DESIGN AND IMPLEMENTATION OF INFRASTRUCTURE PRIVATE CLOUD COMPUTING MODEL.

Victor Aguirre1, Joseph Espinoza2, Yessenia Sigcha.3, Erika Solano 4

[1] Cloud project, Technical University Northern Av. July 17 Ibarra, Imbabura
[2] Cloud project, Technical University Northern Av. July 17 Ibarra, Imbabura
[3]Cloud project, Technical University Northern Av. July 17 Ibarra, Imbabura 4Proyecto cloud, Technical University Northern Av. July 17 Ibarra, Imbabura

vhaguirrec@utn.edu.ec, jsespinozab @ utn.edu.ec, yfsigchap@utn.edu.ec, elsolanot@utn.edu.ec

Summary.*The article presents the process undertaken for the implementation of a project Cloud Computing, managers using open as OpenStack, Eucalyptus and OpenNebula, which allow private cloud models deploy code. In the project implementation process it is based Cloud based on physical characteristics such as network architecture, the respective logical infrastructure of each manager and the dimensioning of each of the platforms to deliver infrastructure as a service to different users. Finally disclosed the results obtained through different tests to students of the Faculty of Engineering of Applied Science at the Technical University of the North.*

Keywords
Cloud Computing, Software as a service, virtualization.

Abstract. *The article presents the process Realized*to Implement a cloud computing project, as OpenStack open using managers, Eucalyptus and OpenNebula, Which allow private Cloud models deploy code. For the implementation of Cloud Computing core project steps is described, as are the physical architecture of the network and Its logical architecture of each respective manager, like how each manager is dimensioned to Provide infrastructure as a service to different users. Disclosed Finally the results of the various tests on the three platforms to the students of the Faculty of Engineering of Applied Science at the Technical University of the North.

Keywords
Cloud computing, Software as a service, virtualization.

1. Introduction

At the Technical University of North project cloud-oriented model private cloud to provide infrastructure as a service to students and teachers is carried out, this initiative came from researchers Engineers Engineering Faculty of Applied Science and PhD. José Luis García Dorado.

Progress and development has formed a perspective based on a need by users to manipulate information from any application based on the concept of cloud computing from the context of the IT location.

It was decided to perform as designing a solution Computing Cloud-oriented infrastructure using free platforms OpenStack, Eucalyptus and OpenNebula. Which provide the service of virtual machines for students of Engineering in Electronics and Communication Networks.

In this article the design and implementation of each of the free software platforms as well as performance testing that was performed to determine the performance one decade will be detailed

2. Design and Implementation

Computing model called cloud computing that allows for the scalability of any virtualized resource within the network as a service.

2.1 Platforms

OpenStack

OpenStack is a virtualization management software open source, which allows users to connect different technologies and components from different vendors and expose a unified API, regardless of the underlying technology. OpenStack, you can manage different types of hypervisors, network devices and services, storage components, using a single API that creates a tissue unified data center, (ORACLE, 2014).

Figure 1. Logo OpenStack
Reference: http://docs.openstack.org/

OpenStack is the collaboration of developers and technologists producing cloud computing platform open standard for both public and private cloud. [1] (OpenStack, 2015).

Key features

This infrastructure manager in the cloud as it is OpenStack has several features that are listed below:

- Installation is somewhat complicated, but also provides a very robust compared to other operators and thus optimal and efficient cloud platform is obtained.
- CompatibleEC2 OpenStack provides an API, which lets you interact with Amazon EC2 API; This is suitable for multi-cloud EC2 environments is the common API.
- It has its own storage system called Swift, which is designed to provide fault tolerance and scalability.
- It offers flexible models to suit the needs of different applications or network user groups. Standard models include flat networks or VLAN to separate servers and traffic.
- Through its management system called Keystone identity performs user authentication and service.

Eucalyptus

Eucalyptus is a software architecture based on Linux that implements scalable private and hybrid clouds within the characteristics of infrastructure.

The platform allows the use of resources (hardware, storage and network) using a self-service interface depending on the needs that are required within certain activities. [1]

Figure 2. Logo Official Eucalyptus
Reference: Enterprise Edition 2.0

Eucalyptus is a computing platform compatible with EC2 cloud and cloud storage
S3 compatible. [2]

Key features

- Eucalyptus provides API support Amazon EC2, S3, IAM, ELB, Auto Scaling and CloudWatch services. It is offering you the ability of a hybrid cloud.
- Control IP addresses allows mapping IPs to VMs dynamically, elastic and controlled manner. [3]
- Is implemented as Infrastructure from
  Service (IaaS), where users provide processing power, storage and
  computational resources. [4]
- It has high flexibility.
- In the implementation allows scalability in performance.
- The virtual machine instances running through a software called hypervisor or VMM (Virtual Machine Monitor).

OpenNebula

OpenNebula is a cloud computing platform under free and open-source software for managing distributed infrastructure of a data center capable of managing virtual servers for the deployment of private, public and hybrid infrastructure as a service implementations.

Figure 3. Logo OpenNebula Reference:
https://www.udsenterprise.com/es/blog/2016/02/22/udsenterprise-soporta-opennebula/

Key features

- OpenNebula is an open-source software to build any type of Cloud.
- It is designed to achieve integration with any network and storage, in order to adapt to existing data centers.
- OpenNebula has the ability to manage storage, networking and virtualization different technologies available today.
- It provides the ability to deploy distributed infrastructure services, combining data center resources as well as remote Clouds, according to deployment policies.

