International Journal of Education and Science Research Review
E-ISSN 2348-6457
Volume-10, Issue-1 Jan-Feb-2023
P-ISSN 2349-1817
www.ijesrr.org
Email- editor@ijesrr.org

STUDY ON CLOUD COMPUTING AND WEB-BASED APPLICATIONS ACCESSIBLE FOR USERS

Sonmali Sandip Shivaji

Research Scholar, NIILM University, Kaithal, Haryana

Dr. Lokendra

Department of Computer Science And Engineering

NIILM University, Kaithal, Haryana

ABSTRACT

The nature of cloud computing is evolving and dynamic, and it is characterized by the virtualization of resources and the use of the Internet to provide them as a service. Virtualization, networking, software, distributed computing, and online services are all components that are included in the phenomenon known as cloud computing. Cloud and computing are the two components that make up this concept. In the former, the Internet and the services it provides to users are reflected. Users are provided with simple and on-demand access to a shared pool of resources, which may include servers, storage, and web-based applications. This accessibility paradigm is referred to as the cloud. The latter, on the other hand, refers to the process of designing or building both hardware and software systems. Multiple data centers and storage services are combined into a single network architecture under the multi-cloud architecture.

***Keywords:**. Components Of Cloud Computing*

INTRODUCTION

Cloud computing environments offer low-cost services or resources, which are determined by the requirements of the users. Because of the ever-increasing requirements of users, it is the duty of the service provider to distribute resources to consumers in an effective manner. Additionally, the efficiency of the cost of resources ought to be optimized in accordance with the demand of the individual. Cloud computing is dynamic and virtualises resources and provides them as a service via the Internet. Google originally introduced the word "cloud" in 2006 to characterise their business strategy for offering services online, and it has since become common. Virtualisation, networking, software, distributed computing, and online services comprise cloud computing.

Components of cloud computing and web-based applications

Cloud and computing comprise this notion. The former shows the Internet and its services. A shared pool of servers, storage, and web-based applications is easily accessible to users. This accessibility paradigm is called cloud. The latter involves developing and constructing hardware and software systems. These systems organise, process, and manage information and run scientific applications on computers.

Cloud computing is "a model for the delivery and consumption of shared, elastic, and persistent computing resources that maximises user experience and elasticity in response to changing demand without requiring centralised management" (servers, networks, storage, applications, and servers). This term is from NIST. Many firms have reduced capital expenses and eliminated long-term management by using cloud computing. It also frees up time and resources for organisations to focus on their strengths rather than maintenance. Customers may employ cloud computing to benefit from a reliable, adaptable, and quality-assured computing environment. When you buy from us, you get these perks. It also lets clients download data from a remote place and access it when needed.

With an Internet connection, consumers are able to utilize cloud computing technologies whenever and whenever they want, regardless of where they are located. This makes it simple for both small and large organizations to use it to store information and access services because to its accessibility.

1 Cloud network

- The entity interconnects cloud services and users.
- It uses encryption and decryption technologies to secure data over the network.

2 Data centers

Data centers are collections of servers that host various kinds of applications. These servers might be located a significant distance away from the client, or they can be positioned a short distance away.

3 Distribution servers

Through the utilization of these servers, users are able to utilize cloud apps or services in the same manner as if they were operating on their own personal computers.

4 Client computers

The client computer acts as a mediator, providing a graphical user interface for cloud interactions. Three main client categories are usually identified:

- **Thick clients -** In the not-too-distant future, it is not out of the question that thick clients will do the majority of their job independently of the central server.
- **Thin clients –** A significant number of thin clients, also known as simply clients, are dependent on servers to provide the bulk of their processing requirements. These clients make the most of the few resources that are available on the host system, despite the fact that their capabilities are limited.

- **Hybrid clients -** Here you can find both wealthy and less wealthy customers. Despite the fact that they are able to handle the bulk of the processing, it is possible that they will need the assistance of a central server in order to finish an essential part of the operation.

5 Consumer/Broker

This entity mediates between the user and service provider. Another advantage is that users may request cloud services from anywhere in the globe. Additionally, it enables safe and secure data transit between customer and supplier.

