Big Data Landscape

Shubham Sharma

Banking Product Development Division, Oracle Financial Services Software Ltd.
Bachelor of Technology Information Technology, Maharishi Markandeshwar Engineering College

***Abstract*- "**Big Data" has become a major source of innovation across enterprises of all sizes .Data is being produced at an ever increasing rate. This growth in data production is driven by increased use of media, fast developing organizations, proliferation of web and systems connected to it. Having a lot of data is one thing, being able to store it, analyze it and visualize it in real time environment is a whole different ball game. New technologies are accumulating more data than ever; therefore many organizations are looking forward to optimal ways to make better use of their data. In a broader sense, organizations analyzing big data need to view data management, analysis, and decision-making in terms of "industrialized" flows and processes rather than discrete stocks of data or events. To handle these aspects of large quantities of data various open platforms had been developed.

***Index Terms*-** Big Data, Landscape,Open Platforms, Technologies,Tools

I. Introduction

In 2012 Gartner defined Big Data as follows "**Big Data are high volume, high velocity, and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization**". Using a big data platform allows one to address the full spectrum of big data challenges. These platforms make use of traditional technologies that are most suited for structured and repeatable task and incorporate them with complementary new technologies that address speed and flexibility and are ideal for unstructured analysis as well as data exploration and discovery.

Open platforms are software systems which have fully documented external application programming interface which allow the use of software in other ways than the original programmer intended without affecting the source code. Open platforms are based on open standards and does not mean they are open source. Big data open platforms are based on similar concepts and various platforms are discussed that provide visualization and discovery of large data sets, monitors big data systems and speeds time to value with analytical and industry specific modules.

Exquisite Example
"THE GOD PARTICLE"

An exquisite example of the enormous amount of data generator
is The Large Hadron Collider which represent about of 150 million sensors delivers 150 million petabytes annual rate or nearly 500 exabytes per day .To put the numbers in perspective this is equivalent to 5×10^{20} bytes per day. Almost 200 times higher than all the sources combined together in the world. To handle this huge chunk of data will be hard with the existing data management technologies. Hence the technology transitions have become imminent.

II. Technology Transition

With the introduction of Big Data platforms there has been a change in analytic techniques of organizations. The focus of the organizations has moved from orthodox methods like trend analysis and forecasting using historic data to its complementary and far better data visualization techniques. More interests had been shown towards scenario simulation and development over standardized reporting techniques. Analytics is emerging as a key to enhance business processes.

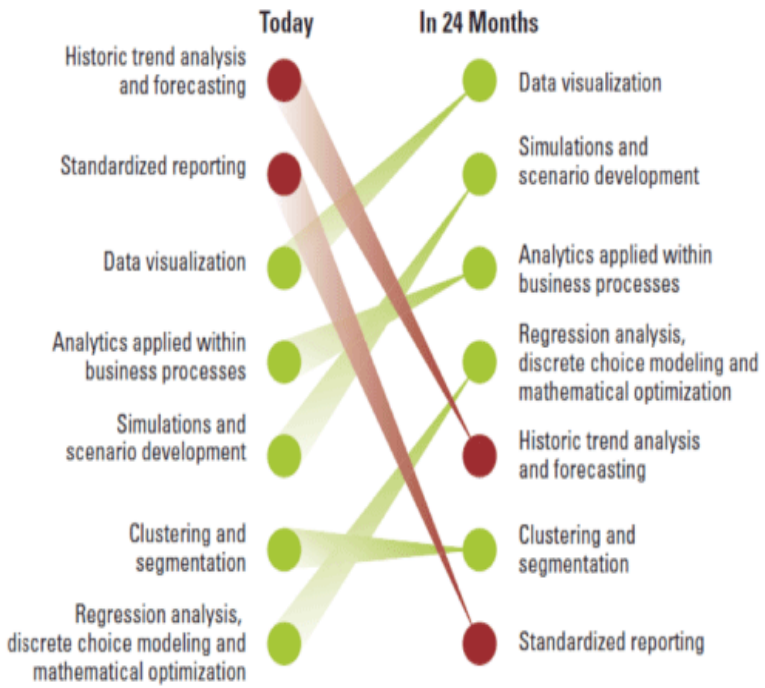

Figure1: Technology Transition

(Big data, Analytics and the Path from Insights to Value, MIT Sloan management review, Winter2011)

III. Classification of Big Data Tools

The Big Data tools landscape is growing rapidly and they can be classified majorly into following area:

1. Data Analysis
2. Databases/Data warehousing
3. Operational
4. Multi value Database
5. Business Intelligence
6. Data Mining
7. Key Value
8. Document Store
9. Graphs
10. Grid Solutions
11. Object Databases
12. Multi Model
13. XML databases
14. Big Data Search.

There are many products available for each classification, which have their own special features to meet the requirements.

Figure2: Big Data Landscape

IV. Big Data Landscape

In order to plan a big data architecture it is important to grasp the knowledge of the current big data landscape and incorporate it into existing infrastructure. In traditional data management structures, the structured information or data was fed into the enterprise integration tool which transferred the collected structured data into data warehouses or operational units. Then different analytical capabilities were used to reveal the data, but the new form of data management structures that inherit big data landscape are designed to meet the velocity, volume, value and variety of requirements. To handle these large data sets, new architectures have been formed that incorporate multi node parallel processing techniques.

Big data landscape has a further classification based on processing requirements and different strategies are proposed for

batch processing and real-time processing. Different technologies through which we can harness big data are :

1. Relational Database Management Systems
2. Massively Parallel Processing
3. MapReduce
4. NoSQL
5. Cassandra
6. Common Event Processing

Relational Database Management Systems

Databases are now using massively parallel processing techniques. These techniques are used to break data into small slots and to achieve faster processing operate them on multiple machines. Databases are acquiring columnar architecture to allow the storage of unstructured data.

Massively Parallel Processing

The data is distributed among a number of nodes for faster processing .The process is done parallel on each machine and the output is collected to deduce the required result. This technology requires knowledge of SQL and expensive hardware to work on.

MapReduce

Map reduce also use the concept of multi nodes and parallel processing .It consists of two function-

- Map - It separates information over multiple nodes which are then processed in parallel.
- Reduce - This function combines the result sets into a final response.
- Massively parallel processing uses SQL queries whereas MapReduce uses java and does not need expensive dedicated platforms.

NoSQL

NoSQL database-management systems are unlike relational database-management systems, in that they do not use SQL as their query language. The idea behind these systems is that that they are better for handling data that doesn't fit easily into tables. They dispense with the overhead of indexing, schema and ACID transactional properties to create large, replicated data stores for running analytics on inexpensive hardware, which is useful for dealing with unstructured data.

Hive

Databases like Hadoop's file store make ad hoc query and analysis difficult, as the programming map/reduce functions that are required can be difficult. Realizing this when working with Hadoop, Facebook created Hive, which converts SQL queries to map/reduce jobs to be executed using Hadoop.

Vendors

There is scarcely a vendor that doesn't have a big-data plan in train, with many companies combining their proprietary database products with the open-source Hadoop technology as their strategy to tackle velocity, variety and volume. Many of the early big-data technologies came out of open source, posing a threat to traditional IT vendors that have packaged their software and kept their intellectual property close to their chests. However, the open-source nature of the trend has also provided an opportunity for traditional IT vendors, because enterprise and government often find open-source tools off-putting.

Therefore, traditional vendors have welcomed Hadoop with open arms, packaging it in to their own proprietary systems so they can sell the result to enterprise as more comfortable and familiar packaged solutions.

Cloudera

Cloudera was founded in 2008 by employees who worked on Hadoop at Yahoo and Facebook. It contributes to the Hadoop open-source project, offering its own distribution of the software for free. It also sells a subscription-based, Hadoop-based distribution for the enterprise, which includes production support and tools to make it easier to run Hadoop.

Since its creation, various vendors have chosen Hadoop distribution for their own big-data products. In 2010, Teradata was one of the first to jump on the Cloudera bandwagon, with the two companies agreeing to connect the Hadoop distribution to Teradata's data warehouse so that customers could move information between the two. Around the same time, EMC made a similar arrangement for its Greenplum data warehouse. SGI and Dell signed agreements with Cloudera from the hardware side in 2011, while Oracle and IBM joined the party in 2012.

Hortonworks

Cloudera rival Hortonworks was birthed by key architects from the Yahoo Hadoop software engineering team. In June 2012, the company launched a high-availability version of Apache Hadoop, the Hortonworks Data Platform on which it collaborated with VMware, as the goal was to target companies deploying Hadoop on VMware's vSphere.

Teradata has also partnered with Hortonworks to create products that "help customers solve business problems in new and better ways".

Teradata

Teradata made its move out of the "old-world" data-warehouse space by buying Aster Data Systems and Aprimo in 2011. Teradata wanted Aster's ability to manage "a variety of diverse data that is not structured", such as web applications, sensor networks, social networks, genomics, video and photographs.

Teradata has now gone to market with the Aster Data nCluster, a database using MPP and MapReduce. Visualization and analysis is enabled through the Aster Data visual-development environment and suite of analytic modules. The Hadoop connecter, enabled by its agreement with Cloudera, allows for a transfer of information between nCluster and Hadoop.

Oracle

Oracle made its big-data appliance available earlier this year — a full rack of 18 Oracle Sun servers with 864GB of main memory; 216 CPU cores; 648TB of raw disk storage; 40Gbps InfiniBand connectivity between nodes and engineered systems; and 10Gbps Ethernet connectivity.

The system includes Cloudera's Apache Hadoop distribution and manager software, as well as an Oracle NoSQL database and

a distribution of R (an open-source statistical computing and graphics environment).

It integrates with Oracle's 11g database, with the idea being that customers can use Hadoop MapReduce to create optimised datasets to load and analyze in the database.

The appliance costs US$450,000, which puts it at the high end of big-data deployments, and not at the test and development end, according to analysts.

IBM

IBM combined Hadoop and its own patents to create IBM InfoSphere BigInsights and IBM InfoSphere Streams as the core technologies for its big-data push.

The BigInsights product, which enables the analysis of large-scale structured and unstructured data, "enhances" Hadoop to "withstand the demands of your enterprise", according to IBM. It adds administrative, workflow, provisioning and security features into the open-source distribution. Meanwhile, streams analysis has a more complex event-processing focus, allowing the continuous analysis of streaming data so that companies can respond to events.

IBM has partnered with Cloudera to integrate its Hadoop distribution and Cloudera manger with IBM BigInsights. Like Oracle's big-data product, IBM's BigInsights links to: IBM DB2, its Netezza data-warehouse appliance (its high-performance, massively parallel advanced analytic platform that can crunch petascale data volumes); its InfoSphere Warehouse; and its Smart Analytics System.

V. Platform Technology that Handles Big Data

VoltDB

VoltDB is a system consisting of a suitable format to a high-performance OLTP environment. The system is not memory-based data processing or SQL, but it performs sequential processing for data split based on stored procedure and reduces lock overhead with communication, helping to configure the high-speed OLTP system through horizontal split for table data.

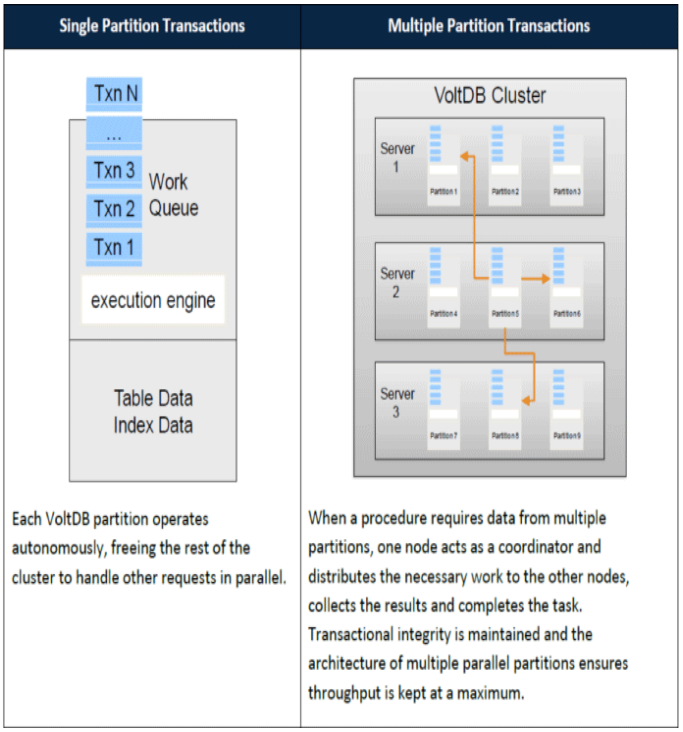

Figure 3: VoltDB architecture.

Figure 3 displays that a certain task that requires to operate in just one partition is executed sequentially in the corresponding partition, and that a certain task that needs to be handled in several partitions are processed by a coordinator. If there are many operations that need to be processed in several partitions, large rows and sizes may not be good.