2.2 Physical Infrastructure

As an initial part of the implementation of the different cloud platforms physical infrastructure owned by the Technical University Northern determining the different elements that intervene in the internal and external access to different platforms to be installed it is determined, as shown in Figure 4 each server which implement the platforms is installed in the data center of Engineering Faculty of Applied Science (FICA) which are connected by the distribution switch Data center to the central building which provides output to internet and connection to other subnets University

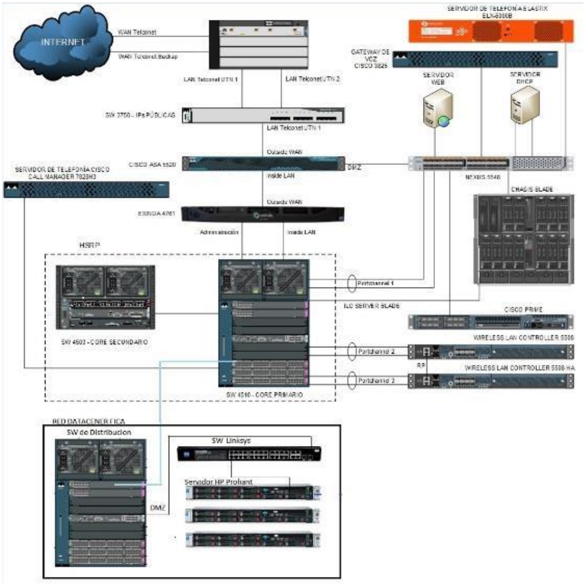


Figure 4. Physical infrastructure and their respective components

2.3 Logical Architecture

Openstack

In Figure 3 the systems and components that account Openstack which meet such different functionalities as presented: storage volumes, the execution of virtual machines; these elements communicate with each other to provide Infrastructure as a Service

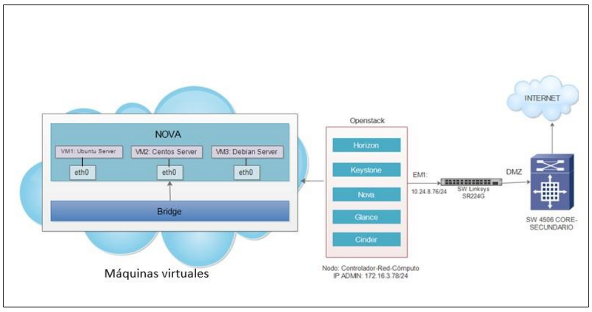


Figure 5. Logical architecture OpenStack Source: own research.

Node: Architecture in which infrastructure as a service was implemented, consists of a single node ("All in One"), in which services such as OpenStack platform will stay: Horizon, Keystone, Nova, Glance and Cinder; turn driving all virtual machines. This node will function Controller, and Network Computing.

Node Controller: Build the Keystone (ID), Glance (OS images) and Horizon (Dashboard), Nova components responsible for the management part of computing services, plus associated services APIs and MySQL database.

Compute node: Nova component is installed, in charge of running virtual machines, the service also runs two hypervisors KVM or Quemu.

Network Node: Connect all network elements, here several agents such as plug-in-agent which handles the routing run, to configure DHCP addresses[1] And also provides internet service.

Virtual Machines: Have a virtual network interface (eth0), the same that is connected to the physical host interface host the em1 through a second logical interface called bridge (Bridge), the communication is managed by the Nova component.

[1] DHCP: The Dynamic Host Configuration Protocol host

Eucalyptus

Eucalyptus is developed in a modular fashion, leading to the five main components that are exposed to high level of functionality within the architecture of Eucalyptus, as shown in Figure 6:

- Cloud controller (CLC)
- Cluster Controller (CC)
- Node Controller (NC)
- Storage Controller (SC)
- walrus

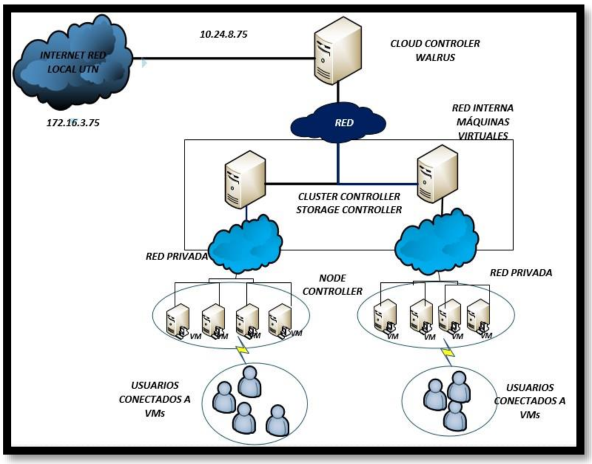


Figure 6. Internal Architecture of Eucalyptus and their respective components
Reference: (HELION, 2013)

- **Cloud Controller (CLC)**

Cloud controller meets its performance in carrying out infrastructure monitoring instances and cloud resources.

- Monitor the availability of resources in various components of the cloud infrastructure.
- To monitor instances that are running and cloud resources.

- **Cluster Controller (CC)**

Cluster Controller (CC) collects status information of the nodes on the platform and performs instances running on them to the CLC.

