optimizing Analytical Workloads. IEEE 2014, International Conference on Contemporary Computing and Informatics.

[10] Kyoo-Sungnoh and Doo-Sik-Lee. Bigdata platform design and implementation. Indian Journal of Science and Technology. 2015 Aug; 8(9).

[11] http://wiki.apache.org/hadoop/JobTracker

[12] Mohammad Hammoud and Majd F. Sakr, “Locality-aware reduce task scheduling for MapReduce," 3rd Int. Conf. on Cloud Computing Technology and Science (CloudCom), IEEE, pp. 570-576, 2011.