ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

Data Injecting Procedures in Big Data by Using Hadoop

A.Swarupa Rani[1] and S. Jyothi[2]

Research Scholar, Sri Padmavathi Mahila Visvavidyalayam, Tirupati, AP, India[1]

Dept. of Computer Science, Sri Padmavathi Mahila Visvavidyalayam, Tirupati, AP, India[2]

ABSTRACT: The paradigm of processing huge datasets has been shifted from centralized architecture to distributed architecture. As the enterprises faced issues of gathering large chunks of data they found that the data cannot be processed using any of the existing centralized architecture solutions. Apart from time constraints, the enterprises faced issues of efficiency, performance and elevated infrastructure cost with the data processing in the centralized environment. With the help of distributed architecture these large organizations were able to overcome the problems of extracting relevant information from a huge data dump. One of the best open source tools used in the market to harness the distributed architecture in order to solve the data processing problems is Apache Hadoop. Using Apache Hadoop's various components such as data clusters, map- reduce algorithms and distributed processing, we will resolve various location-based complex data problems and provide the relevant information back into the system, thereby increasing the user experience. Map Reduce allows people to perform computations of big data-sets, computations that we can't easily perform without this kind of architecture. It is a simple, efficient powerful computing framework. It is very efficient. The idea behind Hadoop is to move computation to data. Hadoop Map Reduce provides a shared and integrated foundation where we can bring additional tools and build up this framework. In this paper, I represent the different types of data ingestion procedures in Big Data by implementing hadoop techniques.

KEYWORDS: Big Data, Hadoop, Map Reduce, Data Ingestion, Hive, Pig, Sqoop, and HDFS.

I. INTRODUCTION

In a study of big data analytics "Hadoop" plays a major role. Hadoop is used for big data purpose, it will work automatically and internally run each and every program by map reduce only. First we can inject the data into hadoop it will be in hadoop distributed file system by using Linux commands. Hadoop read access is good for large data and differentiate sorting order based on distance, Hadoop started in 2006, implementation done on 2009. It is a frame work for no limitation data and stored data. Hadoop is scalable, affordable, flexibility and fault tolerant. It is fully used rack topology. In this hadoop cluster is divided into racks and it will also handling reading, writing and data node failures. Map reduce is a distributing process with FSI (file system image). It divided into active mode and passive node work done by Hadoop. It occurs four types of jobs like Hadoop admin, Hadoop developer, Hadoop tester and Data analyst. Hadoop's strength is that it enables ad-hoc analysis of unstructured or semi-structured data. Relational databases, by contrast, allow for fast queries of very structured data sources. A point of frustration has been the inability to easily query both of these sources at the same time.

II. LITERATURE SURVEY

DataStax Corporation (2013) in white paper Evaluating Apache Cassandra as a Cloud Database describes the cloud databases and growing movement towards cloud computing. As for IDC survey by 2015, 20 percent information will be processed in cloud. This paper describes how apache Cassandra provides a smart platform for the data management in the cloud. Bhat and Jadhav (2010) in research paper entitled Moving towards Non-Relational Databases describes the implementation of RDBMS in different applications and the challenge of databases in cloud based applications. Ho

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

et al. (2011) and Ibrahim et al. (2010) concentrate their efforts on improving performance by changing the data flow in the transition between Mappers and Reducers. Originally, Hadoop employs an all-to-all communication model between Mappers and Reducers. This strategy may result in saturation of network bandwidth during the shuffle phase. This problem is known as the Reducers Placement Problem (RPP). Focusing on the same problem of correlated data placement, Eltabakh et al. (2011) propose Co Hadoop, an extension of Hadoop that allows applications to control where data are stored. CoHadoop addresses Hadoop's lack of ability to collocate related data on the same set of nodes. The approach is designed such that the strong fault tolerance properties of Hadoop are retained.

III. METHODOLOGY

Hadoop is storage plus processing, it can differentiate sequential files with the help of sink marker. In hadoop data ingestion placed a very important role. It means what data you want to bring HDFS (Hadoop distributed file system) and cluster. Data ingestion can be done in four types i.e., Local file system to HDFS, Remote desktop connector, External data base to HDFS and finally Streaming of the data.

3.1 Local File System to HDFS

Connect to the Hadoop cluster whose files or directories you want to copy to or from your local file system. You must run this command before using fs put or fs get to identify the name node of the HDFS. You can copy (upload) a file from the local file system to a specific HDFS using the fs put command.

bin/hadoop fs -get /hdfs/source/path /localfs/destination/path

bin/hadoop fs -copyToLocal /hdfs/source/path /localfs/destination/path

3.2 Remote Desktop Connection

With Remote Desktop Connection, you can connect to a computer running Windows from another computer running Windows that's connected to the same network or to the Internet. For example, you can use all of your work computer's programs, files, and network resources from your home computer, and it's just like you're sitting in front of your computer at work. Here I am connecting and copying files folders from windows xp to oracle virtual box following steps should follows i.e.

Step1: First create a folder on windows xp desktop and select that folder right click on it.

Step2: choose share with specific people and add the folder give the permission for both read and write and share choose done

Step3: Go for command prompt type ipconfig and find the ip address of windows system.

Step4: Go to oracle virtual box menu select places move to network connections.

Step5: Select the server type as windows server and also give the ip address of windows xp.

Step6: finally choose the connect

Using this procedure you can connect easily form windows xp to oracle virtual box and copy what you want on windows xp to oracle virtual box.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

3.3 External database

The Hadoop ecosystem includes other tools to address particular needs. External databases can be divided into two parts i.e. batch processing and streaming of data. Hive is a SQL dialect and Pig is a dataflow language for that hide the tedium of creating Map Reduce jobs behind higher- level abstractions more appropriate for user goals.

Pig

Pig is a platform for constructing data flows for extract, transform, and load (ETL) processing and analysis of large datasets. Pig Latin, the programming language for Pig provides common data manipulation operations, such as grouping, joining, and filtering. Pig generates Hadoop Map Reduce jobs to perform the data flows. This high-level language for ad hoc analysis allows developers to inspect HDFS stored data without the need to learn the complexities of the Map Reduce framework, thus simplifying the access to the data. The Pig Latin scripting language is not only a higher-level data flow language but also has operators similar to SQL (e.g., FILTER and JOIN) that are translated into a series of map and reduce functions. Pig Latin, in essence, is designed to fill the gap between the declarative style of SQL and the low-level procedural style of Map Reduce. Pig properties are lazy evaluation and predefined keywords.

grunt> fs –ls

Found 3 items

drwxrwxrwx- Hadoop supergroup	0 2015-09-08 14:13 Hbase
drwxr-xr-x- Hadoop supergroup	0 2015-09-09 14:52 seqgen_data
drwxr-xr-x- Hadoop supergroup	0 2015-09-08 11:30 twitter_data

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

Hive

Hive is a SQL-based data warehouse system for Hadoop that facilitates data summarization, adhoc queries, and the analysis of large datasets stored in Hadoop-compatible file systems (e.g., HDFS, MapR-FS, and S3) and some NoSQL databases. Hive is not a relational database, but a query engine that supports the parts of SQL specific to querying data, with some additional support for writing new tables or files, but not updating individual records. That is, Hive jobs are optimized for scalability, i.e., computing over all rows, but not latency, i.e., when you just want a few rows returned and you want the results returned quickly. Hive's SQL dialect is called HiveQL. Table schema can be defined that reflect the data in the underlying files or data stores and SQL queries can be written against that data. Queries are translated to Map Reduce jobs to exploit the scalability of Map Reduce. Hive also support custom extensions written in Java, including user-defined functions (UDFs) and serializer - deserializers for reading and optionally writing custom formats, e.g., JSON and XML dialects. Hence, analysts have tremendous flexibility in working with data from many sources and in many different formats, with minimal need for complex ETL processes to transform data into more restrictive formats.

Hive is designed to enable easy data summarization, ad-hoc querying and analysis of large volumes of data. It provides SQL which enables users to do ad-hoc querying, summarization and data analysis easily. At the same time, Hive's SQL gives users multiple places to integrate their own functionality to do custom analysis, such as User Defined Functions (UDFs).

hive -e 'select a.col from tab1 a'

hive -S -e 'select a.col from tab1 a'

```
hive> create database retail;
OK
Time taken: 5.275 seconds
hive>
```

```
hive> describe txnrecords;
OK
txnno   int
txndate string
custno  int
amount  double
category        string
product string
city    string
state   string
spendby string
Time taken: 0.122 seconds
hive>
```

This is the most common way to move data into Hive when the ORC file format is required as the target data format. Then Hive can be used to perform a fast parallel and distributed conversion of your data into ORC. The process is shown in the following diagram:

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

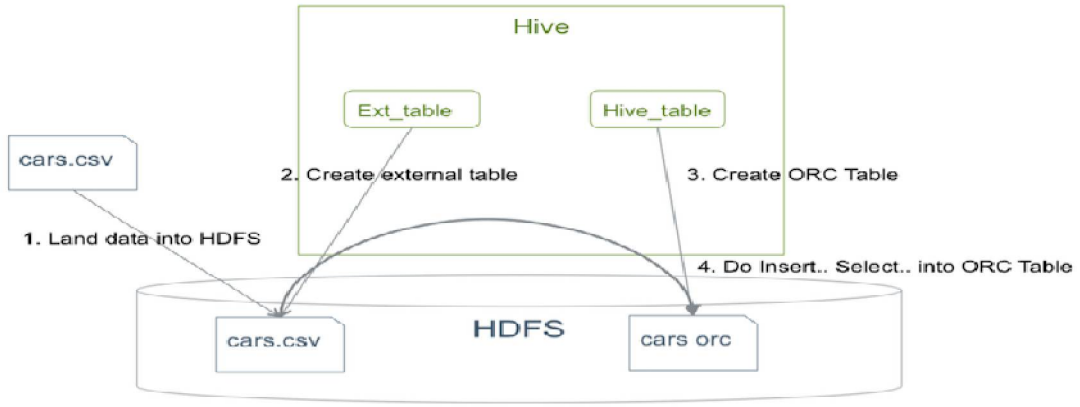


Figure 1.1. Example: Moving .CSV Data into Hive

Sqoop Connectors

All the existing Database Management Systems are designed with SQL standard in mind. However, each DBMS differs with respect to dialect to some extent. So, this difference poses challenges when it comes to data transfers across the systems. Sqoop Connectors are components which help overcome these challenges. Sqoop is a command-line interface application for transferring data between relational databases and Hadoop. Sqoop has connectors for working with a range of popular relational databases, including MySQL, PostgreSQL, Oracle, SQL Server, and DB2. Each of these connectors knows how to interact with its associated DBMS. There is also a generic JDBC connector for connecting to any database that supports Java's JDBC protocol. In addition, Sqoop provides optimized MySQL and PostgreSQL connectors that use database-specific APIs to perform bulk transfers efficiently. It supports:

- Incremental loads of a single table,
- A free form SQL query,
- Saved jobs which can be run multiple times to import updates made to a database since the last import.

Data transfer between Sqoop and external storage system is made possible with the help of Sqoop's connectors.

```
$ sqoop import \
--connect jdbc:mysql://localhost/userdb \
--username root \
--table emp --m 1
sqoop import --connect jdbc:mysql://db.foo.com/bar --table EMPLOYEES
"start_date > '2010-01-01'"   --where

sqoop import --connect jdbc:mysql://db.foo.com/bar --table EMPLOYEES --where
"id > 100000" --target-dir /incremental_dataset --append
```

3.4 Streaming data:

A service for streaming logs into Hadoop. Apache Flume is a distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of streaming data into the Hadoop Distributed File System (HDFS). Flume is a tool/service/data ingestion mechanism for collecting aggregating and transporting large amounts of streaming data such as log files, events (etc...) from various sources to a centralized data store. Flume basically consists

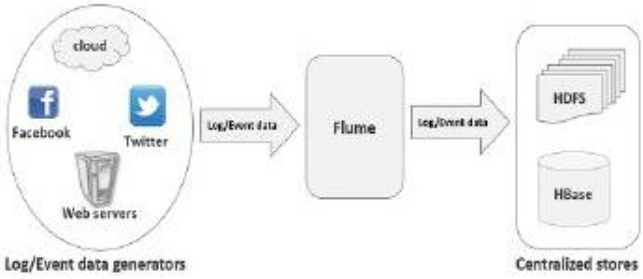
ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

of three main components, Source, Channel and Sink. A flume agent is basically a JVM process which representation of a simple Flume agent listening to a web server and writing the data to HDFS.

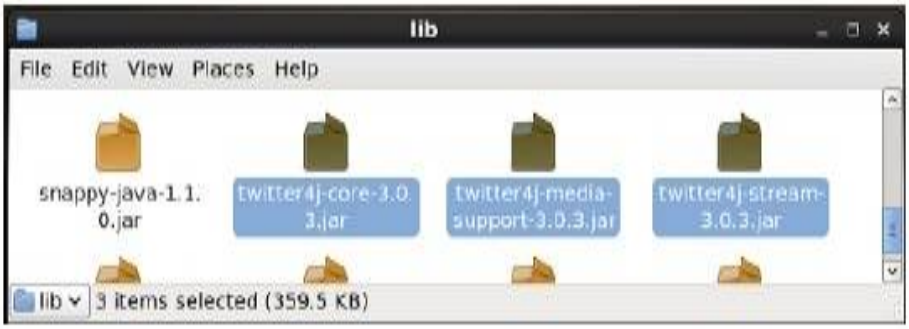


You can use any of them. For example, if you are transferring Twitter data using Twitter source through a memory channel to an HDFS sink, and the agent name id Twitter Agent, then

TwitterAgent.sources = Twitter
TwitterAgent.channels = MemChannel
TwitterAgent.sinks = HDFS

Twitter 1% Firehose Source

This source is highly experimental. It connects to the 1% sample Twitter Firehose using streaming API and continuously downloads tweets, converts them to Avro format, and sends Avro events to a downstream Flume sink. We will get this source by default along with the installation of Flume. The jar files corresponding to this source can be located in the lib folder as shown below.



