Cluster Computing

Debashis Jena

University of Maryland University College

Table of Contents

[1. Abstract 2](#_Toc26079471)

[2. Introduction 2](#_Toc26079472)

[3. History of Cluster Computing 4](#_Toc26079473)

[4. Parallel Computing 5](#_Toc26079474)

[5. Commodity Clusters Classification 6](#_Toc26079475)

[6. Beowulf Node 8](#_Toc26079476)

[7. MapReduce Using Cluster Computing 10](#_Toc26079477)

# Abstract

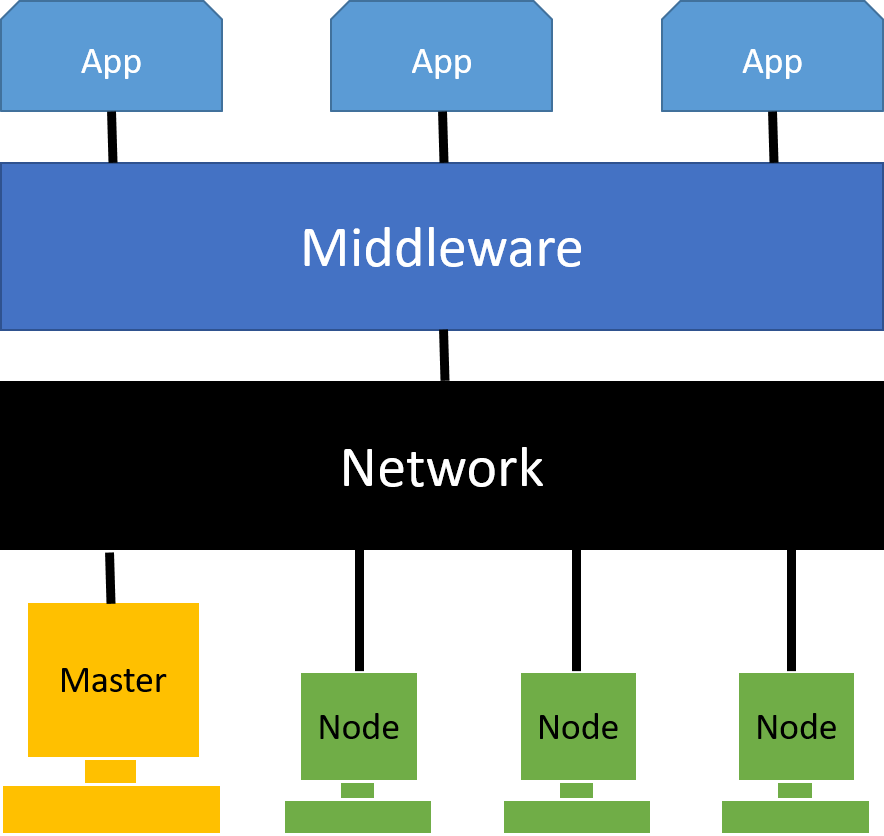
Today, a massive volume of data is both being queried and processed in all the systems. Data can be either structured or completely unstructured. To process them seamlessly, currently, we use MapReduce techniques. Many cloud services use the enhanced or modified version of this technique to handle the huge amount of data. The hardware backbone of this technique completely relies on cluster computing.

This paper gives an overview of cluster computing and the historical perspective of it. Also, MapReduce’s dependency on cluster computing is described in detail. The number of nodes in a cluster can be increased and decreased based on the volume of data being processed. So, this paper gives an insight into the elasticity feature of the MapReduce which is utilized by cloud computing services like Amazon Web Services and Microsoft Azure.

# Introduction

With the evolution of smart devices, IoT and artificial intelligence, the data collected and processed by applications is massive. Data processing by common systems is not possible because of the size of the data. Besides, the data streamed or processed may or may not be structured. This problem required a solution that would process the volume of data asynchronously and flawlessly. The solution was creating a cluster computing structure that would process data with multiple threads.

Cluster Computing is defined as a group of computers connected; tightly or loosely. The computers are called nodes. These nodes can be a group of personal computers or very fast supercomputers. Of the nodes, there will be a master who controls the other nodes, which are called slaves. The slave nodes use parallel computing technology to process a request. They are connected by a high-speed local area network connection (LAN). A clustering middleware software layer connects these nodes and makes them appear as a large cohesive computing unit. This technique makes all the servers process data parallelly and thus reducing the response time by many folds.



