STUDY AND ANALYSIS OF OPTIMIZATION OF RESOURCE PROVISIONING COST AND SCALABILITY IN CLOUD COMPUTING

Project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

In

Computer Science and Engineering/Information Technology

By

Sunidhi Pathania (131236)

Dilpreet Sawhney (131257)

Under the supervision of

Prof. Dr. SP Ghrera

to

Department of Computer Science & Engineering

Jaypee University of Information TechnologyWaknaghat, Solan-173234, Himachal Pradesh

Candidate's Declaration

I hereby declare that the work presented in this report entitled **"Optimisation of Resource Provisioning Cost and Scalability in Cloud Computing"** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2016 to December 2016 under the supervision of **Dr. Satya Prakash Ghrera** (Professor, Brig (Retd.) and Head, Dept. of CSE and IT**).**

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Sunidhi Pathania (131236) Dilpreet Sawhney (131257)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. Satya Prakash Ghrera

Professor, Brig (Retd.) and Head, Dept. of CSE and IT

Dated:

Acknowledgement

It is our privilege to express our sincerest regards to our project supervisor **Prof. Dr. Satya Prakash Ghrera**, for their valuable inputs, able guidance, encouragement, whole-hearted cooperation and direction throughout the duration of our project.

We deeply express our sincere thanks to our Head of Department **Prof. Dr. Satya Prakash Ghrera** for encouraging and allowing us to present the project on the topic "Study and analysis of Optimisation of Resource Allocation Cost and Scalability in Cloud Computing" at our department premises for the partial fulfilment of the requirements leading to the award of B-Tech degree.

At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

Date: 28-04-2017

Sunidhi Pathania(131236)

Dilpreet Sawhney(131257)

Table of Contents

List of Figures

List of Graphs

Chapter1:INTRODUCTION

1.1Introduction

These days, we might hear people discussing “the cloud”. Whether it’s securing many folders and documents in the cloud, hearing music or saving pictures within cloud, many people have started utilizing this term. But for each and everyone, whom of which are not familiar with this term, “the cloud” still refers to that white puffy thing in sky. However, in technology, it’s something different. “The Cloud” is the trendy word for a varied network or one can say remote servers which can be easily accessed with the help of an Internet connection and they store and manage information. In different words, it’s a place where you can store your stuff, other than your personal system. We get to save all of our files either to the systems or on the external HDs. These days, we are having multiple desktop systems, tablets, smart phones, laptop systems from where we may have to access our data. Also some years back, we had to save data to the USB and then we had to either transfer it to some another personal systems or e-mail that data to ourselves so that we could easily open that particular file on another machine. But now, cloud computing allows everybody to easily save data on a remote server so that later on it has been authorized from some another machine that is having an Internet connection. For most of population, authorizing files from somewhere is like taking stuff from “the cloud.”

The term "cloud computing" pertains to executing various system tasks by using services which are discerned completely over the Internet. There are various graphs and figures that are used for pertaining to the various symbols used for displaying the attributes. Web-based e-mail services like Hotmail and Gmail also provide for a cloud computing daemon: users can easily access their e-mail placed in the cloud from any other system having a browser and also an Internet connection, irrespective of the kind of hardware that is present on the system. The emails are hosted on Microsoft's and Google's servers, instead of storing them locally on the client’s system.

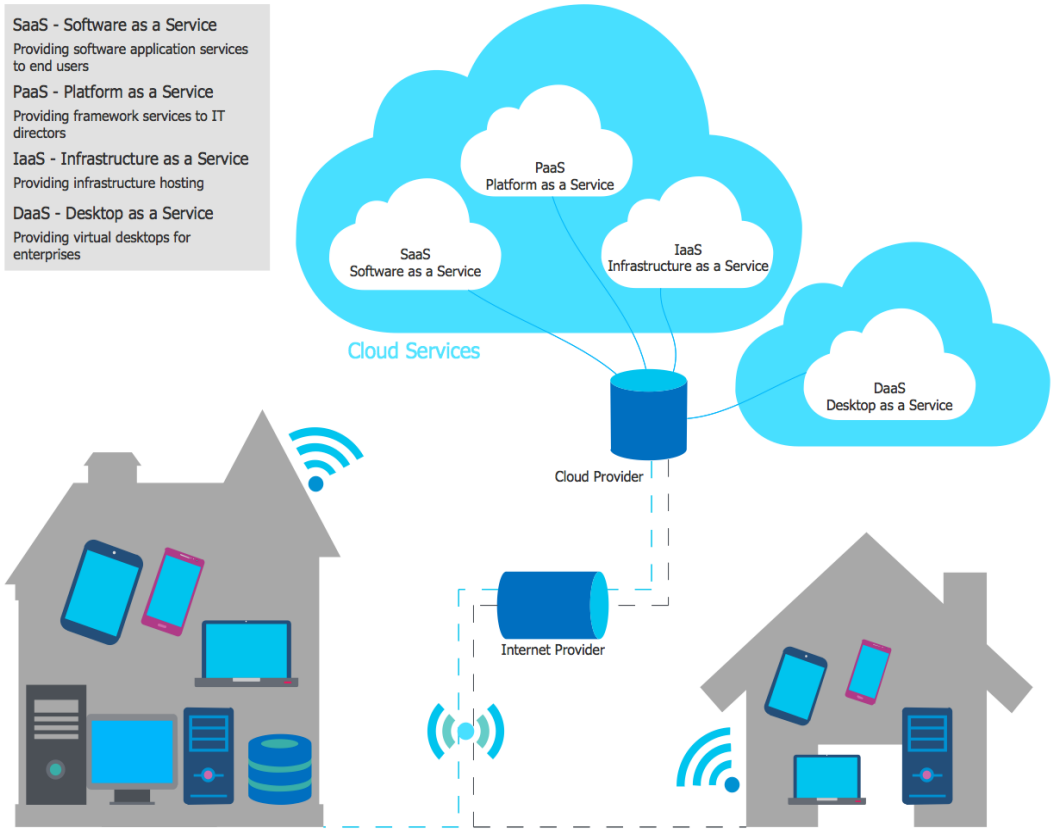


Fig.1.1: Cloud Environment

The apps that were earlier mentioned, were the software solutions provided to us over the Internet, or say in some another words Software as a Service i.e. SaaS. Other cloud computing services are having virtual server storage i.e. Infrastructure-as-a-Service or IaaS, like Amazon Web Services. Software and product development tools i.e. Platform-as-a-Service or PaaS, like Google Apps. Online File Storage is being used for many years and is now becoming very popular as hard drive space cost is even less than a dollar per GB. Asus Cloud Storage, Google Docs, Drop box**,** Microsoft Skydrive, Drop box and many other hundreds of providers offer low cost or free storage for the users.

Figure 1.2 the stack involved in the architecture: it displays all the layers used for delivery of services:

Software as a Service,

Platform as a Service and

Infrastructure as a Service:

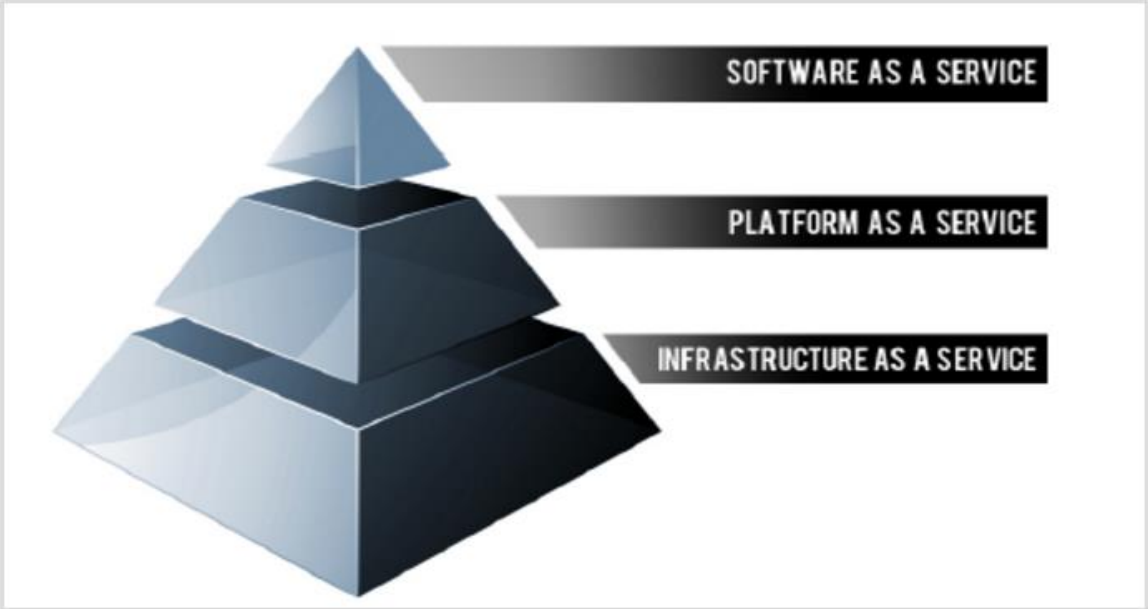


Fig.1.2: Cloud Services

In this report we shall be taking up all the three categories in detail. Hence a simplified approach of differentiating the three categories of Cloud Computing is as follows:

• SaaS apps are purposely designed for the end-users and are delivered over the web
• PaaS is the set of some tools and services that are designed for making coding and deploying tasks quick and efficient
• IaaS is the software and the hardware that powers it all i.e. networks, servers, storage, OSs.

Cloud Computing is a Service: The one and the simplest thing that a system can easily do without any problem is permitting us to keep and access required data. We can easily keep our pictures, movies and music on that. This is an important facility that is given by the cloud. Flickr is an example of cloud computing being given in terms of service. It firstly was used to share photos and the pictures but now is emerging to be a space to store the pictures. Now it's great to have the picture on the system.

Firstly, it helps to get pictures irrespective of where an individual is now or what devices he is currently using. One can put the pictures of your outing to Greece from his own system and also he can get those from any other device or system easily. Second, it lets you share your pictures. You don't burn them to a CD and also don't keep them on the FDs. You can do one thing i.e. give others the address of your account. Also, it has information security facility. If one will save pictures on your system, what will happen in case the HD crashes suddenly? You would regret your inability to save them externally! By keeping the pictures at that place, you are getting information security because backing up of data is there on the web. It is a good practice to have a copy either on the system, a CD or a FD but reality is that you are most probably going to lose the pictures that you are storing at a local place. Now it is that place where grid computing comes into importance. As cloud is already very much used as a very normal space to store the data but also we can use cloud to change the required information. For example: in place of making use of a local system's data base, big companies could give CPU time by paying some money and then using the database that is on web. The main drawback of using cloud computing as a means of providing service is that it needs a connection of internet. Thus, you'll lose all the benefits off if you are having no internet connection.

Cloud Computing is a Platform: OS for the future can be the web if cloud computing is involved. Not exactly true but we always need one local OS. One can say about the web that is upcoming good platform. Platform is the main structure and on that apps can stand. It is the main thing which runs our apps. Windows is also an example of a platform. Mac OS also belongs to the same club also. A platform is not necessarily an OS only. Java also belongs to same club but not an OS. Web is becoming a platform with the use of cloud computing. With some Office 2.0 we are now going to witness many more apps that were some years before used on desktop systems but are now days being converted into the web based apps to facilitate the users. Word processors like Office and Buzzword and Google Docs have become more useful in a similar fashion as the others and is easily replacing software's like MS Office in many ways. With many latest things which are making web apps to keep data locally which makes the on-line words processor available

offline also and Chrome latest browser to push envelope, and Google plays a important role in changing cloud computing into a more useful platform.

The Benefits of SaaS/ Cloud Computing: Software dealers need not to give endless hours for providing the help for the clients over phone, it is easily used and also correct one copy of the end product online which is centrally available. The very first and the major benefit of cloud computing is reduced cost. Users wouldn't have to spend out more on the large up-front costs of completely purchasing the word processing or some other end-user products. Users would have to instead pay some rental fees to access the central copy.

Over last few years cloud computing is growing tremendously, as witnessed by many popular Web apps used now a days, including Skype, Google Voice, Face book, Twitter, LinkedIn, Picassa, YouTube, Flickr, BitTorrent, Mint and many more. Even some of the softwares that were used few years earlier only on the desktop like MS Office have also moved in part to the Web.

Cloud services make businesses and the consumers free from investing a big amount of money in software or hardware on the devices. They reduce maintenance and upgrading needs because all the solutions are Web based and even the systems which are old, can also be used to access cloud services. Cloud computing gives great flexibility for mobile workers especially: professionals can work easily from any of the computing device as long as they are having access to the Web. It makes the collaboration much more efficient as distributed teams or combination of mobile workers and in-office staff can easily work on shared information that is stored centrally in cloud via, for example, online groupware apps.

There are some minor issues and obstacles to cloud computing. An Internet connection is must for taking advantages of the cloud services. Sometimes data may not be accessible when you are offline or due to the disruptions to the cloud services. Some cloud apps like Gmail also have offline capability and others like Unacademy need an Internet connection. The note-taking application like the Ever note does offer a good hybrid solution i.e. with both the phone software or desktop and an online service which helps in

synchronization our notes to the cloud. Despite the availability with the cloud computing, security is the major concern in cloud computing. Many companies and individuals are not cosy in keeping data especially some sensitive or proprietary data on some another server on the Internet. The issues such as trust and reliability are critical factors for cloud services which need to be resolved. Assurances of technologies like cryptography, encryption, privacy protection and more effective solutions for offline accessibility should be taken care of. Cloud computing biggest beneficiaries are remote workers as Web based apps makes us truly mobile and even accomplish the work.

Instead of distributing copies of files to individual devices cloud computing system always keeps the critical information on Internet servers. Video sharing cloud services like Netflix stream data over Internet to a player app on a viewing device instead of sending customer a Blu Ray discs/DVD. In order to avail the services of the cloud Clients need to be connected to the Internet. Video games on Xbox Live for example, cannot be obtained on physical discs but can be played online only and some other games cannot be played without the proper connection. Some researchers and companies are expecting that cloud computing will reach new heights with its popularity. The Chromebook shows how all PCs can be evolved under this trend in future i.e. devices with minimum local storage space and a very few local apps beside Web browser.

Service providers are held responsible for the maintenance, security and installation of the core technology of the cloud. Business customers, companies and individuals prefer cloud model as it reduces their own burden of maintenance infrastructure. These customers most of the time rely on provider for delivering the needed performance levels and reliability. Many home users become dependent on the server provider in this model. Cloud computing systems are pretty much designed to track all the present system resources. This makes providers to charge fees from customers proportional to their network, processing utilization and storage. Some prefer flat rate subscription so as to ensure predictable monthly/yearly costs and other customers may prefer the metered billing method for saving money but others Cloud computing environment may ask the user to send the data over the network/Internet and then cloud can store that data on a

third party system. The privacy and the security risks that are associated with this particular model are weighed against benefits Vs alternatives.

In this particular model, virtualization technologies are used sometimes to allocate resources to the cloud consumers. The consumers specify required software stack i.e. OSs and apps and then package them together into the VMs. The hardware requirements for the VMs can be easily adjusted by consumers. Then those VMs are outsourced to host in the computing environments that are operated by the third party sites that are owned by the cloud providers. Cloud provider is basically held responsible for ensuring that the Quality of Services for running VMs is up to an acceptable level. “Since the resources for the computation work are kept by those providers so the in-total expenditure for ownership that is mainly given to consumers can be easily decreased. In cloud computing, resource allocation method is needed to make the repository of computing resources available to the consumers for the processing of jobs and keeping files.” Cloud providers make available CCs mainly 2 resources provisioning methods which are either for short term on demand basis or long term reservation method. Pay-per use basis is used for charging the pricing in an on demand plan for instance 1 day. So by having the on demand basis allocation the clients can easily allocate the required resources at an instant of time when resources are actually needed in order to satisfy the fluctuating and unpredictable demands of the registered customers. The reservation plan is explained as “In this plan the pricing is actually done by only paying fee once for instance, 1 year, many a times it is done before cloud consumer uses the available computing resources. Also, the charges to use the resources are cheaper than the earlier explained on-demand plan. Thus, consumer can decrease the prices of computing resource allocation while using the reservation plan.” For e.g., the reservation way that was given by Amazon EC2 is capable of reducing total allocating price up to 48% when the resources that are already reserved are completely utilized. In short, it is steady state usage. Here, minimizing both the under-provisioning and the over-provisioning issues because of demand and the price unknowingness in the particular cloud computing area is working as the motivation to learn resource allocating strategy for the various cloud clients.

1.2 Problem Statement:

Cloud computing is growing now a days. One challenge Cloud Providers are facing today are the unpredictable computational demands of the Cloud Consumers. So, choosing the apt capacity increasing policy to satisfy the expectations of the registered customers is a big issue for cloud providers.

In order to get a precise understanding of the clients' demands, cloud providers offer capacity reservation contracts. The benefits of these contracts are many. The providers firstly estimate the appropriate future computational requirements and then provide the computing capacity according to it only. The consumers can depend on reserved computation capacity that is available.

The reservation mechanism balances interests of Cloud consumers and Cloud providers. But reservation fees should be made high enough to avoid exploitation of resources by the consumers.

Also the Cloud consumers should be given incentives so that he can give commitment to the cloud provider and can make reservation in advance for instance by cutting reservation fees from the computed total consumption cost of the registered client. Also, one more reason which can further increase the exploitation of the resources reserved by the clients is the grant of certain amount of additional discounts on the capacity to be reserved.

In reality, we can say that reserving a resource is a kind of a game that the cloud consumer and cloud provider usually play. Both parties seems to play some kind of strategy in that particular game which ultimately maximizes their utility respectively i.e. Maximal computations performed for minimum costs. The strategic decision of the cloud consumer is made with the help of forecast of demands and the amount of capacity that is needed for resource reservation

With the use of reservation method, cloud clients can have their resources reserved easily in advance. So, under-provisioning problem always occurs whenever resources that we

have reserved are not able to completely meet the demands of consumers due to the uncertainty involved in it. Allocating few more resources using the on demand plan can solve this problem so that the extra demand is fulfilled but more expensive prices of the provisioning of resources with the on demand plan can also include some other overall higher costs. Also, if the resources that are reserved are more than the demand made actually then the problem which can arise here is of over provisioning and resource pools some part is going to be underutilized in such a case. The important thing the cloud consumer has to do now is to decrease the in-total price of allocation the resource through decreasing “on-demand” costs and the “over-subscribed” costs of the under provisioning and also the over provisioning methods. For achieving this goal the issue to be resolved the computing resource management optimally.

There are mostly n number of various clients and various n numbers of end users that are involved in the cloud environment. Many times clients often make some kind of requests to the cloud system and that too also, all at once and thus making the entire management of load at a time by the cloud way more difficult.

The load can be:

Central processing unit load,

Memory load,

Network or delay load

This may also in return cause inconvenience to the registered clients by affecting the performance and the efficiency of the given cloud environment or by causing further delay in the response time. Balancing the load effectively can make the cloud computing way more efficient and can also improve the user satisfaction.

1.3 Objective:

To improve the efficiency in the working of the cloud, it is very important to balance the load in the cloud. A good load balancing technique can make the cloud computing more efficient and can also improve the satisfaction among users. We will be giving an approach to balance the incoming load in cloud environment.

Also we will be studying and analysing the various algorithms for scalable resources and will be concluding it by the comparative study of those mentioned algorithms.

1.4 Methodology:

The main use of project methodology is to permit the developer to manage the complete management method issue by taking selections so finding solutions to the matter effectively, whereas making certain the right execution of some particular activities, methods, ways and latest technologies. Basically, a strategy gives a structure in order to explain each thing fully, in a way that a PM can apprehend what should be tried and done so as to give and execute the task in keeping with the timeline, resource available and client requirements.

According to some definition that is mentioned above, a fittingly taken project management method makes it easy for the user to have the subsequent success:

- Wants of stakeholders outlined
- Some kind of one language is used and understood within the team, in order to apprehend their expectations
- Most conflicts are measure noticed and resolved early
- Each task is completed employing a method way
- Expected deliverables created and handed to user
- Problems and mistakes are learned and the ways to solve them are quickly enforced
- Cost estimates measure complete, correct and credible

Here is a simple example of how we can present a project methodology in the hierarchical structure of management:

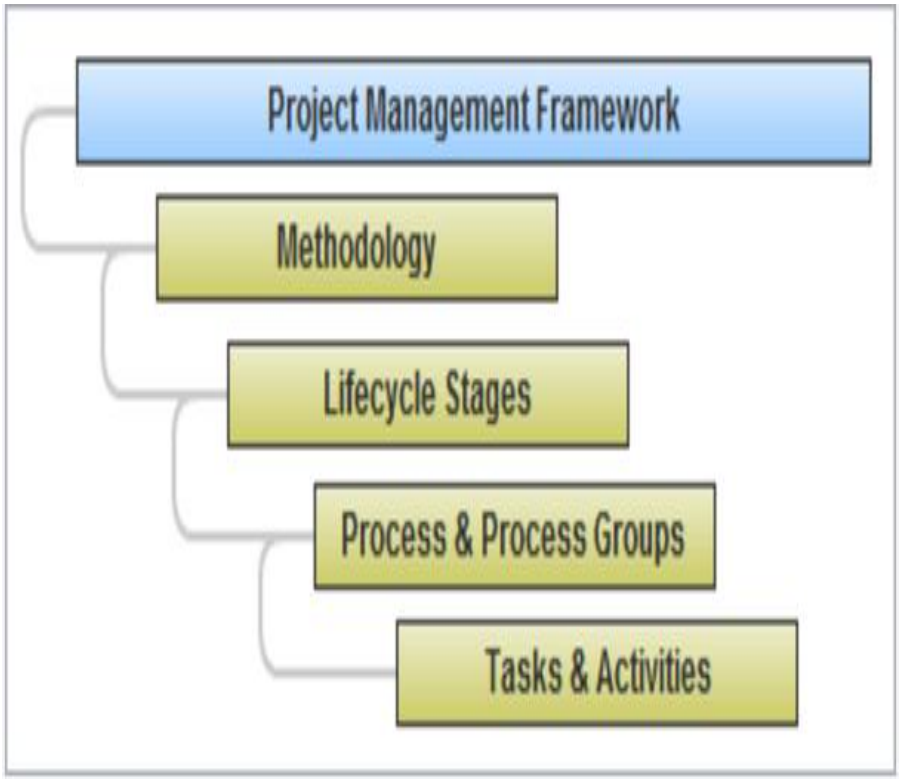


Fig.1.3Hierarchical Structure of project Methodology

Project methodology of all types can be divided into traditional and modern approaches.

Traditional approach has many numbers of subsequent phases within a project management method. Has the characteristics of gradual way to develop, or design or providing the facilities. Here, the thing involves gaining the progress within the execution method and gives advantages of "milestone-based planning and team building". Here, within software system development, the kind of methodology is termed as "*Waterfall*" i.e. various phases of work steps are carried in linear sequence.

The given stages are involved the above method:

- "Initiation i.e. requirements specification"
- "Planning and design"
- "Execution i.e. construction and coding"
- "Control and integration"

- “Validation i.e. testing and debugging”
- “Closure i.e. installation and maintenance”

Modern methodologies are not specialised in sequenced processes however it gives another way into management of project. A number of the strategies are great for the developing software, whereas another ones are often enforced in producing, method improvisation, product engineering, also many more. Now days these approaches use completely varied methods of the management.

A package life cycle model (also referred to as method model) could be a descriptive and represented illustration of the software life cycle. A life cycle model represents all the activities needed to create a software product transit through its life cycle phases. It conjointly captures the order within which these activities are to be undertaken. In alternative words, a life cycle model maps the various activities performed on a software product from its origination to retirement. Completely different life cycle models could map the fundamental development activities to phases in numerous ways. Thus, in spite of that life cycle model is followed, the fundamental activities are enclosed in all life cycle models. Though the activities is also dole out in numerous orders in numerous life cycle models. Throughout any life cycle section, over one activity might also be dole out. For instance, the design section would possibly comprise the structured analysis phase followed by the structured design step.

The development team should establish an acceptable life cycle model for the actual project so adhere to that. While not victimisation of a selected life cycle model the product of merchandise wouldn't be in a very systematic and disciplined manner. Once merchandise is being developed by a team there should be a transparent understanding among team members concerning once and what to try and do. Otherwise it might cause chaos and project failure. This downside may be illustrated by an example. Suppose a code development downside is split into many elements and also the elements are allotted to the team members. From then on, suppose the team members are allowed the liberty to develop the elements allotted to them in no matter approach they like. Its potential that one

member would possibly begin writing the code for his half, another would possibly conceive to prepare the check documents initial, and a few different engineer would possibly begin with the planning section of the elements allotted to him. This might be one among the right recipes for project failure. A code life cycle model defines entry and exit criteria for each section. A section will begin providing its phase-entry criteria are fulfilled. Thus while not code life cycle model the entry and exit criteria for a section can - not be recognized. While not code life cycle models like the classical waterfall model, repetitive falls model, prototyping model, evolutionary model, spiral model etc.) it becomes tough for code project managers to watch the progress of the project.

Waterfalls Model has been used for code development

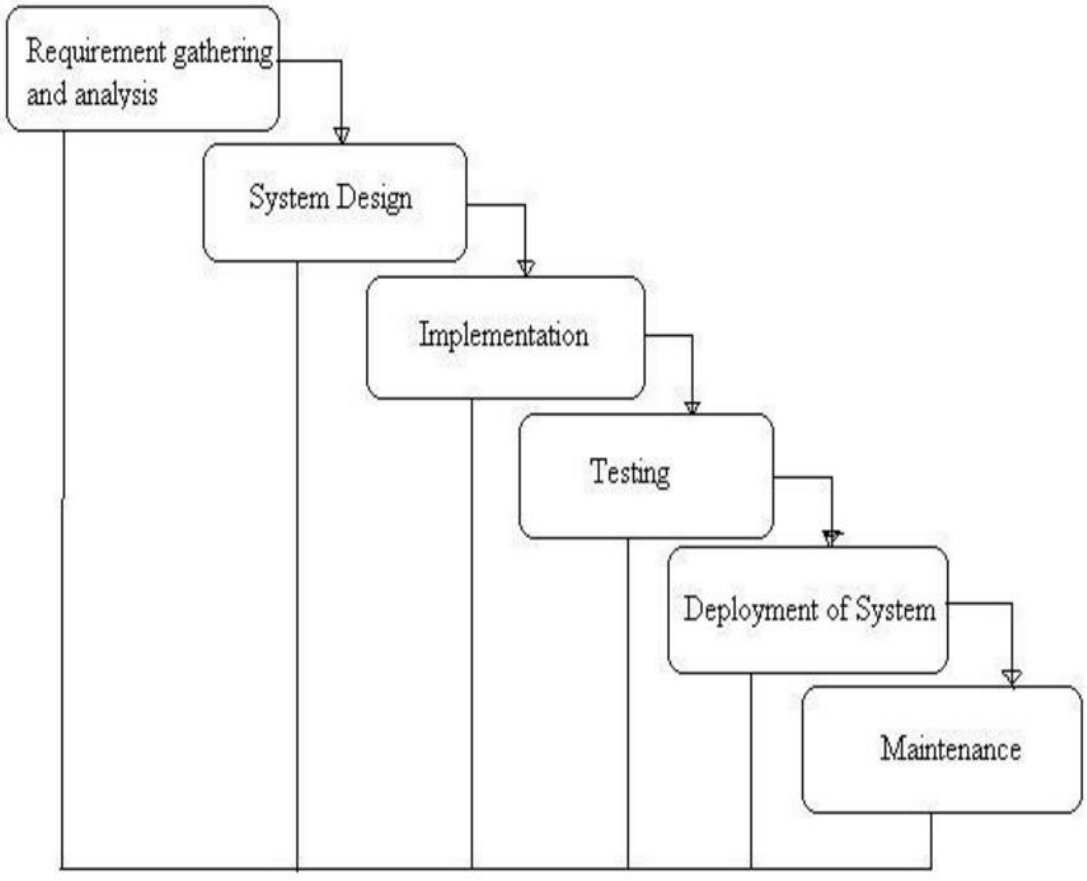


Fig.1.4: Waterfall Model

The waterfall Model was initial method Model to be introduced. It's additionally brought up as a linear sequential life cycle model. It's terribly easy to know and use. During a water model, every part should be completed totally before ensuing part will begin. This kind of model is largely used for the project that is little and there aren't any unsure necessities. At the top of every part, a review takes place to see if the project is on the correct path and whether or not or to not continue or discard the project. During this model the testing starts solely when the event is complete. In water model phases don't overlap.

Merits of waterfall model:

- Waterfall model works well for smaller projects where necessities are understood very clearly
- This model is easy and simple to know and use.
- In this model phases are processed and completed one at a time. Phases don’t overlap.
- It is better to manage owing to the rigidity of the model – every step has specific deliverables and a review activity

Demerits of waterfall model:

- Not apt for the projects where necessities are at a moderate to high risk of varying.
- Once an application is within the testing stage, it is terribly troublesome to travel back and alter one thing that was not going out well in the concept stage.
- Poor model for long and in progress projects
- No operating software is created till late throughout the life cycle.
- High amounts of risk and uncertainty.
- Not an honest model for difficult and object-oriented projects.

We can use the waterfall model when:

- The project is brief.

- This model is deployed only if the wants are fine best known, clear and not to be changed later.
- Product definition is stable.
- Ample resources with needed experience are offered freely
- Technology is known.

Very less client enter action is concerned throughout the process of the merchandise. Once the merchandise is prepared then solely it may be deemed to the clients/ users. Once the merchandise is developed and if any failure happens then the value of fixing such problems are terribly high, as a result of we want to update everyplace from document until the logic

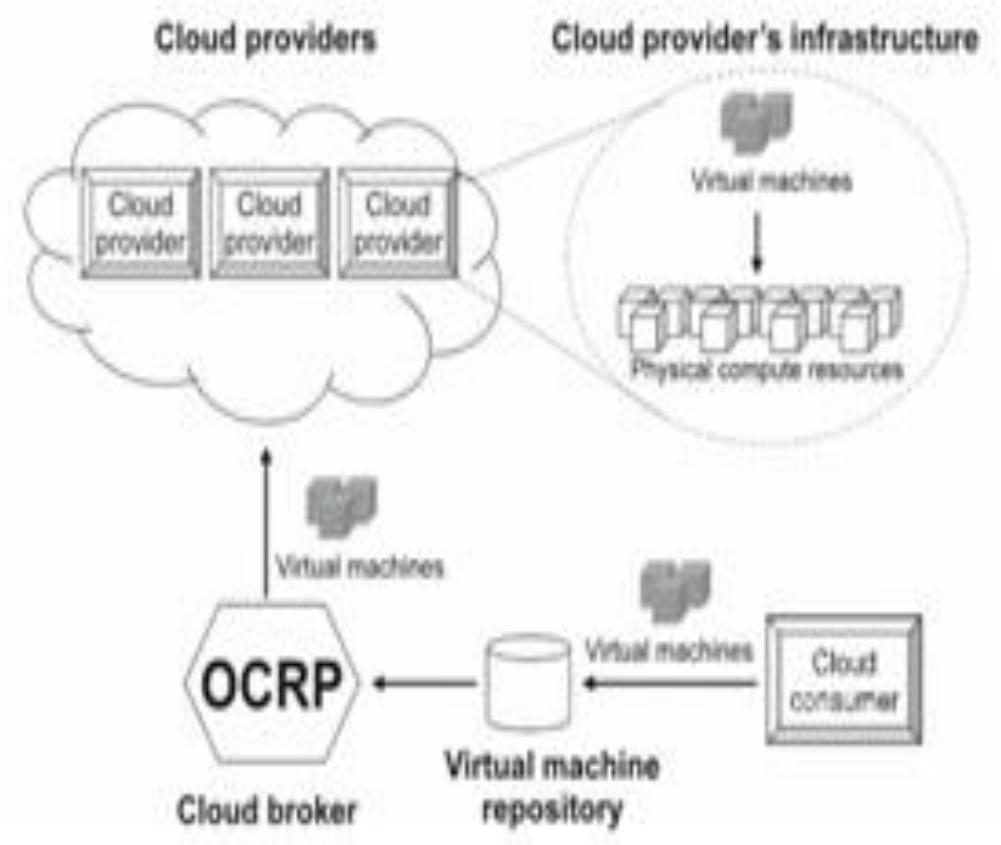


Fig.1.5:System Model of Cloud Computing

The cloud broker considers each reservation and on demand plans for provisioning resources. These resources are maximum times utilized in completely different time intervals, conjointly referred to as provisioning phases. There are basically 3 provisioning phases: reservation, expending, and on-demand phases. These phases with their actions perform in several points of your time as follows. Initially within the reservation section, while not knowing the consumer's actual demand, the cloud broker provisions resources with reservation set up beforehand. Within the expending section, the value and demand is completed, and therefore the reserved resources may be utilised. So, the resources that are reserved can be ascertained to be over-provisioned/under provisioned. In case the requirement is more than the number of resources reserved then the broker pays for extra resources on the basis of on-demand set up and thus, on-demand section starts.

Chapter 2: LITERATURE SURVEY

Literature Survey

Cloud computing is extremely popular now a days because it gives larger pliability and convenience of the resources that too at terribly less value. Key thing to focus on taking into account transferring the apps to public area of cloud computing are the upcoming services provided by cloud computing to possess extensive after effects on the systems and the various organisational networks. Several options which create cloud computing enticing also can different as compared to ancient security methods. Like some rising info technique space, cloud computing ought to be chased rigorously while having some thought to the criticality of information. Coming up with helps to make sure that the computing surroundings is as secure as potential and is in compliance with all relevant structure policies which knowledge privacy is maintained. It conjointly helps to make sure that the agency derives full get pleasure from info technology defrayal.

The protection objectives of a corporation tends to be a key issue for choices concerning outsourcing info technology services and, especially, for choices concerning transitioning an organisations' data, apps, and alternative resources to a pub. cloud computing surroundings. So that effectiveness is maximised and prices minimised, security and confidentiality should be taken care of from the very starting phase in the beginning of the model's developmental life cycle. Making an attempt in dealing with security one time execution and preparation isn't solely rather more tough and expensive, however conjointly a lot of risky. Clouds usually are not responsive to a particular company's security and confidentiality wants. Changes in the cloud computing surroundings are also bonded to fulfil the organization's needs. Organizations ought to need that some elect cloud computing that is made public resolution is organized, installed, and maintained to fulfil their safety and authenticity needs.

2.1Virtualization:

“Virtualisation is that something that is not real, however offers all the things of a true thing. Its code execution of a pc that can run totally different codes sort of real system. Virtualisation is said to cloud, as a result of victimization virtualisation as user will use totally different services of a cloud. The remote information centre can offer totally different services in an exceedingly totally or partial virtualised manner.”

Two kinds of virtualization are found just in case of clouds as given:

Full virtualization

Para virtualization

Full Virtualization: just in case of full virtualisation a whole installation of 1 machine is completed on the other machine. It'll end in a virtual machine which is able to have all the system code that is there within the actual server.

Here the remote data centre delivers the services during a totally virtualised manner. Full virtualization has been triple-crown for many purposes:

a. Emulating hardware on another machine

b. Uninflected users from one another and from the management program

c. Sharing a computing system among multiple users

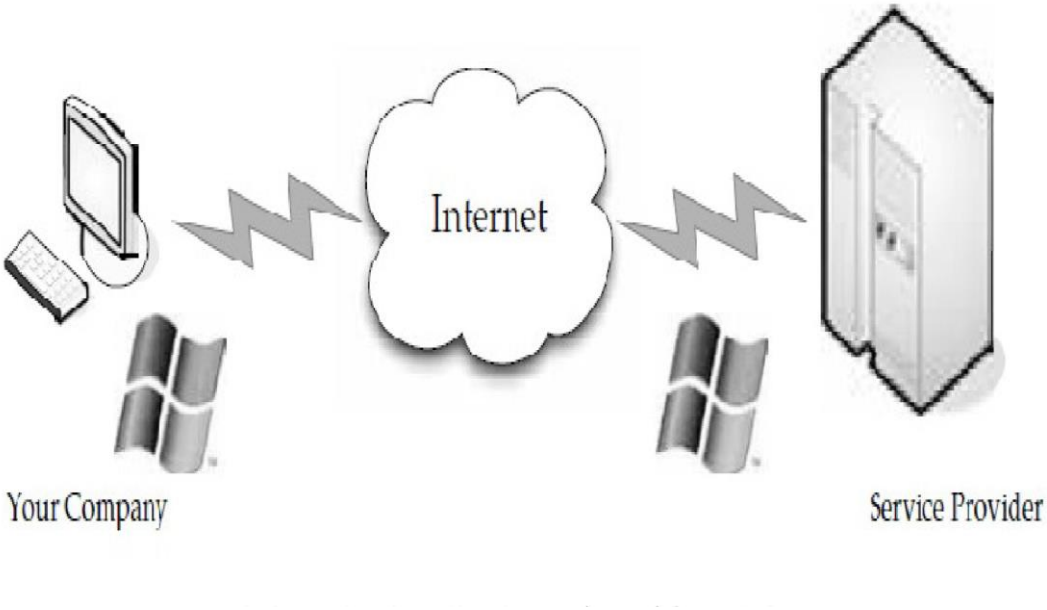


Fig.2.1: Full Virtualization

Para virtualization:

In para virtualisation, the hardware permits many OSs to run on single machine by economic use of system resources like memory and processor e.g. VMware software. Here all the services are not totally made available, rather the services are provided partially.

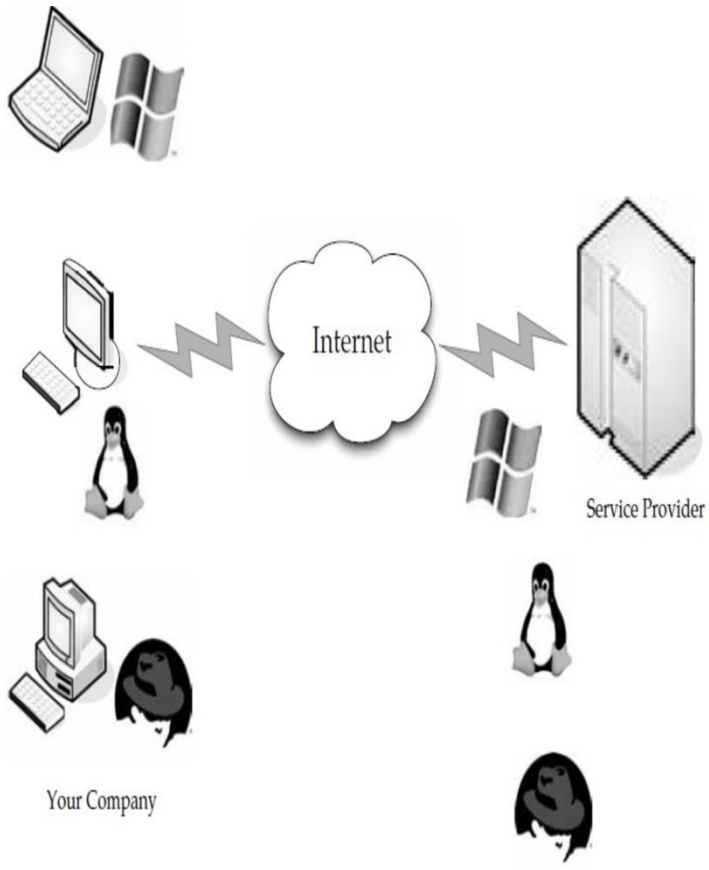


Fig.2.2: Para virtualization

Para virtualization has the subsequent benefits :

Migration:

Because the hardware will be replaced simply, thence migrating or moving the different components of a replacement machine is quicker and easier.

Disaster recovery:

Within the event of a system failure, guest instances are usually moved to other hardware till the machine is repaired or replaced.

Capacity management:

In a virtualized atmosphere, it's easier and quicker to add more disc drive capability and process power. Because the system components or hardware will be moved or replaced or repaired simply, capability management is straightforward and easier.

2.2 Cloud Components:

A Cloud system consists of three major parts like the “customers, datacentre, and various servers”. Every part encompasses one single use and has a selected role.

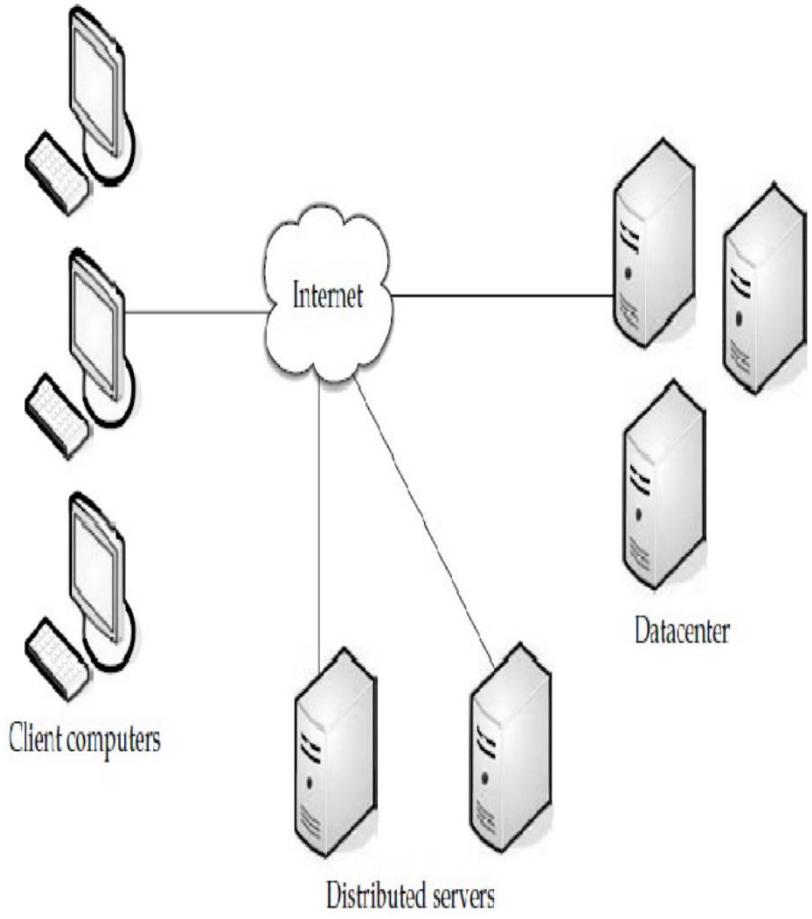


Fig.2.3:Cloud Components

Clients: actual users act with the customers to manage data associated with the cloud.

Currently a thin client is a lot common than rest different ones attributable to the less value, safety, less usage of the power, low noise, simply expendable, fixable extra.

Datacentre: “Datacentre is however a group of servers hosting completely different apps. A user connects to the datacentre to subscribe completely different apps. A datacentre could exist at an oversized distance from the end users.”

Currently, a thing referred to as virtualisation is employed to put in system code that permits multiple instances of virtual server apps.

Distributed Servers : Distributed servers are the components of a cloud that are there throughout the net hosting completely different apps. However whereas if we are executing it from the cloud, it is surely going to make the user realize what it is executing the app from his system.

2.3 Load balancing:

“Work reconciliation could be the technique to divide work among 1 or a lot of servers, interfaces, or different computing facilities. major data centre executions have confidence giant, power pack computing h/w and n/w structure, that have the same old risks related to some device, together with h/w failure, power and network interrupts, and limited resource in case of more needs.”

Stack compromise inside the cloud varies from established thinking on load-adjusting plan and usage by utilizing the accessible servers to play out the work compromise. This accommodates fresh out of the box new open doors and economies-of-scale, moreover as introducing its own particular unmistakable arrangement of difficulties.

It is utilized to shape sure that no current assets are ideally working without moving while others are used. To adjust load conveyance, you'll move it from the supply hubs to moderately tenderly stacked end hubs.

When you this all throughout execution time, it's alluded to as dynamic load compromise — this will be understood each in an direct or unvaried way with regards to the implementation of hub choice:

• inside the unvaried methodologies, a definitive end hub is set along numerous cycle steps.

• inside the direct techniques, a definitive goal hub is picked in one stage.

Other reasonably Load reconciliation technique are often used i.e. the irregular hydraulics Load reconciliation technique , a hybrid technique that takes advantage of each direct and unvaried strategies.

Goals of Load balancing:

The goals of load levelling area unit:

1. to accommodate future change within the system.

2. to enhance the performance well.

3. to take care of the system stability.

4. to possess a backup management in case of failure of system even part

Types of Load balancing algorithms:

Depending on who initiated the method, load levelling algorithms will be of 3 classes as follows:

Sender Initiated: If sender initialise the load levelling rule.

Receiver Initiated: If receiver initiate the load levelling rule.

Symmetric: It is the mix of each sender initiated and receiver initiated.

Depending on the present state of the system, load levelling algorithms will be divided into two classes as given in:

Static: Previous data of the system is required. It doesn't rely upon the present state of the system

Dynamic: No previous data is required. Selections on load levelling are supported by current state of the system. Thus it is much better than the static approach. Here we'll discuss numerous dynamic load balancing algorithms for the clouds of various sizes.

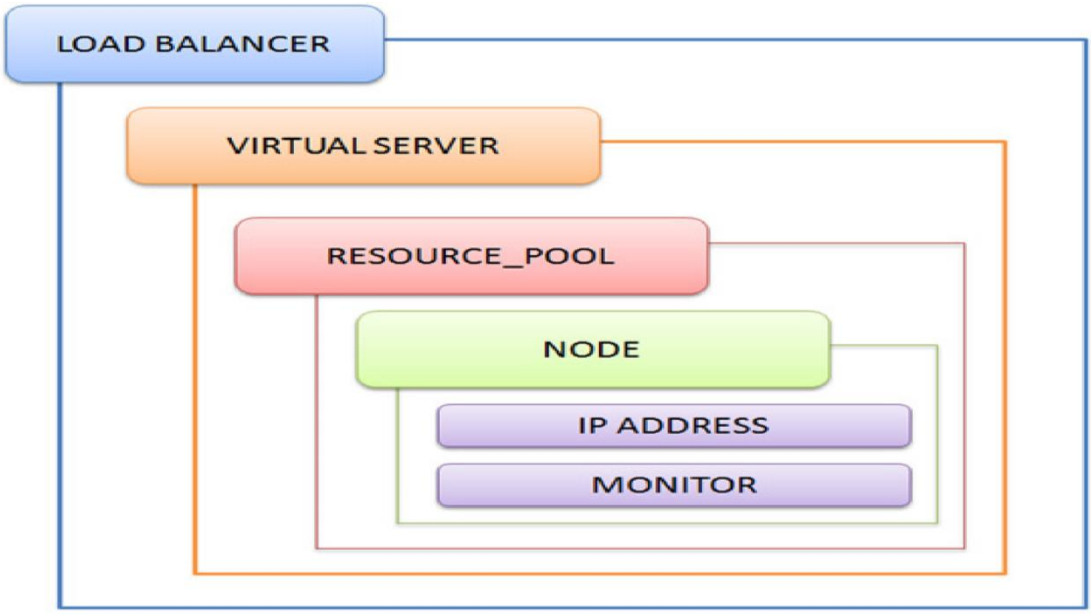


Fig.2.4: Load Balancer

Dynamic Load equalisation algorithmic rule:

During a distributed system, dynamic load equalisation may be drained 2 completely different ways:

Distributed and non-distributed:

Within the distributed one, the dynamic load equalisation algorithmic rule is executed by all nodes gift within the system and also the task of load equalisation is shared among them. In order to attain the load equalisation the interaction among nodes will take 2 forms: cooperative and non cooperative. Within the cooperative form, the nodes will work side-by-side in order to attain a distinctive objective, as an example, to boost the general latency, etc and within the non-cooperative type, every node works severally toward a goal native thereto, as an example, to boost the latency of a neighbourhood task. Non-distributed algorithms generate very few messages as compared to the dynamic load equalisation algorithms as a result of which every node within the system must act with each different node. A benefit of dynamic load equalisation algorithm is that even when

one or a lot of nodes within the system fail, it will not cause the full load equalisation method to stop the progress of, it instead would have an effect on the system performance to some amount. Distributed dynamic load equalisation will initiate great pressure on a system within which every node must interchange standing data with each different node within the system. There is an advantage when most of the nodes act separately with only a few interactions with the other nodes.

In a non-distributed sort, either one node or a multitude of nodes do the task of load equalisation. Non distributed dynamic load equalisation algorithms will take 2 forms: centralized and semi distributed. In centralised type, the load equalisation algorithmic rule is run exclusively by one node within the whole system namely the central node. This node is only accountable for load equalisation of the complete system. The opposite nodes act only with the central node. In semi-distributed type, nodes of the system square measure divided into clusters, wherever the load equalisation in every cluster is of centralized type. A central node is non appointive in every cluster by applicable election technique that takes care of load equalisation at intervals that cluster.

Hence, the load equalisation of the complete system is completed via the central nodes of every cluster.

Centralized dynamic load equalisation takes fewer messages to succeed in a call, because the variety of overall interactions within the system decreases drastically as compared to the semi distributed case. However, centralized algorithms will cause a bottleneck within the system at the central node and conjointly the load equalisation method is rendered useless once the central node crashes. Therefore, this algorithmic rule is most fitted to networks with tiny size.

Policies or Strategies in dynamic load balancing:

There are 4 types of policies involved:

Transfer Policy: The part of the dynamic load equalization algorithmic program that selects employment for transferring from a nearby node to a distant node is said as Transfer policy or Transfer strategy.

Information Policy: The part of the dynamic load equalization algorithmic program answerable for grouping info concerning the nodes within the system is said as info policy or info strategy

Selection Policy: It specifies the processors concerned within the load exchange (processor matching)

Location Policy: The part of the load equalization algorithmic program that selects a destination node for a transferred task is said as location policy or Location strategy.

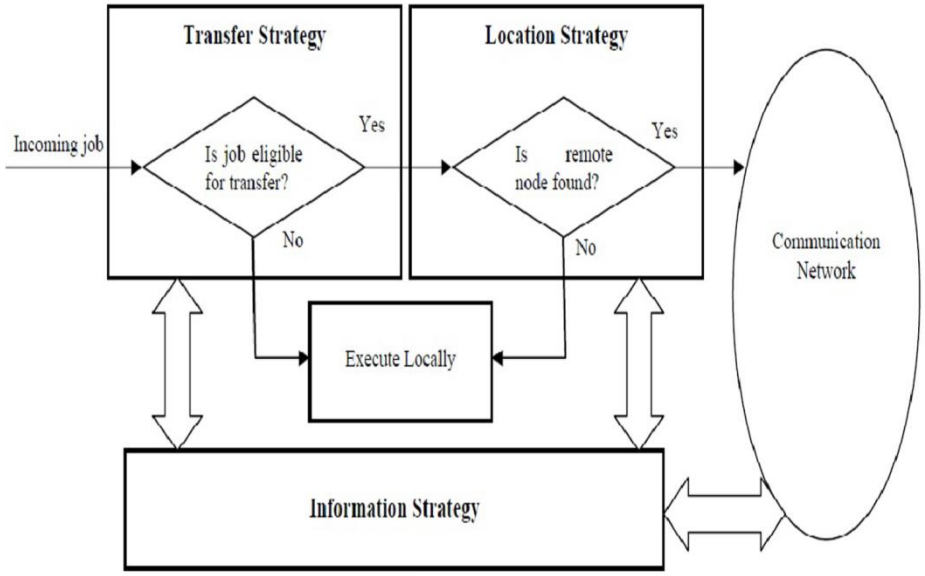


Fig.2.5: Interaction among components of dynamic load balancing algorithm

The computational grid can be a dependable platform that has a huge pool of resources for distributed recursive request. Such platforms are highly efficient as compared to ancient high level computing systems. Moreover, computational grids have different

Hardware and software constraints and necessities than those of ancient superior computing systems. To completely use such computational grid systems, programming and resource management are the key services, wherever problems with system task allocation and processor and network load utilisation represent a lower side for many systems with grids. The methods which contain balanced load and algorithms that equally unfold the load on every computing system, increasing resource utilization and reducing the task processing time.

Generally “Load-Balancing” algorithms are often sub-divided as dynamic or static and localised or centralised. Under centralized load-balancing approach, any one node of the grid system behaves as hardware and takes each and every system of load balancing. Data is distributed from all different nodes to the present node. In the localised load-balancing approach, all nodes of the grid system are concerned for the algorithm choices. So it is terribly processor demanding for every node to get into and synchronously manipulate the non-static state of the entire grid system. In general localised data distribution approaches have been maintaining solely partial data domestically to create sub-optimal choices.

Static algorithms presume that load-balancing choices which will embody the features of the roles and the data computing data ends and therefore network communications are proverbial before. These choices are created probabilistically or deterministically made by the compiler at compile time and stay throughout execution time. The non-dynamic algorithms harbor a terrible con which is, algorithm assumes the characteristics of the communication network and computing resources are all proverbial before and stay constant. Such a pre-occupied thought will not harbour a grid computing atmosphere. In distinction, non-static algorithms comply to utilize the execution time state of data to generate choices pertaining to load balancing. Apart from this, the static load-balancing algorithms are much simpler to apply and has stripped overhead. On the other hand, dynamic load-balancing approaches might impact the higher execution efficiency.

The major con of non-static rules with load balancing is their anticipation towards inconsistency in checking of data performance that the load-balancing algorithm utilizes for its functions. Many of these programming techniques are highly sensitized to

inconsistent results and may commute useless outputs data accuracy is merely slightly but a 100%; in real grid load-balancing environments, moreover, hundred percent accuracy in data is exhausting to attain properly for the acquisition of results.
The questionable hybrid programming is another space that lately have been retrieving some attention. The hybrid balancer of loads tries to mix the advantages of dynamic and static load-balancing programming techniques and therefore reduces their relative inherent disadvantages in terms of dynamic and static equalisation.

It is difficult to differentiate between the definition of static and dynamic load-balancing since it has not been elucidated and many contributors utilize distinguished definitions of dynamic and static algorithms. Also, in distributed systems a hybrid rule for sharing the divided load and was then analyzed. Some studies are done victimising models and game-theoretic algorithms for load equalization in an exceedingly atmosphere of grids. An analysis on load equalization in commodity divided systems, framing these as a non cooperative game with War drop equilibrium because aim has been mentioned before. A lot of recently, a non cooperative game for distributed systems of balanced load additionally bestowed before; victimising the idea of Poisson arrival and exponential service times, a rule used to calculate Nash equilibrium has been arrived at. But, not even one of those papers has this fact in consideration for communication latency in an exceedingly grid atmosphere that will seriously affect the grid systems with irrelevant outputs.

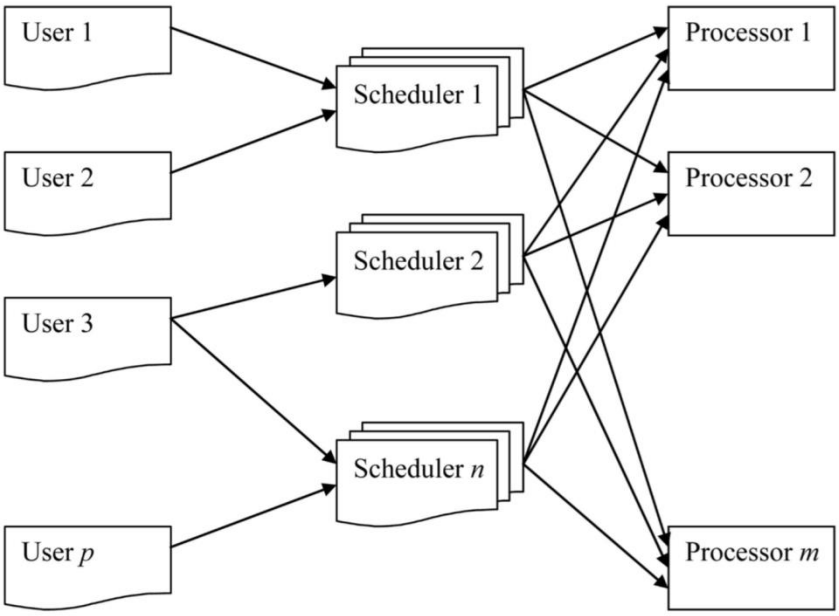


Fig.2.6: Relationship between users, schedulers and processors

2.4 General terms in cloud computing:

User: "This generates tasks to be run by the processors. Every user sends a task to hardware to be scheduled for process. Note that a user could send tasks to one/more hardware i.e. scheduler."

Scheduler: "This receives tasks from a collection of users and assigns them to the processors within the grid system. Each time a task is received from a user, the hardware decides that method or can process the task and send the task to it processor. Ideally, there would be more users than schedulers within the system. As such, the tasks scheduled by the schedulers area unit associate degree combination from several users."

Processor: "This executes and processes tasks sent to that. Every processor incorporates a queue that holds tasks to be executed; every task is then processed on a first-come, first-serve (FCFS) basis."

Fig. tells the connection between users, schedulers, and processors

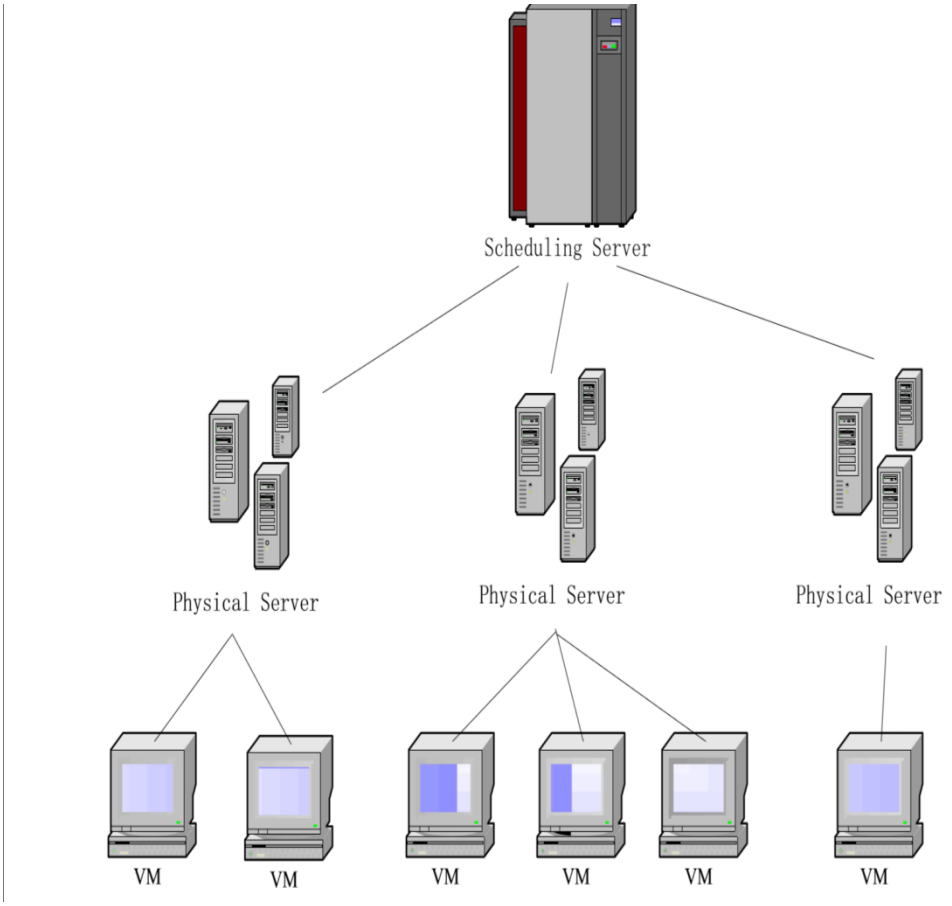


Fig.2.7 System Structure

Currently varied programming systems with distinguished models have harbored plenty of analysis date for proper analysis and ways to handle ever rising scale and complexness in these kind of systems. Honey-bee hunt answer to such problems is investigated as an instantaneous execution of one of these phenomenon. However, a distinguished, based on sampling model that manipulates each and every balanced load close to a world mean live is examined. Finally, associate degree rule for connecting image services by native rewiring is assessed as a way of rising load equalization through non-passive system structuring.

Chapter 3: SYSTEM DEVELOPMENT

3.1 Design

As the distributed computing might be another sort of computation on net. Its few advantages close by some vital issues to be settled to lift duty of cloud environment. The issues are associated with:

"Load administration,"

"adaptation to non-critical failure and"

"diverse security issues in cloud environment"

Amid the research, for most focus is load levelling in distributed computing. The work is regularly CPU stack, storage capacity, postponement or system load. Load levelling, a system of distribution is that the system of distinguished varies the boosted method for each resource utilization and reaction time whereas additionally gripping for a state of problems where a number of data ends square measure loads whereas alternative nodes square measure nothing or accomplishing without any resources.

Load levelling ensures that each one processor within the system or every node within the network will or so the equal quantity of labour at any instant of your time. Several strategies to resolve this downside has been came into existence like:

I Particle Swarm improvement,

II hash technique,

III genetic algorithms and

IV several programming primarily based algorithms.

In this paper, a way was supported by Ant Colony Optimisation to solve the matter of load levelling in given cloud atmosphere.

Also, to prove that the Ant Colony optimisation is better i.e. is more efficient and performs better as compared to some other algorithms, we are here doing comparative study of ACO with some other algorithms such as:

1. Round Robin
2. FCFS
3. Simple genetic algorithm

In all various algorithms, comparative study is done by varying two important parameters of cloud computing and those are:

1. Number of requests made to the cloud by the clients
2. Number of virtual machines available in the Virtual machine repository

3.2. Model Development:

For resource allocation, the resources are allotted in inventory accounting manner ie. FCFS. It a procedure for sorting out and controlling a buffer, wherever the first section, or "head" of the line, is prepared firstly. It's practically equivalent to handle a line with first-come, first-served (FCFS) conduct: wherever the people leave the line inside the request inside what they arrive.

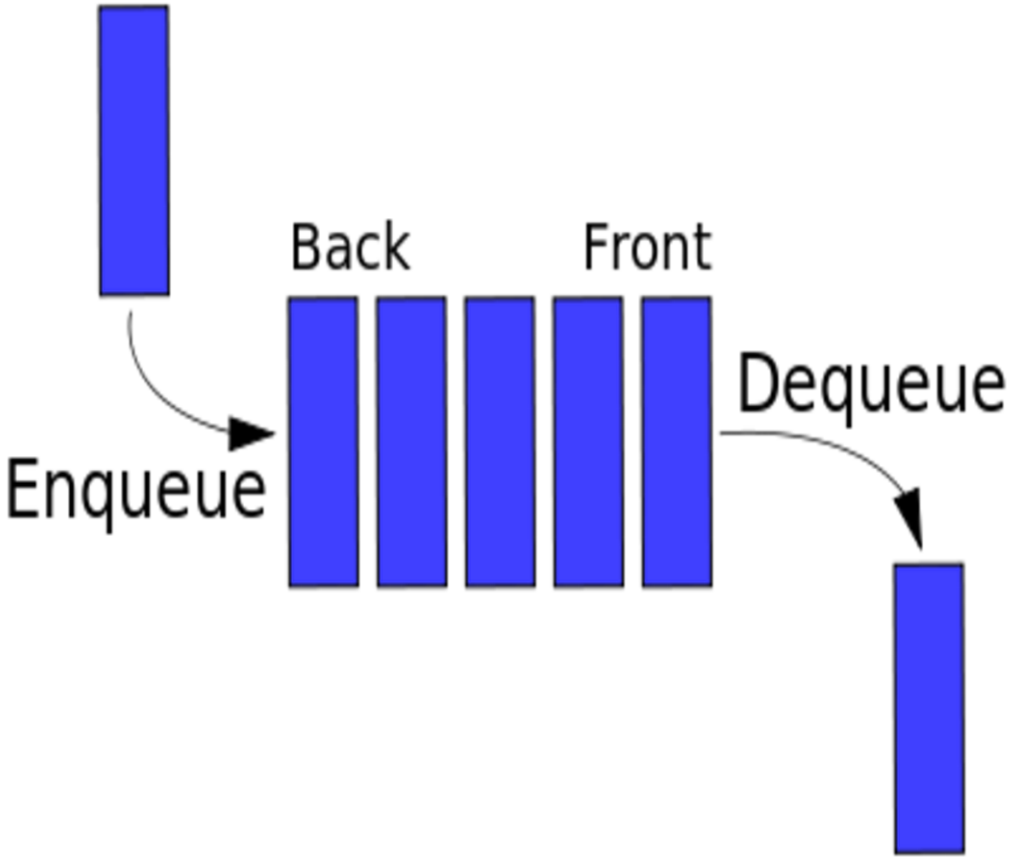


Fig.3.1: Representation of a FCFS queue

Round Robin DNS could be a method for work division, stack levelling, or adaptation to internal failure provisioning different, excess net Protocol benefit hosts, e.g., Web server, FTP servers, by dealing with the name System's (DNS) reactions to deal with solicitations from customer system in line with applicable applied mathematics display.

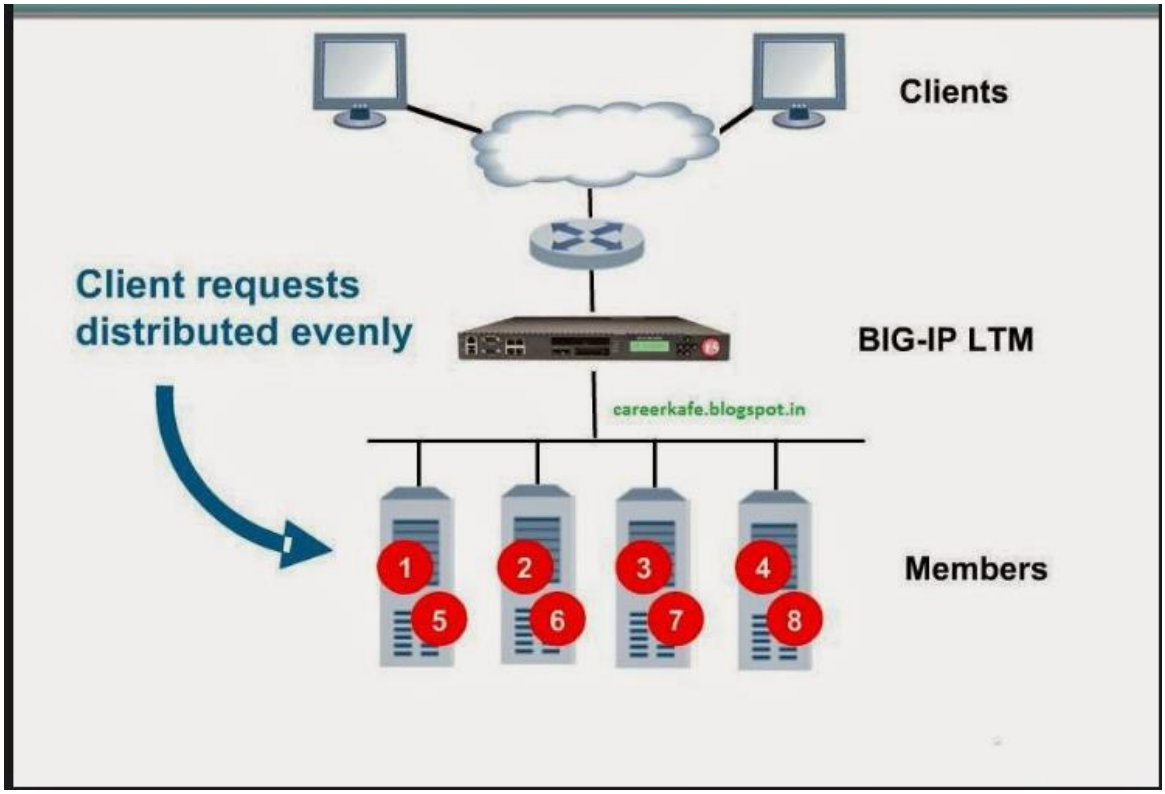


Fig.3.2: Load management using round robin

Simple Genetic algorithm:

The GA process is repeated until the fittest chromosome (optimal way) is obtained or the termination condition (max. ways of iteration) is exceeded. The proposed algorithm is as given below:

Step 1: Randomly initialize a population of processing unit after encoding them into binary strings [*Start*].

Step 2: Evaluate the fitness value of each population using equation 3 [*Fitness*].

Step 3: **While** either maximum number of iteration are exceeded or optimum solution is found **Do**:

Step 3(a): Consider chromosome with lowest fitness twice and eliminate the chromosome with highest fitness value to construct the mating pool [*Selection*].

Step 3(b): Perform single point crossover by randomly selecting the crossover point to form new offspring [*Crossover*].

Step 3(c): Mutate new offspring with a mutation probability of (0.05) [*Mutation*].

Step 3(d): Place new offspring as new population and use this population for next round of iteration [*Accepting*].

Step 3(e): Test for the end condition [*Test*].

Step 4: End.

Ant Colony Optimisation:

Procedure for ACO:

“begin

Initialize the pheromone

while (stopping criterion not satisfied) do

Position each ant in a starting VM

while (stopping when every ant has build a solution) do

for each ant do

Chose VM for next task by

pheromone trail intensity

end for

end while

Update the pheromone

end while

end”

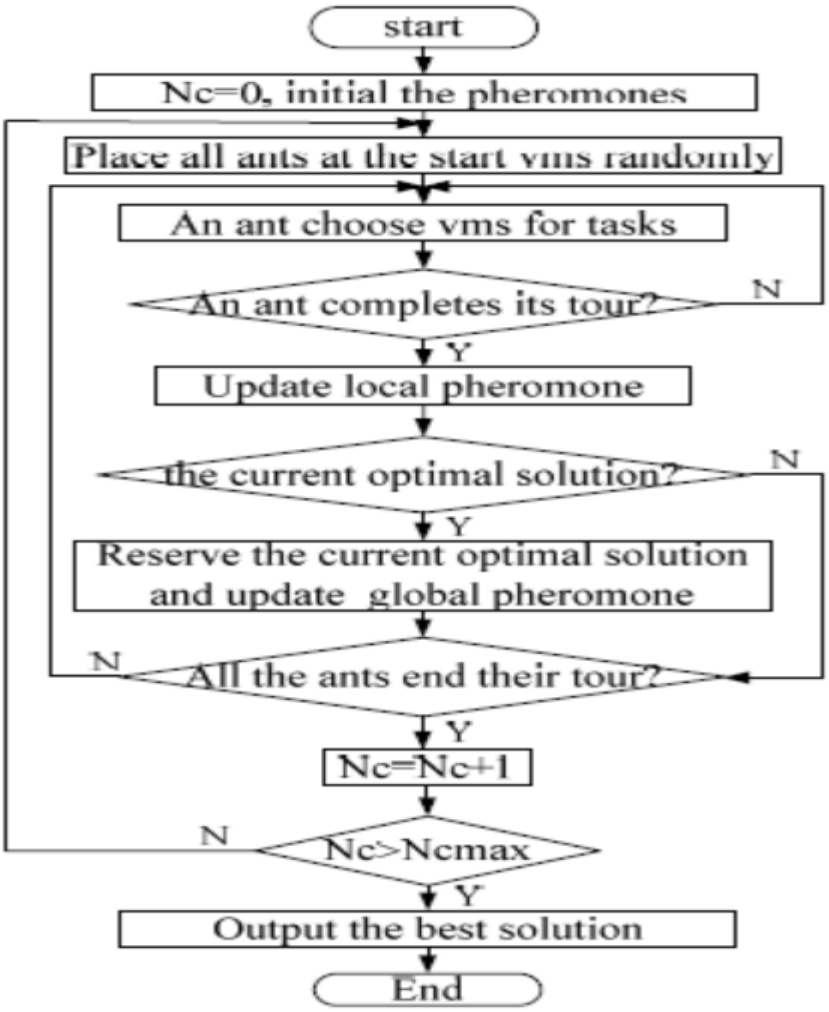


Fig 3.3 Simulations in ACO

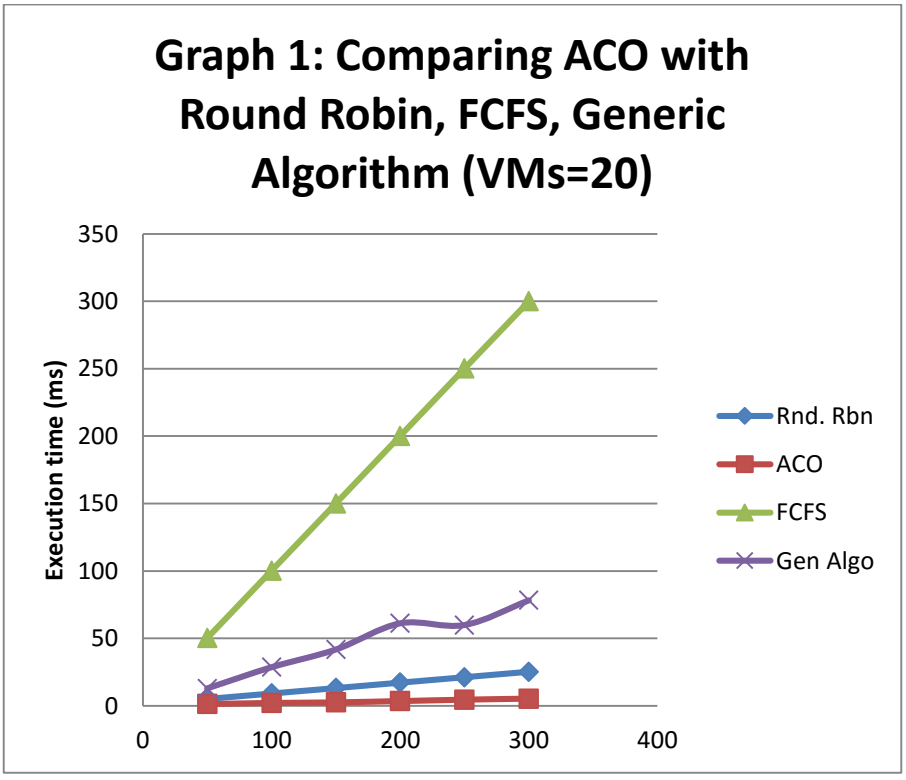
Chapter-4: PERFORMANCE ANALYSIS

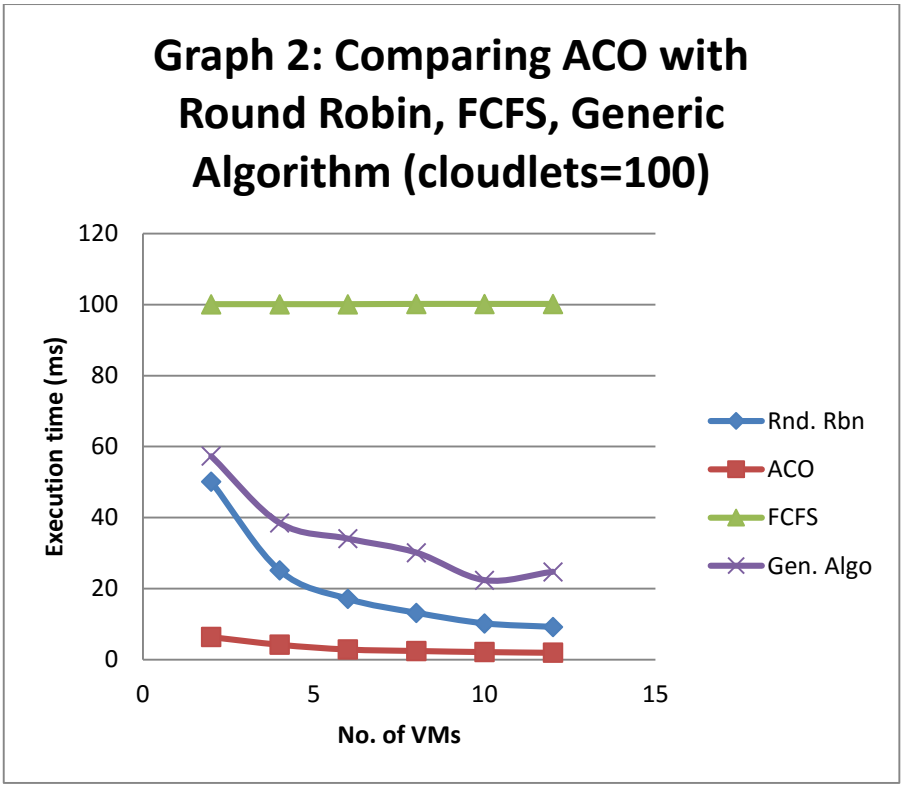
With the use of clousim 3.0.1, we implemented the above mentioned four algorithms for various different cases. The parameters for which reading for various algorithms is being taken are :

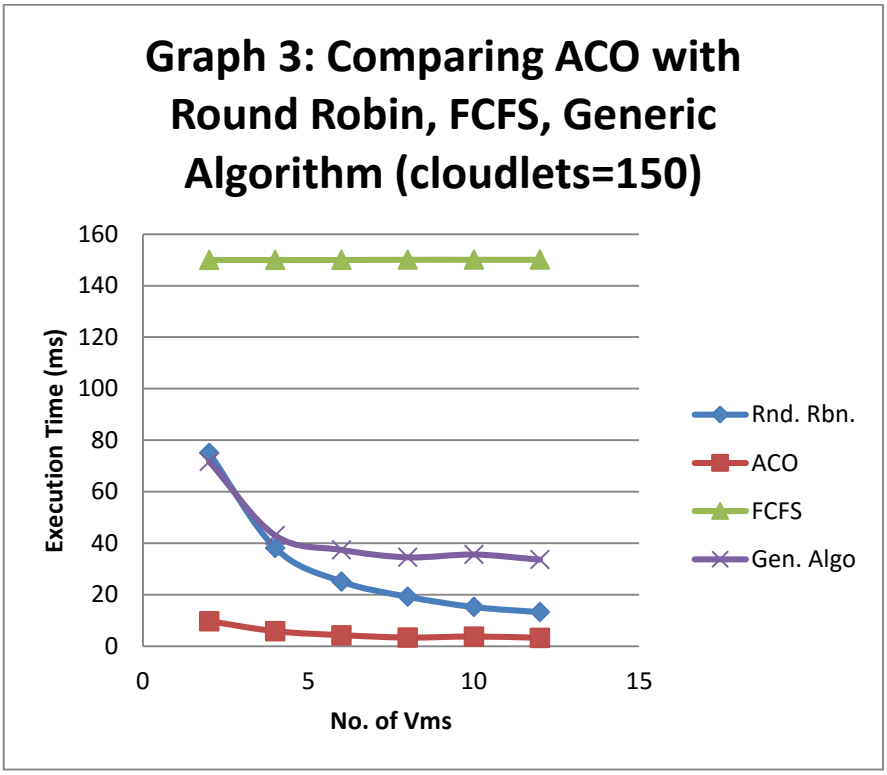
1. Number of requests

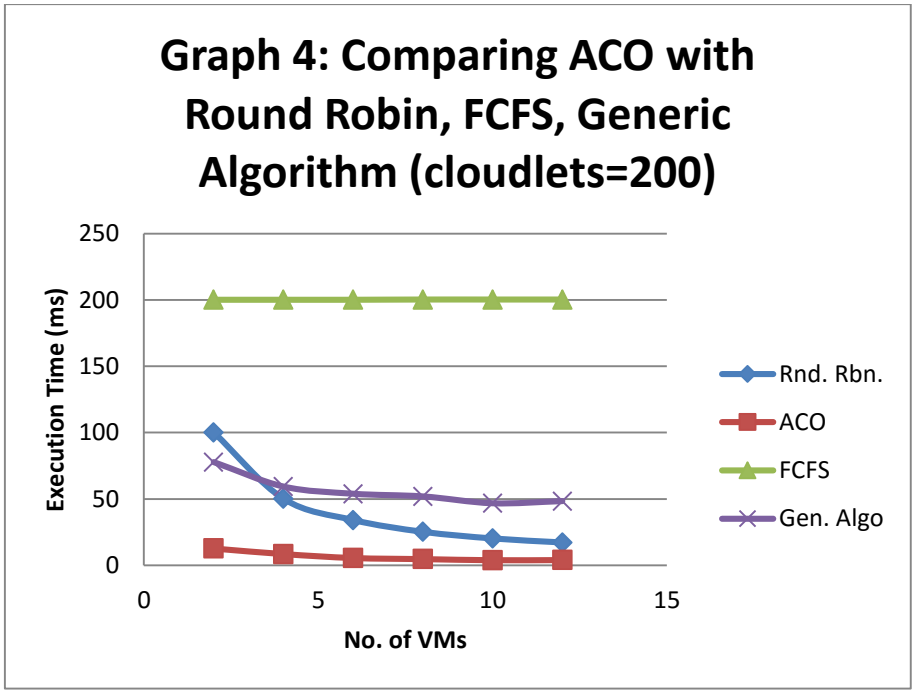
2. Number of virtual machines

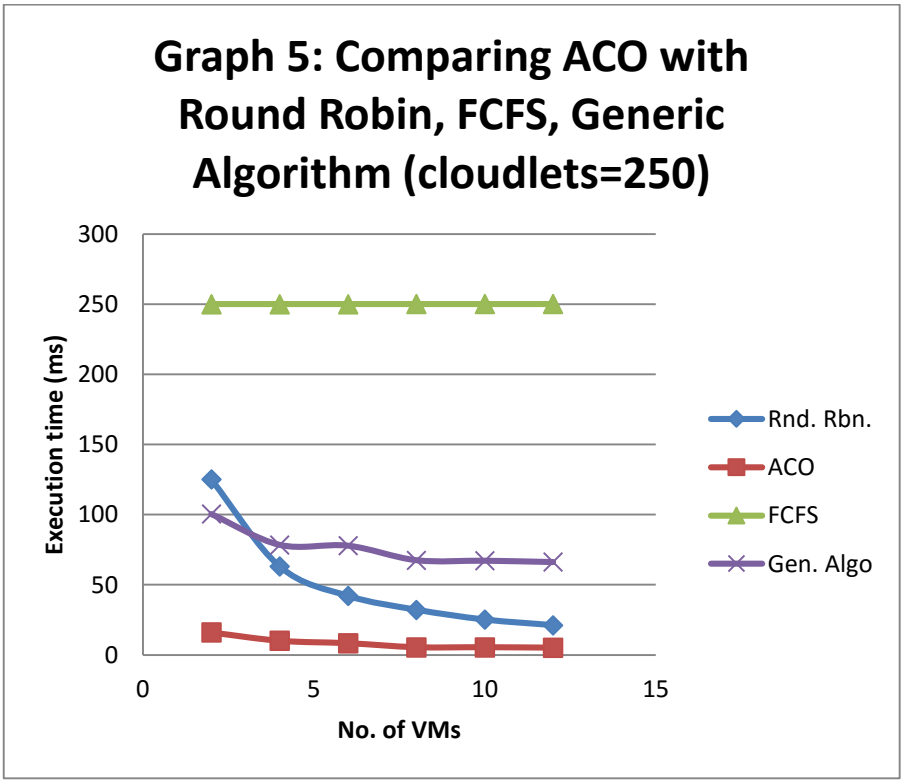
For the different values of these parameters, graphs are plotted for all the four algorithms and a comparative study is carried out.

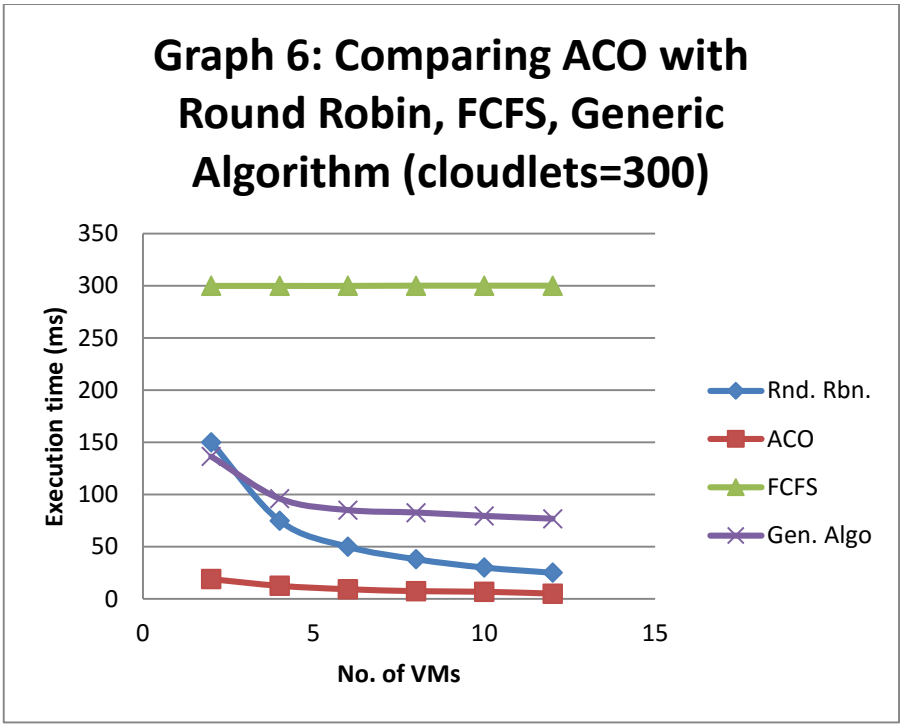


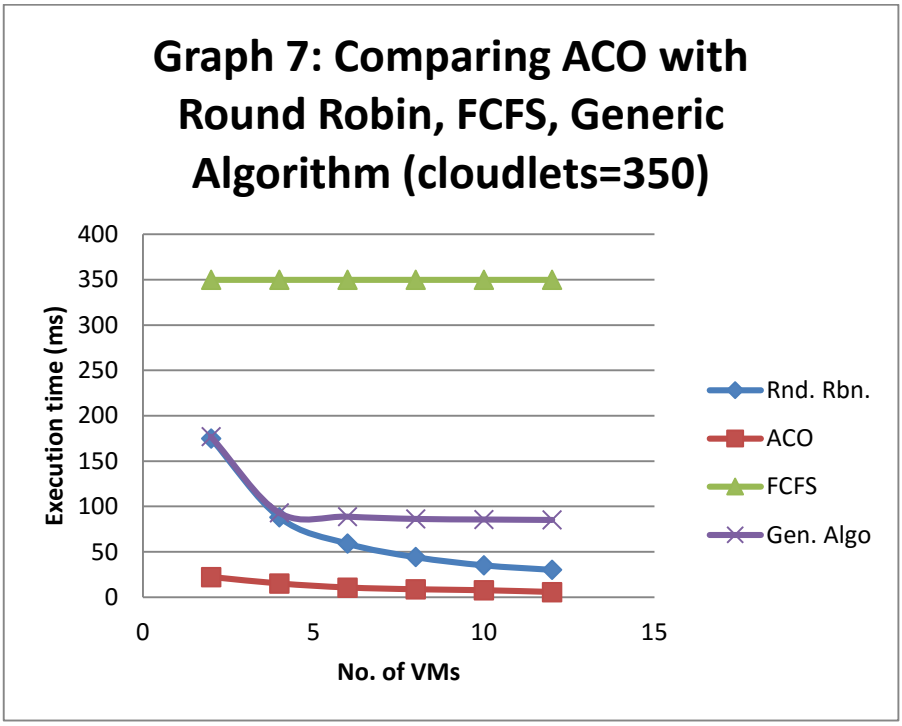

Graph 2: Comparing ACO with Round Robin, FCFS, Generic Algorithm (cloudlets=100)
120
100
80
60
40
20
0
Execution time (ms)
0
5
10
15
No. of VMs
Rnd. Rbn
ACO
FCFS
Gen. Algo

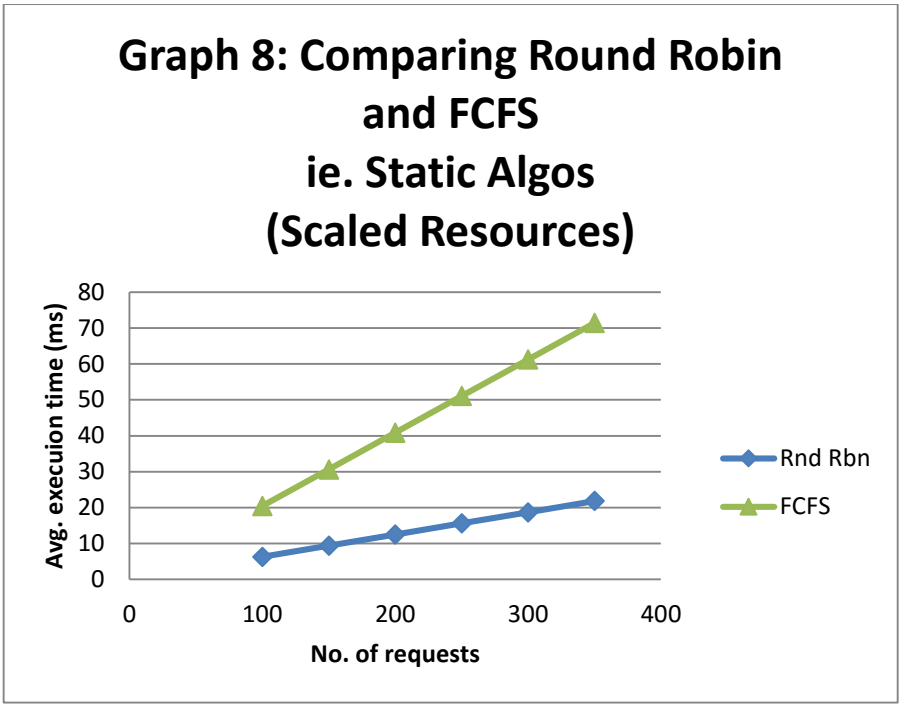

Graph 3: Comparing ACO with Round Robin, FCFS, Generic Algorithm (cloudlets=150)
160
140
120
100
80
60
40
20
0
Execution Time (ms)
0
5
10
15
No. of Vms
Rnd. Rbn.
ACO
FCFS
Gen. Algo

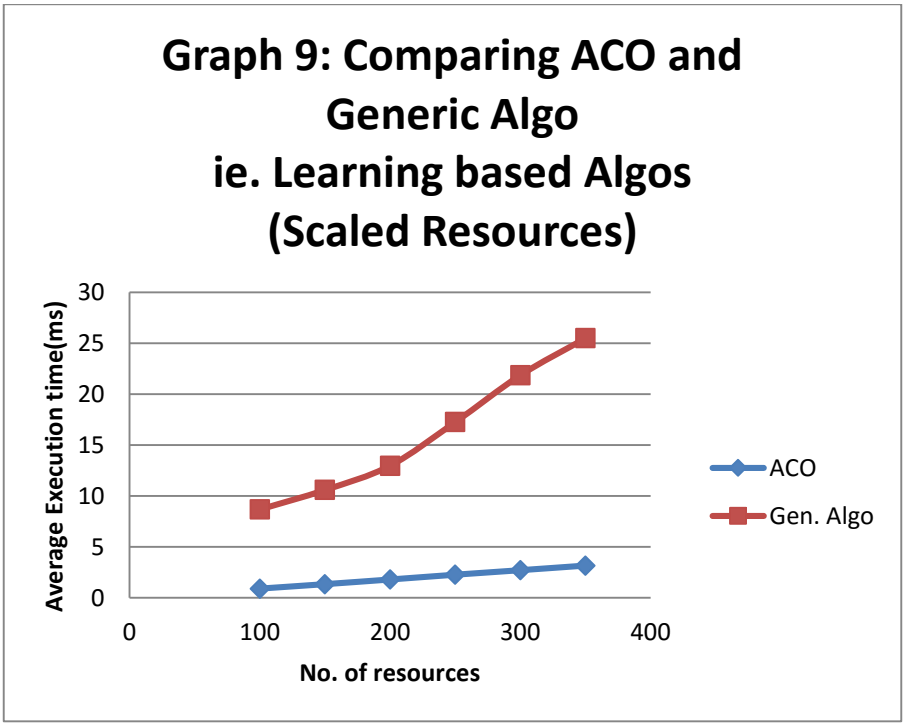

Graph 4: Comparing ACO with Round Robin, FCFS, Generic Algorithm (cloudlets=200)
Execution Time (ms)
250
200
150
100
50
0
0
5
10
15
No. of VMs
Rnd. Rbn.
ACO
FCFS
Gen. Algo

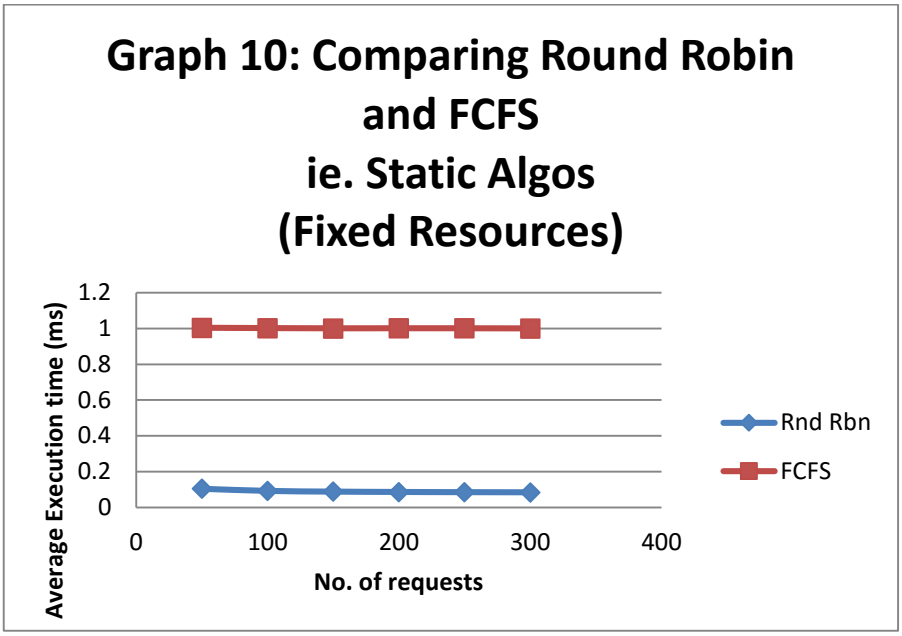

Graph 5: Comparing ACO with Round Robin, FCFS, Generic Algorithm (cloudlets=250)
Execution time (ms)
300
250
200
150
100
50
0
0
5
10
15
No. of VMs
Rnd. Rbn.
ACO
FCFS
Gen. Algo

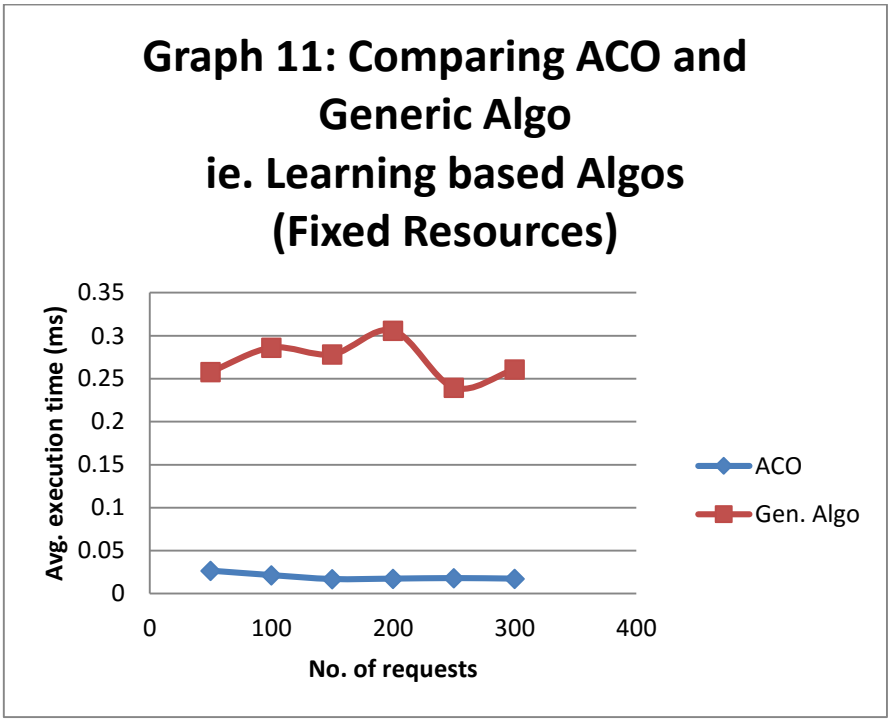

Graph 6: Comparing ACO with Round Robin, FCFS, Generic Algorithm (cloudlets=300)
Execution time (ms)
350
300
250
200
150
100
50
0
0
5
10
15
No. of VMs
Rnd. Rbn.
ACO
FCFS
Gen. Algo

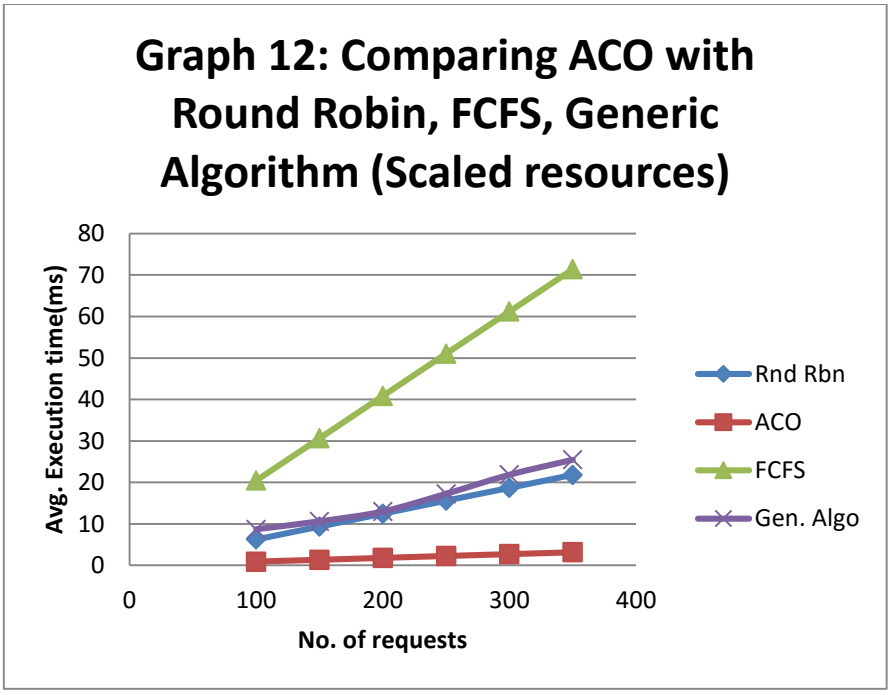

Graph 7: Comparing ACO with Round Robin, FCFS, Generic Algorithm (cloudlets=350)
Execution time (ms)
400
350
300
250
200
150
100
50
0
0
5
10
15
No. of VMs
Rnd. Rbn.
ACO
FCFS
Gen. Algo

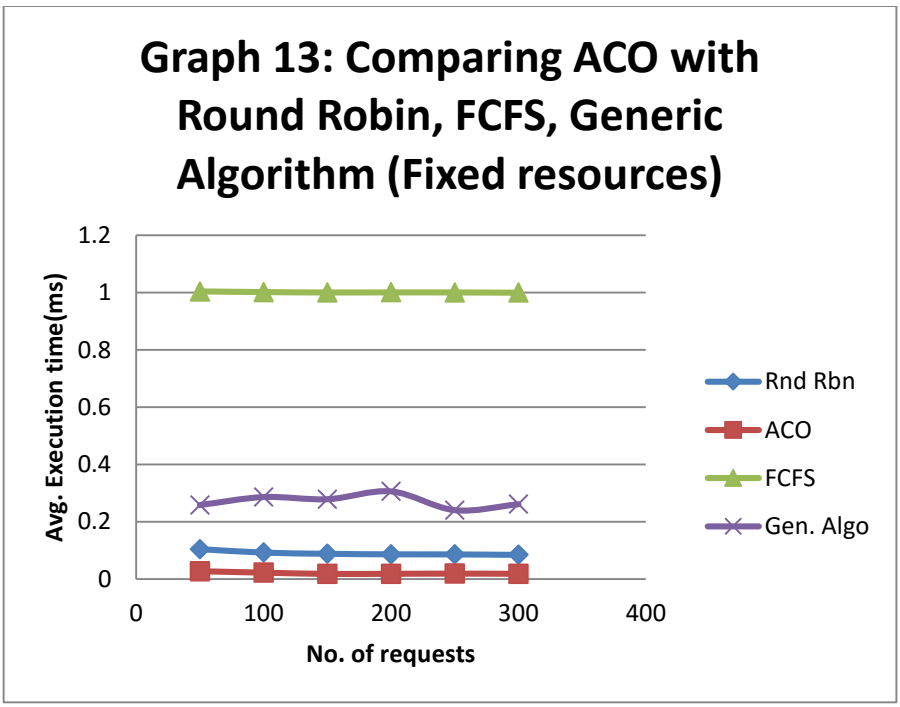

Graph 8: Comparing Round Robin and FCFS
ie. Static Algos
(Scaled Resources)
Avg. execuion time (ms)
80
70
60
50
40
30
20
10
0
0
100
200
300
400
No. of requests
Rnd Rbn
FCFS


Graph 9: Comparing ACO and Generic Algo
ie. Learning based Algos
(Scaled Resources)
Average Execution time(ms)
30
25
20
15
10
5
0
0
100
200
300
400
No. of resources
ACO
Gen. Algo


Graph 10: Comparing Round Robin and FCFS
ie. Static Algos
(Fixed Resources)
Average Execution time (ms)
1.2
1
0.8
0.6
0.4
0.2
0
0
100
200
300
400
No. of requests
Rnd Rbn
FCFS


Graph 11: Comparing ACO and Generic Algo
ie. Learning based Algos
(Fixed Resources)
Avg. execution time (ms)
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
100
200
300
400
No. of requests
ACO
Gen. Algo


Graph 12: Comparing ACO with Round Robin, FCFS, Generic Algorithm (Scaled resources)
Avg. Execution time(ms)
80
70
60
50
40
30
20
10
0
0
100
200
300
400
No. of requests
Rnd Rbn
ACO
FCFS
Gen. Algo


Graph 13: Comparing ACO with Round Robin, FCFS, Generic Algorithm (Fixed resources)
Avg. Execution time(ms)
1.2
1
0.8
0.6
0.4
0.2
0
0
100
200
300
400
No. of requests
Rnd Rbn
ACO
FCFS
Gen. Algo

Chapter-5: CONCLUSIONS

5.1 Conclusions

In this paper we have proposed the Ant colony optimisation algorithm so that tasks scheduling with load balancing can be achieved easily and effectively, and we have evaluated the Ant colony optimisation algorithm in applications with the number of tasks varying from 50 to 350. The experimental result shows that this algorithm balance the entire system load efficiently. If the sizes of the tasks are the same or not, this algorithm can handle all the conditions, and outperforms FCFS, round robin and simple genetic algorithm in cloud computing environment.

5.2 Future Scope

For the future work, there are two points that deserve investigation.

First, all the tasks are mutually independent.

Second, tasks are computationally intensive and it can' happen for real in cloud systems.

Moreover, to incorporate information about task requirements, availability vector should be extended

This methodology doesn't take into account "the fault tolerance" issue. Analysts can further work to involve the "fault tolerance" problems in their future researches.

REFERENCES

[1] Pragnesh K. Patel, “Review on Scheduling based Resource Allocation in Cloud computing”, International Journal for Scientific Research & Development (IJSRD) | Vol. 1, Issue 1, 2013 | ISSN: (online): 2321-0613

[2] Sivadon Chaisiri, Student Member, IEEE, Bu-Sung Lee, Member, IEEE, and Dusit Niyato, Member, IEEE, “Optimization of Resource Provisioning Cost in Cloud Computing”, IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. 5, NO. 2, APRIL-JUNE 2012

[3] Ratan Mishra and Anant Jaiswal, “Ant colony Optimization: A Solution of Load balancing in Cloud”, International Journal of Web & Semantic Technology (IJWesT) Vol.3, No.2, April 2012 DOI : 10.5121/ijwest.2012.320333

[4] Grids Riky Subrata, Member, IEEE, Albert Y. Zomaya, Fellow, IEEE, and Bjorn Landfeldt, Senior Member, IEEE, “Game-Theoretic Approach for Load Balancing in Computational Grids Riky Subrat”, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 19, NO. 1, JANUARY 2008

[5] Jinhua Hu, Jianhua Gu, Guofei Sun, Tianhai Zhao, “Programming A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment ”, 3rd International Symposium on Parallel Architectures, Algorithms and Programming.

[6] Hemant Kumar Mehta, Manohar Chandwani, Priyesh Kanungo, “On Trust Management and Reliability of Distributed Scheduling Algorithms ”, 978-1-61284-260-8/10/$26.00 © 2010 IEEE.

[7] Martin Randles, David Lamb, A. Taleb-Bendiab, “A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing ”, 2010 IEEE 24th International Conference on Advanced Information Networking and Apps Workshops.

[8] Kun Li, Gaochao Xu, Guangyu Zhao, Yushuang Dong, Dan Wang, “Cloud Task scheduling based on Load Balancing Ant Colony Optimization”, 2011 Sixth Annual China Grid Conference

Web References :

https://www.lifewire.com/what-is-cloud-computing-2378249

http://nptel.ac.in/courses/106105087/pdf/m02L03.pdf

http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/

https://en.wikipedia.org/wiki/FIFO_(computing_and_electronics)https://en.wikipedia.org/wiki/Round-robin_DNS
http://www.sciencedirect.com/science/article/pii/S2212017313005318