A SURVEY ON HARDWARE PLATFORMS FOR BIG DATA ANALYTICS

Dr. Rajendra Ganpatrao Sabale, Prof. Pravin P. Kulkarni

1. Ass. Professor, New Arts, Commerce Science College, Ahmednagar

Abstract:
The Aim of this paper is to provide an in-depth analysis of different platforms available for performing big data analytics. We carried out different hardware platforms available for big data analytics. Also, we try to assess the advantages and drawbacks of each of these platforms based on various metrics such as real-time processing, data size supported, scalability, data I/O rate, fault tolerance. Some of the critical characteristics described here can potentially aid the readers in making an informed decision about the right choice of platforms depending on their computational needs.

Keywords: Big data, unit’s scalability, big data analytics, big data platforms.

Introduction:
1. Big Data:
Today current topics to manage data and related to handle data is big data and this is an era of Big Data. Big Data is driving radical changes in traditional data analysis platforms. To perform any kind of analysis on such complex data, scaling up the hardware platforms becomes imminent and choosing the right hardware/software platforms becomes a crucial decision if the user’s requirements are to be satisfied in a reasonable amount of time. We try to research on analysis techniques for big data more than ever before which has led to the continuous development of many different algorithms and platforms.

Today’s day there are several big data platforms available with different characteristics and choosing the right platform requires an in-depth knowledge about the capabilities of all these platforms [1]. Especially, the ability of the platform to adapt to increased data processing demands plays a critical role in deciding if it is appropriate to build the analytics based solutions on a particular platform. We will first provide a thorough understanding of all the popular big data platforms that are currently being used in practice and highlight the advantages and drawbacks of each of them.

While study one will come across a few fundamental issues in their mind before making the right decisions. How big is the data to be processed? Does the model building require several iterations or a single iteration? How quickly do we need to get the results?

Clearly, these concerns are application/algorithm dependent that one needs to address before analysing the systems/platform-level requirements. At the systems level, one has to meticulously look into the following concerns:
Will there be a need for more data processing capability in the future? Is the rate of data transfer critical for this application? Is there a need for handling hardware failures within the application?
In this paper, we will provide a more rigorous analysis of these concerns and provide a score for each of the big data platforms with respect to these issues.

While there are several works that partly describe some of the above-mentioned concerns, to the best of our knowledge, there is no existing work that compares different platforms based on these essential components of big data analytics. Our work primarily aims at characterizing these concerns and focuses on comparing all the platforms based on these various optimal characteristics, thus
providing some guidelines about the suitability of different platforms for various kinds of scenarios that arise while performing big data analytics in practice.

The major contributions of this paper are as: Illustrate the scaling of various big data analytics platforms and demonstrate the advantages and drawbacks of each of these platforms including the software frameworks.

Provide a systematic evaluation of various big data platforms based on important characteristics that are pertinent to big data analytics in order to aid the users with a better understanding about the suitability of these platforms for different problem scenarios.

2. Scaling:

Scaling is the ability of the system to adapt to increased demands in terms of data processing. To support big data processing, we have different platforms incorporate scaling in different forms. From a broader perspective, the big data platforms can differentiated into the following two types of scaling:

i) Vertical Scaling: Vertical Scaling involves installing more processors, more memory and faster hardware, typically, within a single server. It is also known as “scale up” and it usually involves a single instance of an operating system.

ii) Horizontal Scaling: Horizontal scaling involves distributing the workload across many servers which may be even commodity machines. It is also known as “scale out”, where multiple independent machines are added together in order to improve the processing capability. Typically, multiple instances of the operating system are running on separate machines.

