IJCAT International Journal of Computing and Technology, Volume 1, Issue 4, May 2014
ISSN : 2348 – 6090
www.IJCAT.org

Linux Based Cloud OS in Implementing SaaS, PaaS, IaaS for Web Service Provisioning

[1] **Anez Bin Ashraf**, [2] **R.S Shaji**

[1] Department of Information Technology, Noorul Islam University
Thuckalay, Tamil Nadu, India

[2] Department of Information Technology, Noorul Islam University
Thuckalay, Tamil Nadu, India

Abstract - A cloud operating system or cloud OS provides an internet based operating system(OS) that has instances and other provisional computational resources that are deployed in the resource provider's, that is the cloud OS provider server or data center. The cloud operating system provides the environment and feels like normal operating system which can be accessed by any methods. It just need an internet enabled browser, which is just enough to access the operating system located at the server using internet. The model implemented is an open source operating system based on Linux that can be accessed by a web browser. It is enabled by handling the operating system procedures as html or xml requests and responses. As the workload increases, the performance issue needs to be considered. Performance analysis needs to be associated in parallel along with implementing the components. SaaS services like File Splitter, Image Editor and Office Suite, PaaS services like compiler(UTF), Kernel for Scheduling and Kernel for Mutual Exclusion and IaaS services like File Storage, Shared Memory and Shared bandwidth are deployed.

Keywords **- Linux, Cloud, OS, SaaS, PaaS, IaaS, Web Service.**

1. Introduction

Like a server Operating System, a cloud OS is responsible for managing resources. In a server, the OS is responsible for managing the various hardware resources, such as CPU, memory, disks, network interfaces everything inside a servers chassis. It hides the hardware operation details and allows these scarce resources to be efficiently shared. A cloud OS serves the same purpose. Instead of managing a single machines resources, a cloud OS is responsible for managing the cloud infrastructure, hiding the cloud infrastructure details from the application programmers and coordinating the sharing of the limited resources. But unlike a traditional OS, a cloud OS has to do everything at scale.

Cloud Operating System is an open source web desktop following the cloud computing concept. It acts as a platform for web applications written using the cloud computing concepts. It includes a Desktop environment with number of applications and system utilities. It is accessible by portable devices via its mobile front end. The Cloud Operating System lets us upload les and work with them no matter where we are. It contains applications like Word Processor, Address Book, PDF reader, and many more developed by the Cloud vendor. It can be well developed mainly in PHP, XML, and JavaScript.

The OS built is a simple Linux based Operating System that runs just on a Web browser, providing complete access to a variety of web-based applications that allow the user to perform many simple tasks without booting a full-scale Operating System. Because of its simplicity and transparency, Cloud OS can boot in just a few seconds. The Operating System is designed for the systems that are mainly used to browse the Internet. From Cloud the user can quickly boot into the main OS, because Cloud OS continues booting the main OS in the background which can be seen from the browser.

1.1 Web Services

A Web service is a technique that helps two or more devices to communicate each other with network or internet as a medium. Web services can easily associate and readily build a service-oriented architecture. Web services make services in the form of software, infrastructure or platform be accessible over general network protocols irrespective of any system platforms or languages. These services can represent either with the new applications or execute around any sort of existing systems to make them accessible over the network.

For that all we need a web browser with internet connection as the pre requisite. There are two types of people or users depending on this web services. They are providers and consumers. The provider will create a web service and broadcasts the interface and access information to the registry of the web service.[1]

The consumers otherwise known as web service clients access the service from the web using their interface and work it from registry using various valid operations and then ask service provider to access one of its web services.

Whatever the requests made by the client is only in the form of http requests, since the interface they use probably the browser only. Hence after receiving it by the provider, they should adapt those requests in such a way that the original action or method asked by the client should be invoked.