SAP HANA

SAP HANA is a memory-based storage made from SAP. Its characteristic is to organize a system optimized to analysis tasks, such as OLAP. If all data is inside system memory, maximizing CPU utilization is crucial and the key point is to reduce bottlenecks between memory and CPU cache. In order to minimize Cache miss, consecutive data for processing within the given time is more advantageous, meaning that configuration of column-oriented tables could be favorable when analyzing many OLAP.

There are many advantages of the column-oriented table configuration and typical examples are a high data compression ratio and processing speed. In case of the same data domain, several data domains are better for data compression than when they are combined together. Moreover, the configuration enables reducing CPU operations through a lightweight compression, such as RLE (Run length encoding) or dictionary encoding or executing desired operations without a recovery process for compressed data. The following figure shows a brief comparison with Row-oriented and Column-oriented methods.

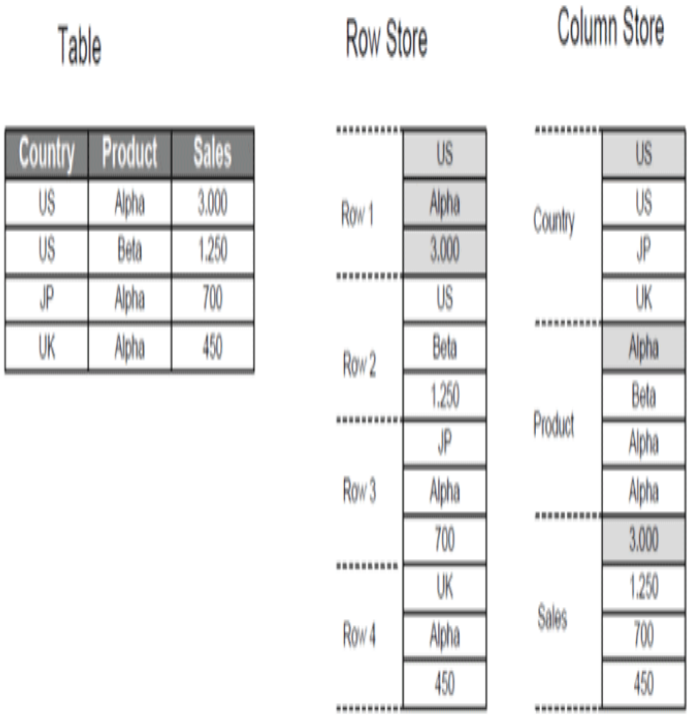

Figure 4: A Comparison with Row-oriented and Column-oriented methods.

Vertica

Vertica is database specialized for OLAP, which stores data on disk via the column method. The Shared-nothing-oriented MPP structure comprises a storage optimized for writing so as to load data fast, a reading storage in a compressed type, and tuple mover that manages bilateral data flow. Figure 5 below helps to understand the Vertica structure.

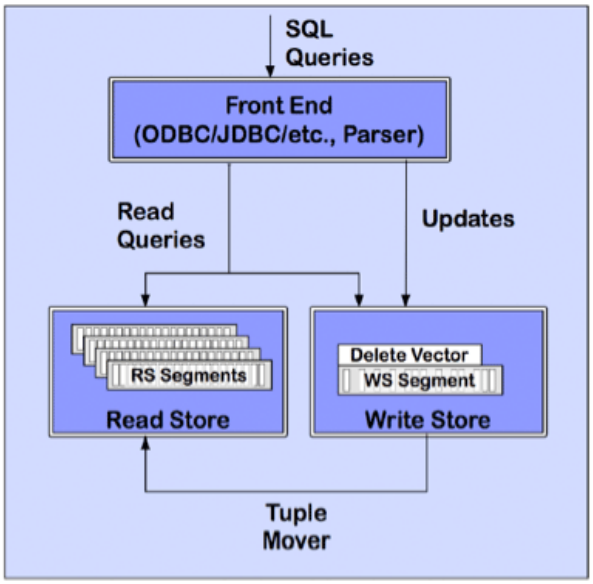

Figure 5: Vertica structure.