6 Resource allocator

The consumer and the cloud infrastructure are brought together by this component, which acts as an interface. Moreover, it places an emphasis on the distribution of resources that are to be made accessible to the end user with the least amount of administration effort possible.

7 Negotiator

Together with the broker, they negotiate the service level agreement (SLA) to set rates and penalties.

Difference between various computing technologies

1.Grid computing

The phrase "grid computing" refers to the process of doing vast computational tasks by coordinating the activities of a large number of dispersed computers that are linked to one another via internet connections. Grid computing, much like cloud computing, makes it simpler to combine several tasks and manage properties that have a large number of tenants. Grid computing may be broken down into three primary categories of entities,

- **Control node -** The entire network is managed by it, and it monitors the resources that are available.
- **Supplier -** In addition, it contributes its resources to the network.
- **User -** Therefore, it makes use of the resources that are made available by the service provider.

When it comes to cloud computing, on the other hand, businesses are able to increase their capacities as quickly as possible without having to install additional infrastructures.

Cluster computing

The phrase "cluster computing" describes a methodology of doing computations in which several interconnected computers act as one. Conversely, cloud computing provides virtualisation technology within a system, which lets the user operate many kinds of services simultaneously.

2. Utility computing

The fact that it provides on-demand resources and charges customers for the services they actually use, rather than charging them a fixed rate, makes it seem very similar to cloud computing. A service provider may maximise resource utilisation and minimise operating expenses via utility-based pricing. The virtualisation of resources is a key differentiator between cloud computing and utility computing.

3. Parallel computing

The method of computing in parallel entails dividing the issue into smaller components and executing each component on several processors simultaneously. Processors in a parallel processing environment are able to talk to each other thanks to the shared memory.

Dominic and Ratnam (2013) Cloud computing has greatly benefited Malaysia's healthcare business. Cloud services may improve cooperation and efficiency between healthcare and health insurance organisations, according to factor analysis. Rahimli (2013) studied Malaysian SMEs and management aspects affecting cloud computing use. Four major problems raised throughout the study were how cost, need, dependability, and security effectiveness affect a business's choice to utilise cloud computing.

Abubakar et al. (2014) They say emerging nations like sub-Saharan Africa favor cloud computing. SMBs were in Nigeria. Cloud computing adoption and emergence were investigated qualitatively. The research found that SMEs in this area are less worried about data, privacy, and security loss than those in the north. However, Nigerian SMEs face service delivery, internet connection cost and stability, cultural, and infrastructure challenges. Security, privacy, and trust issues don't deter SMEs from cloud computing, according to the report.

Steininger et al. (2014) addressed the factors that affect their organisations' public cloud use. The main goal of this project was to rethink and implement the above components. Small and medium-sized firms (SMEs) adopted cloud technology for its flexibility and pay-as-you-go approach, according to Babu and Chakravarty (2014).NTT (2014) advises enterprises to pick a cloud service provider based on a complete cloud computing strategy. The selected partners should also establish a plan and strategy, integrate, deploy, migrate, support, custom development, hosting services, and ongoing management to meet cloud expectations.

OBJECTIVE

1. To investigate Components of cloud computing and web-based applications
2. To study on Difference between various computing technologies

RESEARCH METHODOLOGY

Experimental setup

All of the runs mentioned in this section are performed on Vulcan, an IBM Blue Gene/Q system with a capacity of five Petaflops and 24,576 nodes. The location is inside Lawrence Livermore National Laboratory's (LLNL) Classified Collaboration Zone network. With one GB of RAM allotted to each core, the BG/Q design makes use of 1.6 GHz IBM PowerPC A2 processors that can run one to four hardware threads simultaneously across each of its sixteen cores. An exclusive 5D toroidal connection connects the nodes. This link offers one microsecond latency and a two gigabit per second bidirectional connection capacity. Ten connections, two in each direction, connect each of the system's nodes (A, B, C, D, and E) to ten more nodes. Since the E dimension includes two lengths, the bandwidth available for connecting two nodes in E is twice that of other directions. Depending on different work distributions, the toroidal form and connectivity may change for a given node count while using Vulcan. This circumstance occurs because Vulcan is a distributed computing platform. The work shapes listed in the following table were used to classify the bulk of the runs in this section's data.

