Advances in Computer Science and Information Technology (ACSIT)
p-ISSN: 2393-9907; e-ISSN: 2393-9915; Volume 2, Issue 13; July-September, 2015 pp. 9-14
© Krishi Sanskriti Publications
http://www.krishisanskriti.org/Publication.html

SQL-on-Hadoop: The Most Probable Future in Big Data Analytics

Tererai Tinashe Maposa[1] and Manoj Sethi[2]

[1,2]Delhi Technological University
E-mail: [1]tereraimaposa@gmail.com, [2]manojsethi@dce.ac.in

Abstract—*It is now a well known fact that big data is here to stay. Big organizations and governments are investing millions in big data analytics. No one wants to be left out in this hype for big data. Traditional analytical methods and technologies have fallen short as they cannot scale to the required magnitude. New technologies are being invented by the hour so as to assist organizations in their quest for domination. Hadoop has been on the forefront in being the storage and processing framework for big data. Hadoop has become synonymous with big data; it is the de facto big data technology platform. However, Hadoop is not "the hammer for every nail"; it has limitations. These limitations have hampered the much needed progress in big data analytics. Consequently the scientific community has decided to come up with hybrid platforms that combine the desirable features of the tried and tested DBMS technologies and the new Hadoop. The result of this combination is called SQL-on-Hadoop. The thrust behind this survey paper is to shade more light on SQL-on-Hadoop and highlight their pros and expose the limitations of Hadoop.*

Keywords*: SQL-on-Hadoop, Big Data, Data Analytics*

1. INTRODUCTION

As the world becomes more instrumented, the volumes of data available to the enterprise are growing by orders of magnitude. Those data volumes often hold critical insight for organizations – if only it can be efficiently analyzed, which is no easy task. Big data has truly transformed the face of data management and the requirements of the needed technologies. Traditional database systems and warehouse have not kept the pace. They have become more expensive and rigid. The development of Hadoop has been a welcomed idea.

Hadoop's ability to distribute data and tasks over commodity hardware made it more appealing to the user community. Users can achieve more with less. Additionally Hadoop is open-source its procurement process is much simpler and there is no need for upfront capital expenditure due to the absence of an initial software license. This makes Hadoop a very cost effective solution for data analysis. Hadoop is also fault-tolerant because of replication of data across nodes. This increases the availability of data. The lack of a schema also makes Hadoop flexible. Hadoop can take up anything (unstructured, semi-structured and structured data) because there is no predefine schema. A schema is only imposed until the data is accessed (known as "schema on read") [1].

However, in as much as Hadoop has provided answers to some of the problems that the user community was facing, it is still not "a saint". It has a number of limitations, some of which can be best solved by traditional databases systems. This survey paper aims to expose these limitations and outline how these limitations are being mitigated by the introduction of SQL-on-Hadoop tools.

The rest of the paper is organized as follows: In Section 2, we look at how Hadoop has been accepted by the data analytics community by explaining its technological adoption lifecycle. Then outline Hadoop's limitations in Section 3. In Section 4 we discuss the current dilemma, that of having two excellent technologies which partially fulfill the needs of users. In Section 5 we describe SQL-on-Hadoop technology and highlight the benefits they provide and finally in Section 6 we briefly explain five prominent SQL-on-Hadoop tools.

2. TECHNOLOGICAL ADOPTION LIFECYLE FOR HADOOP

The technology adoption lifecycle model describes the adoption or acceptance of a new product or innovation, according to the demographic and psychological characteristics of defined adopter groups [2]. It provides insight into the current market conditions for a technology and a glimpse into the future. This is achieved by understanding the types of buying personalities, and the percentage of the population they represent, as a technology matures and progresses through its adoption life cycle [3].

According to a report by the Wall Street Journal [4], Wall Street is paying close attention to technologies that help companies derive value from big data. Hadoop is on top of the list of these technologies. Another report by Hortonworks [5] shows that the corporate world has positively responded to Hadoop. The report states that **26%** of respondents of Hadoop have deployed, piloting or experimenting; **11%** plan to invest within a year; and an additional **7%**plan to invest within two years. This is arguably a remarkable response for a new technology. Below is the adoption life cycle curve according to Hortonworks.