- **Node Controller (NC)**

The functionality of Node Controller (NC) is based on the performance of the machines hosting instances. The NC controls the activities of the VM, ie its life cycle as:

- ✓ Execution
- ✓ Inspection
- ✓ Termination

- **Storage Controller (SC)**

The Storage Controller infrastructure to manage instances in Eucalyptus and access to:

- ✓ Storage
- ✓ volumes

- **Walrus**

It is providing priority support for S3 interface and compatibility EBS and stores images of the instances.

OpenNebula

The following image functional blocks representing the elements corresponding to the physical structure of our platform where blocks specified are described:

- Storage
- Running virtual machines
- Network Infrastructure
- frontend

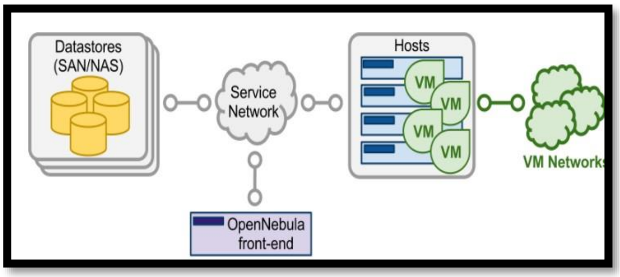


Figure 7. Functional Architecture Reference OpenNebula: http://docs.opennebula.org/4.14/_images/one_high.png

- **The Front-end:**It is the server where all management services, provisioning, monitoring, access, and management platform running. It is the central part of the system.
- **Hosts**: They are servers running the virtualization where instantiate and run the virtual machine (VM) provided by the platform.
- **Datastores:**are the servers where the images corresponding to the instances and virtual machines are stored. Datastores are capable of storing several disc formats and configuration files, kernels, storage volumes, etc.

- **Service Network:** It is part of the network service that allows the interconnection of the Datastores Hosts and Frontend, is also communication to our private network or institutional network and provides us with an internet connection.
- **Virtual Networks:** It is the virtual network infrastructure that allows network services support between virtual machines (VM).

Dimensioning

The sizing process, to analyze the resources has to establish characteristics of equipment capacity and test based on the implementation of the matters oriented applications instances.

The implementation is performed from the present physical resource, for each of the platforms can be seen in Table 1.

NAME SERVER		HP DL360 G9	HP DL360 G9	HP DL360 G9
characteristics	RAM	32 GB	32 GB	32 GB
	PROCESSOR	1x Intel Xeon E52620 V3	1x Intel Xeon E52620 V3	1x Intel Xeon E52620 V3
	HDD	3x450 GB	3x450 GB	3x450 GB
	Trajeta NETWORK	4 X 1 GbE	4 X 1 GbE	4 X 1 GbE
	SIST. OPERATIVE	Ubuntu Server 14.04 LTS	CentOS 6.5	Ubuntu Server 14.04 LTS
FUNCTION		OpenStack Cloud	Eucalyptus Cloud	Cloud Open Nebula
IMAGE				

Table 1. Characteristics of physical servers Source: Own.

The platforms enable compatibility with the following operating systems as shown in Table 2, such as:

SOFTWARE	LINUX	WINDOWS
Platform **openstack** **Eucalyptus** **opennebula**	If there is support: -Opensuse -CentOS -Cirros -Ubuntu -Debian -Fedora	If there is support: -Windows 7 -Windows 8 8 -windows server 32 bits -Windows Xp

Table 2. Platform support with operating systems Source: Own.

2.4 OpenStack

Sizing for the implementation of OpenStack platform is based on requirements of different operating systems be installed for use in applications requiring students of the Faculty of Applied Science Engineering University; in Table 3 shows these requirements.

System operating	Memory RAM	HDD Minimum	HDD Maximum	Processor
Ubuntu	128 MB	500 MB	1 GB	x86 - 2.4 GHz Server
Centos	512 MB	1 GB	2 GB	x86 - 1 GHz **Server** x64 - 2GHz
Debian	1 GB	1 GB	2 GB	x86 - 2GHz **Server**

Table 3. Requirements Operating Systems implemented in OpenStack **Source:** own calculations.

Once the requirements needed to install different operating systems known, we proceed to verify the main features that the physical server where the OpenStack platform will be implemented, as detailed in Table 1.

In order to gauge the exact number of virtual machines to deploy OpenStack platform, you need to know how many CPU has the physical server; whereby the "lscpu" command to view the details of the processor, as shown in Figure 8.

```
Arquitectura:            x86_64
CPU op-mode(s):          32-bit, 64-bit
Orden de bytes:          Little Endian
CPU(s):                  16
On-line CPU(s) list:     0-15
Hilo(s) por nÃºcleo:     2
NÃºcleo(s) por zÃ³calo:8
Socket(s):               1
Nodo(s) NUMA:            1
ID del vendedor:         GenuineIntel
```

Figure 8. Information on the number of cores and socktes on the physical server.

Source: Terminal Server Ubuntu Server 14.04 LTS

#CPUs: Number of CPUs having the physical server that will host the OpenStack platform.

Vel. VM: Processor speed and virtual machine GHz.

Vel Processor: Processor speed in GHz physical server.

According to equation 1, and data obtained in Figure 4, it provides that the maximum number of virtual machines to run simultaneously is 16.

$$\# VM = \frac{16\ x\ 2.4\ GHz}{2.4\ GHz} = \mathbf{16VM}$$

According to Figure 8, it can be seen that the server where the OpenStack platform will be staying has a socket, 8-core and 16 CPUs to be used for the implementation of different virtual machines.

Number of virtual machines

In order to calculate the number of virtual machines that can be created and run simultaneously, the formula of "OpenStack Operations Guide" (Tom et al, 2014) is used.

$$\# VM = \frac{\# CPUs\ x\ Vel\ de\ VM}{Vel\ Procesador} \quad (1)$$

Where: #VM: Number of VMs to work simultaneously.

According to the detailed values in Table 3 and Equation 2 is determined, the dimensioning of the number of virtual machines that can be created in related to the consumption of RAM for each operating system.

TOTAL RAM	RAM SO	#VM	OS
32 GB	1 GB	32	Ubuntu-Server
	1 GB	32	CentOS Server
	1 GB	32	Debian-Server

Table 4. Number of VMs according to RAM memory
Source: Authors.

RAM

This calculation does according to the amount of RAM required for each operating system to be installed on the virtual host and works properly.

$$\# VM = \frac{RAM\_T}{RAM\ \_SO} \quad (2)$$

Where: #VM: number of virtual machines that can be created according to the amount of RAM.

RAM_T: total amount of RAM physical server

RAM _SO: RAM required by the operating system of the virtual machine.

The results in Table 4 indicate that can create up to 32 virtual machines in relation to consumption of RAM that each operating system.

According to the platform OpenStack, the dimensioning of the virtual machines is performed in relation to the flavor that provides when filling the template instantiation, whose value is 2 GB of RAM. Now with this value it proceeds to size the number of VMs that can be created, using Equation 2.

TOTAL RAM	RAM FLAVOR	#VM	OS
32 GB	2 GB	16	Ubuntu-Server
	2 GB	16	CentOS Server
	2 GB	16	Debian-Server

Table 5. Number VM according to the flavors OpenStack
Source: own calculations.

The results in Table 5 indicate that each virtual machine is assigned a RAM of 2 GB, while the hardware has 32 GB total, therefore you can be dimensioned about 16 bodies can be of any operating system because each uses the same resource.

Calculation of Processors

This calculation is done to determine the number of virtual CPUs needed for each operating system to implement virtual machines.

$$\# VCPU = \frac{Vel\_Procesador}{Vel\_VM} \quad (3)$$

Where: #VCPU: Number of CPUs per VM.

Vel_Procesador: Frequency in GHz of the physical server processor

Vel _VM: GHz frequency that will work with the operating system on the virtual machine.

With the data in Table 3 and executing equation 3, it is able to determine the number of CPUs required for each operating system for normal operation.

Processor speed	Speed VM	# VCPUs	OS
2.4 GHz	2.4 GHz 2.0 GHz	1	Ubuntu-Server
		1.2	CentOS Server
	2.0 GHz	1.2	Debian-Server

Table 6. Number of VCPUs by VM
Source: self made

In Table 6, it can be seen that at least one VCPUs is needed for every virtual machine regardless of operating system running on it.

Once you have calculated the number of VCPUs is needed for each operating system, it proceeds to perform the dimensioning of many virtual machines can be created.

$$\# VM = \frac{CPUs}{VCPU} \quad (4)$$

Where: #VM: number of virtual machines.

#VCPU: virtual CPU allocated for each OS

#CPU: CPUs with which account the physical server.

By the results in Table 6 and equation 4, we proceed to calculate the number of virtual machines can be created depending on VCPUs.

#CPUs	#VCPUs	#VM	OS
16	1	16	Ubuntu-Server
	1	16	CentOS Server
	1	16	Debian-Server

Table 7. Number of VM based on the number of VCPUs
Source: Author.

The results in Table 7, indicate that you can create a total of 16 virtual machines, either the operating system you need to view each virtual CPUs per instance consumes.

Here in Table 8 can be seen a little short, where the number of machines that can be created according to different points previously detailed.

TOTAL RAM	TOTAL CPUs	#VM by Formula	#VM by RAM	#VM by VCPUs
32 GB	16	16	16	16

Table 8. Summary of the number of VM to create OpenStack
Source: Author.

In Table 8, it is regarded that the maximum number of virtual machine creation is 16 in relation to the use of formula OpenStack, RAM consumption and the number of VCPUs which is employed in each instance whatever the Linux operating system it runs.

Eucalyptus 2.5

During the sizing process it is required to perform measurements of the current resource and the resource consumed by the platform.

Bandwidth analysis through Exinda

Controls and manages Web access by user or host, point-to-point user groups, servers or individuals traffic.

Through monitoring through Exinda server that owns the Technical University Northern contributed results based on the parameters measured in real time, set forth in Table 9.

Traffic Analysis	Transfer rate (Kbps)	Percentage
Server Execution	1,634	7%
Running 5instances	10,549	43%

Running 10 instances	23,969	95%

Table 9. List of data Exinda Source: Authors.

Analysis RAM

The analysis parameter RAM is display a result server support based on the number of applications and virtual machines running on the platform called Eucalyptus cloud.

Based on the analysis performed under the RAM consumption server and based on the platform, the result is shown in Table 10:

Number of Instances Windows7	RAM consumption	Total value
Initial	9206	31985
5	14626	31985
10	31675	31985

Table 10. Result RAM consumption

The results are shown in Figure 10, where a statistical table was established based on the results.

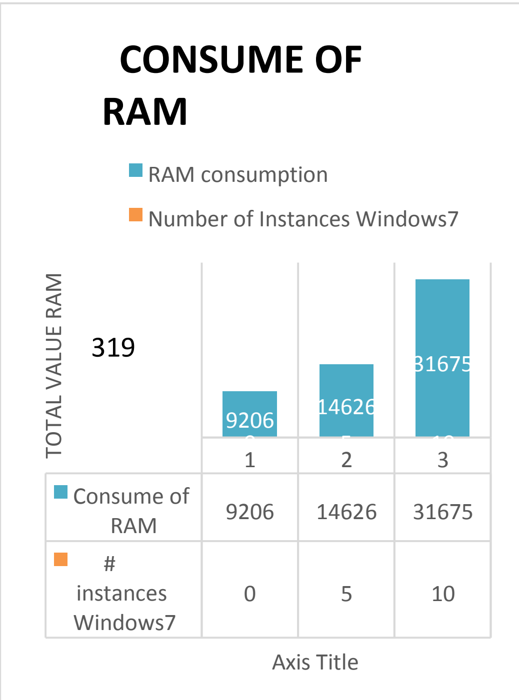


Figure 9. Statistical diagram RAM consumption Reference: Own elaboration based on analysis

Note: The expressed as 0, value determines the consumption based operating system based on the implementation of the platform.

Results of RAM consumption in the instances.

The results observed in Table 11 was obtained from the analysis based on the consumption of RAM, when subjected platform during the execution of a number of instances.

Note: GNS3 is a high-performance tool that requires further action.

H.H

instances Windows7	Consumption RAM	Memory RAM without use	Value Total
running SS.OO	1,1GB	0,9GB	2GB
GNS3 and Packet Tracer	1,87GB	0,13GB	2GB

Table 11. Result RAM consumption instance

RAM consumption based on the peak value

RAM applications = 1,87GB

= 2GB RAM 1vm

10vm = 2GB RAM * 10 = 20 GB

Total = 20 GB RAM

Instances are created under the action of 2GB of RAM, data presented in Table 11, so a diagram is set based on the resource consumption of using a statistical diagram as shown in Figure 11.

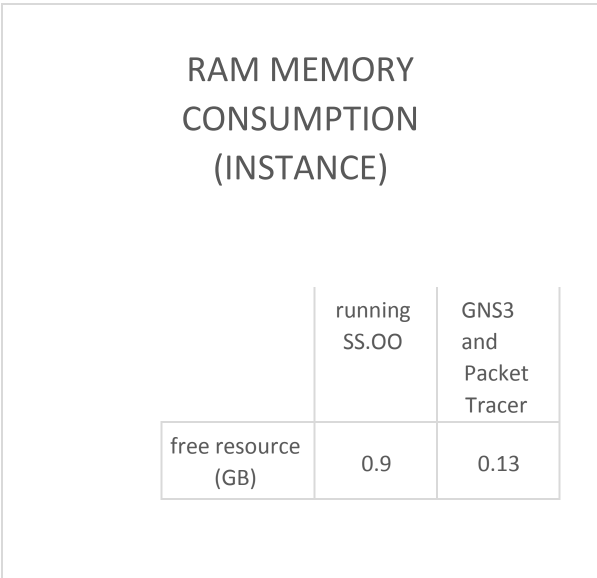


Figure 10. Consumption RAM instances Reference: Prepared based analysis.

CPU consumption

CPU consumption is an analysis system using tools for determining the CPU usage and utilization.

On average, the top provides a real-time system then proceeds to obtain the data during the execution of the platform.

Load average: 0.14; 0.18; 0.18

CPU consumption: + 2.3% US 0.6% + 0.0% wa s = 2.9%

Determines 2.9% of CPU usage on the server, where the platform is executed.

5instances execution, used in applications designed for networking.

Load average: 1.20; 1.38; 1.04

CPU consumption: 25.2% + 6.3% us sy + 5.8% wa = 37.3%

37.3% determines the CPU consumption on the server, where the platform is executed.

10 instances run also found using applications designed for networking.

Load average: 10.06; 6.16; 5.08

CPU consumption: 54.6% + 7.8% us sy + 0.4% wa = 62.8%

62.8% determines the CPU consumption on the server, where the platform is executed.

Analysis of the results of the CPU Consumption

Depending on the results obtained through monitoring and according to established parameters determines the CPU consumption as shown in Table 12, is connected based on running applications and functions of the instances, so it is proportional to its use.

Number of instances	CPU consumption (%)	Unused CPU consumption (%)
Value initial	2.9	97.1
5	37.8	62.2
10	62.8	37.2

Table 12. CPU consumption analysis according to the number of instances.
Reference: Prepared based analysis.

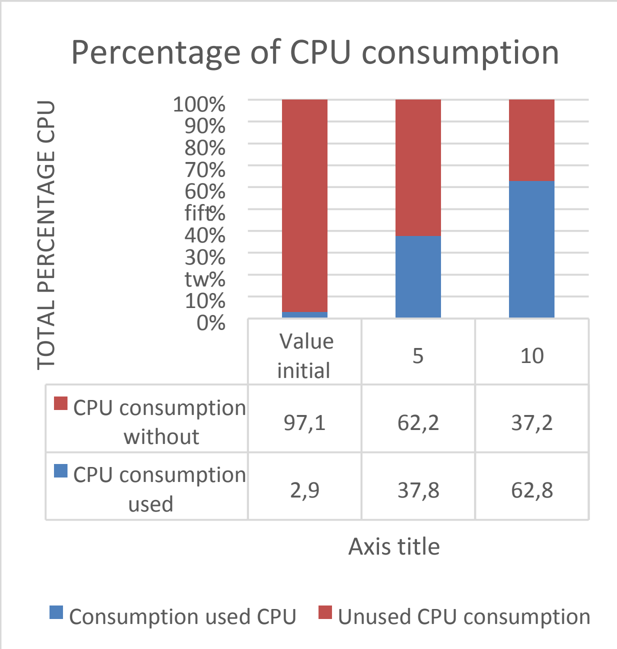


Figure 11. Percentage of CPU consumption Reference: Eucalyptus SA Platform.

The maximum CPU consumption value achieved was 62.8% on 100% provided. This value varies according to usage instances by the student or user, who is playing some kind of application, as shown in Figure 12.

CPU consumption in instances

Parameter analysis measuring CPU consumption based on the resource instance and during application execution is carried out.

Instances based on CPU usage, vary according to applications or activities to which the instances are subjected, therefore, results set based on the consumption of CPU instances, as shown in Table 13.

instances Windows7	CPU consumption (%)
running SS.OO	0-5
GNS3 and Packet Tracer	52-60

Table 13. Consumption of CPU on the instance

Funcionality test

Instances created based on the Eucalyptus platform, a key part of the study of cloud computing that will be used for different functions. Applicability aspects such as offering certain number of instances targeted to a specific function within the Faculty of Applied Science Engineering at the Technical University of the North.

In the case of the Eucalyptus platform is oriented to providing services based on the use of networking tools, providing the academic aspect of solving activities based on student satisfaction and teaching.

2.6 OpenNebula

Dimensioning of RAM and CPU

Table 14 specified requirements for each operating system analyzed, plus server features and accordingly it was possible to get the maximum number of virtual machines that can be deployed on the platform was established for this number is divided core server for the necessary number of cores of each system similarly done with RAM and the smallest value of the two results is the maximum of virtual machines that can be deployed in each operating system.

OS	requirements	
	Processing (GHz)	Memory (GB)
Ubuntu	1	1
Centos	2	1
LinuxMint	1	1
OpenSuse	1	512MB
Fedora	2	2
Windows Xp	0.6	512MB
Windows 7	1.2	2
Windows server	2.4	2

Table 14. Requirements operating systems
Reference: Own

$$N°\ virtuales\ maquinas = \frac{Vel\ CPU * N°\ de\ nucleos\ del\ servidor}{Vel\ CPU\ Operating\ System} \quad (1)$$

$$N°\ virtuales\ maquinas = \frac{Memory\ Server\ RAM}{RAM\ required} \quad (2)$$

According to Equations 1 and 2 can determine the maximum number of virtual machines that can be deployed according to the requirements of each operating system and server resources you have.

Table 14 shows the resources that owns the server, in this case is taken into account the basic resources for the implementation of the platform OpenNebula so for sizing cores 15 2.4 GHz and 30 GB is take detailed RAM, thereby determining each of the different operating systems the maximum number of virtual machines can run simultaneously.

OS	Current resources		MV possible
	nuclei CPU (2.4GHz)	Memory (GB)	
Ubuntu	fifteen	30	30
Centos	fifteen	30	18
LinuxMint	fifteen	30	30
OpenSuse	fifteen	30	30
Fedora	fifteen	30	fifteen
Windows Xp	fifteen	30	60
Windows 7	fifteen	30	fifteen
Windows Server	fifteen	30	fifteen

Table 15. Number of possible virtual machines Reference Prepared

Bandwidth Dimensioning

traffic analysis input and output is performed to determine the total bandwidth (AB) that requires a user to access the platform and the VM and from this determined according to the number of users as is the AB you must be assigned to subnets within the Technical University Northern AB and the minimum that should be the internet connection.

The monitoring is done by Exinda equipment is installed on the UTN, Figure 13 shows that consumption or AB input to the OpenNebula server is 13 kbps and Figure 14 shows that traffic output is 11kbps for a user so an analysis is that the number of students in a classroom that on average is 40 and it is determined that the minimum AB is 991 kbps to ensure access to the platform OpenNebula.

Inbound Conversations				
External IP (User)	Internal IP (User)	Application	Transfer Rate (kbps)	Packet Rate (pps)
Total			13.642	25
FICA-WIRELESS: Auto Catch-all				
172.16.3.74	172.17.42.12	WebSocket[cloudfica.utn.edu.ec:29876]	7.682	14
172.16.3.74	172.17.42.12	Yahoo	5.960	10

Figure 12. Traffic entering the server OpenNebulaFuente: Own

Outbound Conversations				
External IP (User)	Internal IP (User)	Application	Transfer Rate (kbps)	Packet Rate (pps)
Total			11.143	16
FICA-WIRELESS: Auto Catch-all				
172.16.3.74	172.17.42.12	WebSocket[cloudfica.utn.edu.ec:29876]	7.385	10
172.16.3.74	172.17.42.12	Yahoo	3.758	5

Figure 13. Exit Traffic Server OpenNebulaFuente: Own

3. Results

3.1 Openstack

For performance testing platform OpenStack measuring RAM and CPU consumption was performed on the physical server during the simultaneous execution of 5, 10 and 15 virtual machines with Ubuntu Server Operating System.

In Table 16, the result of the total RAM consumption of virtual machines and assigned shown each according to the parameters of each operating systems, it can be concluded that according to the resources allocated to each instance RAM consumption and processing is low, because it is not working with applications requiring greater amount of RAM on each virtual machine.

System	Number of operating virtual	consumption memory allocated	Memory machines RAM	RAM
			By VM	Total
Ubuntu Server	5	2 GB	1.4 GB	7 GB
	10	2GB	1.7 GB	17 GB
	fifteen	2GB	1.7 GB	26 GB

Table 16. consumption RAM 5, 10 and 15 VM Source: Prepared

Table 17 shows the total consumption of processing on the host host used by the virtual machines and assigned to each virtual host according to the parameters established for each operating systems, it can be concluded that according to the resources allocated each instance VCPUs consumption is low, because it is not working with applications that require more processing on each virtual machine.

System vo operati	Number of virtual machines	VCPUs assigns two	consumption ·om VCPUs by VM	Total
Ubuntu Server	5	1 VCPUs	1 VCPUs	5 VCPUs
	10	1 VCPUs	1 VCPUs	10 VCPUs
	fifteen	1 VCPUs	1 VCPUs	15 VCPUs

Table 17. Consumption VCPU in 5, 10 and 15 VM
Source: Prepared

Eucalyptus 3.2

Analysis of results

Analysis result sizing parameters, as can be seen in **Table 18**.

settings	Consume of Equipment (Server)	value Implemented by machine
Bandwidth	24kbps	50 Kbps
RAM	1,89GB	2GB
CPU consumption	62.8%	80%
Statistics Disco	18% to 90%	-------

Table 18. result requirement machines
Source: Prepared

Based on test results it was obtained under the parameters of dimensioning, where it determines that as Eucalyptus private cloud platform, allowed to run applications based on the subject of networking as GNS3, Packet Tracer.

In addition to specifying resources that meet the requirements instances, by appropriate tests to improve execution capabilities, the hardware when subjected gives us the ability to create 8 machines based on windows7 with the characteristics shown in table 17, which meet the parameters allow execution of applications targeted at networking, based on resources.

Settings	SS.OO	# instances
Number of machines	Windows	10
	Linux	4

3.3 OpenNebula

Performance testing for the execution of 5, 10, 15 virtual machines both Linux and Windows operating system determining the CPU and memory consumption is performed when the machines are at rest and when they are being used.

Linux system

In Figure 14 shows the consumption of a Linux which rest gender minimal consumption of 1% of the server, running operating system consumption rose but equally did not exceed the allocated resources remaining in normal operation.

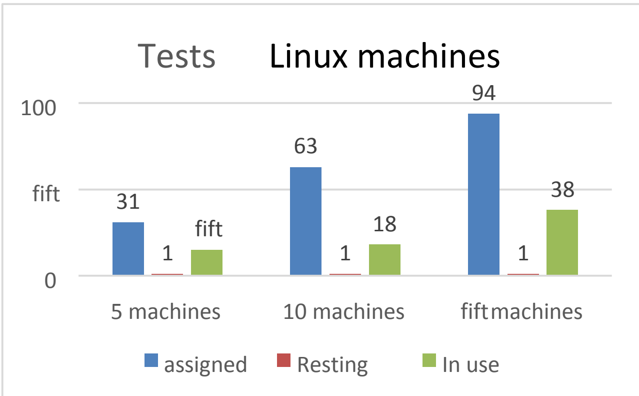


Figure 14. CPU consumption on Linux
Source: self made

In Figure 15 the RAM consumption server being each virtual machine idle and execution is displayed and the correct operation of each system was determined as the consumption of RAM it has not exceeded the allocated resources being maintained at a normal level .

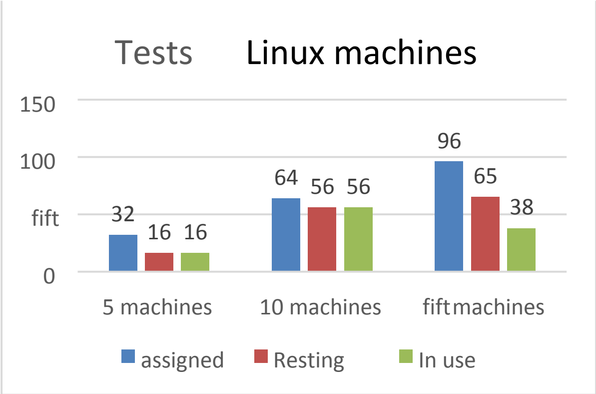


Figure 15. Consumption RAM Systems Linux
Reference: Prepared

Windows Operating System

In Figure 16 one can observe the consumption of 5, 10, 15 virtual machines with Windows system running simultaneously, wherein it is noted that while in idle CPU usage is minimal while when already in use its consumption increases but not exceeding the values assigned maintained in proper operation.

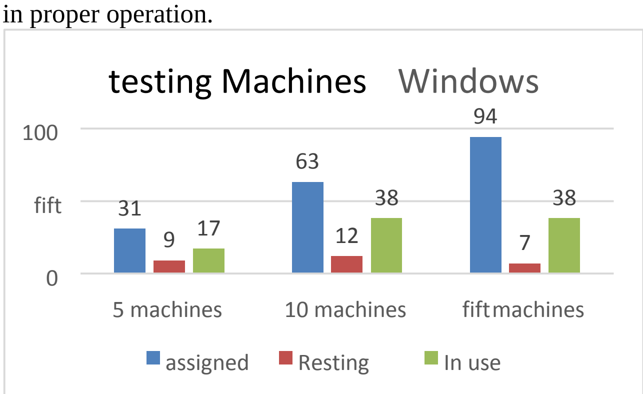
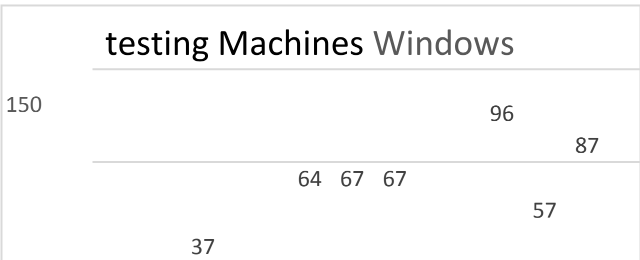


Figure 16. CPU consumption on Windows
Source: self made

In Figure 17 the RAM consumption will have the server to run 5, 10, 15 virtual machines with Windows system where one can see that even at rest the system consumes a large amount of RAM in some cases shown surpassing the resources allocated,



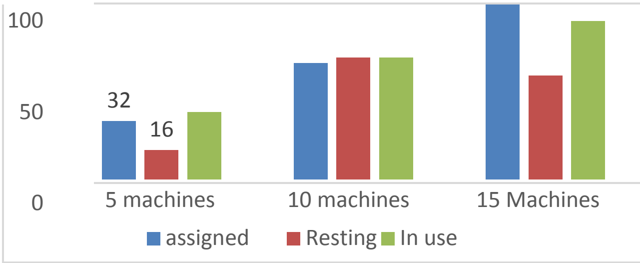


Figure 17. Consumption RAM Systems Windows operating
Reference: Prepared

Four. conclusions

- Infrastructures private cloud under the platform OpenStack, Eucalyptus and openNebula within the Faculty of Engineering of Applied Science at the Technical University North was designed to provide services to students and teachers in the use of virtual hardware designed for different activities of different materials, within the framework of cloud computing.

- The dimensioning was made based on the requirements of computing resources used by each operating system, working with free software their demands are lower and therefore resource consumption is lower, unlike Windows which it requires greater availability of resources for implementation.

- The implementation of this project cloud computing, can provide a new service to the university community, emphasizing to students and teachers, that allows for academic activities from anywhere with internet access to satisfy different needs depending on the resources obtained under the dimensioning.

- Performance tests determining the performance of virtual machines when they are at rest and when in use and concluded that each system is run in a proper way without overloading the server where the platform is housed was performed.

- The OpenNebula platform provides a monitoring system that is built into view administrator web interface with which the administrator may have information about the performance of CPU resources and memory allocated to virtual machines and actual values they are consuming dimensioning helping the administrator of the platform and to better manage it.

Thanks

a thank you is expressed Engineers: Carlos Vásquez and Edgar Maya for their cooperation in the development of the project in a guide provided on the basis of their knowledge and time in this process, besides being the guide for the implementation of the project.

Gratitude extends to the Faculty of Engineering of Applied Science belonging to the Technical University of the North, entities that allowed the development, as part of research projects at the University that contributes to the formation of students.

Bibliographic references

[1] Systems.INC, E. (2010). Cloud Computing Platform, *Administrator's Guide.*United States: Enterprise Edition 2.0. Retrieved on November 7, 2015, Cloud Computing Platform, Administrator's Guide: EucalyptusEE2.0.AdminGuide.1.Master.pdf

[2] Raju, M. (2010, 06). CSS Corp Open Source Services. Retrieved on December 1, 2015, CSS Corp Open Services Source: http://www.csscorp.com/enterprise-it-support/opensource-services.php.

[3] HELION, H. (2013, 09). Eucalyptus Documentation. Retrieved on November 6, 2015, of Eucalyptus Documentation: http://docs.hpcloud.com/eucalyptus/4.1.2/#installguide/euca_components.html

[4] Rao, M. (2015). Cloud Computing. Brooklyn: Editorial Asoke K. Ghosh.

About the Authors

Victor H. C. AGUIRRE

He was born in Ibarra Imbabura province on June 23, 1989. He completed his primary studies in the Esc. Model "President Velasco Ibarra". Highschool were at school, "Teodoro Gomez de la Torre", where he earned a Bachelor of Science Specialization mathematical physicist. He is currently conducting the process of engineering degree in Electronics and Communication Networks at the Technical University of the North.

Yessenia F. P. Sigcha

Born in Quito province of Pichincha on 06 February 1992. He completed his primary studies in the Esc. New Spring and ended in the Esc. Albuja Municipal Alfredo Galindo. Highschool were in the "Ibarra", which ended in 2010 National College, earned the Bachelor of Mathematical Sciences Physical Specialization. Currently, he is conducting its process engineering degree in Electronics and Communication Networks, Northern Technical University - Ecuador.

Erika L. SOLANO T

Born in La Merced in Buenos Aires province of Imbabura on 30 July 1991. He completed his primary studies in the Esc. "Buenos Aires". Highschool were in the "Ibarra", which ended in 2009 National College, earned the Bachelor of Mathematical Sciences Physical Specialization. Currently, he is conducting its process engineering degree in Electronics and Communication Networks, Technical University of the North.

Joseph S. ESPINOZA B.

He was born in Ibarra Imbabura province on 17 August 1992. He did his secondary education at the educational unit Teodoro Gómez de la Torre - Ecuador specializing in mathematical physicist. He attended college at the Engineering in Electronics and Communication Networks at the Technical University of the North. He is currently a volunteer member of the IEEE Student Branch Technical College North where plays the role of Membership Coordinator