Such an arrangement and technique benefits in many ways. It helps to largely reduce the unavailability of these systems and provides larger storage to another desktop workstation or computer. Although cluster computing has been very popular, it is extensively implemented and utilized in Petroleum Reservoir Simulation, Google Search Engine, Earthquake Simulation, Weather Forecasting.

The MapReduce technique uses a cluster computing technique to process the massive data by creating the maps of nodes in a cluster. The master node is responsible to control the data processing in the nodes and accumulate the results after each of the slave nodes has processed the data.

# History of Cluster Computing

The first cluster system that became popular was the Semi-Automatic Ground Environment (SAGE) which built for NORAD under the Air Force contract by IBM in the 1950s. It was created based on the MIT Whirlwind computer architecture. It used a vacuum tube and core memory technologies. SAGE cluster was created by bringing a number of separate standalone systems together and its job was to manage early warning detection of hostile airborne intrusion. In the early age of cluster computing, it used to have loosely coupled computer pairs, with one computer doing the user jobs while the other managed various input/output devices. After the breakthrough of computer networking in the 1970s, it brought significant long-term effects in future cluster computing. In the 1980s, NSA employed a number of 160 interconnected Apollo workstations as a cluster to perform a few specific computational and arithmetic tasks. Digital Equipment Corporation (DEC) created a system consisting of a collection of VAX 11/750 computers connected with each other. DEC coined the term "cluster" in the process of building the system. In 1997, a number of RS/6000s were clustered together to have the required scalability for the famous "Deep Blue" chess championship that featured a computer versus Gary Kasparov, in which the computer won.

Today, on Google’s search site, not only can you search with jumbled up words, but also you can search results using an image as a key. Similarly, the IoTs today process billions of requests parallelly in milliseconds. Google currently processes hundreds of petabytes of data on a daily basis. They take the most advantage of cluster computing to process them. By 2008, they were processing “20 petabytes of data per day through an average of 100,000 MapReduce jobs spread across its massive computing clusters” (Kennedy, 2008).

# Parallel Computing

Basically, parallel computing is the simultaneous use of multiple computer or compute resources to perform a computational task or solve a problem:

* Networks connect multiple stand-alone computers (nodes) to make larger parallel computer clusters.
* A task or problem is split into smaller and independent parts
* A series of instructions are created for each part
* All the instruction sets run simultaneously on different computers
* An overall control/coordination mechanism is employed

For example, Lawrence Livermore National Laboratory (LLNL) is a federal research facility in the United States which has employed parallel computing for the research activities.

A screenshot of a computer

Description automatically generated

The parallelism involved can be briefly explained as follows.

* Many compute nodes are connected together in highspeed LAN connection called InfiniBand network.
* Each of the computers or the compute nodes is a parallel computer in itself.

# Commodity Clusters Classification

Commodity clusters a class of modern-day supercomputers. A commodity cluster is a collection of completely independent computers or computing processors integrated using a commodity off-the-shelf network. The interconnection network used is dedicated to the cluster system and is also commercially available from its manufacturer. There are four subclasses of commodity clusters.

1. Beowulf-class systems
   * Integrated by commercial COTS local area networks or system area networks
   * Run widely available low-cost software for coordinating parallel execution
   * Exceptional price/performance for many applications