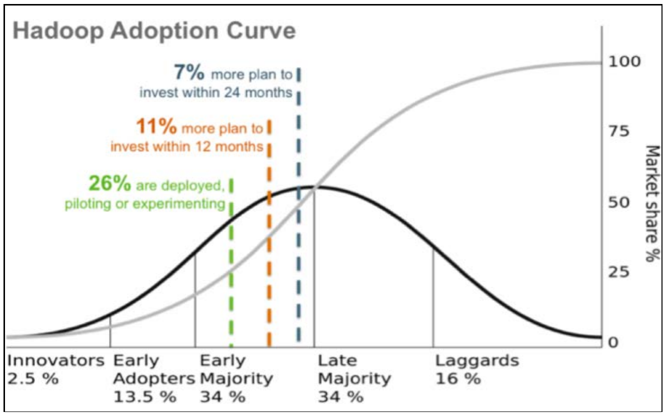


Fig. 1: Hadoop Adoption Curve [5]

This curve shows that the Hadoop is building momentum in the industry. It also shows that Hadoop is going to be at the centre of big data for the foreseeable future.

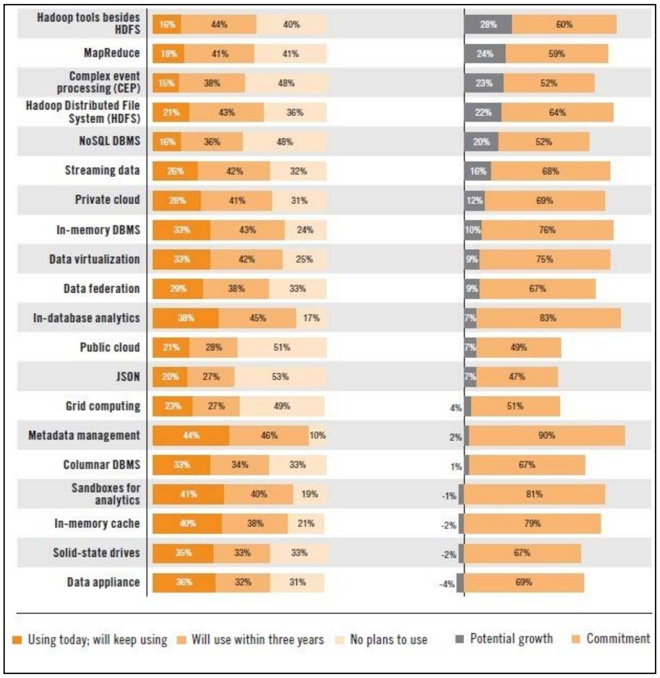


Fig. 2: Technologies and tools currently in use and the most probable ones in future [6]

A survey [6] by the TDIW research institute also confirms the fact that Hadoop is going to dominate big data for a long time. Fig. 2 below is the graph that shows the tools and technologies that organizations are currently using and those they plan to use in future. Fig. 2 is sorted by the "potential growth" column, in descending order. "Hadoop tools besides HDFS" appears at the top of the chart. Hadoop tools have a growth projection of 28% (greater than other options); a strong indicator that they exhibits the highest potential for growth among the options listed.

In the "commitment" column, we also see that 60% of survey respondents have committed to implementing Hadoop tools, whether today or within three years. The prediction deduced from these data points is that future Big Data Management solutions will dramatically increase their usage of Hadoop tools [6].

3. LIMITATIONS OF HADOOP

Of course the future of Hadoop is bright but it has its limitations. These limitations have made some of the users skeptical and hesitant to adopt it in big data analytics. Some of Hadoop's limitations emanate from its strengths and these are:

i. **It is a file system not a database**. This implies three major setbacks;
  - Firstly, there is no notion of transaction consistency or recovery checkpoints. This means that the answer you get from a Hadoop cluster may or may not be 100% accurate, depending on the nature of the job [7].
  - Secondly, Hadoop does not provide easy access to individual records or record sets that are a small subset of the total data [8]. In order to do analysis or exploration of a file in Hadoop, the whole file must be read for every computational process because by its nature there is no predefined data schema or indexes in Hadoop [9]. Hence it is not suitable for interactive, ad-hoc queries required for many applications but for batch processing that aggregates or processes most, if not all, of a massive data set.
  - Thirdly, there is no way to change the data in the files stored in HDFS. There is no such thing as an update or delete function in Hadoop as there is in a database-management system, and no concept of commit data or roll-back data as in a transactional processing database system. Hadoop is a "write-once, read-many" affair [9]. So imagine if you have 100 TB of data and there are a few changes to it. This means that all of it must be reloaded into the HDFS for the data to remain relevant to the organization. This process can be very time consuming hence make Hadoop a poor solution for companies whose data constantly changes.

ii. **Poor performance**; this is as a result of its design.
  - Hadoop was not designed to be efficient. It replicates data which is already big. Each chunk has to be replicated at least three times. This implies that organizations will require at least three times the size of what they really should have.
  - HDFS has no notion of a query optimizer, so cannot pick an efficient cost-based plan for execution. Because of this, Hadoop clusters are generally

significantly larger than would be required for a similar database [9].

iii. **No quality assurance in Hadoop because it is open-source**.

- Big organizations need some accountability on the products that they use in case something goes wrong. Hadoop is open-source which is well known for being highly variable in quality. There is no incentive for software suitability or quality in open-source development. Open-source development is usually an experimental playground for software authors trying out their skills. Due to the unpredictable nature of open-source development, quality assurance is difficult or impossible. The end-result is that only some needs are met, and when they are, it is with a solution of unpredictable usability and quality [7]. This makes Hadoop a high risk tool for organizations that wants to implement Hadoop at an enterprise level.

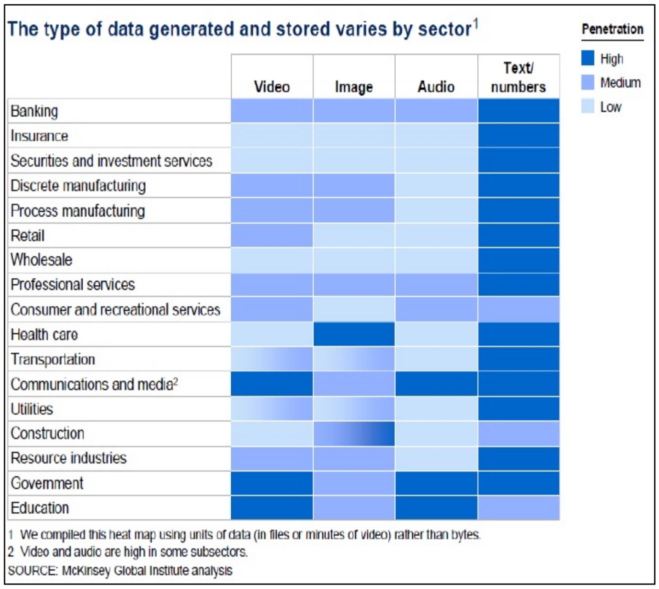
iv. **Lack of Hadoop skills and the steep learning curve**

- Because Hadoop requires complex, specialized MapReduce programs, typically written in Java, to manage its data; programmers with these skills are rare. Hadoop at lowest levels is API based and is difficult to master programming skills at this level because even a simple task can be tedious. This disrupts business continuity. Business needs to continue and cannot be stopped whilst the staff is learning a new technology.

4. THE CURRENT DELIMMA

While Hadoop has been received with enthusiasm and has solved some problematic issues (such as scalability and flexibility) that were overwhelming conventional DBMSs, there are still some areas that these traditional systems out-perform Hadoop (such as performance and user-friendliness). DBMSs benefit from several years of operation; every organization seems to have implemented them. They have been tried and tested, hence they have matured and everyone is comfortable with them. Although DBMSs are now failing to provide all the needed solutions, it is not easy to abruptly replace them with Hadoop because of the following main reasons:

i. Traditional DBMSs still have a lot to offer to the corporate world. Most organizations still have structured data as the dominant type of data. Three separate survey researches [6, 10, 11] discovered that between 85-88% of organizational data units is structured; there is more structured data than unstructured data in terms of units. However unstructured data is more in terms of volume because for instance one video file can be 1GB but for one to have 1GB of transactional data, one needs to have captured millions of records.