Table 1: Shape and connectivity of the partitions allocated on Vulcan (Blue Gene/Q) for different node counts.

#nodes	A	B	C	D	E	Torus or Mesh
128	1	4	4	4	2	Torus in all directions
256	4	4	4	4	1	Torus in all directions
512	4	4	4	4	2	Torus in all directions
1024	4	4	4	8	2	Mesh in D, Torus in rest
2048	4	4	4	16	2	Torus in all directions
4096	4	8	4	16	2	Torus in all directions

The pF3D and MILC algorithms were executed on networks including 128 to 4096 nodes. Our study demonstrates that pF3D and MILC both gain from two threads per core for optimal hardware thread performance, but MILC requires four threads per core. Our previous work with both apps underpins this. Both programs operated in a weak scaling mode only facilitated by MPI to maintain a uniform problem size for each MPI operation. By using mpiP, we determined the time allocated to computation and communication in various MPI functions.

By using a tracing tool that IBM built and tuned for the BG/Q, we were able to calculate the average and maximum number of hops that all communications travelled. An internal library that gives access to network hardware counters was used to collect packet counts for different torus connections. In this study, two mappings on BG/Q provided by the system are compared against a number of partitioning and permutation operations provided by Rubik. MPI ranks are arranged hierarchically by the default mapping on BG/Q, with T (hardware thread ID) coming first, then E, D, and so on. Before moving on to the next node in this mapping process, the node's cores must be filled. The TABCDE mapping is similar to a round-robin mapping and is distinguished by the slowest increment of T. The process moves on to the next core of each node after allocating the MPI ranks for the first core of each node.

pF3D mapping research

At the National Ignition Facility (NIF) at Lawrence Livermore National Laboratory (LLNL), the scalable multi-physics code pF3D is used to model laser-plasma interactions. The objective is to resolve the wave equations pertinent to laser light and backscattered light. The following are two of the most essential phases in any communication process: 1) The propagation of waves and their interaction, and 2) The propagation of light. We use fast Fourier transforms (FFTs) to address the first issue and a sixth-order advection approach to tackle the second.

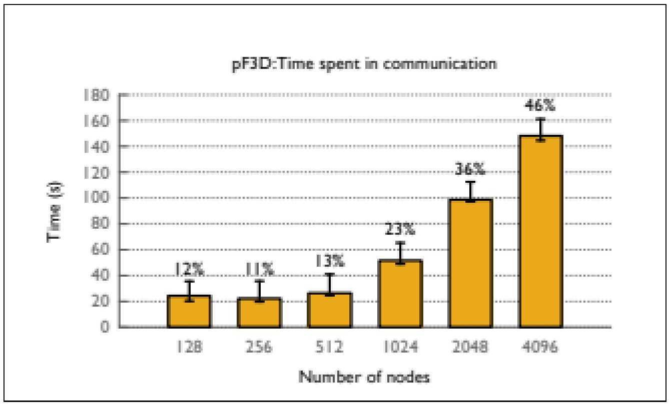

Figure.1: Average, minimum, and maximum time spent in communication by pF3D for weak scaling.

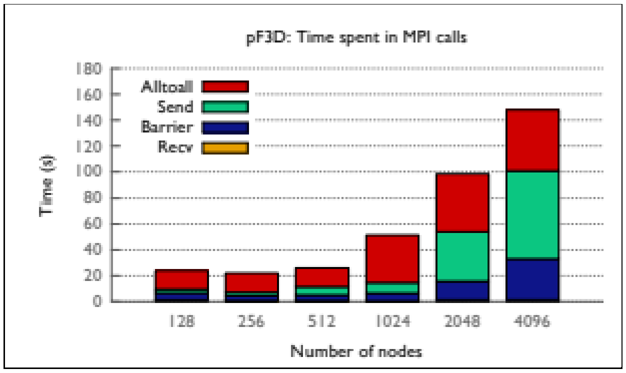

Figure 2: Average time spent in different MPI routines by pF3D for weak scaling.