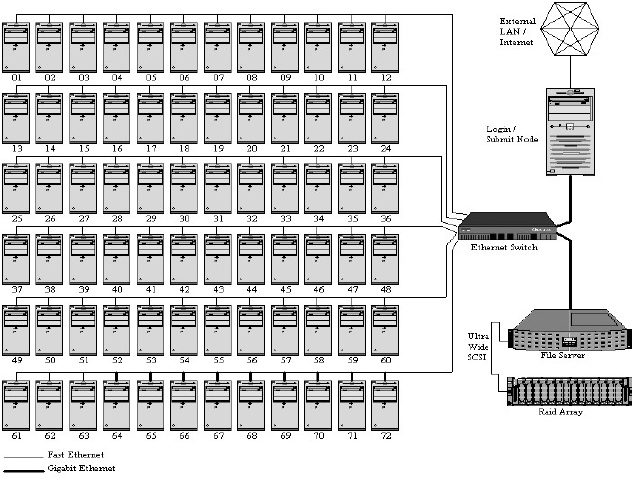


Figure 5.1 Overview of 72-node Beowulf Cluster

1. Workstation clusters
   * Integrated by system area network
   * Tend to be vendor specific in hardware and software
   * Superior performance over MPPs (Massively parallel processors)
   * 2.5 to 4 times costlier than PC-based clusters
2. Superclusters
   * Cluster of clusters
   * Usually integrated by the institution's infrastructure backbone wide area network
   * Typically, within the same internet domain
   * Possesses potential to partner multiple local clusters to achieve large scale computing
   * Highly available with redundancy

A close up of a map

Description automatically generated

1. Cluster farms
   * Integrated by local area networks of PCs and workstations
   * Uses job stream parallelism and distributes queued work
   * Lower performance because of the shared network integration of the resources, unlike dedicated networks used in workstation clusters and Beowulf’s

# Beowulf Node

The basic activities and responsibilities of a node in a cluster can be categorized into below 4 points.

1. instruction execution,
2. high-speed temporary information storage,
3. high-capacity persistent information storage, and
4. communication with the external environment, including other nodes.

To do the above basic activities, the cluster nodes not only need to have basic I/O and processors, but they also need high-speed connectivity between nodes and the external environment. The architecture is more complicated than a general-purpose computer.

A screenshot of a cell phone

Description automatically generated

1. Microprocessor – the logic performed in instruction execution, address translation, memory management including cache needs the processor. Processor speeds may be up to 5GHz.
2. Cache – High-speed clusters take the most advantage of caching. Modern cluster nodes provide up to 16 MB of cache memory.
3. Main memory – Clusters need high-density storage with very quick read/write access. A common node currently has 8GB memory and often optimized for throughput.
4. Hard or Solid-State Drives – Modern clusters utilize the high speed read/write of SSDs.
5. PCI bus – Current PCI buses are mostly with 64-bit data paths at 66MHz.
6. Video Controllers – These are the cards that convert digital signals from the processor into analog signals suitable for driving a video display in a monitor.
7. Network interface controller – an interface that provides interaction between the node and its external environments, such as file servers, printers, terminals, and the Internet.
8. Power supply
9. Cooling systems — typically a fan mounted on the computer box itself

# MapReduce Using Cluster Computing

In early 2000, Google used to utilize the best of web crawling and query frequency technologies for many of its applications. It was adequate for the requirements for the time. However, the engineers at Google foresighted the problem in the future when the data was going to be massive to analyze on. So, they invented the MapReduce method in the year 2004, which utilizes the cluster computing comprising of inexpensive computers. These computers are nothing but the nodes in cluster computing. This process consists of a master and slaves. To explain MapReduce, it has to be split into two independent parts; Map and Reduce.

* *Map* –
  + Processes a (key, value) pair
  + Returns a list of intermediate (key, value) pairs

map (k1, v1) 🡪 list (k2, v2)

* *Reduce –* 
  + Merges all intermediate values having the same intermediate key

reduce (k2, list(v2)) 🡪 list (v2)

The entire process goes through five theoretical steps, which can be described as follows.

1. The master process receives the job configuration, which specifies the location of the input data and other information. This process triggers the job execution.
2. Based on the instructions within the job configuration, the master process starts several mappers and reducer processes in different machines in parallel. Simultaneously, the master process sends another instruction to start a separate process to read the input data from its location, followed by splitting the data and distributing them to the various mappers.

A close up of a map

Description automatically generated

1. Once the mappers receive the data, each of them executes a map function to generate the key/value pairs. Then the pairs are grouped based on the keys.
2. Then the reducer processes are assigned with the pairs with the same keys. They now execute reduce function to merge all the values associated to the same key to create a smaller set of values.
3. The result from each of the reducers are accumulated and the final result is sent to the output.

The MapReduce concept is currently being used in many bigdata systems. The cloud computing companies like AWS, Google and Microsoft Azure utilize this technique to handle massive volume of data.