Greenplum

Greenplum database is a shared-nothing MPP structure, generated based on PostgreSQL. Data to be stored can select Row-oriented or Column-oriented methods accordingly to operations apply to the corresponding data. Data is stored in a server in segment and have availability because of segment unit replication of the log shipping method. A query engine, which was developed based on PostgreSQL, is configured to execute SQL basic operation (hash-join, hash-aggregation) or a map-reduce program so as to effectively process parallel query or map-reduced type programs. Each process node is connected to software-oriented data switch component.

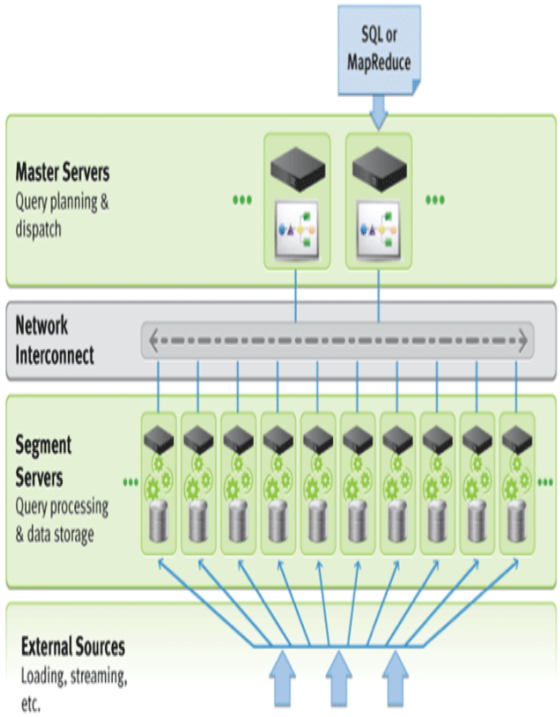

Figure 6: Greenplum architecture.

IBM Netezza Data Warehouse

IBM Netezza data warehouse has a two-tier type architecture consisted of SMP and MPP, called AMPP (Asymmetric Massively Parallel Processing).

A host with a SMP structure operates query execution plan and aggregation results, while S-blade nodes with a MPP structure handles query execution.

Each S-blade is connected by a special data processor called FPGA (Field Programmable Gate Array) and disk.

Each S-blade and host is connected to network that use IP addresses.

Unlike other systems, FPGA has filtering for data compression, record or column; in transaction processing, it enables filtering or transformation functions, such as visibility check during retrieving data from disk memory for real-time processing. When processing large-date, it adheres to the principles (processing close to the data source), which is to

reduce as much unnecessary data as possible from transmission by performing data operation where data is located.

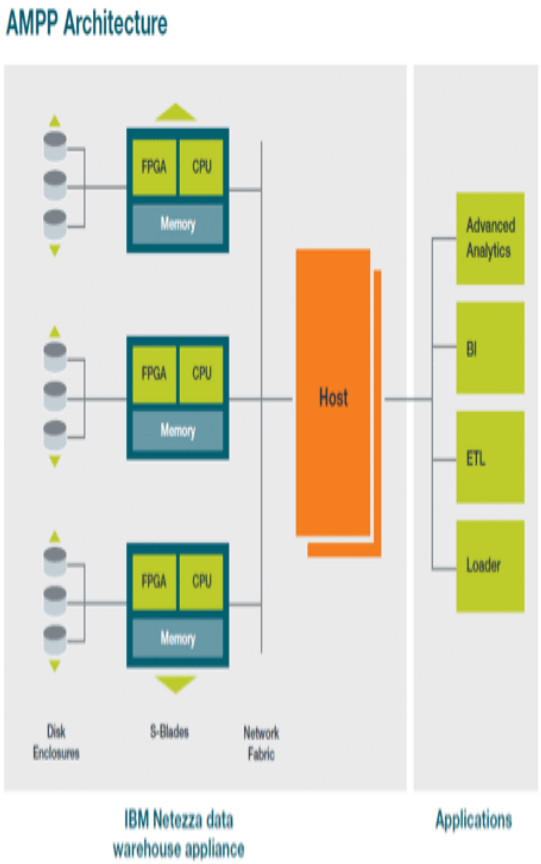

Figure 7: IBM Netezza data architecture.

In addition, companies and organizations that develop parallel DBMS are taken over by IT conglomerates and are in the progress of development in the appliance type. The names of conglomerates and date of acquisition of aforementioned parallel DBMS are shown in the following table:

The names of companies acquired	Database	Year
SAP	Sybase	2010
HP	Vertica	2011
IBM	Netezza	2010
Oracle	Essbase (Hyperian Solutions)	2007
Teradata	Aster Data	2011
EMC	Greenplum	2010

Table 1: Names of companies which acquired parallel RDBMS.

