ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

Manifest for Big Data – Pig, Hive & Jaql

Ajay Chotrani, Priyanka Punjabi, Prachi Ratnani, Rupali Hande

Final Year Student, Dept. of Computer Engineering, V.E.S.I.T, Mumbai, India

Faculty, Computer Engineering, VESIT, Mumbai, India

ABSTRACT: Hadoop is an open-source, Java-based implementation of Google's MapReduce framework. Hadoop is designed for any application which can take advantage of massively parallel distributed-processing, particularly with clusters composed of unreliable hardware. The term “big data” is pervasive, and yet still the notion engenders confusion. Big Data has been used to convey all sorts of concepts, including: huge quantities of data, social media analytics, next generation data management capabilities, real-time data, and much more. The HDFS architecture of Hadoop implements the mapping and reducing of data into clusters and then reducing the space. In this paper we try to have an overview about the open source with Hortonworks PIG, Hive and info sphere.

KEYWORDS*:* Hadoop, MapReduce, BigData, HDFS, PIG.

I. INTRODUCTION

To explore the large data in the organization and to process and analyze the large set of data is important. Big Data provides opportunities for business users to ask questions they never were able to ask before. The need to integrate Big Data techniques with their current enterprise data to gain that competitive advantage. Heterogeneity, scale, timeliness, complexity, and privacy problems with Big Data impede progress at all phases of the pipeline that can create value from data.

The problems start right away during data acquisition, when the data dictionary requires us to make decisions, currently in an ad hoc manner, about what data to keep and what to discard, and how to store what we keep reliably with the right metadata.

The value of data explodes when it can be linked with other data, thus data integration is a major creator of value. Since most data is directly generated in digital format today, we have the opportunity and the challenge both to influence the creation to facilitate later linkage and to automatically link previously created data. Data analysis, organization, retrieval, and modelling are other foundational challenges. Data analysis is a clear bottleneck in many applications, both due to lack of scalability of the underlying algorithms and due to the complexity of the data that needs to be analyzed. Finally, presentation of the results and its interpretation by non-technical domain experts is crucial to extracting actionable knowledge.

II. THE HADOOP ARCHITECTURE

Apache Hadoop is a framework for running applications on large cluster built of commodity hardware. The Hadoop framework transparently provides applications both reliability and data motion.

DOI: 10.15680/IJIRCCE.2015. 0310037

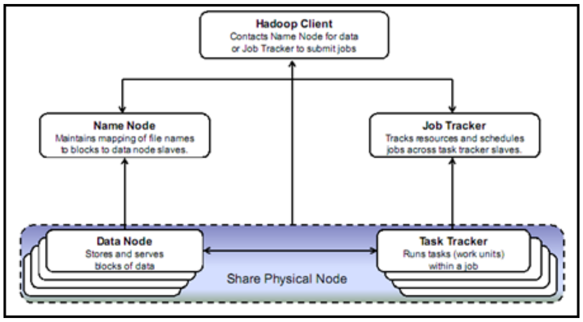

Hadoop Client
Contacts Name Node for data or Job Tracker to submit jobs
Name Node
Maintains mapping of file names to blocks to data node slaves.
Job Tracker
Tracks resources and schedules jobs across task tracker slaves.
Data Node
Stores and serves blocks of data
Task Tracker
Runs tasks (work units) within a job
Share Physical Node

Figure 1: Hadoop -Repository & Refinery

The Hadoop repository is shown in Figure 1. Hadoop implements a computational paradigm named Map/Reduce, where the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the cluster.

In addition, it provides a distributed file system (HDFS) that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both MapReduce and the Hadoop Distributed File System are designed so that node failures are automatically handled by the framework.

As volumes of big data arrive from sources such as sensors, machines, social media, and all the data reliably and cost effectively. When data volumes are huge, the traditional single server strategy does not work for long. Pouring the data into the Hadoop Distributed File System (HDFS) gives architects much needed flexibility. Not only can they capture 10s of terabytes in a day, they can adjust the Hadoop configuration up or down to meet surges and lulls in data ingestion. This is accomplished at the lowest possible cost per gigabyte due to open source economics and leveraging commodity hardware. The architecture of Hadoop is shown below in Figure 2.