Set the classpath variable to the lib folder of Flume in Flume-env.sh file as shown below.

Export CLASSPATH=$CLASSPATH:/FLUME_HOME/lib/*

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 6, Special Issue 13, July 2017

IV. CONCLUSION

The newest wave of big data is generating new opportunities and new challenges for businesses across every industry. The challenge of data integration incorporating data from social media and other unstructured data into a traditional BI environment is one of the most urgent issues facing CIOs and IT managers. Apache Hadoop provides a cost-effective and massively scalable platform for ingesting big data and preparing it for analysis. Using Hadoop to offload the traditional ETL processes can reduce time to analysis by hours or even days. Running the Hadoop cluster efficiently means selecting an optimal infrastructure of servers, storage, networking, and software. Intel provides software as well as hardware platform components to help you design and deploy an efficient, high-performing Hadoop cluster optimized for big data ETL. Take advantage of Intel reference architectures, training, professional services, and technical support to speed up your deployment and reduce risk. The benefits of Hadoop are best realized when it is teamed with a good archival solution that is cost effective, scalable and secure, allowing for speedy data retrieval and analysis. By using Hadoop as an archival platform, organizations can benefit from multiple source connectivity, scalability, cost benefits, and high accessibility of archived data for analytics. These solutions provide an attractive alternative to the delay and pain of archiving data on traditional mass storage devices, as well as to the high costs of modern archival appliances. By leveraging a Hadoop based archival solution; organizations can successfully harness the vast amounts of data at their disposal to realize competitive advantage.

REFERENCES

[1]. Apache Hadoop. Http://hadoop.apache.org
[2]. Steve LaValle, Eric Lesser, Rebecca Shockley, Michael S. Hopkins, Nina Kruschwitz, et al. (2010). MIT Sloan study; results published in MIT Sloan Management Review. “Big Data, Analytics and the Path from Insights to Value”.
[3]. http:www.pcisig.com/news_room/November_18_2010_Press_Release/
[4]. Gartner (2013) CIO Survey. http://www.gartner.com/technology/cio/cioagenda.jsp
[5].http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/big-data-apache-hadoop- technologies-for-results-whitepaper.pdf
[6]. M. A. Beyer and D. Laney, (2012). “The importance of ¨big data¨: A definition,” Gartner, Tech.
[7]. Kiran kumara Reddi & Dnvsl Indira, (2013). “Different Technique to Transfer Big Data : survey” IEEE Transactions on 52(8) 2348 { 2355}.
[8]. Jeffrey Dean and Sanjay Ghemawat, (2010). “MapReduce: Simplified Data Processing on Large Clusters” OSDI.
[9]. Albert Bifet, (2013). “Mining Big Data In Real Time” Informatica 37.
[10].J. Dean, S. Ghemawat, (2014) “MapReduce: Simplified Data Processing on Large Clusters,” In Proc.of the 6th Symposium on Operating Systems Design and Implementation, San Francisco CA.
[11].H.Herodotou,H.Lim,G.Luo,N.Borisov,L.Dong,F.B.Cetin,and S. Babu. Starfish et al., (2011) : A Self-tuning System for Big Data Analytics. In CIDR, pages 261–272.
[12].Hadoop MapReduce Tutorial. http://hadoop.apache.org/common/docs/r0.20.2/
mapred_tutorial.html
[13].Bu Y, Howe B, Balazinska M, Ernst MD (2012). The HaLoop approach to large-scale iterative data analysis. The VLDB Journal 21(2):169–90.
[14].Eltabakh MY, Tian Y, ¨Ozcan F, Gemulla R, Krettek A, McPherson J. Co- Hadoop (2011): flexible data placement and its exploitation in Hadoop. Proceedings of the VLDB Endowment 4(9):575–85.
[15].Hammoud M, Sakr M (2011). Locality-aware reduce task scheduling for MapReduce. In: Third International Conference on Cloud Computing Technology and Science. IEEE. p. 570–6.