NoSQL

In RDBMS, scaling out while supporting ACID (Atomicity, Consistency, Isolation, and Durability) is almost impossible. For storage, data had to be divided into several devices; to be satisfied with ACID that has divided data, you have to use complicated locking and replication methods, which will lead to performance degradation.

NoSQL, a general term for a new storage system has emerged in order to simplify data models for easy definition of shard, which is the basic of distribution, and to make requirements less strict (Eventual Consistency) in a distribution replication environment or constraint isolation.

Since NoSQL is covered many times in our DevPlatform Blogs and there are many places to obtain information, we will not go over the NoSQL products.

Processing Aspects

The key point of parallel processing is Divide and Conquer. That is, data is divided in an independent type and process it in parallel. Just imagine the matrix multiplication that can divide and process each operation. The meaning of big data process is dividing a problem into several small operations, and combine them into a single result. If there is operation dependence, it is certainly impossible to make the best use of the parallel operation. It is necessary to save and process data considering these factors.

Map-Reduce

The most widely known technology that helps to handle large-data would be a distribution data process framework of the Map-Reduce method, such as Apache Hadoop.

Data processing via the Map-reduce method has the following characteristics:

It operates via regular computer that uses built-in hard disk, not a special storage. Each computer has extremely weak correlation where expansion can be hundreds and thousands of computers.

Since many computers are participating in processing, system errors and hardware errors are assumed as general circumstances, rather than exceptional.

With a simplified and abstracted basic operation of Map and Reduce, you can solve many complicated problems. Programmers who are not familiar with parallel programs can easily perform parallel processing for data.

It supports high throughput by using many computers.

The following figure displays the implementation flow of the map-reduce method. Data stored in the HDFS storage is divided to available worker and expressed (Map) a value type, and results are stored in a local disk. The data is complied by reducing worker and generate a result file.

Depending on the characteristics of a data storage, make the best use of locality by reducing the gap between a node which is processing data and source data location by placing worker in the location (based on network switch) where data is stored. Each worker can be implemented in various languages through streaming interface (standard in/out).

Apache Hive

Apache Hive helps to analyze large data by using the query language called HiveQL for data source, such as HDFS or HBase. Architecture is divided into Map-Reduce-oriented execution, meta data information for a data storage, and an execution part that receives a query from user or applications for execution.

To support expansion by user, it allows user specified function at the scalar value, aggregation, and table level.

Analysis Aspects

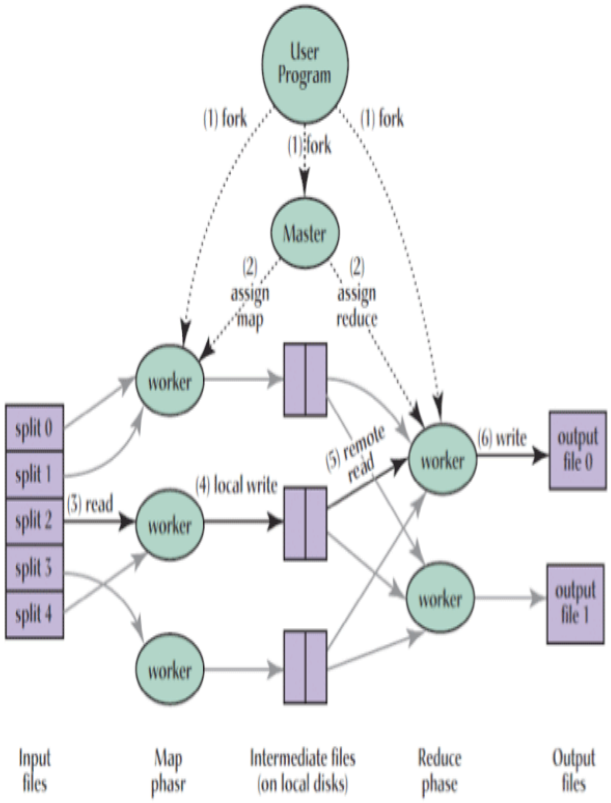

Figure 8: Map-Reduce execution.

We reviewed systems that store big data and procedural/declarative technologies that display processing and how to process large data. Finally, let us look into technology that analyzes big data.

