LOAD BALANCING TECHNIQUES IN CLOUD COMPUTING ENVIRONMENT

Abstract

Virtualization, cloud storage, and networking facilities, plus applications all fall under the rubric of "cloud computing." A cloud is a set of clients, data centres, and distributed servers working stations. this necessitates fault tolerance, rapid provisioning, increased availability, greater performance, and less customer involvement A critical problem is to tackle is an effective distribution algorithm. It could be any one of three: CPU, memory, or delay. Mechanisms that are implemented and operationalized by the receiver of the form We want to develop an efficient load-balancing algorithm that maximises the throughput and minimise the latency of clouds of varying sizes (virtual topology depending on the application requirement). Dynamically allocating system resources can increase application performance in cloud computing. The results of this research show a dynamic resource allocation and dynamic server distribution that is better than the traditional priority algorithm. On the Cloud Network, the experiment was conducted. Our goal is to have as little reaction time as possible. Using the proposed function has allowed VM function scaling and migration to boost application performance.

Keywords: Cloud Computing; Load Balancing, Virtual Machine, Task Scheduling, Computational Load, Scheduling Algorithms, Response Time, Priority Scheduling

Authors

Dr. Sapna Jain Choudhary
PG Head (CSE)
SRGI, Jabalpur.

Priyanka Gupta
Mtech 4th sem,
SRGI, Jabalpur.

I. INTRODUCTION

Load balancing in cloud computing refers to the process of distributing incoming network traffic across multiple servers or resources to optimize resource utilization, improve performance, and ensure high availability. It is a critical component of cloud infrastructure management, allowing efficient utilization of resources and minimizing response time for users.

In cloud computing environments, load balancing can be implemented at different levels, such as:

Network Load Balancing: At the network level, load balancers distribute traffic across multiple servers or resources by using techniques like round-robin, weighted round-robin, or least connections. They operate at the Transport Layer (Layer 4) of the network stack and can balance traffic across multiple data centres or regions.

Application Load Balancing: At the application level, load balancers distribute traffic based on application-specific criteria, such as HTTP headers, session cookies, or application-layer protocols. They operate at the Application Layer (Layer 7) of the network stack, making intelligent decisions based on the content of the network packets.

Load balancers typically monitor the health and performance of backend resources, such as servers or virtual machines, and route traffic only to healthy resources. They can also provide additional features like SSL termination, session persistence, and content caching to further optimize the performance and scalability of cloud applications.

Cloud service providers, such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), offer load balancing services as part of their platform. These services are highly scalable, automatically adjusting to changes in traffic patterns and providing fault tolerance by distributing traffic across multiple availability zones or regions.

Load balancing algorithms used by cloud providers may vary but commonly include round-robin, least connections, source IP affinity, and more advanced algorithms like weighted round-robin, least response time, or least bandwidth. The choice of algorithm depends on the specific requirements of the application and the desired performance characteristics.

Overall, load balancing plays a crucial role in cloud computing environments by ensuring efficient resource utilization, improving scalability, and providing fault tolerance to deliver high-performance and highly available cloud-based services.

Cloud computing refers to the delivery of on-demand computing resources over the internet. It involves accessing and using a variety of services, including servers, storage, databases, networking, software applications, and more, provided by a cloud service provider.

Instead of relying on local servers or personal computing devices, cloud computing allows individuals and organizations to access and use computing resources from remote data

centres. These data centres are typically owned and managed by third-party providers who specialize in delivering cloud services.

Key Characteristics of Cloud Computing

On-Demand Self-Service: Users can provision and access computing resources, such as virtual machines or storage, without requiring human interaction with the cloud service provider.

Broad Network Access: Cloud services can be accessed over the internet using a variety of devices, including laptops, smartphones, tablets, and thin clients, with standard web browsers or specialized client applications.

Resource Pooling: Computing resources are pooled together and shared among multiple users. Providers use virtualization techniques to partition and allocate resources dynamically based on demand.

Rapid Elasticity: Cloud resources can be scaled up or down quickly to meet changing demands. Users can request additional resources or release unused resources as needed, often with near-instantaneous provisioning.

Measured Service: Cloud computing systems automatically monitor and track resource usage, allowing providers to charge users based on their actual consumption. This pay-as-you-go model provides cost transparency and flexibility.

Benefits of Cloud Computing

Scalability: Cloud computing enables easy scalability by allowing users to increase or decrease resource capacity as required. This flexibility ensures that applications can handle varying workloads efficiently.

Cost Savings: Cloud services eliminate the need for upfront investments in hardware and infrastructure. Users can avoid the costs associated with maintenance, upgrades, and physical space, paying only for the resources they consume.

Reliability and Availability: Cloud service providers often have redundant systems and data centers, ensuring high availability and data durability. They also offer Service Level Agreements (SLAs) that guarantee a certain level of uptime and reliability.

Agility and Speed: Cloud computing allows organizations to quickly deploy applications and services, reducing the time to market. Development and testing processes can be accelerated, leading to faster innovation and competitive advantage.

Global Reach: Cloud services are accessible from anywhere with an internet connection, making it easier to reach a global audience and collaborate across geographical boundaries.

Cloud computing has revolutionized the IT industry by providing a flexible, scalable, and cost-effective model for delivering computing resources. It has become an essential

technology for businesses of all sizes, enabling them to focus on their core competencies while leveraging the power and versatility of cloud services.

The three benefits of cloud computing are to make more money, save time, have secure access, and eliminate access restrictions in anywhere, anywhere, and anywhere. It is modern and state-of-of-the-the-the-the-art technology computer science. The word "cloud" uses the term "networking" and the terms "compute" and monetary compensation to describe both means computing and money. It's an internet-based technology, providing locations and accessible and low cost third-party access as service and abundant third-party cloud-based options The Internet offers a lot of different service options for people who want to use virtual computing, including those offered by Amazon Web Services, as well as other cloud services such as Google Cloud. [With improved protection and flexibility,] Becomes more secure and more versatile as the business needs shift. If I call the number, which I will in a moment, there's no guarantee it will be picked up, but, if I call it now, there's a seventy-five percent chance you'll be in four rings

Platform-as-as-a-a-a-Service (PaaS) and Software-as-a-a-a-Service (SaaS) Cloud Computing mean using the Internet to provide a hosting service on which software and/tools are built; SaaS is a collection of commercial products that may or may not additional commercial licences and technology (NIST) standardised method 8" a model which enables IT users to easily have a common pool of customizable resources that is rapidly provisioned and released with minimum effort while it also making those resources available to those who need them on-demand and instantly" (NIST).

I know that it isn't terribly romantic, but as far as getting there on time goes, a corporate ladder is as good as you can get.
The three major key points or highlights of the NIST definition of cloud computing can be stated as follows:

- Software as-a-Service(SaaS)
- Platform-as-a-Service(PaaS)
- Infrastructure-as-a-Service(IaaS)

These services are used by Cloud Service Provider (CSP) in order to offer cloud services to it’s consumers. Along with these services NIST definition is also used to provide following four models [3] :

- Private Cloud
- Hybrid Cloud
- Community Cloud
- Public Cloud

Apart from these services and models an integrated view of following five essential and unique characteristics that can be found in every cloud service:

- On-demand Self Service
- Resource Pooling
- Measured Services
- Rapid Elasticity
- Broad Network Access

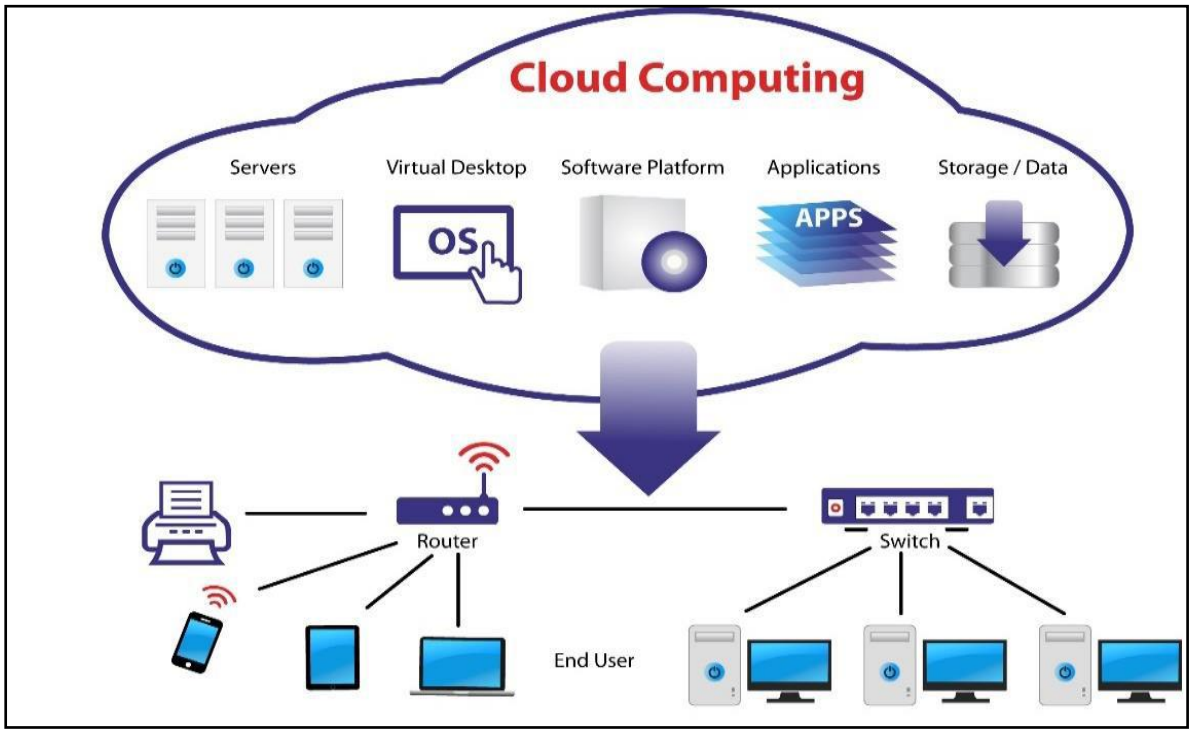

Figure 1.1: Cloud Computing

There are four types of cloud computing deployment models:

1. Private Cloud For a private cloud, cloud-based application and processing environment can provide customers just the set cloud computing model.[4]
2. Public Cloud: It is a third party on the public Internet to have the user computing infrastructure that you want to use offered resources and applications services. [4]
3. Cloud on Community Public Cloud is a coalition that can be divided between them by a number of infrastructures that provides organization, which means that it is shared between applications. [5]
4. Hybrid Clouds Two Clouds Hybrid is a mixture of public and private cloud.

The type of cloud service, which provides three models of data modes, is shown in the figure below:

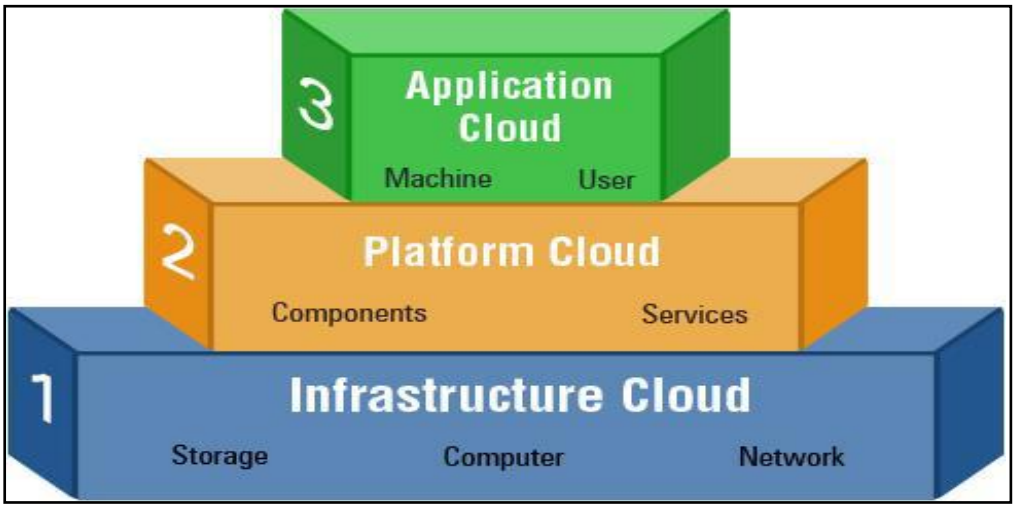

Figure 1.2: Cloud Service Model

The nature of NIST networks, measured on demand for cloud computing services Self-service, resource pooling and rapid elasticity.

1. Benefits of Cloud Computing

The benefits of the cloud computing offerings they are making, some of them are listed below: [6]

- Where to configure (setup) and use applications to build.
- There is no need to use any software and install build cloud-based applications.
- It provides a flexible and scalable reliable service.
- It is self-service, which can be used without repeated use of cloud service provider resources.
- U-Go payment based on cost increases.
- Users can also easily cloud resources and measures anywhere in the cloud.

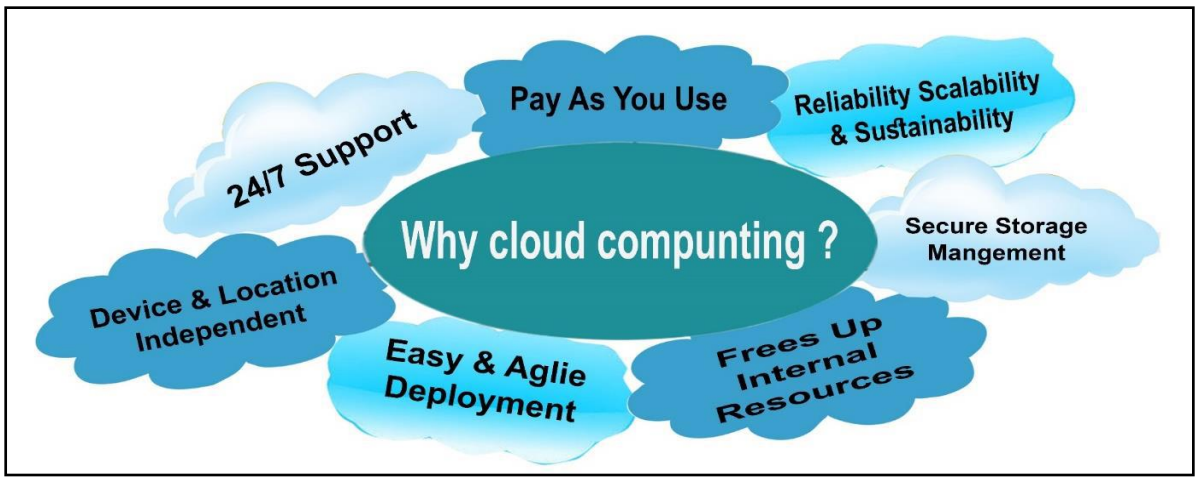

Figure 1.3: Cloud Computing Advantages

II. CLOUD COMPONENTS

As components such as cloud client systems, data centers and distributed servers. An exact point of any element and plays a definite role. [7]

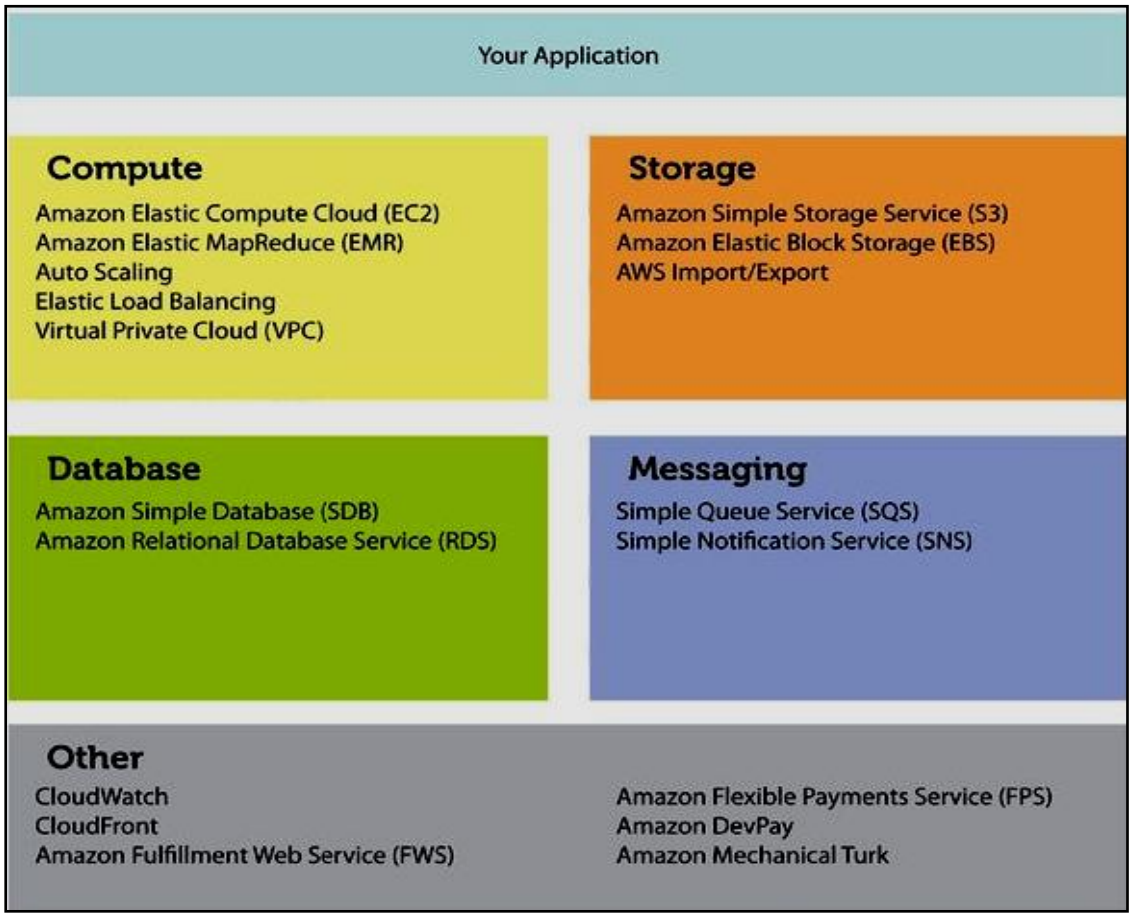

Figure 1.4: Cloud Computing Components

1. **Clients End:** Users to interact with customers managing cloud-related information. Customers fall into three general categories [8]
    - Mobile Phones Smartphone and more.
    - **Thin:** They do not perform their calculations only to show information. Batman does all the work for them. Thin client is no internal memory.
    - **FAT:** Internet such as IE and Mozilla Firefox or Google Chrome connect to web browsers such as a cloud and. When the cost of quality customers, security, low power consumption, low noise, easy and more popular, thin customer for a few more days, so the quality will be improved.

2. **Datacenter:** A collection of various applications of servers build data center. The dataset end user is connected to subscribe to various applications. Datacenter customers may be present over large distances. Now, in a system deployment program, use a technique called virtualization which allows multiple instances of virtual server applications. [9]

3. **Distributed Servers:** The host is part of a distributed server in the Internet cloud for various applications. But when using the application for the cloud, you will see that your machine will get access to this feature. [10]

2. Type of Clouds

Used cloud-based environment domain, the cloud can be divided into three categories.

- Hybrid Clouds
- Private Clouds
- Public Clouds

2.1 Services provided by Cloud Computing

Service that provides a wide range of different types of servers of applications. This will work as a "normal". [1] as three types of cloud services: [11]

- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Hardware as a Service (HaaS) or Infrastructure as a Service (IaaS).

1. Software as a Service (SaaS)

SaaS, using various web applications across some web from different servers. The use of software does not require integration with change, because without this one change or other system there is a lot of requirement. [2] With all the upgrades and patching that the existing infrastructure provider provides. If using the software, the customer must pay. For the software to operate without the need to interact with other systems, it is easy to make it an ideal candidate for software as a service. Clients are not ready to do software creation, but the legislation may also benefit from the need for high capacity applications. [12]

2. Platform as a Service (PaaS)

PaaS offers all the tools required to create apps and services entirely from the Web, without downloading or installing software. The PaaS services are the design, development, testing, installation and hosting of software. Other resources can include team coordination, database integration, and integration of web services, data security, and storage. [13]

III. LOAD BALANCING

Load balancing refers to the process of distributing network traffic across multiple resources, such as servers, virtual machines, or application instances, to optimize resource utilization, enhance performance, and ensure high availability. It plays a critical role in managing and optimizing network traffic in various computing environments, including data centers, cloud computing, and content delivery networks (CDNs).

The primary goal of load balancing is to prevent any single resource from becoming overloaded while ensuring that all resources are utilized efficiently. By distributing traffic across multiple resources, load balancing helps in achieving better performance, reducing response times, and avoiding bottlenecks.

Load balancing can be implemented at different layers of the network stack, depending on the specific requirements and architecture of the system:

Network Load Balancing: At the network layer (Layer 4), network load balancers distribute incoming traffic based on protocols, such as Transmission Control Protocol (TCP) or User Datagram Protocol (UDP), and port numbers. They typically operate using algorithms like round-robin, least connections, or weighted distribution.

Application Load Balancing: At the application layer (Layer 7), application load balancers make distribution decisions based on more advanced criteria, including HTTP headers, URL paths, or session information. They have the ability to inspect and manipulate the content of network packets and can perform additional functions like SSL termination, content-based routing, or application-specific optimizations.

Load balancing algorithms used by load balancers can vary, and the choice depends on factors such as traffic characteristics, resource capabilities, and desired performance objectives. Some commonly used algorithms include round-robin, least connections, weighted round-robin, least response time, and consistent hashing.

Load balancers continuously monitor the health and performance of the resources to which they distribute traffic. If a resource becomes unavailable or performs poorly, the load balancer can automatically detect the issue and stop sending traffic to that resource until it recovers.

Load balancing techniques can also involve session persistence, where subsequent requests from a client are directed to the same resource to maintain session state. This is important for applications that require session affinity or sticky sessions.

Load balancing can be implemented using dedicated hardware or software solutions, or it can be provided as a service by cloud providers. Many cloud platforms offer load balancing services that automatically distribute traffic across instances or resources within their infrastructure, providing scalability, fault tolerance, and ease of management.

Overall, load balancing is a critical component of modern computing architectures, enabling efficient resource utilization, scalability, and high availability for applications and services. It ensures that network traffic is effectively managed, optimized, and delivered to provide a seamless user experience and meet performance requirements.

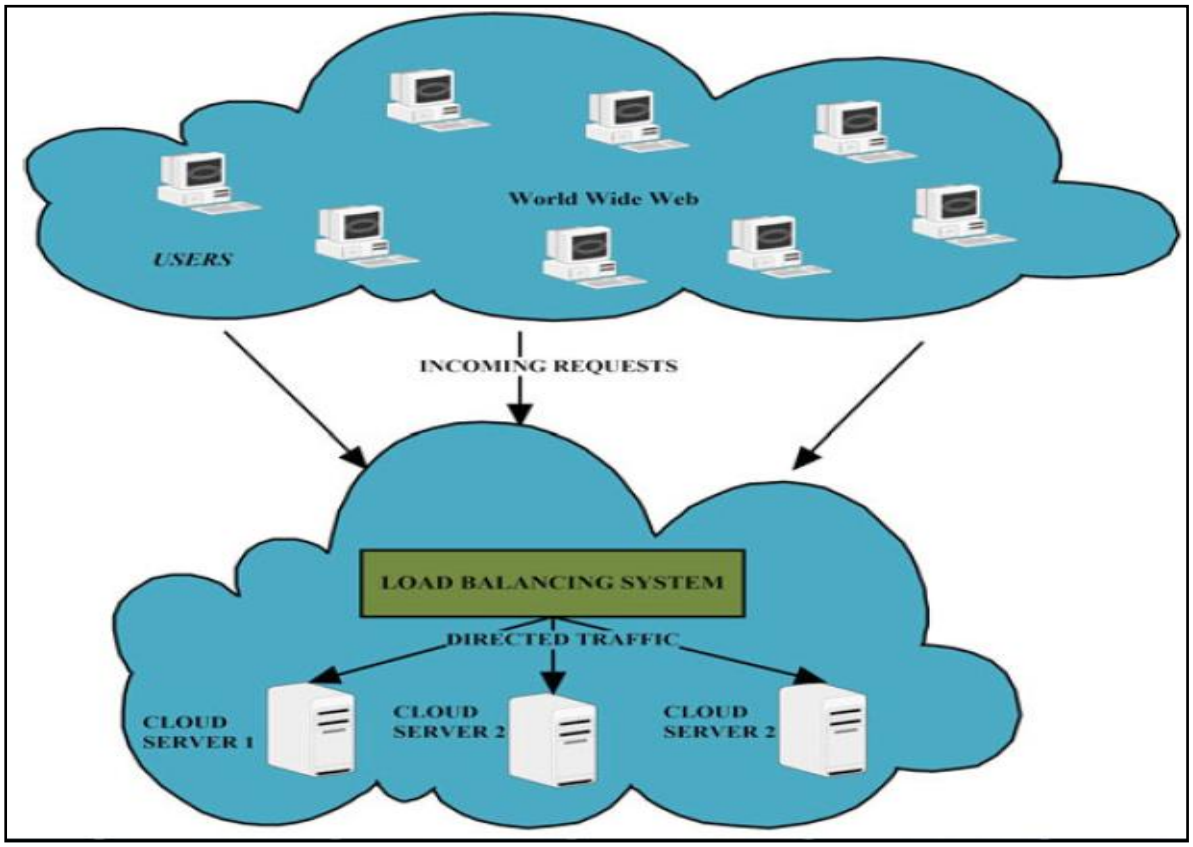

Figure 1.5: Load Balancing System In cloud Computing

1. Goals of Load Balancing

The primary goals of load balancing are to optimize resource utilization, enhance performance, and ensure high availability in computing environments. Load balancing techniques aim to achieve the following objectives:

Efficient Resource Utilization: Load balancing distributes incoming network traffic evenly across multiple resources, such as servers, virtual machines, or application instances. By spreading the workload across available resources, load balancing ensures that each resource is utilized efficiently and effectively.

Improved Performance: Load balancing helps to prevent any single resource from becoming overloaded or overwhelmed with traffic. By distributing the workload, it reduces response times and minimizes the chances of bottlenecks, thereby enhancing the overall performance of the system.

Scalability: Load balancing facilitates the scaling of resources to handle increased traffic or workload. As the demand for computing resources grows, load balancers can dynamically allocate traffic to additional resources, enabling the system to scale horizontally and accommodate higher loads.

High Availability: Load balancing plays a crucial role in ensuring high availability of services and applications. By distributing traffic across multiple resources, load balancers can

detect failures or unresponsive resources and redirect traffic to healthy resources, thereby minimizing downtime and improving the overall availability of the system.

Fault Tolerance: Load balancers can monitor the health and performance of resources, and if a resource becomes unavailable or experiences issues, they can automatically route traffic to other available resources. This fault tolerance mechanism helps maintain service continuity and prevents single points of failure.

Load Distribution: Load balancing algorithms distribute traffic evenly across resources based on various criteria such as round-robin, weighted distribution, least connections, or response time. This ensures fair distribution of workload and prevents any single resource from being overwhelmed.

Flexibility and Adaptability: Load balancers provide the flexibility to add or remove resources dynamically without impacting the availability or performance of the system. They can adapt to changing traffic patterns and adjust the allocation of resources accordingly.

Cost Optimization: By efficiently distributing traffic and optimizing resource utilization, load balancing helps to reduce infrastructure costs. It allows organizations to make the most of their existing resources and avoid over-provisioning, which can result in unnecessary expenses.

Overall, load balancing aims to achieve efficient utilization of resources, enhance performance, provide fault tolerance, ensure high availability, and enable scalability in computing environments. By achieving these goals, load balancing contributes to the efficient and reliable delivery of services and applications to end-users.

IV. LITERATURE REVIEW/ PREVIOUS WORK DONE

Following the effectiveness of the same treatment regardless of whether the success is detected after two days, two weeks, two months, or two years (DCC). The controller was then asked to determine the best virtual machine to manage the workload-balance task, which gave rise to identifying the proper VM in the data centre. As the choke load balancing VM is an array, it is able to protect the availability of the virtual machines. When the VM requires more virtual memory to load or available memory is of a specific to another virtual machine, it is handed over to Cloudlet. If the customer does not get a VM, they must wait their turn. At the same time, VMs must always be placed in a stable state of equilibrium for a better VM experience. A significant drawback is that it operates only in virtual machine data centre environments I know that it isn't terribly romantic, but as far as getting there on time goes, a corporate ladder is as good as you can get.

"A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms" by S. Basha and R. G. Babu: This survey paper provides an extensive review of load balancing algorithms in cloud computing. It discusses various load balancing challenges, categorizes the algorithms based on their characteristics, and evaluates their strengths and limitations.

"A Review on Load Balancing Techniques in Cloud Computing Environments" by A. H. Alsafi et al.: This review paper presents an overview of load balancing techniques in

cloud computing environments. It covers both traditional and state-of-the-art load balancing algorithms, including dynamic, predictive, and application-specific approaches. The paper discusses the advantages and disadvantages of each technique and identifies open research challenges.

"**Load Balancing Algorithms in Cloud Computing:** A Survey" by M. A. Rahman and M. S. A. Latiff: This survey paper provides a comprehensive analysis of load balancing algorithms in cloud computing. It categorizes the algorithms into four groups based on their approach: static, dynamic, heuristic, and game-theoretic. The paper discusses the characteristics, benefits, and limitations of each algorithm and provides insights into future research directions.

"Load Balancing Techniques in Cloud Computing: A Systematic Review" by K. Kaur and A. Rani: This systematic review paper examines load balancing techniques in cloud computing from a practical perspective. It analyzes and compares various load balancing algorithms, including round robin, least connection, and weighted round robin. The paper also discusses factors influencing load balancing decisions and highlights the importance of energy efficiency in load balancing algorithms.

"**Load Balancing in Cloud Computing:** A Systematic Review" by S. R. Saravanan and R. P. Kumar: This systematic review paper explores load balancing techniques in cloud computing, emphasizing the challenges and opportunities in achieving load balancing across cloud infrastructures. It discusses various load balancing algorithms, including centralized, decentralized, and hybrid approaches. The paper also presents a comparative analysis of different load balancing mechanisms.

These papers provide a starting point for conducting a literature review on load balancing in cloud computing. They cover a wide range of load balancing techniques, challenges, and future research directions. By reviewing these works, you can gain insights into the existing approaches, identify gaps in the literature, and form a solid foundation for your own research on load balancing in cloud computing.

Models of service delivery, characteristics, and stakeholders are among the topics of cloud protection. virtualization, isolation, and service orientation have been given greater weight. One is a scalable and sophisticated security configuration based on research has been suggested.

The four layers of cloud computing services are considered to be the "baked in" to the hardware, "running through" the network", "born on the web", and "running through the application" layers It highlights security concerns such as automatic network provisioning, virtual machine relocation, server consolidation, data recovery, and energy management in the cloud. If I call the number, which I will in a moment, there's no guarantee it will be picked up, but, if I call it now, there's a seventy-five percent chance you'll be in four rings

When you outsource cloud computing, data, software, virtualization, complexity, extensibility, and service levels, as well as regulation, take their places. The security model, which includes a centralised management, identity, control, access, and accounting, and privacy, was proposed in order to provide security, trust, user authentication, and policy

management. I know that it isn't terribly romantic, but as far as getting there on time goes, a corporate ladder is as good as you can get.

Virtualization and networking issues are the three big concerns in IaaS. Secure network threats that deal with these three different types of threats have been researched, and practical solutions for each have been proposed in numerous ways, among them are network access control and encryption techniques which are widely considered to be the mainstops for the complete security of the infrastructure. [We learned] what lies at the heart of people's relationship with products they can actually afford, rather than what they felt would lie at the heart of their relationship with things they could buy.

The algorithm maximises energy and assures reliability in data centres. To make things run more smoothly, this algorithm tends to use even more CPU power, making it a heavier on the server, like having three parameters. This dynamic algorithm measures the load for each virtual machine and divides the load by the amount of time it's complete, and allocates the workload to another according to how long and how quickly it runs. He claims to be worth about a billion dollars, but he couldn't prove it because he left no audit trail.

This algorithm is equal and accurate when conducting resource allocation in data centres. This approach considers server load, server resource usage, and task time as three parameters. Using this algorithm, we calculated each virtual machine's load and power consumption and weighted the tasks so that they will go to the VMs with the same percentage completion and also achieve maximum efficiency. He grabbed her gently by the wrist and said, "There are only two kinds of people in this world; those who grab back, and those who don't get grabbed."

Contributors to the resource provider must be well-distributed. Processing data as quickly as possible and spreading the workload across many virtual machines is a Cloud Fabric benefit. Effective load distribution is achieved by this form of resource management technique. This system incorporates the performance distribution (CE) and CPU throttling algorithm. The best part of life, we all discover as we grow up, is to find that we are no longer embarrassed about anything. The hour of awkwardness has passed.

Using successful algorithms to accomplish complex requests is one way to improve machine efficiency. As is has already been mentioned, the resource allocation used for priority virtual machines (PVMs) delivers optimal response and processing time. One thing that load balancing provides is, and something that it must ensure, is that all operations in the network are powered at all times. To reduce the cloud resource consumption and response time, the provision of different priority levels of resources should be done. If I try to judge another book by its cover, I will waste my time and give up far too soon.

Creative say: Burstiness is needed for this process. Recycle Ratio can be used in non-explosive formation. In an attempt to accommodate the user's needs, we use fuzzy authentication. To illustrate, or validate, experimental results show that the algorithm delays the mean response time and increases the average processing time having had two jobs so far, she claims that she has found a role she enjoys more than anything else in the world is still to be in fashion

Honey will ensure virtual machines are all used at the same priority. More pre-loaded storage is better, virtual machines cost-effective, and pre-emptsales behave reasonably well. For a better use of resources, we can minimise the time needed to get the job done and the job cost. He likes doing each day's small jobs differently so he has a better overview of how far he's come and how far he has to go each day.

Consider the concept of weighted load distribution was used in VM DLB. with the round-robin algorithm The virtual algorithm simulation results revealed that the workload is equally distributed among all of the virtual machines

The enhanced load balancing mechanism employed genetic strategies to pick load-balancing mechanisms For instance, design a loading strategy that balances the resources you start with as well as other constraints, such as the exercise programme you must implement. Limitations of the algorithm decrease the algorithm's output significantly. These advanced genetic procedures performances and algorithms work better than traditional approaches. I wonder if someone somewhere out there loves me the way that I love and if he/she will find me as beautiful and fascinating as the others before

The current two-level server distribution architecture on the Tolev cloud (via Global Load Balancing) utilises a system that is a cloud known as Global Load Balancing Framework. This implies that the device load is independently handled between the data centres is seen with this framework. If you stand long enough in the cold, [sometimes] you are likely to get sick of standing still. Furthermore, to achieve this goal, provide customers with high quality of service with low workload or burdened with requests.

A load balancing algorithm serves to distribute the same purpose as computer resources: load dynamic, load dependent data centres, and being essential in relation to computing resources. It is calculated in this article. The proposed data centre algorithm is meant to improve computational performance while simultaneously reducing response time for user requests. To re-frame one's thinking in order to consider something differently is not to think different but to think in new ways, see it in a different light, and have it spark an alternative theory

V. PROPOSED WORKS AND IMPLEMENTATION

1. Proposed Work

Credit-based scheduling is a technique used in cloud computing environments to allocate resources based on a credit system. It aims to provide fairness, prioritize resources, and ensure efficient resource utilization. Here's an overview of credit-based scheduling:

Credit System: In credit-based scheduling, each resource or user is assigned a certain amount of credits or tokens. These credits represent the entitlement or priority of the resource/user to utilize the system resources. The number of credits can be determined based on factors such as resource capacity, user subscription level, or predefined policies.

Credit Consumption: As resources are utilized or tasks are executed, credits are consumed or deducted from the allocated amount. The rate at which credits are consumed can depend

on factors such as resource usage intensity, duration of resource utilization, or the priority level of the task.

Credit Replenishment: Credits are periodically replenished to ensure fairness and prevent starvation. The replenishment rate can be fixed or dynamically adjusted based on system load, resource availability, or predefined rules. Replenishment allows resources with low credit balances to regain their entitlement and ensure that all users get their fair share of resources.

Resource Allocation: When a new task or job is submitted, the scheduler considers the credit balance of the requesting user or resource. Resources with higher credit balances are prioritized and allocated resources first. This ensures that resources/users with more credits have higher access to system resources and helps prevent resource hoarding.

Fairness and Quality of Service: Credit-based scheduling aims to provide fairness in resource allocation by ensuring that all users have an equal opportunity to utilize the system resources. It prevents any single user or resource from monopolizing the resources and promotes a more balanced distribution. Quality of service (QoS) can be enhanced by giving higher credits or priority to certain types of tasks or users based on their requirements or service level agreements (SLAs).

Credit Exchange: In some credit-based scheduling systems, credits can be exchanged or transferred between users or resources. This allows users with excess credits to transfer them to users in need or to obtain additional resources for themselves. Credit exchange mechanisms can further enhance fairness and optimize resource utilization.

Credit-based scheduling provides a mechanism to allocate resources in a fair and efficient manner in cloud computing environments. It helps prevent resource monopolization, ensures a balanced distribution of resources, and prioritizes tasks or users based on their entitlement. By effectively managing credit consumption and replenishment, credit-based scheduling contributes to improved resource utilization and better overall system performance.

Preferred tools are best used for an effective and on-demand load balancing strategy as well as reducing over-subscription and under-de-load. This is correlated with every host in the data centre. Anyone can use any kind of virtual machine (hosted or guest) under the controller datacenter. Using the stack balancing table does not explain the virtual machine parsing requirements until the stack is built. In photography, a focal point (often called the camera's point of aim) is something that focuses attention on one aspect of the image to make it more prominent.


```
  / / Initialize credits for each resource/user
for each resource in resources:
   resource.credits = initial_credits

while tasks_exist:
   // Find the resource/user with the highest credits
   selected_resource = find_resource_with_highest_credits(resources)

   // Get the next task from the task queue
   next_task = get_next_task()

   // Execute the task on the selected resource
   execute_task(next_task, selected_resource)

   // Update credits for the selected resource
   selected_resource.credits -= credit_consumed_per_task

   // Replenish credits periodically
   if time_to_replenish_credits():
      for each resource in resources:
         resource.credits += credits_replenished_per_interval

   // Check if tasks are remaining in the task queue
   tasks_exist = check_for_remaining_tasks()
```

2. Task Size (Min MinAlgorithm)

Successful use of assets can be increased using online adjustment calculations. This is done using unused (passive) property display processor processors, while considerable weight properties. The stack adjustment calculation can be accessed by moving between a lot of employees. The calculation also limited makeup with considerable use of the property. Count Min-min is the beginning of everything. Attempts to set up are also setting some assets and businesses. Remember, to perform the calculation to be mapped, a set of the work set property will be sent to. errand raises the minimum size calculation.

Assets will be distributed in which the task will lead for the shortest time. The effort after completing the work is motivated by the wrong set. Tired air is the result of calculation until the work set. Turn on the task to set T1, T2,..... etc. and set assets R1, R2,….. etc. Asset my distance consumption Ctij as expected time j. It is determined by utilizing the condition 1.

$$Ct_{ij}=Et_{ij}+ rt_j \quad (1)$$

Rtj represents the ready time of resource Rj. Etij stands for execution time of task i. Pseudo Code of MinMin algorithm is represented below.

The principle of the point of calculation is that the idea is not only to increase the length of customer needs. May exist where higher requirements of the business. Need to consider yourself then booking is necessary.

1. **For**allsubmittedtasksintheset;T_i
2. **For**allresources;R_j
3. $Ct_{ij}=Et_{ij}+rt_j$;End **For**;End**For**;
4. **Do**whiletaskssetisnotempty
5. Find task T_k thatcost minimumexecutiontime
6. Assign T_k to the resource R_j whilegivesminimumexpected
7. Complete time
8. Remove T_k fromthetaskset
9. Updatereadytime rt_j for select R_j
10. Update Ct_{ij} forall T_i
11. End**Do**

Theproposedapproachconsiderstwoparameters:

1. UserPriority
2. TaskLength

The calculation depends on the use of the credit card framework. Credit is allocated on the length of each fieldwork and their needs. Genuine actual planning, this credit will be measured.

Tasklengthcredit

Thecloud frameworks to execute a fixed length. The demand for the long haul in place would be off, with the introduction of a group of short-lengths to the company and a surprisingly long run becoming available in the direction of the previous work.

For planning reasons, calculation should be a forward and backward function, which can make it more robust. The credit framework dependent on undertaking length will function as pursues: The initial step is associated with finding the length of each assignment. The subsequent stage is ascertaining the normal of errands length. Starting a long guess can stop step instructions. Assignment turn sets T1, T2, T3 ... etc. 2 are used to find a long term difference to the countries normal length. This information is valuable when it is set in a cluster requesting an extension of the length of employment. The proposed calculation is not so bad with a large long or short length functions. It takes over every function of the center.

$$\mathbf{TLD_i = |len_{avg} - Tlen_i|} \qquad \mathbf{(2)}$$

Where TLDi length difference I, this is done by taking the absolute difference in the length of the ith function and the average price of the work. After ascertaining the difference in the length of each task, credit is assigned to each task. 5 credits in the algorithm and one credit given for each individual position. Before this step 4 different values of array length were found. They create conditions to establish 4 credits. We cannot select only four values. These values should work in long distances. The calculations are given below.

value_1=high_len/5
(3) value_2=high_len/4

(4) value_3= value_2+ value_1
(5) value_4=value_3+value_2
(6)

Wherehigh_len is the highest value of task length. This canbefound by Pseudocode is mentionedbelow.

Forallsubmittedtasksin the set;Ti

|$\mathbf{TLD_i = len_{avg} - Tlen_i}$|

If $\boldsymbol{TLD_i} \leq$ *value_1* t

hencredit

=5

elseifvalue_1 $< \boldsymbol{TLD_i} \leq$ *value_2*

thencredit

=4

elseifvalue_2 $< \boldsymbol{TLD_i} \leq$ *value_3*

thencredit

=3

elseifvalue_3 $< \boldsymbol{TLD_i} \leq$ *value_4*

thencredit

=2

elsevalue_4 $> \boldsymbol{TLD_i}$

thencredit

=1

End **For**

Thecreditsystembasedon tasklength

This algorithmadds creditsbasedon the task length. After thisstep each task will associatedwithacredit(Credit_Length$_i$)

Taskpioritycredit

It is also important to prioritize the tasks for booking errands. Can errand can have different requirements, which are spoken to as values assigned to each undertaking, and the respect for more than one task may be the equivalent.

The estimation of bookings based on the need of the undertaking has the issue of managing errands with comparable need. It does not occur in the proposed approach because,

given the fact that we give credits to each client depending on their needs, the last reservation would be based on aggregate credit, which depends on it.

Failure to book essential priority activity as well. It is obvious the type of property of will requirement which can be equal to any effort and consideration for more than one function, the truth does not differ. The company needs to be focused on issues relating to the real needs of comparative calculations. The suggested solution would be posted on the gross credit required Long and satisfying needs, depending on each given the fact that it arose as an problem in light of the fact that every business we are credited with needing to rely upon.

Forallsubmittedtasksintheset;Ti

Findouttaskwithhighest

priority(PriorityNumber)

Choose division_par

Foreachtask withpriority

Tpri findPri_frac(i)=Tpri

/division_factor setcreditas

Pri_frac

EndFor

ThecreditsystembasedontaskPriority

In the algorithm the first step is the priority number above. The algorithm is chosen to find the Pri frac, the split factor of the second level, for each mission. For example, if the third step in the Divijn part 1000 algorithm calculates Pri frac for each task the highest priority value of a two-digit number is 3 points to be selected as Divijn part 100. By dividing each task by one element of each task partition the priority value can be determined. Finally, a priority credit will be given each for this value (Pri frac).

$$\mathbf{Total\_Credit_i = Credit\_Length_i * Credit\_Priority_i} \quad \mathbf{(7)}$$

Lengths are dependent on the algorithm and priority, in the final stages of the total loan. The total debt is estimated using the equation given above. Credit is dependent on the long process of Credit Length i in the above equation. Credit priority based on previous payment. Eventually, the first job of highest credit quality is scheduled.

VI. IMPLEMENTATION

1. Cloud Analyst

There are some extremely good toolkits that can be used to model a virtual environment for testing the behavior of a large scaled Internet application. But it became obvious that it is much easier to provide an easy to use tool with a simulation level than just a toolkit. Within the initial version of the simulator, the statistical measures generated as output of the simulation follow.

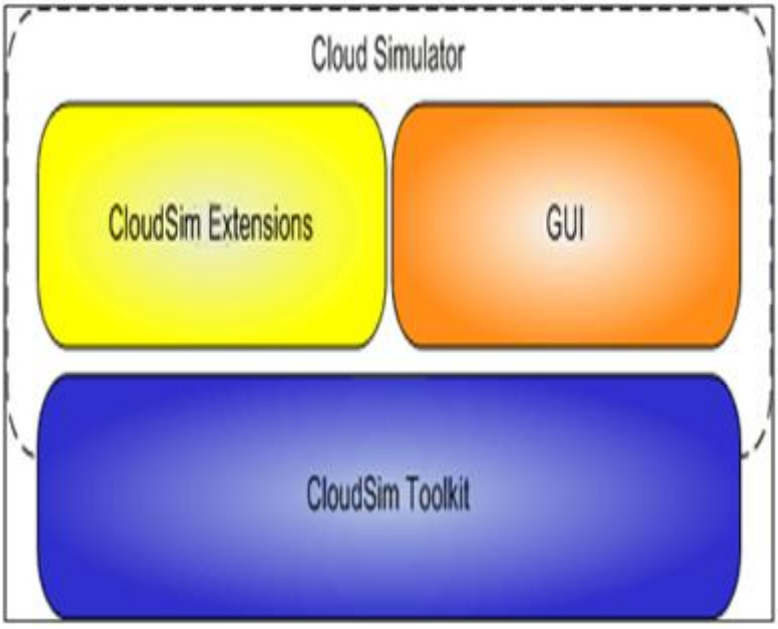

Figure 3.1: CloudAnalyst built on top of CloudSim toolkit

2. Simulation Measured Output

Following are the statistical measures produced as output of the simulation in the initialversion of thesimulator.

1. **Response time of the simulatedapplication**
   - Total mean, minimum and average response time of all simulated user requests.
   - A breakdown of response time by consumer groups across geographic regions.
   - The response time is further broken down by the time showing the trend of improvement over a day's length.

2. **The usage patterns of theapplication**
   - How many people use the program at what point in time from various parts of the world and the cumulative impact of this use on the program hosting data centers?

3. **The time taken by data centers to service a userrequest**
    - The total running time for all simulation requests.
    - The average, minimum and maximum time each data center has to process requests.
    - Pattern of variability in response time throughout the day as the load varies.
    - Business costs.

3.2.2 Technologies Used

- Java – The simulator, using Java SE 1.6, is based 100% on the Java Platform.
- Java Swing – Constructed using Swing modules, the GUI component.
- CloudSim-CloudSim apps are used in CloudAnalyst to model data centers.
- SimJava – Sim Java is the underlying CloudSim simulation platform and several of SimJava '**s features are included in CloudAnalyst direct.**

3.2.3 Use of CloudSimToolkit

3.2.3.1 Functionality Leveraged

CloudSim toolkit includes much of the comprehensive operations that take place inside a data center. This covers

- Simulation of the hardware concept of data centers in terms of physical equipment made up of processors, storage devices, memory and internal bandwidth
- The simulation, development and degradation of virtual machine specifications
- Virtual machine management, allocation of physical hardware resources for virtual machine service based on specific policies (e.g. time-shared and space-shared);
- Simulation of virtual machine execution of user programs or requests (Cloudlet / Gridlet)
- These apps are **used in CloudAnalyst directly.**

3.2.4 CloudAnalyst Domain Model and MainComponents

The figure given here summarizes the main components and domain entities of the CloudAnalyst:

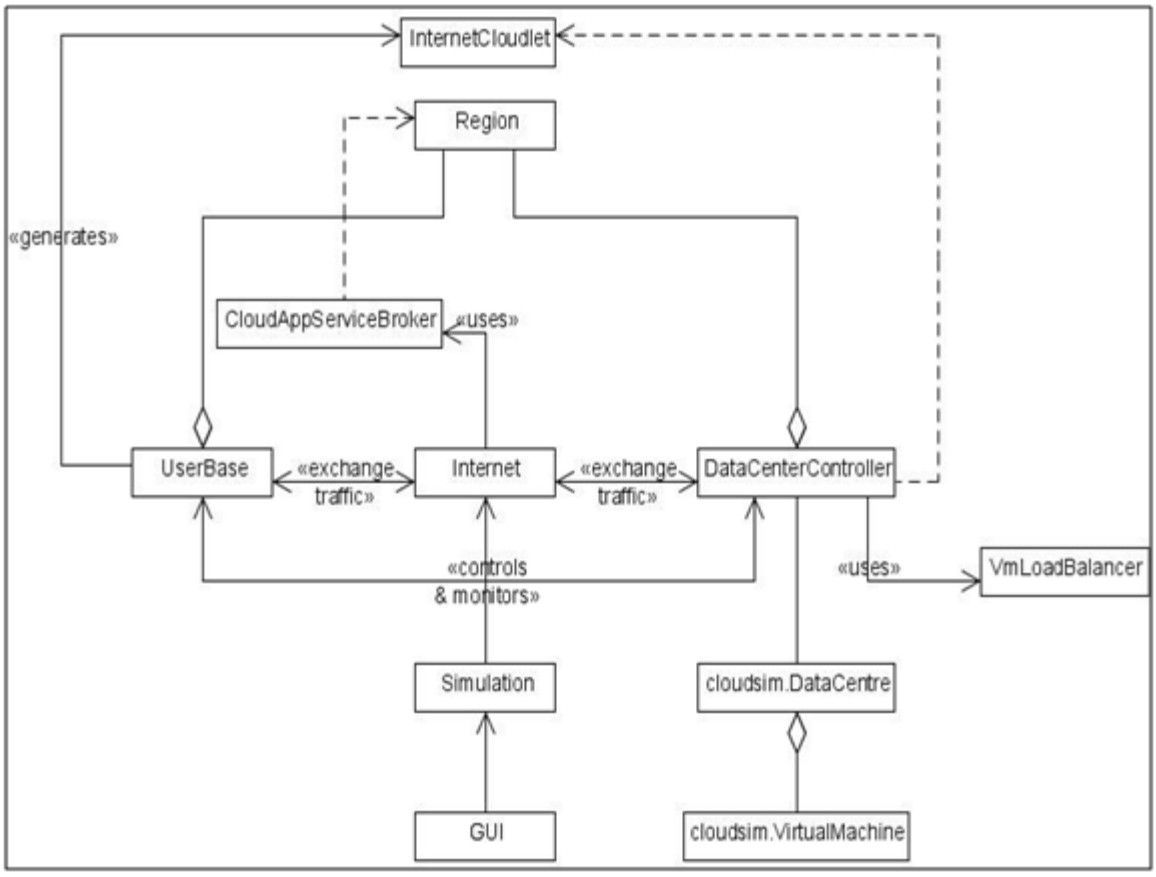

Figure 3.2: CloudAnalyst Domain

Such elements and principles are defined in greater detail in the following section. The three big extensions to CloudSim toolkit implemented in CloudAnalyst are User Base, Data Centre Controller and the Internet. But it is necessary to introduce the idea of "area" before describing those.

1. **Region:** The globe is divided into 6 'Regions' in the Cloud Analyst that correspond with the world's 6 major continents. Most of those regions belongs to other large institutions such as User Bases and Data Centers. This geographic classification is used to preserve a degree of practical simplicity in the Cloud Analyst for the wide scaled simulation being attempted.

2. **Internet:** The Internet Cloud Analyst is an abstraction for the Internet of the real world, adding only the features which are essential to the simulation. This models the routing of Internet traffic across the globe by adding sufficient transmission latency and data transfer delays. Configurable are the transmission latency and the available bandwidth between the six regions.

3. **Cloud Application Service Broker:** The routing of traffic between user bases and data centres is managed by a service broker who determines which data centre is expected to serve the requests from every user base. Current version of Cloud Analyst integrates three different types of service brokers each with different routing policies.

4. **Service Proximity Basedrouting:** In this case the proximity is the shortest route from a network latency based user base to the data centre. In terms of transmission latency the service broker must direct user traffic to the nearest data centre.

5. **Performance Optimizedrouting:** Here the Service Broker constantly tracks the performance of all data centres and guides the traffic to the data centre that it calculates to give the end user the best response time when it is being queried.

6. **Dynamically Reconfiguringrouter:** This is an extension of Proximity-based routing, where the routing logic is somewhat similar, but the service broker has the additional burden of scaling the implementation of the application based on the load it faces. It is accomplished by increasing or reducing the number of VMs allocated in the data centre, as compared with the best processing time ever achieved, according to the latest processing times.

7. **User Base:** A single user base can represent thousands of users, but it is configured as a single unit and traffic created in simultaneous bursts reflecting the user base 's size. The modelers will choose to use a User Base to represent a single user, but ideally they can use a User Base to represent a greater number of users for simulation performance.

8. **InternetCloudlet:** An Internet Cloudlet is a collection of requests from the users. In CloudAnalyst, the amount of requests packed into one Internet Cloudlet is configurable. The Internet Cloudlet holds information such as the size of an execution request instruction, the size of input and output data, the originator and target client I d used for Internet routing and the number of requests.\

9. **Data CenterController:** The Data Center Controller is undoubtedly the CloudAnalyst's most significant person. A single Controller Data Center is connected to a single cloudsim. DataCenter handles and manages data center management tasks such as building and deleting VM and routes user requests obtained through the Internet to VMs. This can also be used as the framework used by CloudAnalyst to access the core features of the CloudSim toolkit.

10. **VM LoadBalancer:** A VmLoadBalancer is used by the Data Center Controller to decide which VM will be allocated to the next cloudlet to load. There are currently three VmLoadBalancers implementing three load balancing policies which can be selected by the modeler as needed.
    a. Load Balancer round-robin-uses a basic round-robin algorithm to assign VMs
    b. Active Monitoring Load Balancer – this version load balancing the tasks between available VM's in such a way that at any given moment, the number of active tasks on and VM is even out.

Throttled Load Balancer-this means that at any given time only a pre-defined number of Internet Cloudlets is assigned to a single VM. If there are more request groups in a data center than the number of available VM's, some of the requests will have to be queued until the next VM is available.

11. GUI: The GUI is implemented as a set of screens that enable the user to:

1. **Define the Simulationparameters**

    a. Define in depth the features of a Data Center including the server farm's comprehensive hardware specification.
    b. Define application deployment requirements, such as how many virtual machines will be
    c. Defines the user bases and their characteristics, such as number of users, peak and off-peak hours of use and traffic generation frequency.
    d. Defines basic features of the Internet including network latency and usable bandwidth.
    e. Define Internet specific characteristics including network latency and available bandwidth.
    f. Simulator output based parameters such as user request grouping variables when messages are sent from User Bases and when messages are allocated to Virtual Machines in the Data Centre.
2. **Save and load simulationconfigurations.**
3. **Execute simulations with the option of cancelling a simulation oncestarted.**
4. **View and save the results of the simulation with graphical outputs whereappropriate.**

3.2.5 Using the Cloud Analyst

Cloud Analyst comes with a robust Java Swing-built GUI. Each segment explains the screens briefly, and how to use them for setting up and running a simulation.

3.2.5.1 Setting up a Simulation

You need to carry out the following steps to set up a simulation. (Please notice that the above screens are discussed in depth in the next section.

1. Defines user bases – Using User Base organizations identify software users, their geographic distribution, and other resources such as user numbers, usage frequency and use trends such as peak hours. It is achieved on the Configure Simulation screen's Key page.
2. Define data centers – Use the Configuration screen's Data Centers tab to identify the data centers that you plan to use in the simulation. Defines all of the data center infrastructure and accounting elements here.
3. Allocate Virtual Machines for the application in Data Centers – Upon creation of the data centers, you need to allocate virtual machines to the simulated application using the Configurations screen's main tab. A data center set out in step 2 above is not included in the simulation unless it is allocated in this step. During this stage, you can assign multiple types of virtual machines to the same data centre.
4. Test and change the relevant parameters in the Configuration Screen 's Advanced tab.
5. Review and adjust the latency and bandwidth of the network matrices on the Internet features screen.

3.2.5.2 SimulatorScreens

Main Screen with SimulationPanel

Figure 3.3: CloudAnalyst Main Screen

The main screen is the first one shown when CloudAnalyst is started. It has a simulation screen with a world map on the right hand side and the main control panel on the left. As stated, the CloudAnalyst is dividing the world into six regions that roughly correspond with the six main continents. Locations of all elements in the simulation are defined for simplicity only by the area.

Control Panel options are:
1. Configure Simulation – takes you to the Configure SimulationScreen
2. Define Internet Characteristics – takes you to the Internet CharacteristicsScreen
3. Run Simulation – Starts thesimulation
4. Exit

At the start the simulator will be loaded with a simple default simulation.

Configure SimulationScreen

The Simulator Settings panel has three tabs.

1. MainTab

The key tab 's configuration choices are to:
1. Simulation time-the simulation period that can be given in minutes, hours or days
2. Application Bases Table-This is a table that lists all the simulation user bases. Every user base has configurable fields which are represented in the table by a single row.
   a. Name
   b. Requests per user perhour
   c. Region

d. Peak hours
e. Data size perrequest
f. Average users during off-peakhours
g. Average users during peakhours

For add or delete user bases from the setup, you can use the Add and Delete buttons next to row.

3. Application Deployment Configuration – This table shows how many virtual machines from the Data Centers tab, along with the specifics of a virtual machine, are allocated for the application in each data center. The following areas are:
    - Data Center – his is a drop-down listing of data center names that were generated in the Data Center tab.
    - Number of VMs – How many VMs from the chosen data center to be assigned to the application.
    - Image Size – a single byte VM file size.
    - Memory – amount of memory available to a singleVM
    - BW – amount of bandwidth available to a singleVM

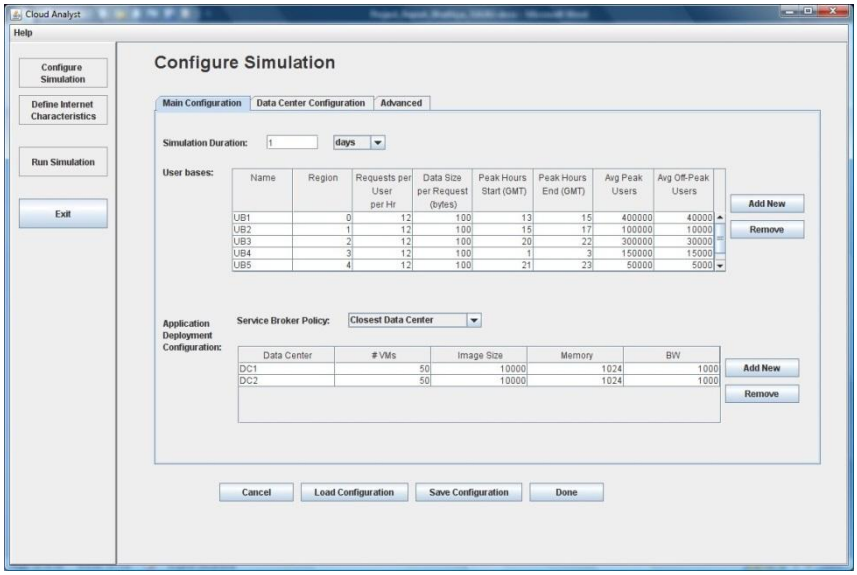

Figure 3.4: Configure Simulation Screen - Main Tab

4. Service Broker Policy – This drop down allows you to choose the brokerage policy between data centers that will determine which data center will receive traffic from which user base. Possible policies are:
    - All requests from that user base are sent to the datacenter with the least network latency (regardless of network bandwidth) from a specific user base.
    - This policy aims to balance the charge between data centers while overloading one data center.

The "Save Configuration" button lets you save the generated configuration as a file. Simulation files are saved using an extension to.sim. Likewise you can load a previously saved simulation configuration using the "Load Configuration" function.

1. Data Center Tab

The Data Centre tab lets you describe the data centre configuration (see Figure below). The table at the top lists the data centres and you can add or delete data centres into the configuration by using the Add / Remove buttons. The fields for the parameters are:

- Region
- Name
- Virtual Machine Monitor(VMM)
- Operating System – e.g.Linux
- Architecture – Architecture of the servers used in the data center. e.g.X86
- Storage cost perGb
- Cost per 1Mb Memory Hour
- Cost per VMHour
- Data Transfer cost per Gb (both in andout)
- Number of servers

'By choosing a data centre from this table a second table with the information of the server machines in the data centre appears below it. You may define the parameters for each system according to the fields available.

- MachineId
- Memory
- Storage
- Available networkbandwidth
- Number of processors
- Processor speed(MIPS)
- VM allocation policy (time shared/spaceshared)

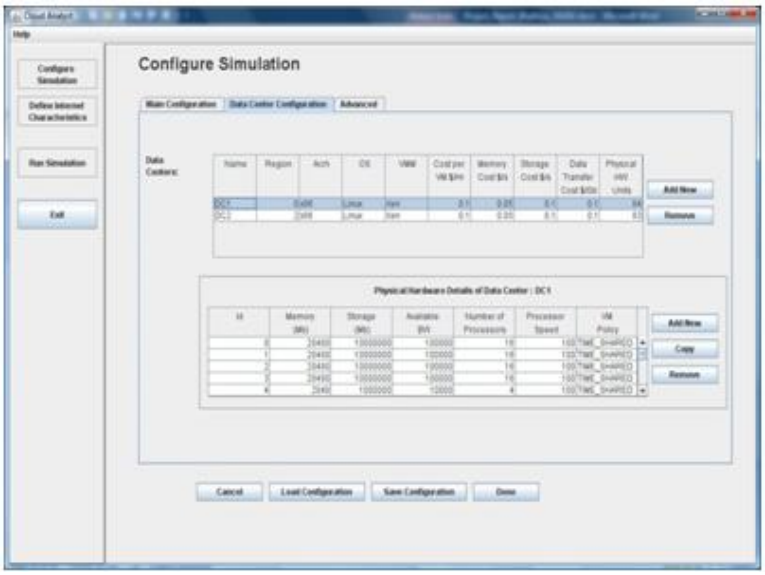

Figure 3.5: Configure Simulation Screen - Data Centre

2. Advanced Tab

The Advanced tab includes several essential parameters relevant to the whole simulation.

a. User Grouping Factor (in User Bases) –This parameter informs the simulator how many users should be handled for traffic generation as one single package. The number given here is being used as the number of requests that a single Internet Cloudlet can serve. In the ideal situation this parameter should be 1, expressed independently by each individual consumer. Yet that unrealistically increases simulation time.

b. Request Grouping Factor (in Data Centers) – This parameter informs the simulator how many demands for processing will be handled as a single entity. That is to say, so many requests are bundled together and allocated as units to one VM.

c. Again this should ideally be equal to 1. But this may also be seen as the number of concurrent threads that can be handled by a single program instance (VM).

d. Executable instruction length (in bytes) – This is the key parameter that influences a request's execution length. It is the same attribute used in GridSim as in GridletLength.

e. Load balancing policy – The strategy for load balancing used by all data centers when allocating requests to virtual machines. Policies Available are:

- Round-robin (RR)
- Equally Spread Current Execution Load – The load balancer keeps track of how many cloudlets each VM is currently processing, and attempts to even out the active load.
- Throttled – The load balancer throttles the number of requests that a single VM is allocated to. For the throttling algorithm see section 3.5.1.1.

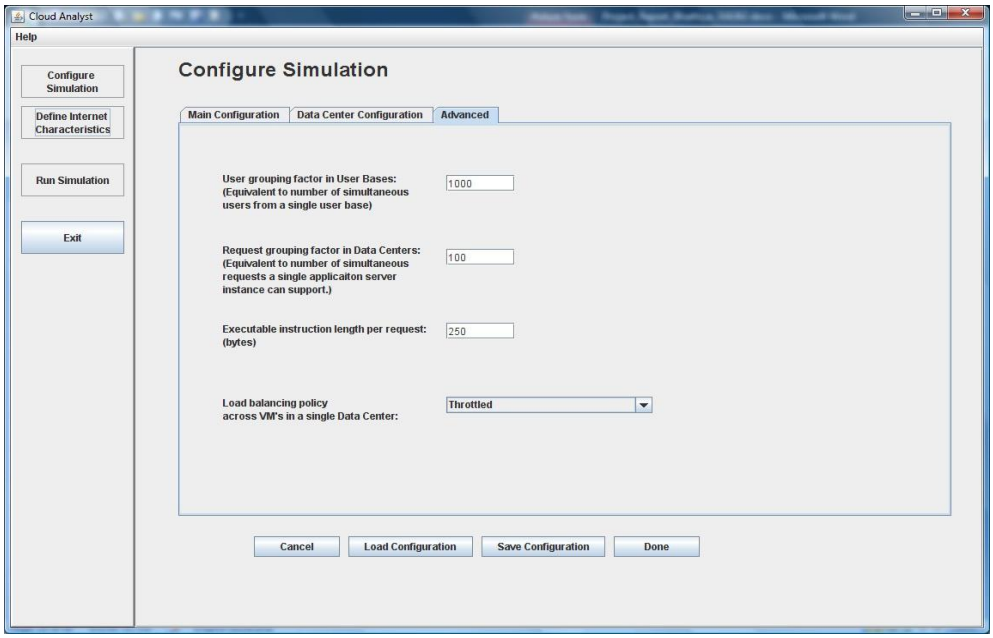

Figure 3.6: Configure Simulation Screen - Advanced Tab

iv. Internet CharacteristicsScreen

The Internet Features Panel can be used to set thresholds for internet latency and bandwidth. For those two groups it presents two matrices.

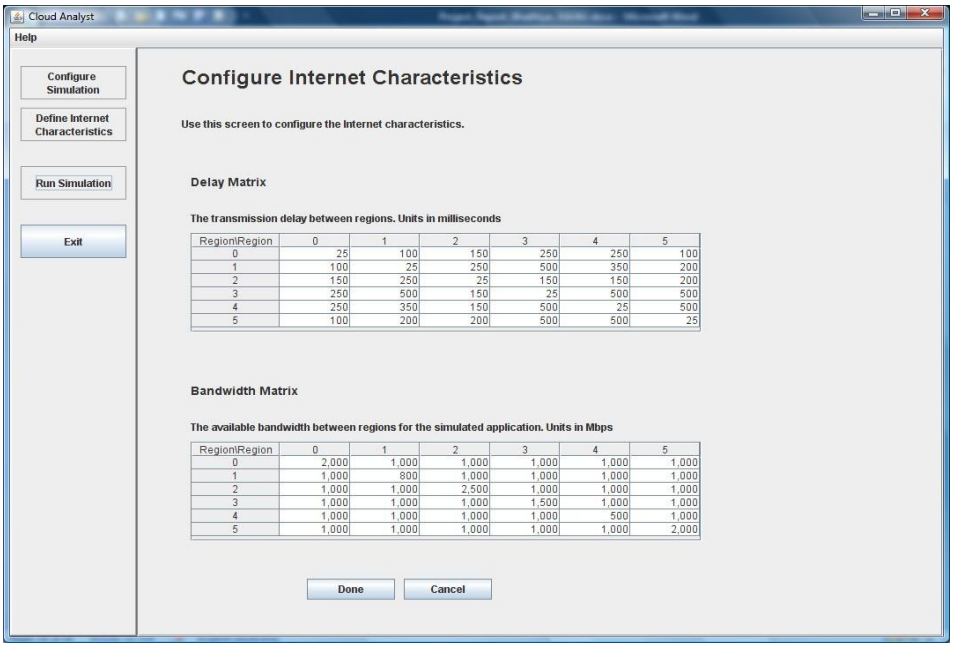

Figure 3.7: Internet Characteristics Screen

VII. RESULTSANDCOMPARISONS

4.1.1 Running a Simulation

By using the above screens to construct a simulation configuration successfully, the user must go back to the main screen and conduct the simulation by selecting the "Run Simulation" from the control panel. The simulation screen displays a basic animation that displays which user bases send messages to all data centers.

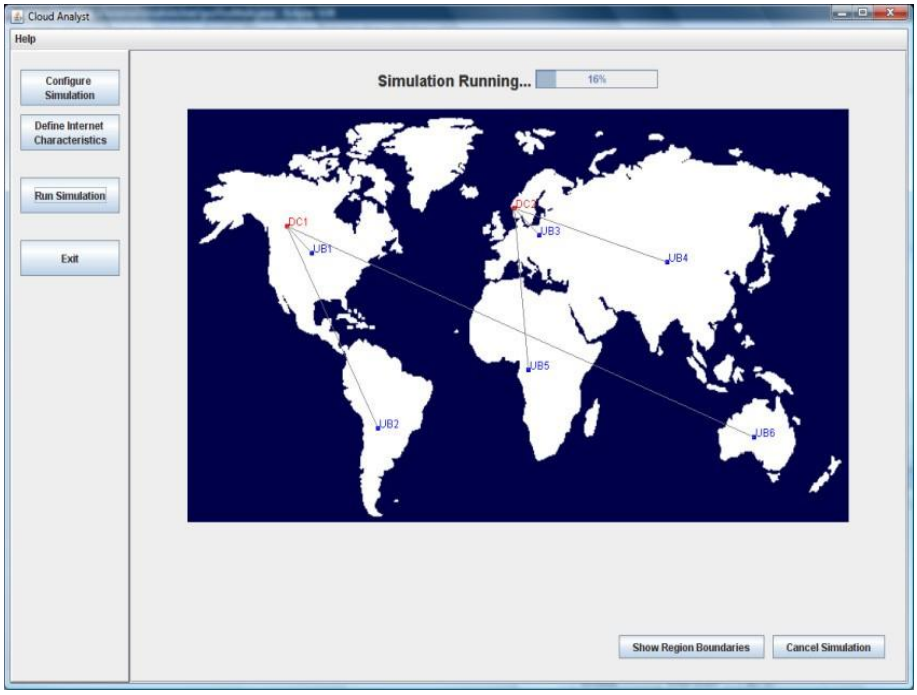

Figure 4.1: Simulation Panel During a Simulation

4.1.2 Results Screen

When the simulation is complete, the key response times are shown next to each user base on the simulation screen. Summary results can be accessed by clicking on the "Show Summary Results" button which will appear at the bottom right corner of the screen after the simulation has been completed.

The results screen shows the data from the simulation obtained. This covers:

1. A description of the average response time (for all user bases).
2. Response time in tabular format per user base.
3. User-based Response time in graphical format split into the 24 hours a day.
4. Request time of operation in tabular format by each data center.
5. Service time request by data center in graphical format, broken down into 24 hours a day.
6. Registration of data centers (number of requests served) in graphical format, broken down in 24 hours a day.
7. Cost details

VIII. CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

A systematic analysis of different balancing strategies has been carried out in this investigation. the current algorithms are static, dynamic, super-compositional, and hierarchical. The goal of these algorithms is to cut response time while using the resources as efficiently as possible. Better output can be obtained by restricting the previous algorithm's search criteria. We could use more innovation to further increase the efficiency of our cloud solutions. This study supports the findings obtained from the latest resource allocation methods as well.

5.2 Future Scope

In the future, we will expand our implementation as a whole to CPU, bandwidth, RAM as a parameter from the use of CPU as a parameter. This will increase load calculation performance which will reduce VM migration and energy consumption.

REFERENCES

[1] [1] Violetta, "Load Balancing in Cloud Computing", International Journal of Computer Applications (0975 – 8887) Volume 117 – No. 978-1-5386-4340-2/18/$31.00 ©2018 IEEE.
[2] [2] Mamta Khanchi, Sanjay Tyagi, "An Efficient Algorithm for Load Balancing in Cloud Computing" International Journal of Engineering Sciences & Research Technology, June, 2016.
[3] [3] Divya Garg, Urvashi Saxena," Dynamic Queue Based Enhanced HTV Dynamic Load Balancing Algorithm in Cloud Computing", International Journal of Innovative Research in
[4] Science, Engineering and Technology, Vol. 5, Issue 1, January 2016.
[5] [4] Sachin Soni, Praveen Yadav, "A Load Balancing Approach to Minimize the Resource Wastage in Cloud Computing", International Advanced Research Journal in Science, Engineering and Technology, Vol. 3, Issue 3, March 2016.
[6] [5] Navtej Singh Ghumman, Rajesh Sachdeva," an efficient approach for load balancing in cloud computing using composite techniques", International Journal of Research in Engineering and Applied Sciences, volume 6, issue 2 February, 2016.
[7] [6] G.Suryadevi, D.Vijayakumar, R.SabariMuthuKumar, Dr. K .G. Srinivasagan, "An Efficient Distributed Dynamic Load Balancing Algorithm for Private Cloud Environment", International Journal of Innovative Research in Science, Engineering and Technology, Volume 3, Special Issue 3, March 2014.
[8] [7] Sally F. Issawi, Alaa Al Halees, Alaa Al Halees, "An Efficient Adaptive Load Balancing Algorithm for Cloud Computing Under Bursty Workloads", Engineering, Technology & Applied Science Research Vol. 5, No. 3, 2015.
[9] [8] Khushbu Zalavadiya,Dinesh Vaghela , "Honey Bee Behavior Load Balancing of Tasks in Cloud Computing", International Journal of Computer Applications (0975 – 8887), Volume 139
– No.1, April 2016.
[10] [9] B. Nithya Nandhalakshmi , Mahalingam, "Efficient Load Balancing in Cloud Computing Using Weighted Throttled Algorithm", International Journal of Innovative Research in Computer and Communication Engineering ,Vol. 3, Issue 6, June 2015.

[11] [10] Er. Pooja Er. Vivek Thapar,"An Enhanced Virtual Machine Load Balancing Algorithm for Cloud Environment", International Journal of Emerging Research in Management &Technology, ISSN: 2278-9359 (Volume-5, Issue-5),May 2016

[12] [11] Po-Huei Liang1 and Jiann-Min Yang. "Evaluation of two level Global Load Balancing Framework in cloud environment" International Journal of Computer Science & Information Technology (IJCSIT) Vol 7, No 2, April 2015.

[13] [12] Navtej Singh Ghumman, Rajesh Sachdeva," an efficient approach for load balancing in cloud computing using composite techniques", International Journal of Research in Engineering and Applied Sciences, volume 6, issue 2 February, 2016.

[14] [13] Er. Rajeev Mangla, Er. Harpreet Singh," Recovery and user priority based load balancing in cloud computing", International Journal of Engineering and Science and Research, February 2015.

[15] [14] Harish Chandra, Pradeep Semwal, Sandeep Chopra," load balancing in cloud computing using a novel minimum makespan algorithm", International Journal of Advanced Research in Computer Engineering & Technology ,Volume 5, Issue 4, April 2016.

[16] [15] Saher Manaseer, Metib Alzghoul, Mazen Mohmad, "An Advanced Algorithm for Load Balancing in Cloud Computing using MEMA Technique", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-3, January 2019.

[17] [16] Durga Patel, Rajeev Kumar Gupta and R. K. Pateriya, "Energy-Aware Prediction-Based Load Balancing Approach with VM Migration for the Cloud Environment", © Springer Nature Singapore Pte Ltd. 2019.

[18] [17] S.Mohinder, R,Ramesh, D.Powar, ''Analysis of Load Balancers in Cloud Computing'', International Academy of Science, Engineering & Technology, vol.2, May 2013.

[19] [18] Poulami Dalapati1, G. Sahoo,"Green Solution for Cloud Computing with Load Balancing and Power Consumption Management"- International Journal of Emerging Technology and Advanced Engineering,Vol3:2013

[20] [19] Saurabh Kumar Garg and Rajkumar Buyya, " Green Cloud computing and Environmental Sustainability".(references)

[21] [20] Sidhu A, S.Kinger, "Analysis of Load Balancing techniques in Cloud Computing", Council for innovative research international Journal of Computer & Technology, vol.4, March-April 2013.

[22] [21] Sumalatha M.R, C. Selvakumar, T. Priya, R. T. Azariah, and P. M. Manohar, "CLBC-Cost effective load balanced resource allocation for partitioned cloud system", Proc. International Conference on Recent Trends in Information Technology (ICRTIT), 2014, 1-5.

[23] [22] The Amazon Elastic Compute Cloud (Amazon EC2), http://aws.amazon.com/ec2/

[24] [23] Yilin Lu," A Hybrid Dynamic Load Balancing Approach for Cloud Storage", 2012 International Conference on Industrial Control and Electronics Engineering 978-0-7695-4792-3/12 © 2012 IEEE.

[25] [24] Yuvapriya Ponnusamy, S Sasikumar,"Application of Green Cloud Computing for Efficient Resource Energy Management in Data Centres", International Journal of Computer Science and Information Technologies,Vol3:2012.

[26] [25] Ruhi Gupta. "Review on Existing Load Balancing Techniques of Cloud Computing." International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 2, February 2014.

[27] [26] Jinhua Hu," A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment", 3rd International Symposium on Parallel Architectures, Algorithms and Programming 978-0-7695- 4312-3/10 © 2010 IEEE (published)

[28] [27] Jayant Baliga, Robert W. A. Ayre, Kerry Hinton, and Rodney S.1Tucker, Fellow IEEE, "Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport".

[29] [28] S. Kapoor, and C. Dabas, Cluster based load balancing in cloud computing, Proc. Eighth International Conference in Contemporary Computing (IC3), 2015, 76-81.

[30] [29] Klaithem Al Nuaimi," A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms", 2012 IEEE Second Symposium on Network Cloud

[31] [30] Angona Sarker,Ali Newaz Bahar,SM Shamim, "A Review on Mobile Cloud Computing", International Journal of Computer Applications (0975 – 8887)