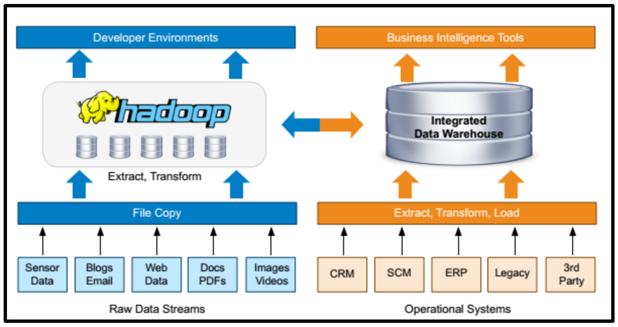


Figure 2: An Enterprise Architecture of Hadoop

Hadoop is an Apache open source project that provides a parallel storage and processing framework. Its primary purpose is to run MapReduce batch programs in parallel on tens to thousands of server nodes. MapReduce refers to the application modules written by a programmer that run in two phases: Map and Reduce.

III. THE FILE SYSTEM HDFS

Underlying all of these components is the Hadoop Distributed File System (HDFS™).This is the foundation of the Hadoop cluster. The NameNode structure and the cluster formation of HDFS are shown below in Figure 3. HDFS file system manages how the datasets are stored in the Hadoop cluster. It is responsible for distributing the data across the data nodes, managing replication for redundancy and administrative tasks like adding, removing and recovery of data nodes.

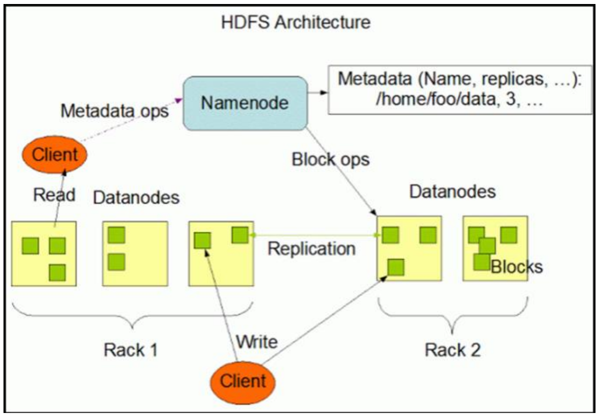


Figure 3: HDFS

Apache Hadoop can be useful across a range of use cases spanning virtually every vertical industry. It is becoming popular anywhere that you need to store, process, and analyze large volumes of data.

Examples include digital marketing automation, fraud detection and prevention, social network and relationship analysis, predictive modeling for new drugs, retail in-store behavior analysis, and mobile device location-based marketing. The MapReduce structure is shown in Figure 4 below.

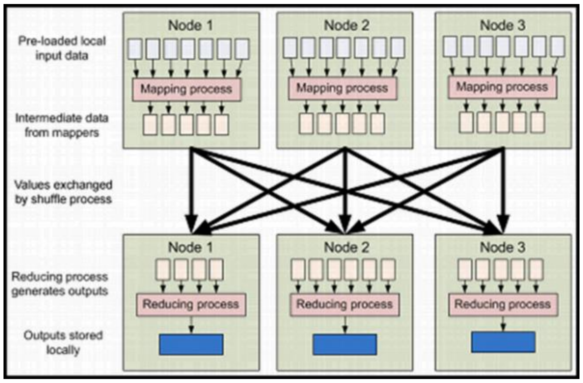
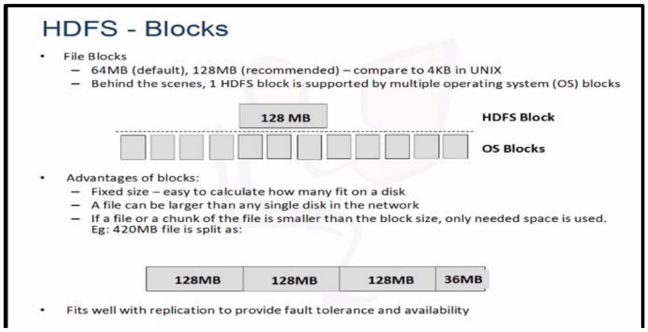


Figure 4: MapReduce

IV. WRITING FILES TO HDFS

The blocks are replicated to multiple nodes. The data is secure and safe even if there is any node failure. The HDFS block structure is shown in Figure 5 and 6.



HDFS - Blocks

- File Blocks
  - 64MB (default), 128MB (recommended) – compare to 4KB in UNIX
  - Behind the scenes, 1 HDFS block is supported by multiple operating system (OS) blocks

128 MB
HDFS Block
OS Blocks

- Advantages of blocks:
  - Fixed size – easy to calculate how many fit on a disk
  - A file can be larger than any single disk in the network
  - If a file or a chunk of the file is smaller than the block size, only needed space is used. Eg: 420MB file is split as:

- Fits well with replication to provide fault tolerance and availability

Figure 5: HDFS Block

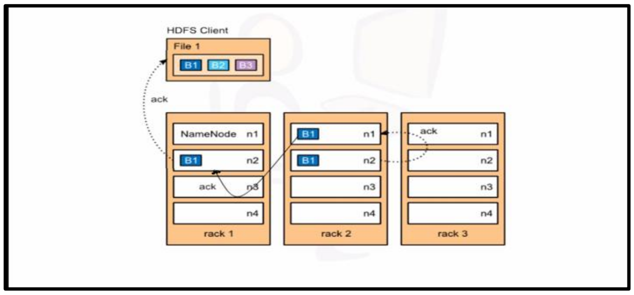
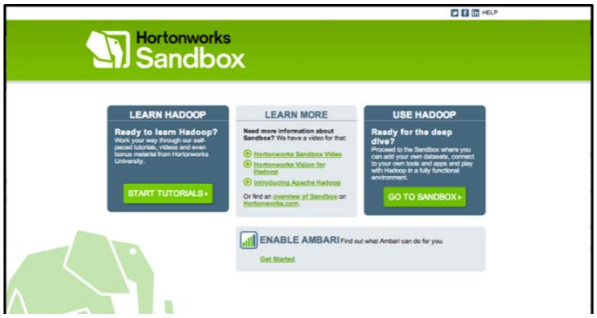


Figure 6: Writing File

V. THE SANDBOX

1. Once the virtual box is completely installed.
2. Go to local host 127.168.0.0:8888, to go to Hortonworks Sandbox and click on GO TO SANDBOX as shown in Figure 7



LEARN HADOOP

Ready to learn Hadoop? Work your way through our self-paced tutorials, videos and even bonus material from Hortonworks University.

START TUTORIALS ›

LEARN MORE

Need more information about Sandbox? We have a video for that:

- Hortonworks Sandbox Video
- Hortonworks Vision for Hadoop
- Introducing Apache Hadoop

Or find an overview of Sandbox on Hortonworks.com.

USE HADOOP

Ready for the deep dive? Proceed to the Sandbox where you can add your own datasets, connect to your own tools and apps and play with Hadoop in a fully functional environment.

GO TO SANDBOX ›

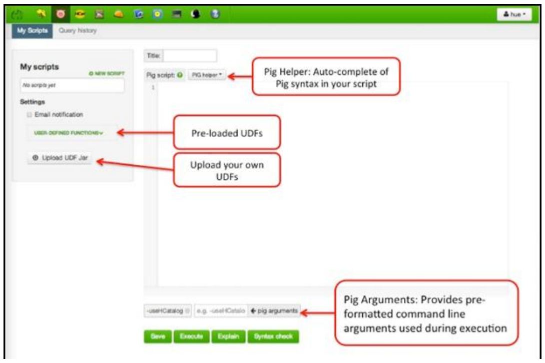
ENABLE AMBARI Find out what Ambari can do for you.