1 We compiled this heat map using units of data (in files or minutes of video) rather than bytes.
2 Video and audio are high in some subsectors.
SOURCE: McKinsey Global Institute analysis

Fig. 3: Heat Map for types of data generated by various sectors [11]. Approximately 85% of all sectors still generate more units of structured data (text/numbers) than unstructured.

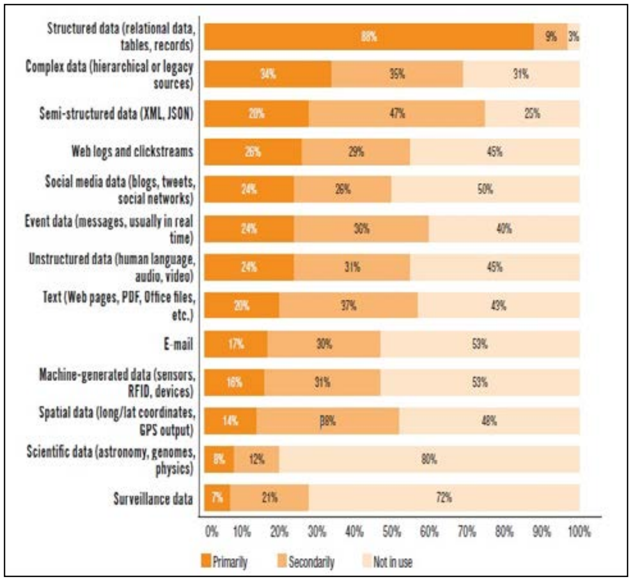


Fig. 4: Types of data constituting big data. At 88%, structured data is by far the most managed data type today, according to the survey [6].

ii. Hadoop is still a “new baby” and still evolving. It is too risky to replace traditional DBMSs with Hadoop at the moment because Hadoop has not matured enough to be adopted into the corporate world. Additionally to its immaturity, Hadoop cannot do everything and has some costly limitations as we have highlighted above.

iii. Cristian Molaro in the IBM Data Magazine [12] claims that the structured data in organizations is the one that matters the most because most of the unstructured data are “cutie selfies” which in most cases have no business value. Most organizations, decision makers, and business analysts and other information consumers, can derive better business intelligence from information derived from big data when it is organized in a defined and structured manner rather than an unstructured one.

These reasons indicate a strong need for traditional DBMSs even if they also have their limitations.

5. SQL-ON-HADOOP - “UNITY IS POWER”

SQL-on-Hadoop is a new class of analytical application tools that combine well known SQL-style querying with newer Hadoop data framework elements. By supporting established SQL, SQL-on-Hadoop allows a wider range of developers as well as business analysts to be comfortable around the “dreaded” Hadoop. SQL-on-Hadoop tools inherit merits from both worlds and provide the users with a hybrid solution. The limitations of both Hadoop and traditional DBMS are addressed in these hybrid tools because they complement each other.

Since SQL was originally intended for relational databases, it has to be modified to work with Hadoop. There are three distinct techniques in which SQL can be executed on Hadoop. (It is beyond the scope of this paper to discuss these techniques in depth.)

i. Connectors that translate SQL into a MapReduce format;
ii. "push down" systems that forgo batch-oriented MapReduce and execute SQL within Hadoop clusters; and
iii. Systems that apportion SQL work between MapReduce-HDFS clusters or raw HDFS clusters, depending on the workload.

The introduction of SQL on Hadoop opens a new door of opportunities as it combines two great inventions. Benefits of implementing SQL-on-Hadoop include:

i. **Business Continuity:** Existing BI tools most of which use SQL can be integrated with SQL-on-Hadoop tools with relative ease. No major rewrites are required for the existing tools to be productive [8]. Business will not have to be disrupted whilst organizations dismantle existing solutions and set up new platforms.
ii. **Productivity:** Since most analysts can “speak” SQL, enterprises can use their existing human resources and will not need to hire new skilled programmers or retrain their current staff, hence ensuring business continuity. New personnel may need to get accustomed to organizational cultures before they can become productive. Most organizations do not have this luxury [6].
iii. **Standard compliance:** SQL has been around for quite some time and as such has matured. It allows analysts to enforce known standards in data analytics unlike Hadoop which is still in its infancy stages. Standards are necessary for quality assurance. Also standard user-friendly interfaces can be made available to the less technically gifted personnel thereby increasing productivity.
iv. **Interactive queries:** Response time in Hadoop is not suitable for ad-hoc interactive queries; however the introduction of an SQL layer improves the response time which is crucial to promote data exploration, rapid prototyping and other tasks. (Not all SQL-on-Hadoop tools have desirable response time.)
v. **Scalability:** It is apparent that data volumes will keep increasing therefore organizations require a solution that is “future proof”; one that is scalable. With Hadoop as the data repository, SQL-on-Hadoop tools can easily scale.
vi. **Flexibility:** The flexibility that lacked in traditional systems which are intended only for structured data is now available in SQL-on-Hadoop tools. This flexibility is made possible by the use of HDFS as the central repository (aka, Data Lake). Data from diverse data sources; both structured and unstructured can now be accommodated [13]. This flexibility has also eased the data loading process by lessening the demands imposed on the ETL process. Traditional systems require careful data filtration and aggregation due their predefined structure; however this is no longer the case in SQL-on-Hadoop tools since HDFS can take up any kind of data; both clean/complete and dirty/noisy data.

6. EXAMPLES OF SQL-ON-HADOOP TOOLS

In this paper we cannot discuss each and every SQL-on-Hadoop in existence. We will however briefly outline only five we think are the most prominent ones.

i. Apache Hive [14]

Hive is the first SQL-on-Hadoop. It is considered one of the de-facto tools and is installed on almost all Hadoop installations. It is an open-source Java project which translates SQL to a series of Map-Reduce jobs which execute on standard Hadoop task trackers. It largely supports MySQL syntax and categorizes datasets using familiar database/table/view conventions. Hive provides an SQL-like query interface called Hive-QL, which roughly mimics MySQL.It allows metadata distribution via a central service and also provides JDBC drivers.

Queries performed with Hive are usually very slow because of the bottleneck associated with using Map-Reduce. Apache Tez a new back-end for Hive which provides fast response times that is currently unachievable using Map Reduce is being developed by Hortonworks.

ii. Cloudera Impala [15, 16]

It is a product of Cloudera, one of the market's dominant distributors of Hadoop. Impala is an open-source 'interactive' SQL query engine for Hadoop. The interactive SQL it provides is 4-35 X faster than Hive. It provides a way to write SQL queries against your existing Hadoop data. It is different from Hive in that it uses its own daemons to execute the queries instead of Map-Reduce. These daemons have to be installed alongside data nodes. Impala supports HIVE-QL support (ANSI---92 standard SQL queries with HiveQL) and ODBC drivers

iii. Presto [17]

Presto was built by Facebook and open-sourced in 2013. Presto is an 'interactive' SQL query engine for Hadoop written in Java. Presto has the ability to query data without moving it from its original residence, such as from Hive, relational databases, Cassandra, or even proprietary data stores. A single Presto query can combine data from multiple sources, allowing for analytics across your entire organization. Presto is targeted at analysts who expect response times ranging from sub-second to minutes. Facebook uses Presto for interactive queries against several internal data stores, including their 300PB data warehouse. Other big organizations such as Dropbox and Airbnb also use Presto in their daily operations.

iv. IBM BigSQL [18]

BigSQL is a product by IBM which has its own Hadoop Distribution called BigInsights. BigSQL enables IT professionals to create tables and query data in BigInsights using well-known SQL statements. Programmers use both standard SQL syntax as well as SQL extensions created by IBM to make it easy to harness certain Hadoop-based technologies. It supports both JDBC and ODBC client access from Linux and Windows platforms. The LOAD command in BigSQL can read data directly from several relational DBMS systems as well as from files stored locally or within the BigInsights distributed file system. The SQL query engine supports joins, unions, grouping, common table expressions, windowing functions, and other familiar SQL expressions.

v. HAWQ [13,19]

HAWQ is a parallel SQL query engine that amalgamates the key technological advantages of the industry-leading Pivotal Analytic Database with the scalability and convenience of Hadoop [19]. HAWQ reads data from and writes data to HDFS natively. HAWQ delivers industry-leading performance and linear scalability. HAWQ provides users with a complete, standards compliant SQL interface. HAWQ has been designed from the ground up to be a massively parallel SQL processing engine optimized specifically for analytics with full transaction support. HAWQ breaks complex queries into small tasks and distributes them to query processing units for execution.