# Cluster Lifecycle in EMR within AWS

EMR is the Elastic MapReduce technology within AWS (Amazon Web Services). The term “elastic” refers to the easy and horizontal scaling of the clusters within the cloud service. A successful EMR cluster goes through a specific life cycle as illustrated in the following figure.

A close up of text on a white background

Description automatically generated

Figure 8.1 Lifecycle of a cluster in EMR

1. Amazon EMR supports cluster types of Hadoop and Spark. As the first step, EMR starts clusters of any of the specified types. During this, the cluster state is “STARTING”.
2. As the next step, the user-defined actions such as installing other applications or any environment setup, are run on the clusters. This state of the cluster is called “BOOTSTRAPPING”.
3. In case of a bootstrap failure or any process failure, the cluster goes to the “SHUTTING\_DOWN” state and then to “FAILED”.
4. However, on successful completion of the bootstrap phase, the cluster goes to the “RUNNING” state, where all the job configuration steps are run sequentially.
5. Depending on the configuration, the cluster may go to a state called “WAITING”, after processing all the steps. This is the case when the cluster has been configured as a long-running one. It can be done by disabling the “auto-terminate” feature or by using the KeepJobFlowAliveWhenNoSteps parameter in the API.
6. When the step execution is complete and there is no instruction for WAIT, then the cluster goes to the “SHUTTING\_DOWN” state and then to “COMPLETE”.
7. In case of a manual termination, the cluster goes to the “SHUTTING\_DOWN” state followed by “TERMINATED”.

# Conclusion

Cluster computing is not only about having a large number of computers connected via a high-bandwidth network connection. It is also about taking the most advantage of the parallel computing ability of not depending on a shared-memory system. The cloud computing companies have come up with many techniques that have made the clustering technology provide high-availability and optimum redundancy. For example, the EMR clusters within AWS have nodes with auto-scaling and auto-recovery technology features. They also provide load balancing feature which makes each node balanced equally within a cluster.

Cluster computing has aided the data analysis within many scientific applications so much already. This is highly critical for the military systems and astronomical systems. Similarly, many modern business intelligence systems (e.g. Tableau) use cluster computing to support the data warehousing system to analyze a high volume of data, which helps companies target the right customers for marketing.

As quantum computing evolves and becomes a reality, cluster computing can take advantage of their size and availability.

**References**

1. Antonopoulos, Nick & (eds), Lee Gillam. (2010). Chapter 7 - A Peer-to-Peer Framework for Supporting MapReduce Applications in Dynamic Cloud Environments, In *Cloud computing: principles, systems and applications*. [Books24x7 version] Available from [*http://common.books24x7.com.ezproxy.umuc.edu/toc.aspx?bookid=36581*](http://common.books24x7.com.ezproxy.umuc.edu/toc.aspx?bookid=36581)
2. Barney, B. (n.d.) Introduction to Parallel Computing. Retrieved from <https://computing.llnl.gov/tutorials/parallel_comp/>
3. Das, D. (2018, February 1) What Is Amazon EMR? Retrieved from [*https://thirdeyedata.io/amazon-emr/*](https://thirdeyedata.io/amazon-emr/)
4. Kennedy, N. (2008, January 8) Google processes over 20 petabytes of data per day. Retrieved from <https://www.niallkennedy.com/blog/2008/01/google-mapreduce-stats.html>
5. Lopez, P. & Baydal, E. (2018, July) Teaching high-performance service in a cluster computing course. Retrieved from <https://www-sciencedirect-com.ezproxy.umuc.edu/science/article/pii/S0743731518301059?via%3Dihub>
6. Sterling, T. (2002). *Beowulf cluster computing with linux*. [Books24x7 version] Available from [*http://common.books24x7.com.ezproxy.umuc.edu/toc.aspx?bookid=3410*](http://common.books24x7.com.ezproxy.umuc.edu/toc.aspx?bookid=3410).
7. No Author. (2002) Clustering: A basic 101 tutorial. Retrieved from <https://www.ibm.com/developerworks/aix/tutorials/clustering/clustering.html>
8. No Author. (n.d.) What is Cluster Computing? Retrieved from <https://www.educba.com/what-is-cluster-computing/>