1.2 Linux OS

OS is in otherwise known as an operating system. It is the software on a computer that enables applications and the computer operator to access the devices on the computer to perform desired functions. The operating system (OS) relays instructions from an application to, for instance, the computer's processor. The processor performs the instructed task, and then sends the results back to the application via the operating system. As an open operating system, Linux is developed collaboratively, which aims that not a single company is solely responsible for its development or ongoing support. Companies participating in the Linux economy share research and development costs with their partners and competitors. This spreading of development burden among individuals and companies has resulted in a large and efficient ecosystem and unheralded software innovation.

Kernal: All operating systems have kernels, built around the architectural metaphor that there must be a central set of instructions to direct device hardware, surrounded by various modular layers of functionality. The Linux kernel is unique and flexible because it is also modular in nature.[3] Modularity is desirable because it allows developers to shed parts of the kernel they don't need to use. Typically a smaller kernel is a faster kernel, because it isn't running processes it does not need.

1.3 SaaS

The SaaS otherwise known as software as a services model has no physical need for indirect distribution since it is not distributed physically and is deployed almost instantaneously. Earlier companies that provide SaaS built their own economic model without including partner remuneration in their pricing structure. When the idea is conceived, it was not easy for traditional software creators to enter into and SaaS model of their existing application.

The first reason is that SaaS model cannot assure income structure as that of the normal case. The second reason is that continuing to work with a network distribution application will certainly decrease their product margins and it will let competitors of their product to grow up in pricing.

What make the SaaS model to work and get managed is when then combine the indirect sales model with their own existing business model, and it is attracted to those clients who readily want to switch and sharpen themselves in the ever increasing economy of information technology.

1.4 PaaS

Platform as a service in brief known as PaaS is one type of category of cloud computing services that provides a computing platform and a clear stack that offers services. Along with all other service models like software as a service and infrastructure as a service, it is a service model of cloud computing.

In this model, unlike the SaaS, the client builds the software using tools and libraries provided by the PaaS facilitors. The client dominantly controls the entire software development from the starting to last and all sorts of configuration settings[2]. The vendor provides the networks, servers, storage, and other services except the factor that the client should deliver the request and response using high bandwidth.

PaaS offerings facilitate the working of applications without the complexity and cost of pricing the costly apps and managing the underlying hardware and software and provisioning hosting capabilities. There are various types of PaaS vendors and dealers, though all sort of application hosting and deployment environment is handled, along with many integration services.

PaaS offerings also include options for designing an application, developing an application, implementing and testing the applications. The other services include collaboration of services, integration of web service, marshaling, integration of database, security, scalability, storage, persistence, state management, version of application, instrumentation of application, and lastly facilitation of developer community.

1.5 IaaS

In this most basic cloud service model, cloud provides other computers as physical or more often as virtual machines, raw (block) storage, firewalls, load balancers and networks. IaaS providers supply these resources on demand from their large pools installed in data centers. Local Area Networks including IP addresses are part of it. For the wide area connectivity, the Internet can be used or in carrier clouds dedicated Virtual Private Networks or in brief VPN can be configured. To deploy their applications, cloud users then install operating system images on the machines as well as their application software. In this model, it is the cloud user who is responsible for patching and maintaining the operating systems and application software.

2. Background Works

A normal network consists of a processor and I/O nodes connected through a fabric made up of cascaded switches and routers. I/O units can range in complexity from a single device to a large, memory-rich redundant array of independent disks (RAID) subsystem.

The network can be subdivided into subnets interconnected by routers.
A subnet consists of end nodes and switches and forms an administrative domain for InfiniBand network management. All transmissions begin or end at a channel adapter (CA). Each processor contains a host channel adapter (HCA) and each peripheral has a target channel adapter (TCA). CAs appear as nodes to the fabric and provide one or more ports to which the data is sent[5].

A single process multiple data model can be pictorially represented as in figure 1.

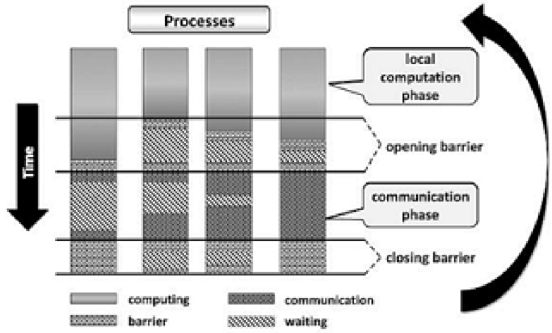

Fig. 1 Single process multiple data model.

A stochastic model is highly attractive and is one such alternative because relative cost of solving the model while concerned with covering large parameter space is much low. However, these types of stochastic analytic models are assumed not to scale well when dealing with the rising complexity associated with any type of cloud service architectures[6].

In IaaS Clouds, when a request is processed by from the client end, the particular service will be executed. For all the services that can be deployed by the system, there will be a predefined image. That particular image will help in execution of one or more virtual machine instances or a pre-deployed virtual machine. It can have the provision to be made available to the client or can be easily customized. [7]

Virtual machines are deployed on any of the physical machines associated with the cloud where each of those physical machined may contain or can be shared by multiple virtual machines. The virtual machines will request resources for CPU, RAM, and disk capacity. The creation, functioning and execution of such resources may find delays, when this type of process is executed. This delay can be easily reduced by various resource optimization and allocation techniques available.

Public cloud providers offer a high level of services at different performance levels that matches with different application requirements. The Amazon Web Services is one such instance currently constitutes twelve instance types with variable amount of memory, storage and networking capacity.[8] The performance of service processing is relatively given in Elastic Compute Units. Elastic compute units or ECU corresponds approximately to the equivalent CPU capacity of a 1.0–1.2 GHz 2007 Opteron or 2007 Xeon processor.

The resource provisioning in an operating system is depicted in fig 2.

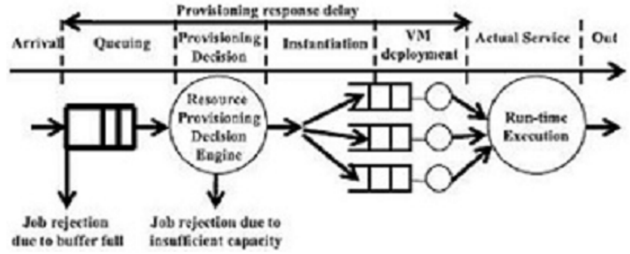

Fig. 2 Resource provisioning in cloud

The goal is to develop Unix and the Kernel-based Virtual Machine hypervisor in combination with the existing

InfiniBand driver stack to set up a virtualization fundamentals for any of the high performance cloud computing systems.[9] At present the focus is on to the InfiniBand technology, the concepts and ideas can be interchanged to other cluster interconnects or future works.

3. Modeling and Architecture

3.1 Linux System

The Linux system used for this workspace is Linux Mint, which is actually created out of Ubuntu. The more advantage of Linux Mint is that of its multimedia code support, both audio and video. The main advantage of Linux mint than Ubuntu is that of transparency.

The Linux Mint is carved out of Ubuntu with a better intention of ease of use it provides. The applications and repositories it uses are of highly helpful for any level of customers to operate without any complexity of Ubuntu.

Linux was originally developed as a free operating system for Intel x86-based personal computers. It has since been ported to more computer hardware platforms than any other operating system. It is a leading operating system on servers and other big iron systems such as mainframe computers and supercomputers.

As per a recent study, more than 95 percent of the world's 500 fastest supercomputers run some variant of Linux, including all the 44 fastest. Linux also runs on embedded systems, which are devices whose operating system is typically built into the firmware and is highly tailored to the system; this includes mobile phones, tablet computers, network routers, facility automation controls, televisions and video game consoles. Android which is a widely used operating system for mobile devices is built on top of the Linux kernel.

The development of Linux is one of the most prominent examples of free and open source software collaboration. The underlying source code may be used, modified, and distributed commercially or non-commercially by anyone under licenses such as the GNU General Public License. Typically, Linux is packaged in a format known as a Linux distribution for desktop and server use. Some popular mainstream Linux distributions include Debian, Ubuntu, Linux Mint, Fedora, Arch Linux, and the commercial Red Hat Enterprise Linux and SUSE Linux Enterprise Server. Linux distributions include the Linux kernel, supporting utilities and libraries and usually a large amount of application software to fulfill the distribution's intended use.