Table 1: Comparisons, the advantages and drawbacks of horizontal and vertical scaling:

<table>
<thead>
<tr>
<th>Scaling</th>
<th>Advantages</th>
<th>Drawbacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal scaling</td>
<td>➔ Can scale out the system as much as needed</td>
<td>➔ Limited number of software are available that can take advantage of horizontal scaling</td>
</tr>
<tr>
<td></td>
<td>➔ Financial investment to upgrade is relatively less</td>
<td>➔ Software has to handle all the data distribution and parallel processing complexities</td>
</tr>
<tr>
<td></td>
<td>➔ Increases performance in small steps as needed</td>
<td>➔ Requires substantial financial investment</td>
</tr>
<tr>
<td>Vertical scaling</td>
<td>➔ Most of the software can easily take advantage of vertical scaling</td>
<td></td>
</tr>
</tbody>
</table>

Horizontal scale out gives users the ability to increase the performance in small increments which lowers the financial investment. Also, there is no limit on the amount of scaling that can done and one can horizontally scale out the system as much as needed. In spite of these advantages, the main drawback is the limited availability of software frameworks that can effectively utilize horizontal scaling.

3. Horizontal scaling platforms:

Some of the prominent horizontal scale out platforms include peer-to-peer networks and Apache Hadoop. Recently, researchers have also been working on developing the next generation of horizontal scale out tools such as Spark [2] to overcome the limitations of other platforms. We will now discuss each of these platforms in more detail in this section.
3.1 Peer-to-peer networks:

Peer-to-Peer networks [3] involve millions of machines connected in a network. It is a decentralized and distributed network architecture where the nodes in the networks (known as peers) serve as well as consume resources. It is one of the oldest distributed computing platforms in existence. Usually Message Passing Interface (MPI) is the communication scheme used in such a setup to communicate and exchange the data between peers. The major bottleneck in such a setup arises in the communication between different nodes. Broadcasting messages in a peer-to-peer network is cheaper but the aggregation of data/results is much more expensive. In addition, the messages are sent over the network in the form of a spanning tree with an arbitrary node as the root where the broadcasting is initiated.

4. Comparison of different Platforms:

We will now provide a more detailed comparison of different platforms using the (Diamond) ratings, where 5 diamonds correspond to the best possible rating and 1 diamonds corresponds to the lowest possible rating for any given platform for a particular characteristic. **Table 2: Comparisons the different platforms based on the following characteristics**: scalability, data I/O performance, fault tolerance, real-time processing, data size support and the support for iterative tasks. Clearly, the first three characteristics are system/platform dependent and last three are application/algorithm dependent. We will provide more details about each of these characteristics and evaluate the “goodness” of each platform for that particular characteristic by providing the star ratings for each of the platform with respect to these characteristics. It should be noted that, a rating of two diamonds for a platform should be interpreted as better than having a one diamond and does not necessarily mean that it is two times better than the platform with a one diamond. Similarly, if a platform X is rated with four diamond, platform Y is rated with two diamond, and platform Z is rated with one diamond, then it should be interpreted as platform X is significantly better than platform Z compared to platform Y. It should also be noted that it is almost impossible to quantify this significance in a general scenario and one can only do so in the context of a specific application or a problem at hand.

**Table 2: Comparison of different platforms (along with their communication mechanisms) based on various characteristics.**

<table>
<thead>
<tr>
<th>Scaling type</th>
<th>Platforms (Communication Scheme)</th>
<th>System/Platform</th>
<th>Application/Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Scalability</td>
<td>Data I/O performance</td>
</tr>
<tr>
<td>Horizontal</td>
<td>Peer-to-Peer (TCP/IP)</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Virtual clusters (MapReduce/MPI)</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Vertical</td>
<td>HPC clusters (MPI/MapReduce)</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Web: [www.irjms.in](http://www.irjms.in)  Email: [irjms2015@gmail.com, irjms.in@gmail.com]
5.1 Scalability:

Scalability is defined as the ability of the system to handle growing amount of work load in a capable manner or its ability to be enlarged to accommodate that growth. In our case, scalability is considered to be the ability to add more hardware (scale up or scale out) to improve the capacity and performance of a system.