The domain of pF3D is partitioned across MPI processes using a three-dimensional Cartesian grid. This discussion pertains to the process grid with fixed dimensions of 32 for the X-axis and 16 for the Y-axis, as applied to the input problem used. An increase from 8 to 256 planes in Z corresponds with a rise in program process count from 4,096 to 131,072. Numerous one-dimensional Fourier transformations (FFTs) are distinct from the two-dimensional Fourier transform (FFT) during wave propagation and coupling. A collection of

Fourier transforms (FFTs) includes processes with congruent X and Z coordinates, as well as another collection that comprises processes with congruent Y and Z orientations. The advection signals are sent in the Z direction when interconnected processes of the XY planes interact. In the FFT phase, MPI All totals are used across 32 and 16 sub-communicators, respectively, whilst MPI Send and MPI are utilised during the advection phase.

Performance debugging: baseline performance

We first investigate pF3D using mpiP to determine communication's temporal contribution and the importance of the two processes. Figure 1 shows MPI process messaging lengths for different node counts including average, minimum, and maximum. Communication contributes to runtime as shown by the percentage labels above each vertical bar in the application. We expected limited scalability to have a constant communication time, however it increases, especially after 1,024 nodes, reaching 46% of the total time when 4,096 nodes are examined. A careful analysis of the temporal distribution across multiple MPI operations shows that MPI Alltoall in the FFT phase, MPI Send in the advection phase, and MPI Barrier dominate messaging performance (Figure 2). Mainly these three MPI functions affect performance. 32-byte all-to-alls occur across sub-communicators, but 8-kilobyte communications between process pairs are uniform. Advection sends 256- and 384-KB messages. Our 200-millisecond transmission time exceeds our expectations. At 4,096 nodes, we spend time at an MPI Barrier. Network difficulties cause communication imbalance symptoms, according to our research. Barrier-stalled processes are called congestion. We expect an AI-driven mapping system to reduce this time.

RESULT

In the Tilt mapping, we create three-dimensional tiles inside the five-dimensional torus partition and incline the BC planes within the three-dimensional sub-tori around B. This procedure significantly improved performance on Blue Gene/P, but seems to have little impact on BG/Q.

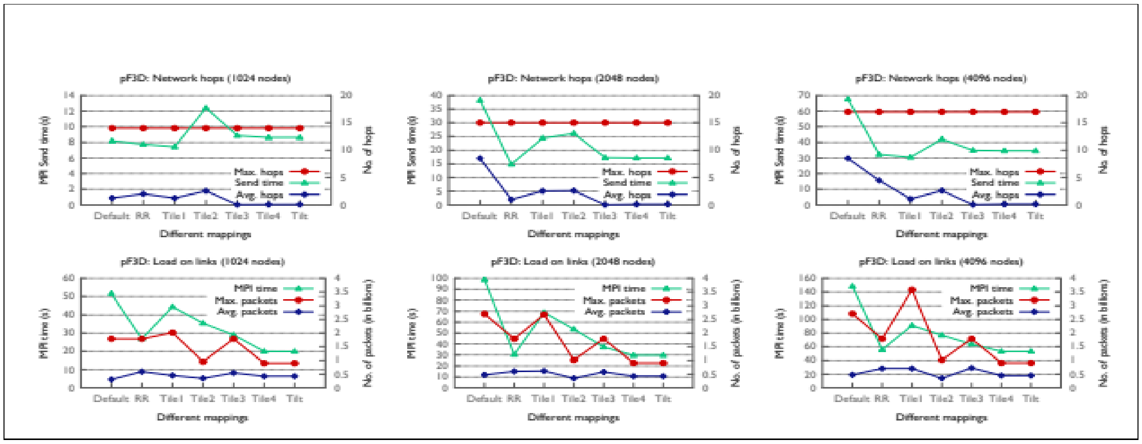

Figure 3: pF3D displays that compare the MPI time to the average and maximum load on network connections (bottom) and the time spent on point-to-point operations to the average and maximum hops (top) (Note: y-axis has a different range in each figure).