3.2 Cloud Operating System

The cloud systems are mainly available to achieve a certain goal, like wise for completing a task, for example document conversion from one format to another format like doc to pdf, etc. There are at present too many web sites available that clearly aimed to provide the service more specifically.

These sites are providing services named as SaaS or software as a Service. Moreover, there are web sites that provide PaaS or platform as a service. The sites those provide such services are like providing compiler to compile and test the codes.

The limitation with those sites or services is they are providing compiler in such a way that the user inputs and values should be given in the code itself. The dynamic way of handling the user inputs and providing the outputs are still a problem.

One of the main issues behind implementing such services is performance issue. When the load increases, it is quite difficult to handle the things and hence it clearly affects the performance of the system largely.

Another issue is that of the security threats. Many security threats can pose serious problem with the server system and hence there are chances with which the entire system may fail, Thereby losing the personal preferences and private data of the users.

Another concern about the cloud environment is handling of many users at the same time. There may be many people simultaneously accessing the same service with different settings and preferences. The instantiation of the service depending on the user is thus a tedious task.

Most important factor that should be taken in to consider is the virtualization of the services depending on the tasks and users. The virtualized environment should me framed out for the users which clearly provide an abstract environment.

If anything fails while in the course of doing the task itself, care should be taken. There are times in which the virtual machines may fail while a task is in progress. In that times, the effective methods by which the migration of one virtual machine is needed and that is highly significant.

Whenever a new user try to access the system, the resource allocation is another complex process. The system should calculate the resource utilization either using static methods or using dynamic approaches.

In certain conditions where only one service is providing, then just static means of resource allocation is needed. There are areas or sites where multiple services may be accessed by the users simultaneously. In those applications, static resource provision is not feasible. The resources may be allotted on a dynamic way that specifically aims at handling the demand of users on a timely basis.

The system works in such a way that can handle all the above policies very clearly. The SaaS services work on top of IaaS and PaaS. The services File split, image edit and office app can very easily accessed and highly user friendly. The cloud operating system needs certain pre-requisites before it can be initiated. One of it is MySQL database server package. It needs to be installed and configured into the operating system. MySQL is the prime choice of database for many applications.

The next thing needed is WAMP web server to host certain files needed for the client system to work smoothly from a web browser. The procedural diagrams of different applications are as follows.

In this system, mainly 5 applications of SaaS, 3 of each PaaS and IaaS are being implemented. The office suite, image editor and file splitter are the 3 applications from the SaaS. File storage, shared memory and the shared bandwidth are the 3 applications proposed to be implemented from IaaS point of view. Moreover, compiler (supports UTF), Kernel for scheduling and kernel for mutual exclusion are the services intended to be executed in the proposed cloud operating system as an application to PaaS.