5.2 Data I/O Performances:

Virtual clusters using the Spark framework receive 3 Diamonds since Spark makes use of system memory which gives it an edge over Hadoop. Although system memory is fast, the data transfer between different nodes still takes place over the network. This makes the network access a bottleneck for data I/O. In addition, the communication over the network degrades the performance.

Data I/O performance refers to the rate at which the data is transferred to/from a peripheral device. In the context of big data analytics, this can be viewed as the rate at which the data is read and written to the memory (or disk) or the data transfer rate between the nodes in a cluster. GPU and FPGA receive 5 stars since they have high throughput memory and the data I/O operations are extremely fast.

Peer-to-peer systems are the worst in this category and will receive only 1 diamond. These systems use disks for the data access. In addition, the unmanaged and complex network scheme makes it very inefficient to aggregate the results over a single node and makes network communication even slower compared to the virtual clusters. These flaws degrade the data I/O performance.

5.3 Real-time Processing:

Real-time processing of a system is its ability to process the data and produce the results strictly within certain time constrains. Real-time responses are often delivered in the order of milliseconds and sometimes microseconds depending on the application and the user requirements.

5.4 Fault Tolerance:

Fault tolerance [40] is the characteristic of a system to continue operating properly in the event of a failure of one or more components. Since we created this table with intent to compare the platforms of similar capacity, we additionally consider the chances of failure in a system and give a high rating if system failures are extremely rare even though it may not have any fault tolerance mechanism. This enables us to make an unbiased comparison between unreliable systems with fault tolerance and reliable hardware with not so good fault tolerance mechanism.

5.5 Data size supported:

Data size support is the size of the dataset that a system can process and handle efficiently. In this category, peer-to-peer networks will receive 5 Diamonds since they can handle even petabytes of data and can theoretically scale out to unlimited number of nodes. Virtual clusters and HPC clusters can handle terabytes of data and thus will receive 4 Diamonds. Virtual clusters can scale up to tens of thousands of nodes and frameworks like Hadoop and Spark are capable of processing and handling such large datasets.

6] How to choose a platform for big data analytics?

The star ratings provided in Table 2 gives a bird’s eye view of the capabilities and features of different platforms. The decision to choose a particular platform for a certain application usually
depends on the following important factors: data size, speed or throughput optimization and model development. We will now provide more details about each of these factors.

6.1 Data size:

The size of data that is being considered for processing is probably the most important factor. If the data can fit into the system memory, then clusters are usually not required and the entire data can be processed on a single machine. The platforms such as GPU, Multicore CPUs etc. can be used to speed up the data processing in this case. If the data does not fit into the system memory, then one has to look at other cluster options such as Hadoop, Spark etc. Again, Hadoop and Spark clusters can handle large amount of data but Hadoop has well developed tools and frameworks although it is slower for iterative tasks.

6.2 Speed / throughput optimization:

We here speed refers to the ability of the platform to process data in real-time whereas throughput refers to the amount of data that system is capable of handling and processing simultaneously. We must clear that goal is to optimize the system for speed or throughput. If one needs to process large amount of data and do not have strict constraints on the processing time, then one can look into systems which can scale out to process huge amounts of data such as Peer-to-Peer networks, Hadoop etc. These platforms can handle large-scale data but usually take more time to deliver the results. These platforms are capable of processing the data in real-time but the size of the data supported is rather limited.

Conclusion:

This paper surveys various data processing platforms that are currently available and discusses the advantages and drawbacks for each of them. Several details on each of these hardware platforms along with some of the popular software frameworks such as Hadoop and Spark are also provided. A thorough comparison between different platforms based on some of the important characteristics (such as scalability and real-time processing) has also been made through diamond based ratings. This article provides the readers with a comprehensive review of different platforms which can potentially aid them in making the right decisions in choosing the platforms based on their data/computational requirements. This study will provide a first step to analyse the effectiveness of each of the platforms and especially the strengths of them for handling real-world applications.

References: