Suresh Gyan Vihar University, Jaipur
International Journal of Converging Technologies and Management (IJCTM)
Volume 4 , Issue 1 , 2018
ISSN: 2455 – 7528

Optimal Resource Management in the Cloud Environment- A Review

R.S.Rajput , and Anjali Pant
Assistant Professor, Department of Mathematics, Statistics and Computer Science, College of Basic Sciences, G. B. Pant University of Agriculture & Technology, Pantnagar, India
Head of Applied Science Department, Govt. Polytechnic College, Shaktifarm, Uttarakhand, India

Abstract- Resource management in Cloud computing is demanding, and a new domain of research. In the present study, we have reviewed some critical components of Resource management like Cloud resources, components of cloud resources management, Resource management techniques and, Reference architectures of cloud computing environment. We also have to work on performance indices and some techniques that are useful for performance evaluation.

Keywords-Cloud computing, Resource management, Performance indices, Reference architectures of cloud computing environment, Queuing Theory, Jackson Network, JMT, CloudSim.

I INTRODUCTION

As stated in, cloud computing infrastructures consist of services that are offered and delivered through a service center, such as data center, that can be accessed through user interface such as the web browser, thin client, mobile app, remote desktop, terminal emulator, etc., anywhere in the world.

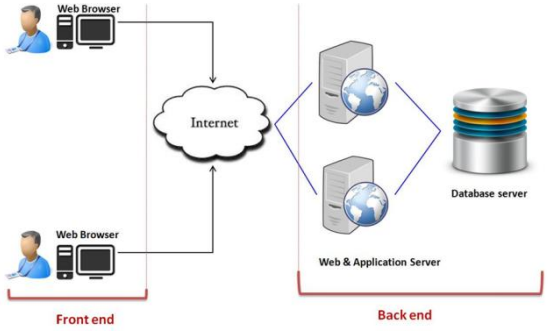


Figure 1: Cloud Computing System [20]

A cloud computing system, is divide it into two sections: - the front end and the back end. They connect to each other through a network, usually the Internet. The front end is the part seen by the client, i.e. the cloud user. This includes the client's network and the cloud resources accessed as a service. Backend part is cloud itself; It is a pooling of computing resources like servers, storage, memory in the physical or virtual form. Availability Zones of AWS is one of example of cloud backend.

Cloud computing system has four basic deployment models, three service models, and four key attributes. Deployment models of cloud computing system are defined on the type of access to the cloud, i.e., how the cloud is located? Cloud computing systems have four types of access: Public, Private, Hybrid and Community.

- Public Cloud: are developed, deployed and maintained by a third part service. The services within the public cloud have been developed for the public use.
- Private Cloud: services are developed, deployed, maintained and maintained for single enterprises. The private cloud provides more security and greater control than public clouds.
- Community Cloud: a cloud that is developed for sharing of resources by the several organizations. These clouds are designed for specific purpose like for security requirements.
- Hybrid Cloud: a cloud that is built by combining the above of three deployment models. These clouds provide the features of that clouds from whom that it is made.

Service model of Cloud computing is divided into three broad service categories: infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS).

- Infrastructure as a service: IaaS providers such as AWS supply a virtual server instance and storage, as well as application program interfaces (APIs) that let users migrate workloads to a virtual machine (VM). Users have an allocated storage capacity and start, stop, access and configure the VM and storage as desired. IaaS providers offer small, medium, large, extra-large, and memory- or compute-optimized instances, in addition to customized instances, for various workload needs.AWS and Microsoft Azure are some provider of IaaS.
- Platform as a service: In the PaaS model, providers host development tools on their infrastructures. Users access those tools over the Internet using APIs, Web portals or gateway software. PaaS is used for general software development and many PaaS providers will host the software after it's developed. AWS, Microsoft Azure, GAE are some examples of PaaS.
- Software as a service: SaaS is a distribution model that delivers software applications over the

Internet; these are often called Web services. Users can access SaaS applications and services from any location using a computer or mobile device that has Internet access. Sales Force, Google are some example of SaaS.

The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state or datacenter). Examples of resources include storage, processing, memory and network bandwidth.

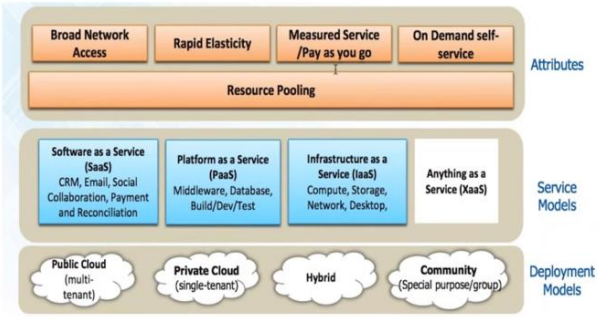


Figure 2: Cloud computing model [4]

A cloud computing system has four basic attributes: Broad network access, rapid elasticity, measured service and on-demand self-service.

- Broad network access: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops and workstations).
- Rapid elasticity: Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.
- Measured service: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth and active user accounts). Resource usage can be monitored, controlled and reported, providing transparency for the provider and consumer.
- On-demand self-service: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

Resource Management

All items used in the scenario of cloud are resources. Resources include computing, storage, networking and energy, resources directly or indirectly associated with the set of cloud applications. A resource is any physical or virtual component of limited availability within a computer system. Every device connected to a computer system is a resource. Resources can be categorized into two sub parts:-

Physical resources: CPU, Memory, Storage, Workstations, Network elements, Sensors/actuator

Logical Resources: Operating System, Energy, Network throughput/bandwidth, Information Security Protocols, API, Network loads, delays

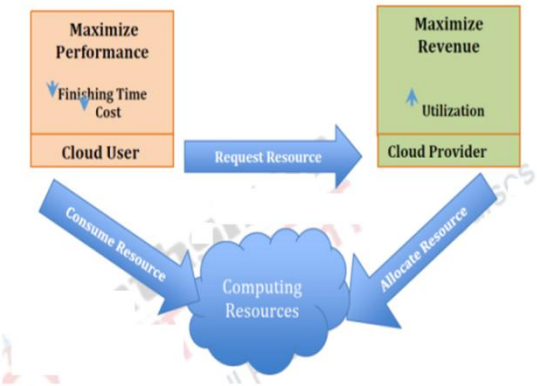


Figure 3: Resource usage scenarios in cloud [32]

Resource management is the process of allocating and monitoring the allocated as well as free resources. In the context of resource management there are two entities, first is the cloud providers and the second is cloud users. The major objective of the cloud providers is to provide an efficient and effective resource utilization maximizing the revenue earned while staying within the boundaries of service level agreements (SLAs). It is the job of the cloud provider to procure and maintain such facilities from where the resources and leased out to the cloud user. The cloud user's concerns are to maximize the allocated resource utilization with minimum cost.

The goal of resource management in Information technology is to provide high availability of resources, sharing of resources, fulfilling time variant service model, providing efficiency, and reliability on resource usage. From the cloud computing perspective, resource management is a process which effectively and efficiently manages above mentioned resources as well as providing QoS guarantees to cloud consumers.

Objectives of Cloud Computing Resource management

Suresh Gyan Vihar University, Jaipur
International Journal of Converging Technologies and Management (IJCTM)
Volume 4 , Issue 1 , 2018
ISSN: 2455 – 7528

Table 1: Objectives of Resource management of Cloud Computing

S. No.	Objective Group	Objectives
1	Performance	Response Time, Uptime, Throughput, Jobs under waiting, Fault Tolerance
2	Financial	Price, Income, Cost
3	Environment	Energy, Peak power, Thermal, CO_2 emission
4	Others	Reliability, Security, legal compliance

The first group of objectives of resource management of cloud computing system is the response time for a user request, and the throughput for a service provider. These objectives can be further modified as the maximum completion time for a batch of tasks, and the mean flow time, which is defined as the mean time spent by tasks in the system. SLAs often promote service uptime as a critical performance metric, i.e. the percentage of time that the service is available to its users. Uptime can be related to the success rate, which is the ratio of the number of successfully executed tasks to the total number of tasks.

The second group of objectives of resource management is cost-related. As in other economic settings, cloud customers are interested in obtaining the best performance for the lowest price. On the other hand, the providers' goal is to maximize their income, by maximizing revenue and minimizing costs.

The third group of objective of resource management is the minimization of the environmental impact. Environmental objectives include the minimization of electrical energy use, the peak power draw of the systems, cooling costs, and CO_2 emissions.

The fourth group of objectives is factors resulting from highly-distributed and shared cloud environments, including reliability of scheduling on heterogeneous infrastructures, security in the case of distribution and multi-tenancy, and, last but not least, the appropriate location of computation and storage of data for legal compliance.

Process of Resource Management

The cloud computing perspective, resource management is a process which effectively and efficiently manages above mentioned resources as well as providing QoS guarantees to cloud consumers. Resource Management has two sequential phases.

Phase-I Process: It is initial resource assignment, in a manner that resources are requested by application (on behalf of cloud consumers) first time. Figure 4 shows several sequential steps which need to be followed for completion of this phase.

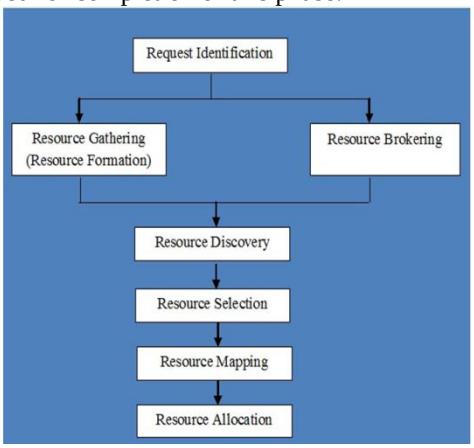


Figure 4: Resource Management in Phase I process [30]

- **Request Identification:** This is the first step of phase 1. In this step, various resources will be identified by cloud providers.
- **Resource Gathering / Resource Formation:** After identification of resources in step 1, gathering or formation of resources will take place. This step will identify available resources. This step may also prepare custom resources.
- **Resource Brokering:** This step is negotiation of resources with cloud consumers to make sure that they are available as per requirement.
- **Resource Discovery:_**This step will logically group various resources as per the requirements of cloud consumers.
- **Resource Selection:** This step is to choose best resources among available resources for requirements provided by cloud consumers.
- **Resource Mapping:** This step will map virtual resources with physical resources (like node, link etc) provided by cloud providers.
- **Resource Allocation:** This step will allocate / distribute resources to the cloud consumers. Its main goal is to satisfy cloud consumers' need and revenue generation for cloud providers.

Phase-II Process: Phase II of resource management is done at regular intervals once phase I is completed. It is also known as periodic resource optimization. Periodic resource optimization is presented as a process for two different categories of resources which are non-virtualized resources and virtualized

resources. The non-virtualized resources are also called as physical resources. For both categories of resources, periodic resource optimization contains similar steps. The only difference is that virtualized resources can be assembled together as per the resource requirement and can be disassembled also. So periodic resource optimization for virtualized resources contains two steps more compared to non-virtualized resources which are Resource bundling and Resource fragmentation.

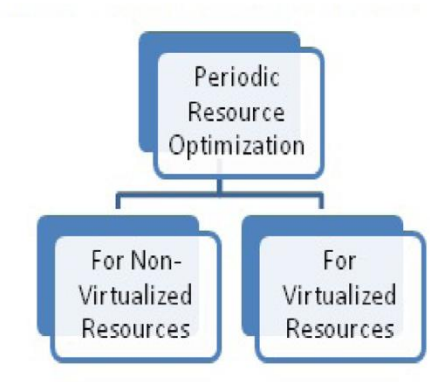


Figure 5: Periodic Resource Optimization [30]

For Non-virtualized Resources

- **Resource Monitoring:** Resource Monitoring is the first and crucial step in periodic resource optimization. Various non-virtualized cloud resources are monitored to analyze utilization of resources. This step will also monitor availability of free resources for future purpose. The major issue with cloud resource monitoring is to identify and define metrics/parameters for it.
- **Resource Modeling / Resource Prediction:** This step will predict the various non-virtualized resources required by cloud consumer's applications. This is one of the complex steps as cloud resources are not uniform in nature. Due to this non uniformity, it is very difficult to predict resource requirement for peak periods and as well as for non-peak periods.
- **Resource Brokering:** This step is negotiation of non-virtualized resources with cloud consumers to make sure that they are available as per requirement.
- **Resource Adaptation:** As per the requirements of cloud consumers, non-virtualized cloud resources can be scaled up or scaled down. This step may increase cost from cloud providers' perspective.
- **Resource Reallocation:** This step will reallocate / redistribute resources to the cloud consumers. Its main goal is to satisfy cloud consumers' need and revenue generation for cloud providers.
- **Resource Pricing:** It is one of the most important steps from cloud providers, and cloud consumers perspective. Based on cloud resource usage pricing will be done.

Figure 6: Periodic Resource Optimization for Non-Virtualization Resources [30]

For Virtualized Resources

- **Resource Monitoring:** Resource Monitoring is the first and crucial step in periodic resource optimization. Various virtualized cloud resources are monitored to analyze utilization of resources. This step will also monitor availability of free resources for future purpose. The major issue with cloud resource monitoring is to identify and define metrics / parameters for it.
- **Resource Modeling / Resource Prediction:** This step will predict the various virtualized resources required by cloud consumers applications. This is one of the complex steps as resources are not uniform in nature. Due to this non uniformity, it is very difficult to predict resource requirement for peak periods and as well as for non-peak periods.
- **Resource Brokering:** This step is negotiation of virtualized resources with cloud consumers to make sure that they are available as per requirement.
- **Resource Adaptation:** As per the requirements of cloud consumers, virtualized cloud resources can be scaled up or scaled down. This step may increase cost from cloud providers' perspective.
- **Resource Bundling:** As per the requirement various non-virtualized resources can be bundled into virtualized resources.
- **Resource Fragmentation:** Various virtualized resources needs to be fragmented to make non virtualized resources free. After this step various non-virtualized resources can be bundled in to virtualized resources as a part of resource bundling.
- **Resource Reallocation:** This step will real-locate / redistribute resources to the cloud consumers. Its main goal is to satisfy cloud consumers' need and revenue generation for cloud providers.
- **Resource Pricing:** It is one of the most important step from cloud providers and cloud consumers

perspective. Based on cloud resource usage pricing will be done.

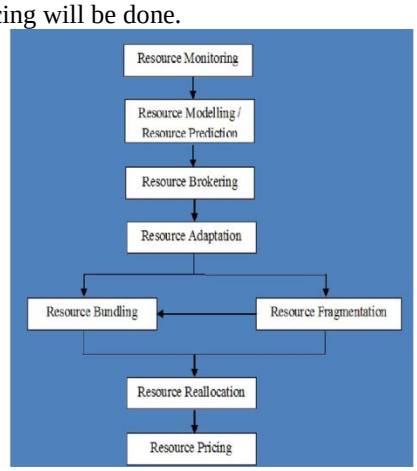


Figure 7: Periodic Resource Optimization for Virtualized Resources [30]

Reference diagram of Cloud computing environment

Reference architectures are logical arrangement of physical and/or virtual cloud resources in a specific manner to fulfill cloud computing requirements. Some of research interested reference architecture diagrams as under:

1. Redundant three-tier architecture
2. Multi-datacenter architecture
3. Auto scaling architecture
4. Scalable architecture with memory base
5. Scalable multi-tier architecture with memory cached
6. Scalable queue-based setups
7. Scalable multi-cloud architecture
8. Failover multi-cloud architecture
9. Multi-cloud disaster recovery architecture
10. Cloud and dedicated hosting architecture

Redundant three tier architecture: Any production environment that is launched in the cloud should also have a redundant architecture for failover and recovery purposes. Typically, use a Server Array for application tier. There may be some scenarios where application is not designed to auto scale, in such cases create a redundant multi-tiers architecture where it have redundancy at each tier of reference architecture. The figure-8 describe a redundant three tier architecture of cloud computing system, there are two load balancer servers, two application servers, as well as master and slave database servers. A redundant architecture will help protect site/application from system downtime.

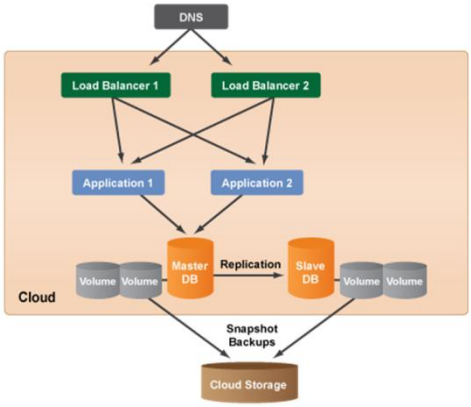


Figure 8: Redundant three tier architecture [19]

Multi-Datacenter Architecture: If cloud infrastructure is spread multiple datacenters, which is an advantage of redundancy and protection. Each datacenter in a cloud is designed to be an isolated segment inside the same geographical cloud. So if a power/hardware/network failure occurs in one datacenter, the other datacenters will be unaffected. The figure-9 describes Multi-Datacenter architecture of cloud computing system.

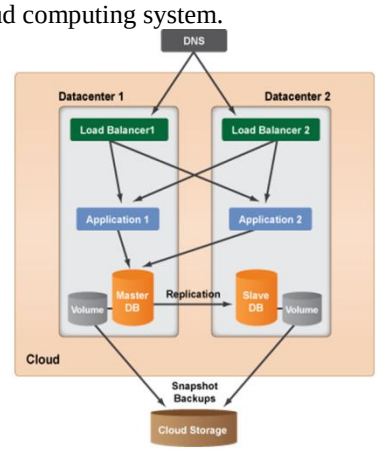


Figure 9: Multi-Datacenter Architecture [19]

Auto Scaling Architecture: If cloud computing system is designed with the ability of grow or shrink the number of running server resources (VMs) as the demands of application over time. Auto scaling is most commonly used for the application tier for enhancing performance of cloud applications. The figure 10 and figure 11 describes Auto Scaling architecture of cloud computing system.

Suresh Gyan Vihar University, Jaipur
International Journal of Converging Technologies and Management (IJCTM)
Volume 4 , Issue 1 , 2018
ISSN: 2455 – 7528

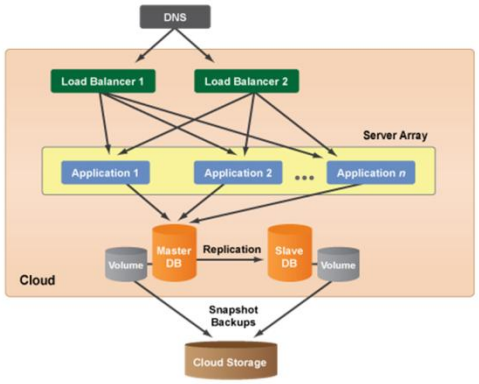


Figure 10: Auto Scaling Architecture [19]

Scalable Architecture with Membase: Membase (or Couchbase) type data is a need of Web 2.0 application. Companies such as Facebook, Google and Amazon.com are using NoSQL data. If cloud system is not using conventional master-slave as MySQL, Oracle, MS-SQL Server, and use Membase node in database tire, this type of architecture is called Scalable with Membase. The figure-12 describes Scalable architecture with Membase of cloud computing system.

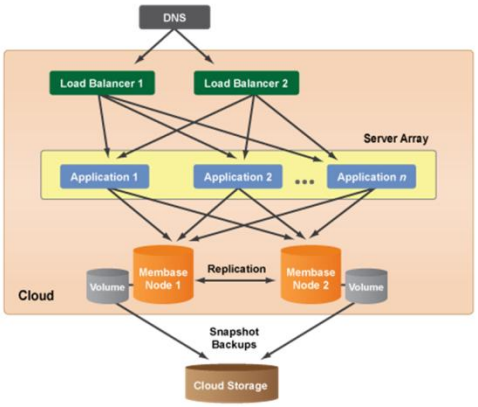


Figure 12: Scalable Architecture with Membase [19]

Scalable Multi-Tier Architecture with Memcached: For applications/sites that require lots of reads from the database and serve a lot of static content, a Memcached layer present in cloud system architecture to offload a read-heavy database. Memcached is an open source distributed memory object caching system that's ideal for speeding up dynamic web applications by alleviating database load. In the example diagram below, the application servers can still make writes to the database, but many commonly used objects will be retrieved from one of the Memcached servers instead of the Master-DB server. The figure-14 describes Scalable architecture with Membase of cloud computing system.

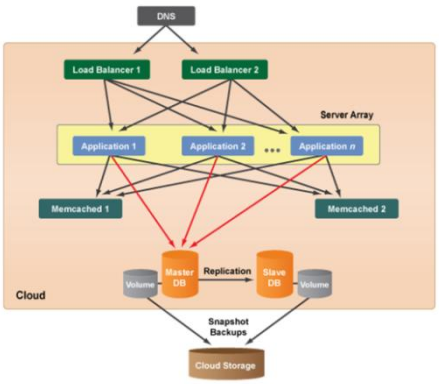


Figure 13: Scalable Multi-Tier Architecture with Memcached [19]

Scalable Queue-based Setups: This type of architecture is developed integration of scalable cloud architecture with scalable grid based system .Grid is basically a queue based setup. Grid application is enhancing performance of application server tire .Figure 15 is descried scalable queue base setup.

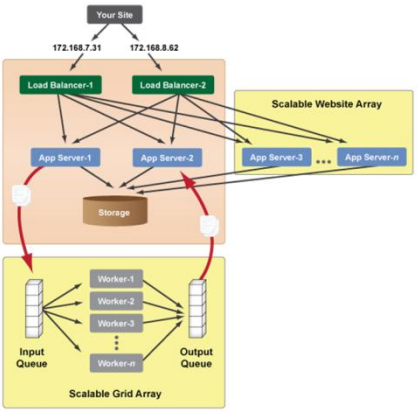
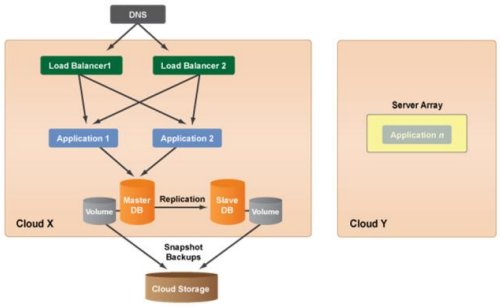


Figure 14: Scalable Queue-based Setups [19]

Scalable Multi-Cloud Architecture: The Multi-Cloud Architecture diagram given below, the flexibility of primarily hosted application in the private cloud infrastructure but also auto scale out into a public cloud for additional server capacity, if necessary. In this architecture basically virtualized whole data centre.



Figur
e 15: Scalable Multi-Cloud Architecture [19]

Failover Multi-Cloud Architecture: A Multi-Cloud Architecture has a facility to allow easily migrate sites/applications from one cloud to another cloud during when one cloud not working, is known as failover-cloud architecture. In this architecture also virtualized whole data centre for failover purpose.

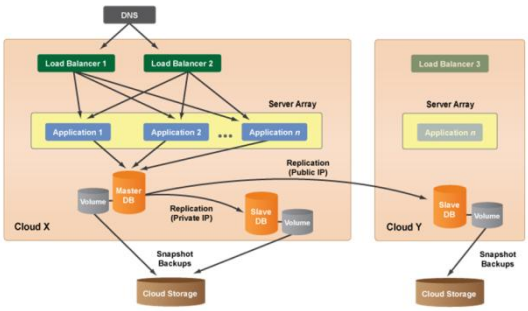


Figure 16: Failover Multi-Cloud Architecture [19]

Multi-Cloud Disaster Recovery Architecture: In the figure 19, the production environment is currently hosted in Cloud X. Primary (snapshot) backups are periodically being taken of the database. However, you can also take a manual Logical volume management (LVM) backup of the database to a supported cloud storage service (e.g. Remote Object Storage (ROS) such as an Amazon S3 bucket or Google Cloud Storage container. So, if you need to perform a database migration or there is a cloud outage, you can use the LVM backup to re-launch your database server into a different cloud/region. The LVM backup can then be used to-restore your database and re-establish a redundant database setup in the cloud of your choice. (Note: The LVM backup can either be used to restore the database to a volume or locally to an instance depending on whether or not the cloud supports the use of volumes and snapshots.)

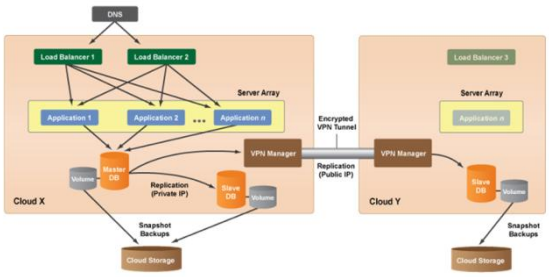


Figure 17: Multi-Cloud Disaster Recovery Architecture [19]

Cloud and Dedicated Hosting Architecture: Another type of hybrid cloud solution architecture is to leverage a public/private cloud's resources along with existing servers from an internal or external datacenter. For example, perhaps your company has strict requirements around the physical location of your database server because it contains sensitive user information or proprietary data. In such cases, even though the database cannot be hosted in a cloud infrastructure the other tiers of your application or site are not subject to the same levels of restrictions. In such cases, can use platform to build a hybrid system architecture using a virtual private network (VPN) solution to create a tunnel for secure communication across a public IP between cloud servers and dedicated servers.

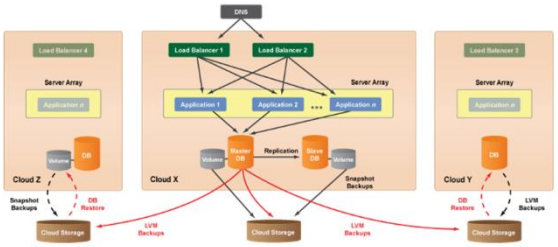


Figure 18: Cloud and Dedicated Hosting Architecture [19]

II RELATED WORK

Rodrigo N. Calheiros, Rajiv Ranjany and Rajkumar Buyya [8], identified some significant problems that exist with efficient provisioning and delivery of applications using Cloud-based IT resources. These problems concern various levels, such as workload modeling, virtualization, performance modeling, deployment, and monitoring of applications on virtualized IT resources. They proposed an analytical performance model using queueing network (M/M/1 and M/M/∞) and time series (ARMA), which modeled the IaaS, and monitoring data from running VMs. The goal of the model is to meet QoS, service

time, the rejection rate of requests and utilization of available resources. The research gap identified in this work is queueing network models M/M/1, and M/M/∞ are standard models, some other models also available to describe more realistic situations. Above model not suitable for the study of component levels of the cloud environment.

Neelam Sah, S. B. Singh, and R.S. Rajput [9] investigated the reliability characteristics of a system based on the concept of a web server. They worked on different types of failures like partial, complete, and repairable. They developed mathematical models and discussed different reliability measures like availability, reliability, M.T.T.F. and cost analysis. The research gap identified here models can be extended to the cloud environment.

Rahul Ghosh andVijay K. Naik [11], Cloud service providers are looking for ways to increase revenue and reduce costs either by reducing capacity requirements or by supporting more users without adding capacity. Over-commit of physical resources without adding more capacity is one such approach. Researchers estimate the risks associated with over-commit, they describe a mechanism based on the statistical analysis. They used secondary data of CPU usage collected from an internal private cloud and show that the proposed approach is useful and practical. Statistical approached applied to the analysis on resources, e.g., memory, disk, VMs, and network. The research gap identify is researchers used secondary data; secondary data can be categories as test data, data for training and data for validation for better analysis.

Jiayin Li, Meikang Qiu, Zhong Ming, Gang Quan, Xiao Qin, Zonghua Gu [12], found in case of significant client demands, it may be necessary to share workloads among multiple data centers, or even various cloud providers. The workload sharing can expand the resource pool and provide even more flexible and affordable resources. They presented a resource optimization mechanism using graph theory and proposed a dynamic scheduling algorithm namely DCLS and DCMMS for resource optimization mechanism. The research gap identify is, graphs are mathematical structures that used to model pairwise relations between objects. Layered graphs architecture will be more suitable, like the first layer of graphs covers data centers, the second inner layer of graphs covers the inside of a data center and the third layer of graph covers inside a machine.

Rahul Ghosh, Francesco Longo and Vijay K. Naik, Kishor S. Trivedi [13], developed a scalable stochastic analytic model using continuous time Markov chains (CTMC) and queuing model (M/M/1 and M/M/∞) for performance measurement of IaaS Cloud. The study conducted on the proposed stochastic model with the effects of workload (e.g., job arrival rate, job service rate) and system capacity (PMs per pool, VMs per PM) for the revaluation of IaaS performance. The research gap identify in this work is queueing network models M/M/1, and M/M/∞ are standard models, some other models also available to describe more realistic situations.

Dario Bruneo [14], developed an analytical model using Markov Modulated Poisson process (MMPP) and evaluates the performance of cloud resources. Performance parameters in this study were availability at a time, instance service probability at a time. The research gap identified is model applicable only for IaaS, the model can be extended, for PaaS and SaaS.

Kangwook Lee, Ramtin Pedarsani, and Kannan Ramchandran [15], proposed queuing models M/M/n, that describes the efficient utilization of cloud resources in the redundant type request. Researchers explained with the help of an analytical model. The research break identified is here used only some standard models M/M/2, and M/M/n, a study of component level explanation also required.

Mohamed Eisa, E. I. Esedimy and M. Z. Rashad [16], proposed a model for cloud scheduling based on multiple queuing models (M/M/S) allow improving the quality of service by minimizing execution time per jobs, reducing waiting time, and reducing the cost of resources. The proposed improved scheduling algorithms based on queuing theory. Experimental results indicated that model increases utilization and reduce waiting time. The research gap identified here queueing network models M/M/S are standard models, some other models also available to describe more realistic situations and work extended only horizontal scaling, vertical scaling also proposed.

Parvathy S. Pillai and Shrisha Rao, [17], proposed a resource allocation mechanism for machines on the cloud, based on the principles of the uncertainty principle of game theory and found that the method of resource allocation using game theory is better for resource utilization. Here research gap identified is various experiments conducted in this work, but they use only four types of VM instances it may be generalized.

Danilo Ardagna, Giuliano Casale, Michele Ciavotta, Juan F Pérez and Weikun Wang [18], conducted a survey focused on QoS aspects like performance, reliability, and availability of cloud systems. Here research gap identified is in the present survey described various queuing theory methods that are applicable for capacity allocation, admission control,

and load balancing. An extension of current study for the importance of any system component and the importance of its position also is required.

Eliomar Campos, Rubens Matos, Paulo Maciel, Igor Costa, Francisco Airton Silva and Francisco Souza [21], investigated the process of VMs instantiation, an essential activity in cloud computing systems, and intensively used by elasticity mechanisms such as the Eucalyptus auto-scaling feature. For this, proposed a full factorial experiments design followed by measurements and analytical modeling. They analyzed the effects and relevance of three factors cache, VM type, and EMI size considering the total time of instantiation. The times for completion of intermediate phases of the instantiation were also measured and analyzed. The research gap identified is the experiment conducted using physical machines, those machines have own capacity. The experiments may be extended for generalizing machines.

K. Shyamala and T. Sunitha Rani [22], summarized resource allocation methods, and methods have been discussed to provide an overall picture of resource allocation and assumed that efficient resource allocation could optimize cost, time and power consumption. It can also minimize the underutilization of resources, balance load, request loss and leasing cost. Here research gap identified that, for the better explanation, some analytical models be required.

Deepa Mani et al., [27], proposed a model for the reliability of VM using Markov chain and evaluated its performance. The research gap identified here some components of the system have more importance compare to another components. To estimating overall reliability, the importance of components of a system cannot be overlooked.

Sandeep K. Sood, Rajinder Sandhu [23], proposed a model which formulates a personalized mobile cloud environment for all mobile cloud users. Mobile cloud users have three entities viz. mobile customer MC, cloud service provider CSP and independent authority IA. The artificial neural network techniques used to select CSP. The essential component of the proposed model is the representation of cloud resources in a two-dimensional matrix termed as resource provisioning RP matrix. A past resource usage stored in RP matrices which can be used by the neural network for future prediction. Accurate billing calculation can also be done using RP matrices over a desired period. The accuracy of the proposed model vastly depends on finding a pattern in cloud resource usage by an MC. The research gap identified here the problem will be generalized to all types of cloud resources not only for mobile.

Deborah Magalhas et al., [24], proposed a web application model to capture the behavioral patterns of different user profiles and to support analysis and simulation of resources utilization in cloud environments. Model validated using graphics and statistical hypothesis methods. The research gap identified model may also be validated using weblog and data mining techniques.

D Chitra Devi et al., [25], proposed an analytical model using linear programming model for performance evolutions of the cloud system. Here research gap identified is the model can be an extended component level of the cloud system. Some other linear models like sequencing problem, transportation problem method can be applied to cross-validation.

Wanbo Zheng et al., [26], proposed a queuing theory-based model performance determination of IaaS. The research gap Identified as they used standard queueing network models M/M/C some other models also be used to describe more realistic situations. M/M/C model not suitable for the study of component levels of any queue.

Anil Mukherjee et al., [28], proposed a cloud resource allocation model based on stochastic linear programming. The author proposed a time preference (value of service at different points of time) based stochastic integer linear programming model to allocate the cloud resources among the cloud users intending to maximize the revenue of cloud providers from the spot market. The research gap identified is the model can extend performance evolutions.

Deval Bhamare et al., [29], presented an analytical study of the placing service function chains over the virtualized platform in a multi-cloud. The focus of the work is on reducing the total delays to the end users and total cost of deployment for service providers in inter-cloud environments. To achieve this, researchers aim to reduce the inter-cloud traffic between virtual function instances. The problem has been solved using an integer linear programming ILP methodology. Here is research gap identify the study of the components level of the system will be required.

R. S. Rajput, Dinesh Goyal and S. B. Singh [31], proposed an analytical model for redundant three-tier cloud computing system and, investigated system up to components level of the cloud system using concepts of queuing model with Jackson network. The aim of work is to evaluate the performance of cloud system. The research gap identified is in the present study used all components as M/M/1 queue; different

queuing models can also be applied to incorporate realistic situations. The model developed using probabilistic routing; other types of routing may also be applied to enhance the practical applicability.

III PERFORMANCE INDICES

Performance evaluation techniques fall into two categories: measurement techniques, and predictive techniques; with the latter category comprising mathematical analysis and simulation. Measurement methods require real infrastructure to be available for experimentation. In the present study, we discussed for performing the second option. Some performance indices for our consideration as under:-

- **Average number of job requests (of a station):** number of the tasks at a station, both waiting and receiving service.
- **Average number of job requests in the queue (of a station):** number of the tasks at a station waiting for service.
- **Average queue time (of the station):** average time spent on a job waiting in a station queue. It does not include the service time.
- **Average response time (of the station):** average time spent on a station by a job request for a single execution of job request (sum of queue time and service time)
- **Utilization (of the station):** percentage of time a station is used (i.e., busy). It ranges from 0 (0%), when the station is idle, to a maximum of 1 (100%), when the station is constantly busy. In delay stations, the utilization is computed as the average number of job requests in the station, and thus it may be greater than 1.
- **Bottleneck station:** The station with the highest utilization in the system.
- **System Throughput (of the entire cloud computing system):** Rate at which job requests departs from the system.
- **System response time (of the whole cloud computing system):** average duration a job request spends in the system to receive service as well as waiting for service from the various stations it visits. It may be obtained summing the response times of all the stations.
- **Expected number of job requests in the system (of the entire cloud computing system):** average number of job requests in the system. It may be obtained summing the average number of job requests of all the station.
- **Horizontal Scaling**: Increasing or decreasing capacity of particular queue (or VM)**.**
- **Vertical Scaling:** Add parallel queue (or VMs) if load increase or delete/stop queues (or VMs) if queue unused.
- **Fault Tolerance**
- **Importance of a component in the system**
- **Importance of a components' position in the system**

IV METHODOLOGY

We are discussing Queuing theory, Jackson network theory, JMT simulator, CloudSim simulator; these are useful for performance analysis of cloud environment.

Queuing Model

A Queue represents a general service facility. Queuing theory uses mathematical tools to predict the behavior of queuing systems [1, 6]. The range of applications has grown including telecommunications, computer communication, manufacturing, air traffic control, military logistics, the design of theme parks, and any others area that involve service system whose demands are random.

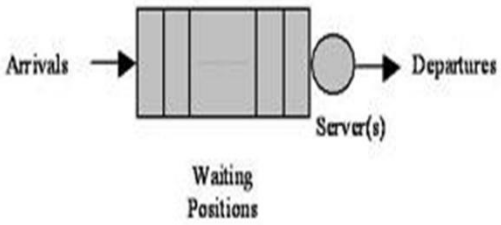


Figure 19: Queuing System

A queuing system consists of a stream of arriving customers, a queue, and a service stage. To model such system, the following basic elements are needed:

- A stochastic process describing the arrivals of customers
- A stochastic process describing the service or departures of customers
- Numbers of servers (m)
- System capacity (K)
- Size of customer population (N)
- Queues disciple (Z)

A queuing system is described using the notation A/B/m/K/N/Z, where A and B specify the distributions of the inter-arrival and service times,

Jackson network

Jackson network is a network of queues [2, 10] Jackson formula is used to calculate inter-arrival rate of a queue.

Suresh Gyan Vihar University, Jaipur
International Journal of Converging Technologies and Management (IJCTM)
Volume 4 , Issue 1 , 2018
ISSN: 2455 – 7528

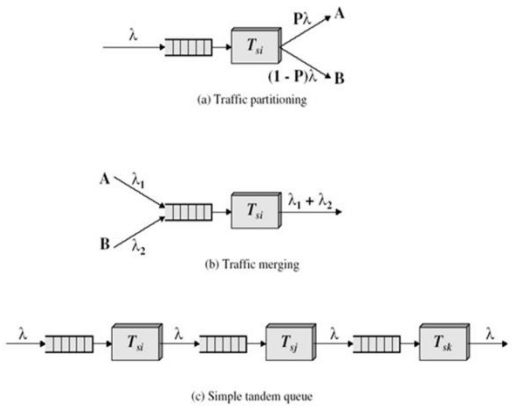


Figure 20: Jackson Networks

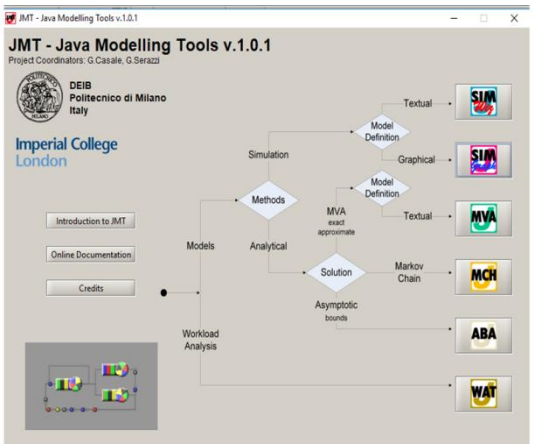


Figure 21: JMT Java Modelling Tools

Jackson Formula

$$\lambda_i = \gamma_i + \sum_{k=1}^{m} \delta_{ki}\lambda_k$$

Where

- λ_i: Calculated job request arrival rate at station i
- γ_i: External job request arrival rate at station i
- δ_{ki}: Probability a job request from station k goes to station i

Java Modelling Tools (JMT)

Java Modelling Tools (JMT) is a suite of applications developed by Politecnico di Milano and Imperial College London and released under GPL license. Java Modelling [3] Tools is a free open source suite consisting of six tools for performance evaluation, capacity planning, workload characterization, and modeling of computer and communication systems. The suite implements several state-of-the-art algorithms for the exact, approximate, asymptotic and simulative analysis of queuing network models, either with or without the product-form solution. Models can be described either through wizard dialogs or with a user-friendly graphical interface. In the JMT suite, a discrete-event simulator for the analysis of queuing network model is provided. Two user interfaces are available: alphanumerical JSIMwiz and graphical JSIMgraph. JSIMgraph is a user-friendly graphical tool of JMT. It allows an easy description of network structure, as well as a simplified definition of input and execution parameters. In the present study, we have used JSIMgraph for investigation of the performance of redundant three-tire architecture Cloud computing system.

CloudSim

The CloudSim toolkit supports both system and behavior modeling of Cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. [8] It implements generic application provisioning techniques that can be extended with ease and limited effort. Currently, it supports modeling and simulation of Cloud computing environments consisting of both single and inter-networked clouds (federation of clouds). Moreover, it exposes custom interfaces for implementing policies and provisioning techniques for allocation of VMs under inter-networked Cloud computing scenarios. Several researchers from organizations, such as HP Labs in U.S.A., are using CloudSim in their investigation on Cloud resource provisioning and energy-efficient management of data center resources. The usefulness of CloudSim is demonstrated by a case study involving dynamic provisioning of application services in the hybrid federated clouds environment. The result of this case study proves that the federated Cloud computing model significantly improves the application QoS requirements under fluctuating resource and service demand patterns

V CONCLUSION

In the present review work, we have reviewed about 32 articles. Our work focused on cloud resources, cloud resources management, resource management process and reference architecture of cloud environment. We have also discussed some performance indices that apply to the performance evaluation of cloud computing environment. Some tools and techniques also discussed in this article that is useful to performance modelings like queuing theory, Jackson network, JMT, and CloudSim.

VI REFERENCES

1. Swarup K., Gupta P.K., and Manmohan 1996, Operation Research, New Delhi: Sultan Chand & Sons.
2. Bose Sanjay Kumar 2001, An introduction to the queueing system, Kluver/Plenum Publisher.
3. M.Bertoli, G.Casale and G.Serazzi 2009, JMT: performance engineering tools for system modeling. ACM SIGMETRICS Performance Evaluation Review, Volume 36 Issue 4, 2009, 10-15, ACM press.
4. Barrie Sosinsk 2011, Cloud Computing Bible, Wiley Publishing, Inc.
5. Rajkumar Buyya and Karthik Sukumar 2011, Platform for building and Deploying Application for Cloud Computing, CSI Communication, May 2011, 6:11
6. Panneerselvam R 2011, Operation Research, New Delhi: PHI Learning Private Limited.
7. Rodrigo N Calheiros et. al., 2011,Virtual Machine Provisioning Based on Analytical Performance and QoS in Cloud Computing Environments , IEEE International Conference on Parallel processing, Page No.295-304
8. Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar A. F. De Rose, and Rajkumar Buyya, 2011, CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms, Software: Practice and Experience (SPE), Volume 41, Number 1, Pages: 23-50, ISSN: 0038-0644, Wiley Press, New York, USA, January, 2011.
9. Neelam Sah, S. B. Singh, and R.S. Rajput, 2011, Stochastic analysis of a Web Server with different types of failure, Journal of Reliability and Statistical Studies, Vol. 3, Issue 1(2010): 105-116
10. Ebrahim Mahdipour et al., 2012, Performance evaluation of an Important sampling technique in a Jackson network, International Journal of System Science 1:11.
11. Rahul Ghosh et. al., 2012, Biting Off Safely More Than You Can Chew: Predictive Analytics for Resource Over-Commit in IaaS Cloud, Conference Paper June 2012 https://www.Researchgate.net/publication/261526165
12. Jiayin Li et. al., 2012, Online Optimization for scheduling for scheduling preemptable tasks on IaaS cloud systems, Journal.of Parallel and Distributed Computing., Vol.72, Page No.666-677
13. Rahul Ghosh et al., 2013, Modeling and performance analysis of large-scale IaaS Clouds, Feature Generation Computer System, Vol. 29, Page No.1216-1234
14. Dario Bruneo., 2013, A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems, IEEE Transactions on Parallel and Distributed System, Vol. 25, No. 3
15. Kangwook Lee et. al., 2014, On scheduling Redundant Request with Cancellation Overheads,
16. Mohamed Eisa et. al., 2014, Enhancing Cloud Computing Scheduling based on Queuing Models, International Journal of Computer Applications Vol. 85, No. 2, Page No.17-23
17. Parvathy S Pillai et. al., 2014, Resource Allocation in Cloud Computing Using Uncertainty Principle of Game Theory, IEEE System Journal
18. Danilo Ardagna et. al., 2014, Quality-of-service in cloud computing modeling techniques and their applications, Journal of Internet Services and Applications, 5:11
19. http://docs.rightscale.com/cm/designers_guide/cm-cloud-computing-system-architecture-diagram.html
20. http//eid100nujhatn.blogspot.in/2015/10/all-you-need-to-know-about-cloud.html
21. Eliomar Campos et al., 2015, Performance Evaluation of Virtual Machines Instantiation in Private Cloud, IEEE World Congress on Services
22. K. Shyamala et al., 2015, An Analysis on Efficient Resource Allocation Mechanisms in Cloud Computing, Indian Journal of Science and Technology Vol 8(9), Page No.814-821
23. Sandeep K. Sood et al., 2015, Matrix-based Proactive resource provisioning in the mobile cloud environment, Simulation Modelling Practice and Theory 50 (2015) 83-95
24. Deborah Magalhaes et al., 2015, Workload modeling for resource usage analysis and simulation in cloud computing, Computer and Electrical Engineering, Vol. 47, No. 1, Page No.69-81
25. D Chitra Devi et. al., 2016, Load Balancing in Cloud Computing Environment using Improved Weighted Round Robin Algorithm for Nonpreemptive Dependent Task, The Scientific World Journal, Vo20l6. 2, Page No.1-14
26. Wanbo Zheng et al., 2017, Percentile Performance Estimation of Unreliable Issa Clouds and Their Cost-Optimal Capacity Decision, IEEE Access, Vol. 15
27. Deepa Mani et al., 2017, Availability Modelling of Fault-Tolerant Cloud Computing System, International Journal of Intelligent Engineering & Systems, Vol.10, No. 1, Page No.154-165
28. Anil Mukherjee et al., 2017, Users' Time preference based stochastic resource allocation in cloud spot market: cloud provider's perspective, Conference Publication
29. Deval Bhamare et al., 2017, Optimal Virtual Network Function Placement in Multi-Cloud Service Function Changing Architecture,

Computer Communications, Vol 102, No. 1, Page No.1-116.
30. Swapnil M Parikh et. al., 2017, Resource Management in Cloud Computing: Classification and Taxonomy, arXiv:1703.00374[cs.DC]
31. R. S. Rajput, Dinesh Goyal and S. B. Singh (2018), Study of performance evolution of three tier architecture based cloud computing system, 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT), MNIT Jaipur, India 2018.
32. MHRD NME-ICT, Govt of India, e-PG Pathshala, http://www. http://epgp.inflibnet.ac.in