The overall interaction diagram is depicted in figure.3

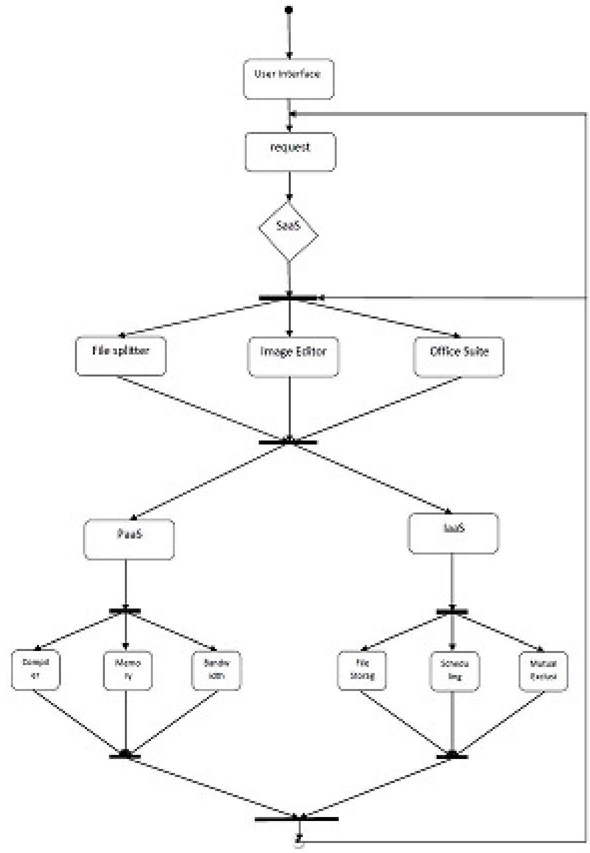

Fig. 3 Overall interaction diagram

The office word contains general purpose word editor that mostly handles with the features that can be done by M.S Word or openoffice suite or Libre suite. The office suite contains the daemon of open office suite.

Procedural diagram for presentation app is as in figure 4.

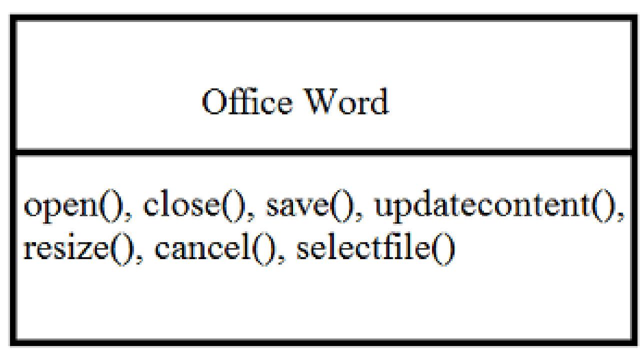

Fig. 4 Procedural diagram of word application.

The presentation suite is a replica of show of the openoffice org. The presentation helps to create slides, view presentation in a new pop up window. The spreadsheet, presentation and word comes together with the package under office suite.

It helps for a hassle free preparation of any presentation in all concerns. The presentation can be exported or imported from the local system by the help of easy daemons. The daemon used here is that of openoffice created by openoffice.org.

The managing user as a part of mutual exclusion is shown in fig.5.

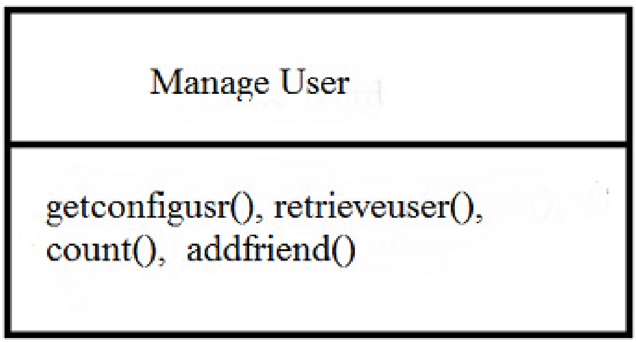

Fig. 5 Procedural diagram for mutual exclusion

This is an app that falls under the IaaS or infrastructure as a service. The kernel for mutual exclusion is highly needed for the effective and smoother functioning of the operating system. The managing user depicts as such. The administrator can easily create, delete, and manage any number of users. But a user can manage only the user itself. This is being achieved by the mutual exclusion.

4. Implementation

The system is purely a Linux based operating system that can work from a browser. For that the operating system be installed in a web server. The user accounts can be created by the administrator. Every successful log in takes the individual to their own desktop. They can access the system as if they were doing it from the normal systems.

They can have their own files and customization as they can have in the normal operating systems. The operating system mainly concentrates on certain services like SaaS, PaaS and IaaS. The services related with image handling, office purposes including spreadsheet, writer, presentation is also included.

From the infrastructure service, file system, Kernel for mutual exclusion and sharing of memory is implemented. The file system is actually a virtual file system. Virtual file system is created from the physical system. As and whenever needed, depending on the number of users and the number of processes, it is being created from the physical system.

