Performance Enhancement of Database Driven Technique using Cynosure Method in Cloud

Deepak Begrajka
Computer Science
Medicaps Institute of Technology and Management, Indore,(M.P)

Avini Sogani
Computer Science
Medicaps Institute of Technology and Management, Indore,(M.P)

Arpit Jain
Information Technology
Medicaps Institute of Technology and Management, Indore,(M.P)

ABSTRACT

Cloud computing is characterized by illusion of infinite computing resources, shared infrastructure,elimination of an up-front commitment by cloud users, ability to pay for use as needed, and secernent its users and operators. These above mentioned characteristics inflict new challenges to database applications entertained in the cloud, some of them are: (i) how to isolate database users,(ii)how to price database services, and (iii) how to manage resource demands and database performance in relational database system and shared infrastructure. To provide a system which balances these aspects and to serve quick and exact results to isolated users is a biggest challenge. The current way to approach this problems and solutions provided by present techniques based on virtual machine and query optimization does not completely fulfill your desire to properly address these challenges. So in this paper after a survey in global cloud network it has been discovered that some meaning full points and provide new direction to research to handle these problems and propose that they above challenges share a common need for accurate prognostic models of performance and resource utilization.

Keywords

Cloud, Isolation, optimization.

1. INTRODUCTION

Humans are always so ambitious to achieve more and improve upon the way they interact with one another. No matter where one looks there are the evidences of new inventions and creation of new technologies that make life easier. As they move forward it is clear to see that as advancement in technology increases, what might have been a dream in the past has now become a possibility and in some cases a reality. And one of the such dream was cloud computing which is know reality. Cirrus, Nimbostratus, Stratocumulus, Contrails, Amazon, Billow, Mammatus, Orographic, Altocumulus and Pileus are all types of what? Yes you may be scratching your head on this question; so to assist you the answer is—Clouds. Cloud computing is emerging as an important IT strategy as enterprises strive to deliver more services to the tenants across the wide network. Cloud computing is an architectural model that leverages standardization and consolidation to allow effective and safe sharing of pooled resources. The key attributes of cloud are shared infrastructure, and a layer of indirection that unlinks the responsibility of running the infrastructure that use it. This key attributes simplifies some aspects of application deployment, as the vendor of cloud manages its security, reliability, mapping of software processes onto physical machines, and above all these due to share infrastructure cloud has economic benefits. Cloud services are provided on a concept called virtualization. Virtualization is a concept where a fraction of CPU, disk, and memory resources on a physical machine are dedicated to each application running on it. Virtual Machines (VMs) provide “hard isolation” because there is little or no sharing of resource or there is no interaction between applications hosted on the same physical machine. Unfortunately, this way of providing cloud service is not-ideal for performance-sensitive applications like database for four key reasons:

1. Accounting to be eased off, cloud service providers require customers to do capacity planning in terms of virtualized hardware resource instead of measurable performance analyses of database.
2. Hard isolation prevents multiplexing and sharing of resource when multiple databases are co-located on a machine, causing considerable over-provisioning and adequate loss of performance.
3. Higher degree of multi-tenancy and the decoupling of cloud vendors from app-developers make tuning/provisioning of the DBMS in the cloud more than in conventional database deployment.
4. In a cloud configuration users share the same underlying physical infrastructure. Without network isolation, users could intentionally or unintentionally consume a large part of the network, intrusively see data on the network that does not belong to them, or invoke side-channel user attacks.

These four limitations are crucial reasons why many businesses are unenthusiastic to use the VM based database as a service. They user should believe that, to completely realize potential of cloud computing in the database universe, an alternative approach is needed. One part of the solution is to avoid VMs, which introduce a performance bottleneck in cloud. By using the same DBMS to host multiple databases owned by different tenants of the cloud, it is feasible to avoid some of the shortcomings of a VM-based approach.

Simply rejecting VMs and running multiple databases in the same DBMS instance, however, does not address the four limitations discussed above. In this paper, it has been argued and suggested that what is needed is innovation in several key technical areas:

Resource Attribution and Application level: New research is required to provide mechanisms for mapping high-level analysis such as transaction throughput and dormancy into underlying resources, e.g., CPU cores, RAM capacity, Disk I/O, network bandwidth. This will enables service providers to expose billing schemes and capacity planning tools that are more understandable to the tenants, and provide them better schema to deal with.

Soft Isolation: The performance drawbacks of VM based hard isolation are needed to be abstained, the paper need to provide substitute mechanisms that achieve soft isolation, that is, consent to resource sharing when the system is underutilized, but emphasize quotas when necessary. The paper suggest this could be done as a closed loop admission control in the DBMS, which in turn requires a deep understanding and knowledge of the performance resource tradeoff of tenants' workloads.

Multi-user, Multi-machine DBMS Tuning: For high performance workload specific DBMS tuning is crucial. In a decoupled, multi-user setting such as a DBaaS, neither provides nor tenants have the right combination of visibility/access/expertise to perform proper tuning and provisioning. In present DBaaS offerings, tenants are allowed to select one of a small number of fixed configuration. The above mentioned way is inefficient, and the paper suggest for new, automated solution that can achieve workload specific DBMS tuning in a multi-user fashion, where user observe multiple workloads running on a pool of multiple machines, extract performance/resource characteristics of each of each work load (without ever observing it in isolation), and solve problems like DBMS placement and DBMS tuning (i.e., selecting group of workloads and assigning them to DBMS configuration in an optimal way). And this criteria can be called as multi-user, multi-machine DBMS tuning.

Network Traffic Isolation: There might be cases in the cloud environment where certain tenant needs to be isolated on its own network for various reasons. For example, the traffic isolation can be used to provide an initial layer of security, higher bandwidth for specific users, implement specialized charge back policies, or to support tiered networks. Other examples include isolating network traffic for LAN based backups, ftp, or replication traffic.

Network Security Isolation: Networks in a consolidated environment must ensure that the database traffic is secure and authenticated against trusted tenants. Network security is built on top of network isolated traffic, and can be implemented using encryption (SQL*Net, TLS/SSL, or https) or authentication; i.e., allow or deny database service access using validation rules.

Building such models is tricky, as the performance of given work load or transaction varies depending on other transaction running in the system at the same time. For example, a given transaction may do many I/O operations when running alone, but fetch few pages in the presence of other transaction that reads the same data. Alternatively, a given transaction might do more I/O operations in the presence of another transaction that access many pages and put pressure on the buffer pool. This paper suggest some methods and models for resource and performance prediction. In this paper, with the help of this one can solve the performance modeling problem for transactional workloads. Then describe how such models might be used to tackle the four technical challenges describe above.

2. LITERATURE SURVEY

[1] A paper on performance isolation of network virtualization for cloud by professor Sung-Won Ahn and Chuck Yoo Dept. of Computer Science and Engineering, Korea University[1]. They focused on how network virtualization can be done efficiently and how this can mitigate the hardware restriction problem. They provided the concept of virtual router in virtual network environment which suggest the coexistence of various virtual networks which provides various different services on a physical network, and manages these virtual network efficiently. In this paper they have implemented virtual network that can provide network isolation and control bandwidth allocation with SR-IOV on Intel 82599 10G NIC. Hence performance isolation can lead to balance of bandwidth allocation for better quality of service. It can provide security, isolation, efficient resource management and can lower the cost, provide flexibility, manageability on virtual network.

[2] A paper on Performance Isolation for Cloud Datacenter Networks by Professor Alan Shieh, Srikanth Kandula, Albert Greenberg, and Changhoon Kim[2]. They focused on the design space for achieving performance isolation between tenants. They pointed following problems like Datacenter cannot keep up with the numbers of the users and the VM churn observed in cloud database centers; datacenter impose static bandwidth limits to obtain isolation at the cost of network. To mitigate this concerns the provide a tool called Seawall, which is a scalable performance isolation system for cloud data center networks that fairly allocates network capacity between users, achieves elastic, network utilization, and is robust against malevolent users. And Seawall doesn't require any special network support, so it can be deployed in existing data center to improve both of these metrics.

[3] A paper on Business Driven Cloud Optimization Architecture by Professor Marin Litoiu, Joanna Ng, Gabriel Iszali, Murray Woodside, and Johnny Wong[3]. They focused on several facets of optimization in cloud computing and propose an architecture for addressing those challenges. Their architecture takes into account various stakeholders in the cloud. Their architecture supports self management by automating most of activities pertaining to optimization: monitoring, analysis and prediction, planning, preparation and execution. Their architecture coordinates the separate problems through virtualization of resources in the optimization. It supports linear programming to balance complex business factors and come at least close to optimal profits at all layers. It provides self adaptation through feedback mechanism at each layer.

[4] An oracle white paper on April 2012 present some factors on network isolation in private database cloud [6]. They focused on tenant isolation on multi tenant configuration in private database cloud. Proper isolation results the fair and safe use of the shared resources. If proper isolation not present tenants may intentionally or unintentionally abuse shared resources and can lead to security threats.

There are various techniques like physical and logical isolation to provide varying degree of isolation. And the level of isolation to deploy will largely depends on tenants need. They have provided some technologies and techniques available to Database cloud Providers to implement end to end isolation for network traffic.

3. METHODOLOGIES

Several techniques are used for isolation, optimization and cost some of the important methods are as follows below.

3.1 Method for Isolation

3.1.1. XEN Virtualization

This virtualization is divided into two parts such as full-virtualization and para-virtualization. VM-ware is an example of full-virtualization. It does not require modification of guest operating system and most work is operated as software. And full virtualization has low performance. Para-virtualization typical example is XEN. XEN increases virtualization by modifying guest operating system code. It manage instruction by using hyper call that is similar to system call. Therefore, there are no additional conversion operations during guest operating system operations.

Advantages:-

- The biggest advantage of XEN virtualization is that it allows tenant to get the most efficient use possible out of the given server. With XEN virtualization user can run all the computers off to one server saving us a great deal of money.
- With the help of XEN virtualization vendors maintenance cost is greatly reduced. The main reason for this is that if

something goes wrong they have to fix one computer since all of the others are running off of this one. This also reduces the downtime since all you have to do is fix or replace one computer to get the whole system back up and running.

- The technique used by XEN virtualization is safer than other. Since you will only need to be able to handle security on one server it will be a much easier task than it would otherwise be.

Limitations:-

- For *86 tenants, a maximum of 16GB memory per tenant.
- Higher level of expertise required.
- Possible increase in licensing fees.
- Additional backups, monitoring and seller support.
- A maximum of 254 devices per guest.
- A Maximum of 15 network devices per visitors.

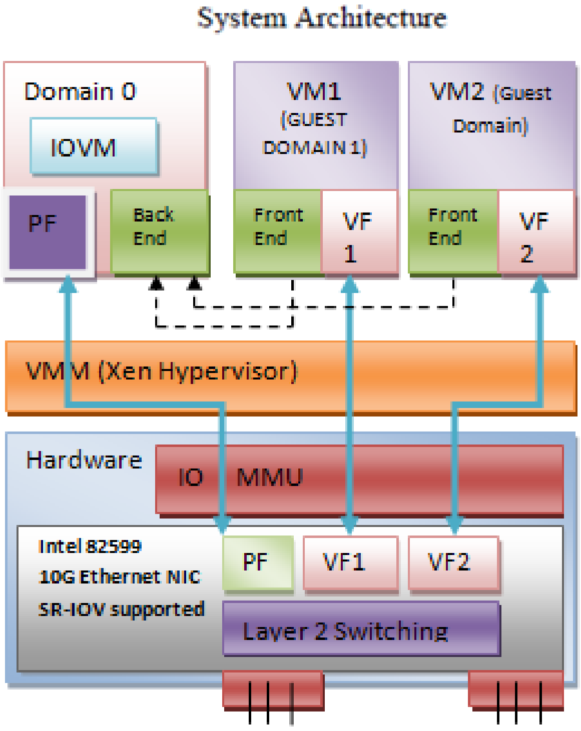

Fig. 3.1:- XEN Virtualization Architecture.

3.1.2. SR-IOV (Single-Root I/O Virtualization)

One of the new I/O virtualization technique which uses Intel-VT in Intel is SR-IOV. This method is sort of hardware assisted virtual machine to solve performance degradation and allows that a guest operating system can access physical Network Interface Card directly as a result, increases performance of virtualized network I/O which infects to performance of network virtualization significantly. Furthermore, it classifies clearly role and definition of Physical Function and Virtual Function. Hence, it is possible that each guest operating system can use isolated virtualized Network Interface Card without Influence from other virtual network.

Advantages:-

- Large number of virtual machine per server are possible to run with this technique, which reduces the need for hardware and the resultant costs of space and power required by hardware devices.
- It creates the ability to start and add or remove servers independently, increasing flexibility and scalability.
- It adds the capacity to run different operating systems on the same host machine, again reducing the need for discreet hardware.

Limitations:-

- Live migration is presently unsupported. As with PCI passthrough, alike devices configurations are required for live (and offline) migrations. Without alike devices configurations, visitors cannot access the passed-through devices after migrating

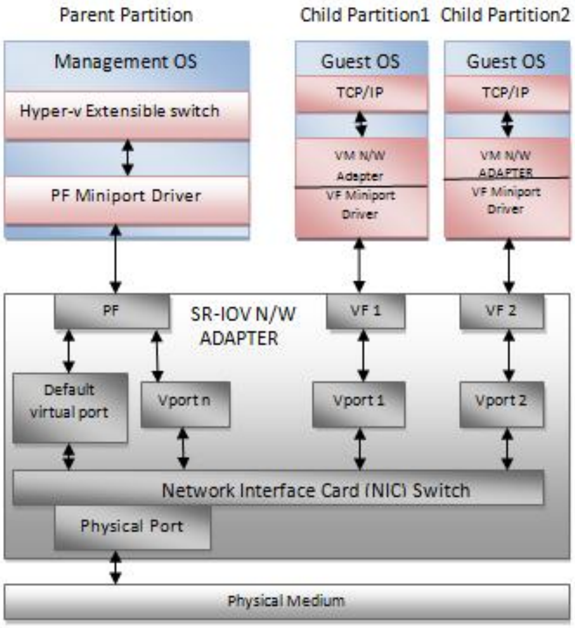

Fig. 3.2:- Single-Root I/O Virtualization Model

3.1.3. Software Router:

A software router architecture is configured by software routing application on operating system on hardware. This system is studied as a solution of saving cost. There are many studies in software router. For instance, the click ensures freely configurable routing schema by making it possible to change module dynamically. And the Open flow ensures managing dynamic routing flow table by separating data and routing control. Moreover, the XORP can support varied routing protocol such as uni or multi cast.

Advantages:-

- It has flexibility, manageability, and scalability.
- Direct connectivity ameliorates the response time of cloud-hosted applications.
- Private WAN integration ameliorates security, performance, and predictability.
- Enterprise control, visibility, and policy uniformity reduce security risks.
- Feature uniformity and product familiarity ameliorate operational efficiency.
- Expansion of the data centre network to a cloud simplifies application on boarding.

Limitations:-

- Microsoft Hyper-V has concerns with tagged packets, so VLAN (dot1Q and QinQ) will not work on Microsoft Hyper-V.
- When the Cisco CSR 1000V is installed on the Microsoft Hyper-V, the interface numbers can change after Microsoft Hyper-V fails over to a new server, or restarts after a live migration.
- This concern is not seen if the maximum number of interfaces is configured.

- On XEN Server 6.1, the paravirtual drivers for the CSR 1000V will not work without a certain for the XEN Server host. This is harmful to performance.

3.1.4. Orthogonal Tenant isolation Mechanism

This System provide the same features as traditional database system (e.g., support for full SQL, transactions, and recovery). Cipherbase is based on architecture with secure co-processors. Also provide data confidentiality. In terms of confidentiality, the Cipherbase system supports various levels of encryption (from no encryption to strong encryption) and different end-to-end security settings so that the right level of confidentiality can be selected for all data.

In general, there are three ways to build secure database systems:

- Encryption during rest: Data is stored encrypted on a commodity storage system (e.g., on hard disks) and shipped to a trusted domain the data is decrypted, processed, and probably cached. The trusted domain may be located on the same premises (or even within the same box) as the (untrusted) storage system or remotely.
- Secure Servers: Data is stored and processed in the cloud environment on specially designated secure nodes.
- Fully Homomorphic Encryption: The Encryption of data is done in such a way that any operation (e.g., addition, multiplication, comparisons) can be performed directly on the data without decrypting it. In this way, data can be processed using conventional untrusted servers. The encrypted results of a query from the untrusted system are shipped to the (trusted) client.

Limitations:-

- It is incomplete, brittle and in order to get functionality you are bounded to give up security.

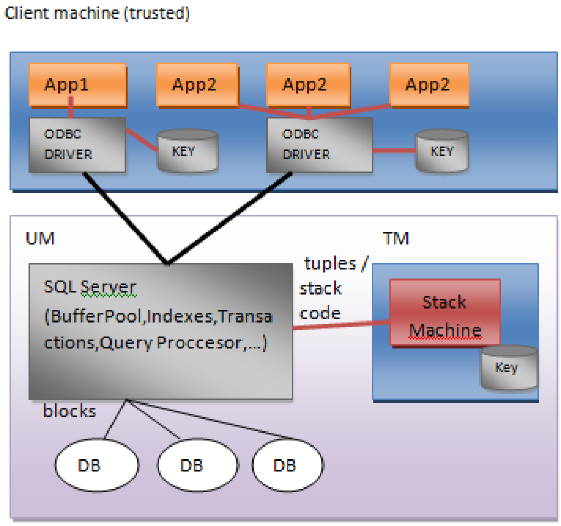

Fig. 3.3:- Cipherbase Architecture.

3.1.5. Linux Process Isolation Mechanism

The Linux kernel provides namespaces to ensure process isolation and cgroups to control the system management. SELinux is used to control severance between the host and the container and also between the single containers. The Libvirt toolbox provides an interface for construction and management of containers.

3.2. Method for Optimization

3.2.1. MapReduce

With the help of this programming paradigm system lends itself well to scale-out and leverage several machines to parallely process data. In the MapReduce it usually consists of 3 phases-map, copy and reduce. The input data is split into chunks of 64MB size (by default). In the map phase, a user defined function operates on every chunk of input data producing intermediate key-value pairs which are stored on local disk. A single map process is invoked to process a single chunk of input data.

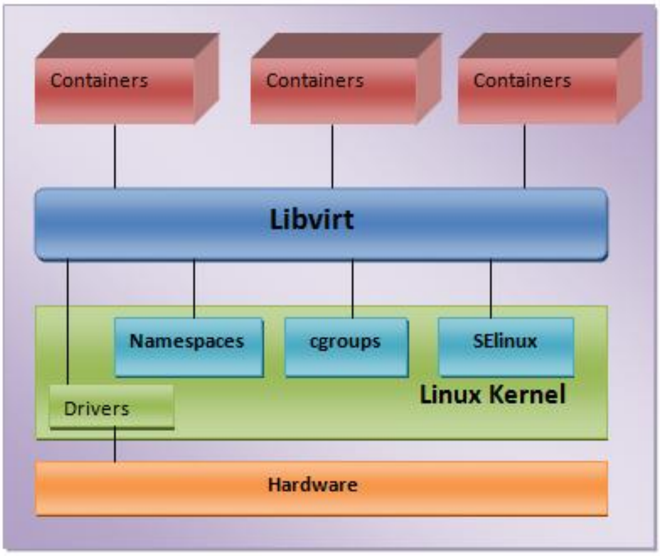

Fig. 3.4:- Linux Process Isolation Model.

In the copy phase, the intermediate key-value pairs are transferred to the location where a reduce process would operates on the intermediate data. In reduce phase, a user defined reduce function operates on the intermediate key-values pairs and generates the output. A single reduce process is invoked to process a range of keys.

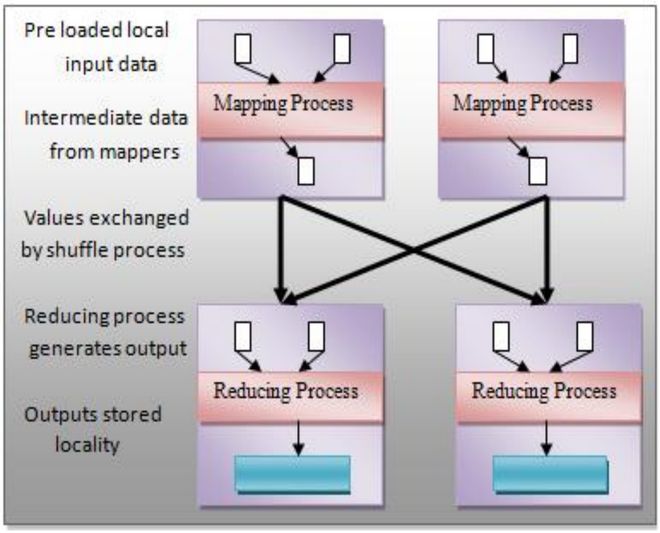
Pre loaded local input data
Intermediate data from mappers
Values exchanged by shuffle process
Reducing process generates output
Outputs stored locality
Mapping Process
Mapping Process
Reducing Process
Reducing Process

Fig. 3.5:- MapReduce Architecture.

Advantages:-

- MapReduce model is easy to use, even for novice programmers without experience with distributed systems, since it conceals the details of parallelization, locality optimization, and load balancing.
- A large diversity of problems are easily expressible with the help of MapReduce computations.

- MapReduce facilitates scaling of applications across large clusters of machine comprising thousands of nodes, with fault-tolerance built-in for ultra-fast performance.

Limitations:-

- In this model Computation depends on formerly computed values.
- The index produced in the Map step is one dimensional, and the reduce step must not generate a large amount of data or there will be a serious performance degradation.
- This algorithm depends on the shared global state.

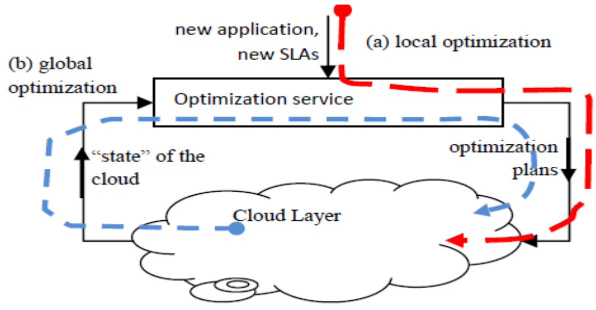

Fig. 3.6:- Two Optimization Scenarios in Cloud.

3.2.2. Hierarchical Dictionary Encoding strategy

While the query execution operation is performed, data is either *eagerly* decompressed when it is read into main memory, or data *lazily* stays compressed in main memory and is decompressed on demand only.(Refer Fig 3.7).

3.2.3. Huffman coding

Huffman coding algorithm is carried out by using the probability distribution system of the alphabet of the source to develop the code words for symbols. The frequency allotment of all the characters of the source is calculated in order to calculate the probability distribution. You should first calculate the possibility and according to the possibilities the code words are assigned. Shorter code words for the higher probabilities and longer code words for smaller probabilities are assigned. For this job a binary tree is created using the symbols as leaves according to their probabilities and paths of those are taken as the code words. Two families of Huffman coding have been proposed: Static Huffman Algorithms and Adaptive Huffman Algorithms. Static Huffman Algorithms compute the frequencies first and then generate a common tree for both the compression and decompression processes.

3.2.4. The Shannon Fano Algorithm

This is another alternative of Static Huffman Coding algorithm. The only distinction is in the creation of code word. All the other processes are equivalent to the above mentioned Huffman Coding Algorithm.

3.2.5. Arithmetic Encoding

This is the technique in which, a code word is not used to represent a symbol of the text. Instead it uses a fraction to represent the entire source message. The probability of occurrence and the cumulative probability of a set of symbols in the source message are taken into account. The cumulative plausibility range is used in both compression and decompression processes.

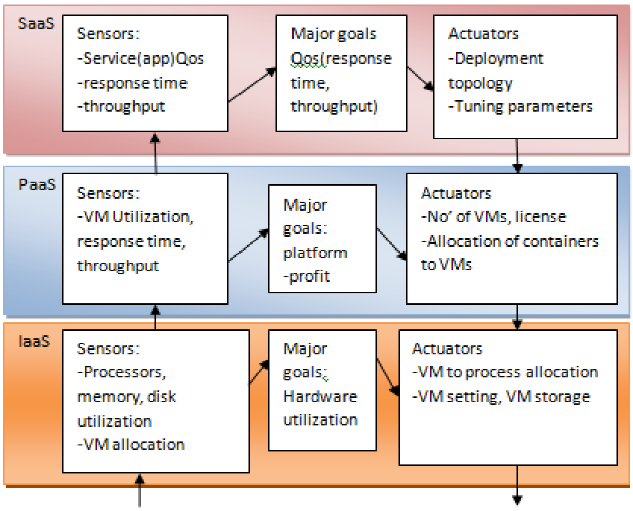

Fig. 3.7:- Conceptual Optimization in Cloud.

3.2.6. Measuring Compression Performances

There are various criteria to measure the performance of a Compression algorithm depending on the nature of application. When measuring the performance the main concern would be the space efficiency. The time efficiency is another factor. The presentation depends on the type and the structure of the input source. Additionally the compression behaviour depends on the category of the compression algorithm: lossy or lossless. If the lossy compression algorithm is used to compress a fussy source file, the space efficiency and time efficiency would be higher than that of the lossless compression algorithm. Thus measuring a general presentation is difficult and there should be different measurements to evaluate the performances of those compression families.

There are some other techniques to evaluate the performance of compression algorithms. Compression time, computational intricacy and probability distribution are also used to measure the effectiveness.

3.3. Method for Cost

3.3.1. Database as a Service (DBaaS)

Database as a Service (DBaaS) is an architectural and operational approach enabling IT providers to deliver database functionality as a service to one or more consumers. Database as a Service architectures support the following necessary capabilities:

- Consumer-based provisioning and management of database instances by means of on-demand, self-service mechanisms;
- Automated monitoring of and compliance with provider-defined service definitions, attributes and quality of service levels;
- Fine-grained metering of database practice enabling show-back reporting or charge-back functionality for each individual consumer.

In addition to these obligatory characteristics, it is likely that DBaaS architectures will naturally support granular service elasticity, secure multi-tenancy, access by means of a broad range of non-proprietary devices and mechanisms, automated supply management, and integrated capacity planning. While these architectural characteristics are certainly significant, they are equally essential for traditional database service architectures and therefore are not specific to what makes Database as a Service new and unique

.

3.3.2. Amazon Relational Database Service

Amazon Relational Database Service (Amazon RDS) is a web service that makes it painless to set up, operate, and scale a relational database in the cloud. It offers cost-efficient and resizable capacity while managing time-consuming database administration tasks, freeing you up to spotlight on your applications and business.

Amazon RDS automatically scraps the database software and backs up your database, accumulating the backups for a user-defined retention period and enabling point-in-time recovery. You advantage from the flexibility of being able to scale the compute resources or storage capacity associated with your Database Instance (DB Instance) via a single API call.

3.3.3. Amazon Elastic Compute Cloud (Amazon EC2)

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale calculating easier for developers.

Reserved Instances – Reserved Instances give you the option to make a small, one-time payment for each instance you want to reserve and in turn receive a significant discount on the hourly charge for that instance. There are three Reserved Instance kinds (Light, Medium, and Heavy Utilization Reserved Instances) that enable you to balance the amount you pay upfront with your effective hourly price.

4. PROPOSED CYNOSURE METHOD

After diligently scrutinizing all the methods and different ways adopted by various users and cloud vendors, This paper has come up with a solution which will not only optimize system performance, will also isolate various files from one another and also save searching time of tenant and also prevent haphazard solution while searching and the provided solution will be to the point, So as you know time is money, and hence in a way you will be able to save your money too. The description of method is divided into two parts as you know tenant upload their files in cloud and also download/search files in cloud, So the cynosure method covers this two point that how the uploading and downloading can be govern in adroit way so that the optimization, isolation, and time constraint for search can be reduced.

Over concept is basically, it should have the central memory divided into small chunks, instead of storing all the files at random memory location they should be stored in a specified manner. Like certain text file is of various format like .pdf, .rtf, .doc, .txt and etc. So saving them randomly any random memory location can be done easily but while traversing them can be quite onerous so it can consume enough time and also user will be provided haphazard resulted with same file name. So it will be quite cumbersome for him too, to select exact same file from the list of file which has analogous file name but can have different file format, or other possibility. So instead of storing them randomly in memory, memory should be divided into small chunks of .pdf, .txt, and etc. So traversing and storing can both be easily maintained and user time can be efficiently saved.

File Uploading:- Concept of file uploading is as same as usual with some amendments user can increase the old concept of storing work more effectively. Tenant will attach the file by providing file name and format, this will be done at the client side now at the server side server will extract the file extension and will find the address of the block of file extension allocated in cental memory unit. The address of the block of file extension will be found with the help of the concept of hash table.

Hash table consist of **hash function** which compute index into an array of slots or buckets, out of them the correct value is found out. The concept of hash function is used because it will assign each key to unique slot, perhaps this condition is rarely achieved in practice (habitually some keys will hash to same slot). So, most hash table assume **hash collision** which assumes different keys which are assigned by the hash function to same slot- will occur and will be accommodated in some possible way. Hash table are more efficient than search.

Steps of Uploading File (See Fig 4.2)**:-**

- Tenants attach File on client side.
- Server extracts file extension from file and search its address in hash table.
- If address of extension block found save it at that memory location, if not found save it at other extension block.

File Searching:- Searching of file is quite oppressive task, as user search for a certain file as searching traverse the complete central memory, so it take a lot amount of time, and after that also there are chances that user does not get the desired result as it provide all the format file present and hence user has to manually search for his file and after going through a lot of files there are chances that he doesn't get the desired output.

So in this file searching technique which is quite unique and provide the desired file you search for in most of the cases. As tenant enters the file name he will be provided checkboxes of different categories like .pdf, .txt, .doc, .others, etc. He will select the checkbox according to the desired file extension he want. It ends here for client side now at the server side, server get the file format from the corresponding selected checkbox, hash table is again used to search the address of the corresponding selected checkbox file format, now as you get the file format block search the file in the that block of memory with various searching algorithm which are present in cloud or DBA can also handle this task, after searching if the file is found display the desired output to the tenant and if no such file is found hence display no such file exist.

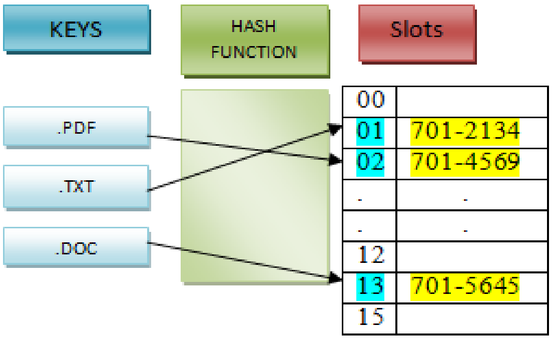

Fig. 4.1:- Hash Table.

This is for single file format search if user mentions various file formats in single search by selecting multiple checkboxes then with the help of concept of pipelining the searching will be applied in different file format blocks parallel and display all such desired output to the user.

Steps of File Searching (Refer Fig 4.3)**:-**

- Tenant will enter the file name and select the corresponding checkbox.
- Server will extract file format and search for address in hash table.
- Search file with file name in file format blocks of memory.
- If found display file, not found display data not exist.

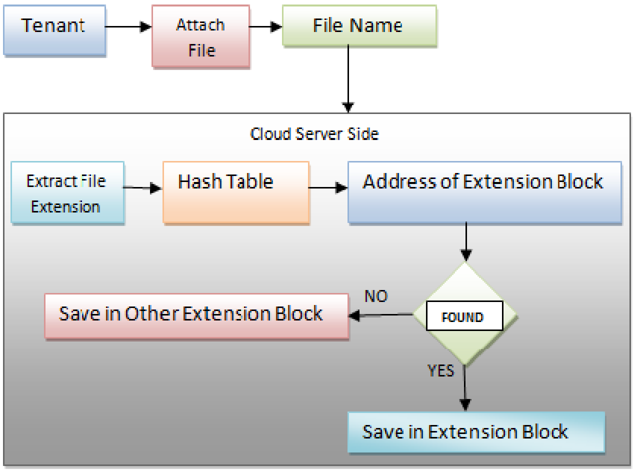

Fig. 4.2:- Flow Chart For Uploading File.

5. RESULT

After deeply scrutinizing all the facts, this paper has come up some exciting results accuracy in cynosure method. In cynosure model on applying search for a file, like .Pdf or etc. The Result obtained are 40% more efficient, than the cloud present way of providing output. Cynosure method, also arrange the cloud memory hierarchy in a proper way like, .pdf have separate block of memory, and like wise other blocks, this will not only decrease the traversing time, but also optimize different types of file from one another. This method will also reduce server side complexity of searching and storing the file, and will completely support all the present features of cloud.

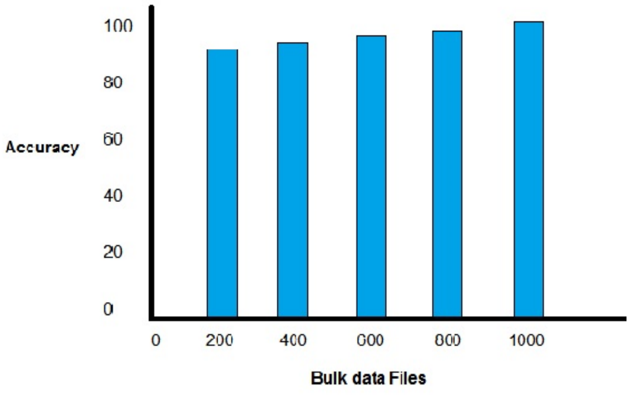

Fig. 5.1:- Performance analysis on bulk data.

The above figure 5.1 shows the data analysis using Cynosure method which describes how it performance varies according to bulk data file storage. This method concludes the best result over bulk data files as shown in tabular description in figure 5.2.

Table 5.2:- Tabular description of Performance analysis.

Files	**200**	**400**	**600**	**800**	**1000**
Accuracy	**91.7**	**91.9**	**92.3**	**92.5**	**92.8**

6. CONCLUSION

In this paper, user came across various current methods for maintaining isolation, optimization, and cost. The paper discusses there advantages and disadvantages and their effect in cloud services and performance. The paper argues that current methods to cloud computing also suffer from their own advantages like infrastructure sharing and also decoupling of services from vendors of the services are why cloud are popular, but despite this fact these features also disable performance sensitive approach. How to increase the searching speed, how to optimize tenant search, how to provide the accurate results? These questions are answered in this paper. This paper provide a model which will guide how searching and uploading should be done in cloud. Finally the paper has come up with an example model named as cynosure method which is discussed above.

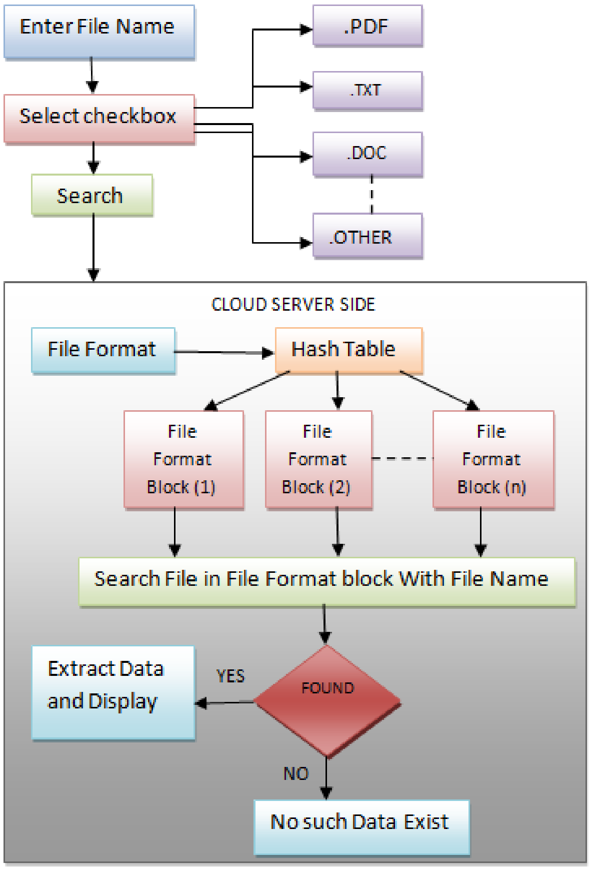

Fig. 4.3:- Flow Chart For Searching a File.

Through this cynosure method it has been concluded that it can provide the exact result to the demanding tenants. This will reduce the complexity to search exact result and also save the time. This provides the systematic approach also at server side to store and serve to the multiple tenants. During this work it has been discovered some several future scope which can be considered as well to increase the efficiency of cloud.

7. FUTURE SCOPE

- How can be the wide spreading infinite data can be managed and unused data can be deleted from cloud is one of the high pounding topic to study.
- How can be the security issues of isolation of data of tenant is alarming.
- How data space for individual user can be increased?
- How Database tuning can be made more efficient with user and server.

8. REFERENCES

[1] Sung-Won Ahn, Chuck Yoo, "Performance Isolation of Network Virtualization for Cloud Computing," KSII The third International Conference on Internet (ICONI), December 2011.

[2] Alan Shieh, Srikanth Kandula, Albert Greenbeerg, Changhoon Kim, "Seawall: Performance Isolation for Cloud Datacenter Networks," Cornell University and Microsoft Research.

[3] Marin Litoiu, Joanna Ng, Gabriel Iszali, "A Business Driven Cloud Optimization Architecture," SAC'10, March 22-26, 2010, Sierre, Switzerland.

[4] Impact of Cloud on Database? http://www.networkcomputing.com/cloudcomputing/virtualization-cloud-having-littleimpac/232500669

[5] Effect of Algorithm on Efficiency of Cloud? http://www.redorbit.com/news/technology/1112801906/algorithm-helps-make-cloud-computing-more-efficient/

[6] An Oracle White Paper, "Network Isolation in Private Database Clouds," April 2012.

[7] M.Ravi Kumar, S.Manoj Kumar, "Multi Benfit's Through Compression for Large Data Stored In Cloud," International Journal of Engineering And Computer Science ISSN:2319-7242 Volume 2 Issue 4 April,2013 Page No. 991-996.

[8] Arvind Arasu, Spyros Blanas, Ken Eguro, Raghav Kaushik, Donald Kossmann, Ravi Ramamurthy, Ramaratnam Venkatesan, "Orthogonal Security With Cipherbase," 6th Biennial Conference on Innovative Data System Research (CIDR ' 13) January 6-9 2013, CA, USA.

[9] Zhiyuan Chen, Johannes Gehrke, Flip Korn, "Query Optimization In Compressed Database Systems," ACM SIGMOD 2001 May 21-24, Santa Barbara, California, USA.

[10] Karthik Kambatla, Abhinav Pathak, Himabindu Pucha, "Towards Optimization Hadoop Provisioning in the Cloud".

[11] Mozafari, C. Curino, S. Madden, "Performance and Resource Modeling in Highly-Concurent oltp Workloads," Technical report, MIT, February 2012.