The process of finding meaning in data is called KDD (Knowledge Discovery in Databases). KDD is to store data, process/analyze the whole or part of interested data in order to extract progress or meaning value, or discover facts that were so far unknown and make them into knowledge ultimately. For this, various technologies are comprehensively applied, such as artificial intelligence, machine learning, statistics, and database.

GNU R

GNU R is a software environment comprising program languages specialized for statistics analysis and graphics (visualization) and packages. It ensures a smooth process of vector and matrix data so as to be optimized for statistical calculations in terms of language. You can easily acquire desired statistics process library because of the R package site known as CRAN (Comprehensive R Archive Network). It can be touted as an open source in the field of statistics.

In the past, R used to put data to be processed into the memory of a computer for analyzing using a single CPU. There has been much progress due to ever increasing data to be processed

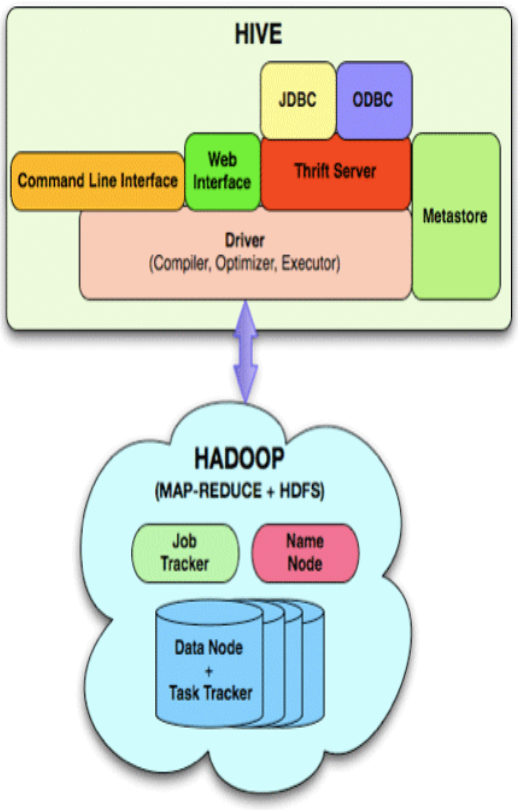

Figure 11: HIVE architecture.

REFERENCES

[1] Steve LaValle, Eric Lesser, Rebecca Shockley, Michael S. Hopkins and Nina Kruschwitz (December 21, 2010), "Big data, Analytics and the Path from Insights to Value".

[2] James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, Angela Hung Byers (May 2011), "Big data: The next frontier for innovation, competition, and productivity".

[3] Richard Winter (December 2011), "BIG DATA: BUSINESS OPPORTUNITIES, REQUIREMENTS AND ORACLE'S APPROACH" (PDF).

[4] Michael Stonebraker, Nabil Hachem, Pat Helland, "The End of an Architectural Era (It's Time for a Complete Rewrite)", VLDB 2007 (PDF).

[5] Michael Stonebraker et al., "One Size Fits All? – Part 2: Benchmarking Results", CIDR 2007 (PDF).

[6] Daniel J. Abadi et al., "Integrating Compression and Execution in Column-Oriented Database Systems", SIGMOD '06 (PDF).

[7] "VoltDB | Lightning Fast, Rock Solid".

[8] "SAP HANA".

[9] "Real-Time Analytics Platform | Big Data Analytics | MPP Data Warehouse".

[10] "Greenplum is driving the future of Big Data analytics".

[11] "Data Warehouse Appliance, Data Warehouse Appliances, and Data Warehousing from Netezza".

[12] Jeffrey Dean and Sanjay Chemawat, "MapReduce: Simplified Data Processing On Large Clusters", CACM Jan. 2008 (PDF).

[13] Mihai Budiu (March 2008), "Cluster Computing with Dryad", MSR-SVC LiveLabs (PPT)

[14] Hung-chih Yang et al., "Map-Reduce-Merge: Simplified Relational Data Processing on Large Clusters", SIGMOD '07 (PPT).

[15] "Welcome to Apache Hadoop".

[16] "The R Project for Statistical Computing".

AUTHORS

First Author – Shubham Sharma, Bachelor of Technology Information Technology, Maharishi Markandeshwar Engineering College, Associate Consultant, Banking Products Development, Oracle Financial Services Software Ltd. , shubhhbk.sharma@gmail.com.