Numerous deductions and observations may be derived from these scaling graphs. The all-to-all and transmit procedures may be performed more efficiently if the application is strategically tiled onto the torus. This seems to be the first trend. The time spent in the barrier has decreased, indicating reduced network congestion and/or a reduction in communication imbalance. Employing all dimensions of the torus enhances the efficacy of torus tiles relative to cubic tiles in pF3D computations. This transpires because the messages, particularly the all-to-alls, may transmit their traffic more comprehensively. Ultimately, when improved mappings are implemented, the time needed for transmissions decreases; nonetheless, it eventually reaches a plateau above a certain threshold. Section 3.1.3 delineates more research on this topic. Tile4 surpasses the default ABCDET mapping in overall performance by reducing communication time by 52% on 1,024 nodes and by 64% on 4,096 node

Table 2: Details of communication patterns.

Communiqué Design	Quantity of Procedures	Mails per Procedure (TDC)	Note Scope (KB)
Shapeless Net	8,847,361	6 - 20	512
Organized Network	80 × 48 × 48 × 48	8	2,048
Numerous toward numerous	180 × 128 × 384	127	100
Uniform Spread	8,847,360	6 - 20	512

Round Robin Routers (RRR): Similar to the RRN scheme, the RRR scheme uses a round-robin approach to assign routers to tasks instead of assigning individual nodes to specific jobs.

Kamil and colleagues found that a processor's topological degree of communication (TDC) closely correlates with its communication partners. They investigate many popular applications and find that their TDC may be 400–250,000. Table 2 patterns were used to analyse a cross-section of common communication methods and account for a comparable variety of TDC. In Section 3.2.4, the anticipated outcomes for each pattern are examined.

We may generate communication graphs for each pattern by executing them using AMPI, which enables us to run more MPI processes than the actual cores, or a simple sequential program that replicates their communication topology. We can build each pattern's communication graph by running them over AMPI, which lets us run more MPI processes than the physical cores can manage, or by replicating their communication structure using a simple sequential program.

CONCLUSION

This conclusion tests high-performance computer networks and their communication flows under different application patterns. In-depth investigations of communication stack components have been reported. This includes the application's communication pattern, runtime communication support, injection policy, routing protocol, and many other components. This concludes also examines how several environmental variables, such as network architecture and job placement policies, affect application communication performance. These case studies taught us that optimising communication should not be done alone. Instead of optimising one element of communication, employ a holistic strategy that examines all components and configurations to obtain the greatest performance. Thus, you may significantly increase our communication.

REFERENCES

[1] S. Kamil, L. Oliker, A. Pinar, and J. Shalf, “Communication requirements and interconnect optimization for high-end scientific applications,” IEEE Trans. Parallel Distrib. Syst., vol. 21, no. 2, pp. 188–202, Feb. 2010.

[2] B. Duzett and R. Buck, “An overview of the ncube 3 supercomputer,” in Frontiers of Massively Parallel Computation, 1992., Fourth Symposium on the, Oct 1992, pp. 458–464.

[3] C. Leiserson, “Fat-trees: Universal Networks for Hardware-Efficient Supercomputing,” IEEE Transactions on Computers, vol. 34, no. 10, October 1985.

[4] The Connection Machine CM-5 Technical Summary, Thinking Machines Corporation, 245 First Street, Cambridge, MA 02154-1264, October 1991.

[5] X. Yuan, “On nonblocking folded-clos networks in computer communication environments,” in Parallel Distributed Processing Symposium (IPDPS), 2011 IEEE International, May 2011, pp. 188–196.

[6] “Lonestar supercomputer at TACC,” https://www.tacc.utexas.edu/systems/lonestar.

[7] “Stampede supercomputer at TACC,” https://www.tacc.utexas.edu/stampede/.

[8] “Wikipedia entry on torus,” http://en.wikipedia.org/wiki/Torus.

[9] M.Blumrich, D.Chen, P.Coteus, A.Gara, M.Giampapa, P.Heidelberger, S.Singh, B.Steinmacher-Burow, T.Takken, and P.Vranas, “Design and Analysis of the Blue Gene/L Torus Interconnection Network,” IBM Research Report, December 2003.

[10] Y. Ajima, S. Sumimoto, and T. Shimizu, “Tofu: A 6d mesh/torus interconnect for exascale computers,” Computer, vol. 42, pp. 36–40, 2009.

[11] Cray Inc., “Cray XE6 Specifications,” http://www.cray.com/Assets/PDF/products/xe/CrayXE6Brochure.pdf, 2010.