Apart from these five we have mentioned, there are other countless SQL-on-Hadoop tools. Some are open-source whilst others have been licensed. More examples of the these tools include Apache Drill, HBase, Shark, SparkSQL, Hadapt, Apache Tajo, Apache Phoenix, Stinger and Oracle BigData SQL. Most are still under development and are being improved regularly.

7. CONCLUSION

The creation of SQL-on-Hadoop provides the data analytics community with a glimpse of what the future holds. SQL-on-Hadoop is a combination of arguably the two biggest inventions in data management. SQL is the dominant and preferred language for data processing due to its maturity and user-friendliness. On the other hand Hadoop has solved some of the most troublesome issues and as such has won the hearts of many and is set to be the central/key ingredient for future platforms. Their partnership is therefore an attractive one and is likely to be the probable choice for most organizations in the foreseeable future. The potential of these SQL-on-Hadoop tools has also attracted the attention of industry's biggest organizations and most of them have invested millions in the development of SQL-on-Hadoop tools; such as IBM's BigSQL, Presto by Facebook, Dremel (Apache Drill) by Google, HAWQ by Pivotal, Oracle BigData SQL by Oracle, Apache Tez by Hortonworks and Impala by Cloudera. This is a strong indicator of where the future is inclined.

REFERENCES

[1] Jean-Pierre Dijcks and Martin Gubar, Oracle Inc, Business Intelligence Journal, *"Integrating SQL and Hadoop"*, 2014.

[2] Geoffrey Moore, *"Inside the Tornado"*, Harper Collins Publishing, 1995.

[3] Wohlers Associates, INC, *"Rapid Prototyping & Tooling State of the Industry"*, 2002.

[4] *http://blogs.wsj.com/cio/2015/03/31/corporate-hadoop-adoption-is-growing-barclays-report-says/*

[5] *http://hortonworks.com/blog/enterprise-hadoop-adoption-half-empty-or-half-full/*

[6] Philip Russom, *"Managing Big Data"*, TDWI research.

[7] Paraccel Inc Whitepaper, *"Hadoop's Limitations for Big Data Analytics"*, 2012.

[8] Splice Machine Whitepaper, *"Splice Machine: SQL-on-Hadoop Evaluation Guide"*, 2013.

[9] Bob Palmer, Senior Director, SAP National Security Services *"Hadoop: Strengths and Limitations in National Security Missions"*, 2012.

[10] Cynthia M. Saracco, IBM Silicon Valley Lab, *"Big SQL: A Technical Introduction"*, 2014.

[11] Marko Grobelnik, Jozef Stefan Institute, Ljubljana, Slovenia, *"Big Data Tutorial"*, May 2012.

[12] *http://ibmdatamag.com/*.

[13] Lei Chang, Zhanwei Wang, Tao Ma, Lirong Jian, Lili Ma, Alon Goldshuv Luke Lonergan, Jeffrey Cohen, Caleb Welton, Gavin Sherry, Milind Bhandarkar, Pivotal Inc, *"HAWQ: A Massively Parallel Processing SQL Engine in Hadoop"*, 2014.

[14] *https://hive.apache.org/*

[15] Justin Erickson, Senior Product Manager, Cloudera *"Cloudera Impala: A Modern SQL Engine for Hadoop"*, 2014.

[16] John Russell, O'Reily e-E-Book, *"Cloudera Impala"*.

[17] *https://prestodb.io/*

[18] Scott C. Gray, Fatma Ozcan, Hebert Pereyra, Bert van der Linden and Adriana Zubiri, IBM Software Group, *"SQL-on-Hadoop without compromise"*, 2014.

[19] Pivotal HD, Whitepaper, *"HAWQ: A True SQL Engine for Hadoop"*