Enhancing the Performance of Cloud-based Application Systems for Robust Retrieval of Information Using Various Tools And Techniques

Simrit Kaur Sohal
Chandigarh Engineering College, Mohali, India

Harmanbir Singh Sidhu
Assistant Professor, Chandigarh Engineering College, Mohali, India

Abstract – Cloud computing is defined as a computing application that delivers computing resources from a large computing pool to the end users as a service over a network i.e., generally internet. The integral part of maintaining cloud is performance monitoring of cloud. Since performance monitoring should monitor the capability of each component in delivering the services but largely resources are virtualized in cloud. Therefore it is possible to estimate the performance of physical infrastructure by resource utilization but not with virtual components because of shared and dynamic resource allocation. Presently many big cloud-based firms are working hard on the performance of cloud-based applications to make it more efficient to the customers. But, still there are some issues to be worked upon. This paper deals with the various tools and techniques for fast retrieval of information from the cloud with the objective to develop a new methodology in cloud computing using tools like C,C++or MATLAB.

Index Terms – Cloud Computing, Performance, Quality of Service, Performance Monitoring.

1. INTRODUCTION

1.1. Cloud Computing

In computing era, cloud computing is considered to be a great revolution and the key idea behind this is, based on the demand that users make resources from large pool can be distributed irrespective of users location. Cloud computing is based on pay-as-you-use model. That is, users are to pay only for the resources being used or reserved. These resources can be computing platforms, infrastructure or data storage. Cloud computing is the cost effective way to access resources. Thus, it is not entirely new concept and is driven from some technologies such as grid computing, utility computing, autonomic computing, and cluster computing, and distributed computing.

Cloud computing helps large scale business firms to maintain their data and applications on internet rather than their in-house server systems. This can relieve the operational burden from the organization for managing, maintaining and controlling the data and applications.

Some of the leading platforms are IBM Cloud Computing, Oracle, Google App Engine (GAE), HP Cloud-enabling Computing, Amazon Web Services (AWS), Microsoft and Salesforce.com are in use today.

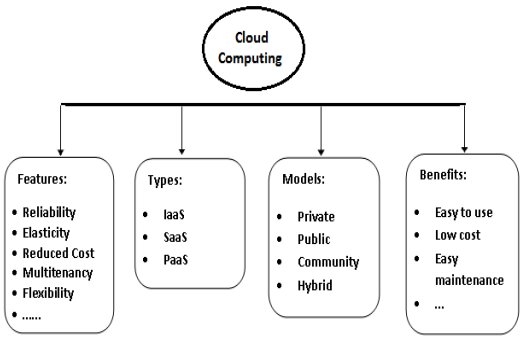

Fig.1 View of Cloud

A cloud provider utilises the virtualization technologies in their server systems and customers are charged upon the amount of resources used or reserved. Virtualiztion helps to achieve greater system utilization and at the same time lowers the total cost of ownership.

1.2. Server systems

Following are the reasons why company's don't prefer to maintain their own server systems:

- Data accessibility is limited.
- Data security is insufficient.
- Maintenance costs are not calculable.
- Computing power is not flexible.
- Issues with performance, capacity and scalability.

1.3. Cloud Characteristics

Various characteristics of cloud computing are as follows:

- On-demand service

- Provides application programming interface (API)
- Easy to maintain and manage
- Reduces costs and capital expenditure to large extent
- Provides rapid elasticity
- Flexible and scalable
- Device and location independent
- Resource pooling to serve multiple users

1.4. Reasons to fear cloud Computing

There are several risks that are given below:

- Lack of transparency
- Network dependent
- Centralization
- Reliability and performance issues
- Security and compliance issues
- Cloud service provider viability
- High value cyber-attack targets
- Risk of data leakage

1.5. Performance Monitoring

The integral part in maintaining cloud is performance monitoring. There are several tools available for monitoring the cloud but no standard tool is available that cover all the parameters of performance. Therefore, Service Level Agreement (SLA) form is crucial part of cloud environment for availing cloud facility. Performance can be defined as the quality of service (QoS) that service provider provides to the client over the network i.e., internet. There are many ways to calculate the performance of the network because of different nature, design and topology used. As in case of circuit-switched networks, the performance can be calculated with the help of queuing modes. Therefore, performance metric can be used to improve the performance and provides the customer with the best value service.

2. PRESENT WORK

2.1. Scope

Google App Engine (GAE) and Amazon Web Services (AWS), the two leading platforms, performance analysis with traditional web servers is obtained by httperf measurement tool and planet lab testbed. The metrics selected for Google App Engine are Round Trip Time (RTT) and Network throughput with parameter data size and number of requests per planet lab node. Whereas, metrics selected for Amazon Web Service is memory bandwidth and parameters are the action of application including four levels i.e., Copy, Add, Scale, and Triad. Therefore, in Google App Engine, it is analyzed that the network throughput strongly depends on data size. Hence it is clear that network throughput is inversely proportional to data size. Thus concluded in Google App Engine if we wisely choose data size then it is possible to get reasonable performance whereas in Amazon, infrastructural architecture is to be considered deeply to enhance performance.

However if we somehow manage to reduce the data size sent over the network then the network throughput would be increased which further improves the performance of cloud-based applications. Therefore the proposed study will be much more elaborative considering different searching and sorting techniques so as to measure the retrieval time and enhancing cloud's performance. Hence there's a need to develop new methodology for fetching data of components in robust manner.

2.2. Problem Formulation

In present days, to make the cloud based applications more efficient to the customers in terms of performance, continuous work is done. But still, a number of performance issues exists on which if worked upon can enhance the cloud performance to great extent. So, our main concern is to overcome the problem either by improving cloud-based applications architecture or designing an algorithm that works upon the parameters of the performance to improve overall performance of the cloud-based application and provides the QoS to the customer that they are paying.

2.3. Objectives of the Study

- Review performance aspects in cloud computing.
- On the basis of attributes, comparing performance models/cloud providers.
- To improve the performance of cloud-based applications in terms of fast retrieval, design an algorithm.

3. PROSPOSED SOLUTION

The proposed problem can be solved by using programming language(C/C++) and the Apriori Algorithm which is cluster related algorithm. The very basic and simple physics concept is applied that states the time is directly proportional to distance provided speed of data retrieval remains the same. Various searching and sorting algorithms that would be retrieved from database using Apriori Algorithm will be used and thus measuring Euclidian distance. The performance calculated in cloud-based application systems is based on timer and genetic algorithm.

3.1. Techniques

Following are the various techniques along with pros and cons that are used in this project:

Sr.No.	Technique	Pros.	Cons.

1.	Random Selection	it is easy to learn few lines of code works very well on already sorted lists, or lists with just a few permutations	not effective for large numbers of sorting elements
2.	Serial Selection Algorithm	easy to implement requires no additional storage space it performs well on a small list	poor efficiency when dealing with a huge list of items its performance is easily influenced by the initial ordering of the items before the sorting process
3.	Serial Insertion Algorithm	it is easy to learn few lines of code efficient for small data sets stable, does not change the relative order of elements with equal keys	not effective for large numbers of sorting elements needs a large number of element shifts as the number of elements increases the performance of the program would be slow
4.	Bucket Algorithm	the user knows the range of the elements time complexity is good compared to other algorithms	you are limited to having to know the greatest element extra memory is required
5.	Shell Algorithm	only efficient for medium size lists 5 times faster than the bubble sort and a little over twice as fast as the insertion sort, its closest competitor	it is a complex algorithm and its not nearly as efficient as the merge, heap, and quick sorts is significantly slower than the merge, heap, and quick sorts
6.	Radix Algorithm	if it's implemented in Java, it would be faster than QuickSort or HeapSort stable, i.e. it preserves existing order of equal keys is quite good on small keys	Does not work well when you have very long keys, because the total sorting time is proportional to key length and to the number of items to sort. • You have to write an unconventional compare routine. It requires fixed size keys, and some standard way of breaking the keys into pieces
7.	Merge Algorithm	• is stable • Can be applicable to files of any size • Sequential reading through each run while merging and writing the sorted record. Switching from run to run only necessary.	• it requires an extra array • is recursive
8.	Quick Sort	• Does not need additional memory	• is recursive

9.	Heap Algorithm	• it does not use recursion • works just as fast or any data order	• slower that quick and merge sort • memory requirement, it requires both an array and a heap of size n • not stable
10.	Counting Algorithm	• is stable • is fast	• unsuitable for strings • memory requirement, it requires both an array and a heap of size n • not recommended on large sets of data.

Table 1. Techniques for this project

4. CONCLUSION

The fast and robust information retrieval has been playing a relevant role in the current competitive environment, aiming to improve the business efficiency. The comparison with the traditional web services highlighted the performance and the security issues. Therefore, the research question was how to enhance the performance of the cloud-based application systems for the fast and robust retrieval of the data. Aiming to answer the research question this study thoroughly investigated the cloud performance issues and highlighted the important concepts, which should be specially analyzed when evaluating the performance of cloud-based application systems. The application of above techniques along with Apriori Algorithm helps improving performance of cloud-based applications for data retrieval from proxy cloud.

5. FUTURE SCOPE

The work done so far is not the end, there is a huge potential for future research to get more and more companies involved in cloud solutions and therefore a chance to increase the performance of the cloud-based application systems. For further research, it can be extended using various other tools and techniques, applied over different environments and platforms for the faster and robust retrieval of data and thus enhancing the performance. Instead of using the basic C language and the Turbo compiler, one can make use of MATLAB compiler for the better normalization technique. The work can be extended for the various other simulations like JSIM, NS2, NS3, WLAN, etc. for better optimization of the data, thus obtaining better results.

REFERENCES

[1] Keith Jeffery [ERCIM], Burkhard Neidecker-Lutz [SAP Research], "The Future of Cloud Computing: Opportunities for European Cloud Computing Beyond 2010 Expert Group Report", Version 1.0.
[2] Clarke Alan, "Managing the performance of Clouds and Cloud-based Applications", IEEE CRQ 2012
[3] Google App Engine, http://code.google.com/appengine
[4] He Chao, "Performance Analysis based on two Leading Computing Platforms: Google App Engine and Amazon Web Service", http://www1.cse.wustl.edu/~jain/cse567-11/ftp/clouds/index.html
[5] Heger Dominique A., "A Performance Centric Introduction to Cloud Computing", http://www.dhtusa.com/media/CldPerf.pdf
[6] Hill Zach and Humphrey Marty, "A Quantitative Analysis of High Performance Computing with Amazon's EC2 Infrastructure: The Death of the Local Cluster", 10th IEEE/ ACM International Conference on Grid Computing, Vol pp, Oct 13-15 Year 2009
[7] "httperf – Wikipedia, the free encyclopedia",http://en.wikipedia.org/wiki/Httperf.
[8] Jain Raj, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling", Vol pp 203-220, pp 381-389, Year 1991
[9] Amazon Elastic Compute Cloud, http://aws.amazon.com/ec2
[10] "Planetlab testbed" , http://www.planet-lab.org.
[11] R. Nathuji, C. Isci and E. Gorbatov, "Exploiting platform heterogeneity for power efficient data centers", IEEE Conference on Automonic Computing (ICAC), Vol pp, June 2007
[12] Singh Dr. Gurdev, Sood Shanu and Sharma Amit, "CM-Measurement facets for Cloud Performance", International Journal of Computer Applications (IJCA), Vol 23- No.3, June 2011
[13] Vineetha V., "Performance Monitoring in Cloud", www.infosys.com
[14] Venkataraman Vinod, Shah Ankit and Zhang Yin, "Network-based Measurements on Cloud Computing Services", http://www.cs.utexas.edu/~vinodv/files/cc-measure.pdf
[15] Xiong Kaiqi and Perro Harry, "Service Performance and Analysis in Cloud Computing", http://www4.ncsu.edu/~hp/Kaiqi10.pdf.

Authors

Simrit Kaur Sohal received her B.Tech in Computer Science & Engineering from Quest Group of Institutions, Mohali, India in 2013. Currently she is pursuing M.tech in Computer Science & Engineering from Chandigarh Engineering College, Landran, Mohali, India. Her area of interest is applied in computers, networking and cloud computing.

Mr. Harmanbir Singh Sidhu has completed his B.tech and M.tech in Computer Science and Engineering. Currently he is faculty at Department of Computer Science & Engineering, Chandigarh Engineering College, Landran, Mohali, India. He has presented and published many papers in National and International Conference.