Get Started

Figure 7: Hortonworks Sandbox

VI. DATA PROCESSING WITH PIG

Pig is a high level scripting language that is used with Apache Hadoop. Pig excels at describing data analysis problems as data flows. Pig is complete in that you can do all the required data manipulations in Apache Hadoop with Pig. In addition through the User Defined Functions (UDF) facility in Pig you can have Pig invoke code in many languages like JRuby, Jython and Java. Conversely you can execute Pig scripts in other languages. The result is that you can use Pig as a component to build larger and more complex applications that tackle real business problems. The Pig structure is shown in Figure 8.Pig can ingest data from files, streams or other sources using the User Defined Functions (UDF).



Pig Helper: Auto-complete of Pig syntax in your script

Pre-loaded UDFs

Upload your own UDFs

Pig Arguments: Provides pre-formatted command line arguments used during execution

Figure 8: The PIG

VII. DATA PROCESSING WITH Jaql

The query language adopted by the IBM Big Data ecosystem for Java Script Object Notation (JSON). It supports both structured and non-structured queries and allows you to select, join, group, and filter data that is stored in HDFS.

1. Jaql's query language was inspired by other programming and query languages, including Lisp, SQL, XQuery, and Pig
2. Jaql is a functional, declarative query language that is designed to process large data sets.
3. For parallelism, Jaql rewrites high-level queries, as appropriate, into "low-level" queries consisting of MapReduce jobs.

The importance of Jaql is

1. The real beauty of the language is that it transparently exploits massive parallelism using Apache Hadoop's MapReduce processing.
2. Jaql code can be packaged in a way that fosters reuse. It is extensible, allowing functions written in other languages, for example Java, to be invoked from within your Jaql code.

VIII. CONCLUSION

Hadoop and the data warehouse will often work together in a single information supply chain. When it comes to Big Data, Hadoop excelsin handlingraw, unstructured and complex data with vast programming flexibility.

Data warehouses also manage big structured data, integrating subject areas and providing interactive performance through BI tools. It is rapidly becoming a symbiotic relationship. Some differences are clear and identifying workloads or data that runs best on one or the other will be dependent on your organization and use cases.

The future work holds as comparing the data working on sandbox and working with IBM info sphere on Linux. We plan to implement transferring of data between the two nodes of Hadoop on the same machine. The different nodes will be created on the same machine different drives.

REFERENCES

[1] Gaurav Vaswani,Hadoop – A Big Data Layout, Thinkquest, 2nd International Conference "Contours of Digital Technology", ICCDT-2015, Vol No 2 Issue No 2.

[2] Anuradha Bhatia, Gaurav Vaswani,BIG Data – A Review, International Journal of Engineering Sciences & Research Technology [2102-2106] Bhatia, 2(8): August, 2013],ISSN: 2277-9655.

[3] Gaurav Vaswani, Ajay Chotrani, Hitesh Rajpal, Sandbox- An Application Tool for Hadoop ,et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2) , 2014, 1255-1258.

[4] Anuradha Bhatia, Gaurav Vaswani, "Hadoop" An Open Source OS – A Review, International Journal of Computer Science Research & Technology (IJCSRT) ISSN: 2321-8827.

[5] Gaurav Vaswani, Anuradha Bhatia, "A Real Time Approach with BIG Data – A Review", Volume 3, Issue 9, September 2013 ISSN: 2277 128X International Journal of Advanced Research inComputer Science and Software Engineering.

[6] White, T., Hadoop: The Definitive Guide. O'Reilly Media, May 2009.

[7] Lin, J. and Dyer, C., Data-Intensive Text Processing with MapReduce. To be published by Morgan and Claypool. Pre-press edition available at http://www.umiacs. umd.edu/~ jimmylin/MapReduce-book-20100219.pdf, February 2010.

[8] The Hadoop Distributed File System: Architecture and Designand Implementations ed.). Amazon.

DOI: 10.15680/IJIRCCE.2015. 0310037