Any number of users can work, where each user will be allotted with their resources. Depending on the users and requests arrived, app handles and access the resource. When each app starts working, it will be instantiated depending on the need. Hence every user will get an image of the app working inside it. That image will work for the particular user.

Almost all mutual exclusion algorithms employ a static approach to invoke mutual exclusion. These algorithms always take the same course of actions to invoke mutual exclusion that doesn't matter what is the state of the system. The implemented system works in a dynamic way by deploying shared memory, shared bandwidth and mutual exclusion by using effective hyper visor agents.

The system takes a small time for the initial load. After that it takes the user credentials to check the authentication process. Rather than the traditional OS, that needs a lot of infrastructure, the cloud OS doesn't need that at all. Only a system is needed that have a browser with internet connection only. The cloud OS is incorporated with PaaS services like Unicode Support, Scheduling of the process and handling of mutual exclusion as well. Moreover the IaaS services like File Storage, Memory Sharing and bandwidth Sharing are also well implemented.

The cloud OS is incorporated with PaaS services like Unicode Support, Scheduling of the process and handling of mutual exclusion as well. Moreover the IaaS services like File Storage, Memory Sharing and Bandwidth Sharing are also considered. The system works in such a way that can handle all the above policies very clearly. The SaaS services work on top of IaaS and PaaS. The services File split, image edit and office app can very easily accessed and highly user friendly.

4.1 SaaS Services

From the SaaS services, typically 5 applications had been created. Those applications are Word, Presentation, Image Editor, Chess Game, Public Chat.

The Word and Presentation applications work with the support of daemons provided by the open office suite. Hence, when an access is required the daemon will communicate with the open office org system, thereby executing the system.

The image editor can do with the basic operations as well it provide the image viewing also. Image editor application included in the proposed system supports opening and saving of all major picture file formats, including GIF, JPEG, PNG, TIFF and BMP. Full support is provided in handling transparency for GIF, PNG and BMP files. It also has its own proprietary file format if needed, which preserves all layer information and can be used as a way to provide end users with editable templates instead of forcing them to create images from scratch as well.

The chess game is deployed to play the chess game against the system itself. The added feature of chess game is that of as follows.

- All the movements will be listed out. The system's movement will be listed soon after that of movement of the client is displayed.

- Another feature is that undo operation is clearly specified and can be executed by the player. When an undo operation is called, then the last two movements will be back. Specifically the movement of the system and the client. The latter operation precedes that of former operation.

Public chat is like a board system, where any of the registered users can make their voice heard by others.by default all the users will be a part of the system and anyone one is free to share their opinion and that works as a discussion forum.

4.2 PaaS Services

PaaS offers and facilitate the deployment of applications without the cost and complexity of buying and managing any of the underlying hardware and software and hence provisioning hosting capabilities. There are various types of PaaS providers. However, all over application hosting and a deployment environment, along with various integrated services. Services offer varying levels of scalability and maintenance.

Unicode Support in the system- The Unicode support is made available for the system so that any one from any region can access it.

Filenames: Filenames specified on the command line or in compiler directives (such as include) may contain Unicode characters.

Source code files: Unicode characters are now supported in identifiers, macros, string and character literals, and in comments. Universal character names are also now supported. Unicode can be input into a source code file in the following encodings: UTF-16 little endian, UTF-16 big endian, UTF-8 with BOM.

Output: During compilation, compiler outputs diagnostics to the console in UTF-16. The characters that can be displayed at the users console depending on the console window properties. Compiler output redirected to a file is in the current ANSI console code page.

Memory Sharing: Shared memory is memory that may be simultaneously accessed by multiple programs with intent to provide communication among them or avoid redundant copies. Shared memory is an efficient means of passing data between programs. Depending on context, programs may run on a single processor or on multiple separate processors.

Sharing of memory is done a dynamic way with which every user could easily get the resources whenever they want. There are times a user needs the memory for a single application to work. In those times, memory will be allotted as per his needs. There are other situations where a user work with multiple applications and undergo multiple tasks. In those times, the system will provide the memory request raised by that application for that user and the sharing will be done accordingly.

4.2 IaaS Services

File Storage:File storage is based on highly virtualized infrastructure and has the same characteristics as cloud computing in terms of agility, scalability, elasticity and multi-tenancy, and is available both premises and on-premises.

The physical memory is divided into many number of virtual memory units known as virtual file systems. This virtual file system can be easily handled and every user can easily access and use it.

Kernel for Scheduling: Job scheduling is one of the major activities performed in almost all of the computing environments. Cloud computing is one the upcoming latest technology that has a drastic growth. Thus to efficiently increase the working of cloud operating system environments, job scheduling is one the tasks performed in order to gain largely in all the aspect.

The goal of scheduling algorithms in the proposed cloud operating systems is spreading the load on processors and maximizing their utilization while minimizing the total task execution time. Job scheduling is one of the most famous optimization problems that clearly plays a key role to improve flexible and reliable systems. The main purpose is to schedule the jobs to the adaptable resources in accordance with adaptable and permitted time that involves finding out a proper sequence in which jobs can be executed under any sort of transaction logic and other constraints.

There are various types of scheduling algorithm exist in the distributed computing system. Many of them can be applied in the cloud environment with suitable verifications. The main advantage of job scheduling algorithm is to achieve a high performance computing and the best system throughput. Traditional job scheduling algorithms are not able to provide scheduling in the cloud environments.

Kernel for Mutual Exclusion: Kernel for Mutual Exclusion is planned to be included in the proposed operating system caters the needs that is to achieve the effective control of the various processes under all the constraints. A system managing the users is created as a part of the mutual exclusion. Only the administrator could access it. Others could not access it.

4. Conclusions

The operating system is based on Linux O.S, which is free and open source. Linux is the most famous open source operating system which is available. The software as a service, platform as a service and infrastructure as a service are being implemented in the cloud operating system. 3 services from each of the SaaS, PaaS and IaaS are planned to be added to the system.

Other than this certain features have been added to improve the functionality of the entire cloud operating system. The services are mostly targeted for business related purposes.

Moreover to improve the overall performance of the cloud operating service, effective resource provisioning and load balancing algorithms have been formed. Resource provisioning has been done in both static and dynamic approach that is chosen based on the purpose. Load balancing has been done on a dynamic way. Migration algorithm has been used on an effective way such that the load balancing satisfies the processes, physical machines and virtual machines to a complete extent.

References

[1] S. H. J. H. E. J. C. L. I. P. A. W. C. Clark, K. Fraser, migration of virtual machines, 2012.
[2] A. G. Delvar and Y. Aryan, synthetic heuristic algorithm for independent task scheduling in cloud systems, November 2011.
[3] H. P. K. Xiong, performance and analysis in cloud computing, November 2013.
[4] Z. Xiao, W. Song, and Q. Chen, resource allocation using virtual machines for cloud computing environment technique, October 2012.
[5] D. E. A. Iosup, N. Yigitbasi, the performance variability of production cloud services, October 2012.
[6] L. Quan, W. Yuan-Zhou, and Z. Yong-hui, graph representations and a converging algorithm of the resource virtualization for cloud computing, January 2013.
[7] C. H. C. Evangelinos, computing for parallel scientific hpc applications: feasibility of running coupled atmosphere ocean climate models, January 2013.
[8] D. M. A. Gupta, of hpc applications on cloud, 2012.
[9] V. K. N. K. S. T. Rahul Ghosha, Francesco Longo, and performance analysis of large scale iaas clouds, 2012.

Anez Bin Ashraf received the bachelor's degree in Information Technology from the University of Calicut, Kerala in 2012. Presently he is pursuing his M.Tech in Software Engineering from from Noorul Islam University, Tamil Nadu. His research interests include cloud operating systems, failure handling in cloud, etc.

R.S Shaji received Master's degree in Computer Science and Engineering from Pondichery University in 2002. Currently working as Professor in Information Technology Department, Noorul Islam University, Tamil Nadu.