



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**PROCESO PARA LA IMPLEMENTACION DE INFRAESTRUCTURA DE
“CLOUD COMPUTING” Y SUS VENTAJAS SOBRE LA
INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES**

Luis Fernando Juárez Avila
Asesorado por el Ing. Mario Enrique Sosa

Guatemala, agosto de 2011

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**PROCESO PARA LA IMPLEMENTACION DE INFRAESTRUCTURA DE
“CLOUD COMPUTING” Y SUS VENTAJAS SOBRE LA
INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

LUIS FERNANDO JUAREZ AVILA

ASESORADO POR EL ING. MARIO ENRIQUE SOSA

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, AGOSTO DE 2011

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympto Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Ing. Miguel Angel Dávila Calderón
VOCAL IV	Br. Juan Carlos Molina Jiménez
VOCAL V	Br. Mario Maldonado Muralles
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Pedro Pablo Hernández Ramírez
EXAMINADOR	Ing. Edgar Estuardo Santos
EXAMINADOR	Ing. Oscar Alejandro Paz
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

PROCESO PARA LA IMPLEMENTACION DE INFRAESTRUCTURA DE “CLOUD COMPUTING” Y SUS VENTAJAS SOBRE LA INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha julio de 2010.



Luis Fernando Juárez Avila.

Guatemala 28 de Marzo de 2011.

Ing. Carlos Azurdia
Carrera ingeniería en ciencias y sistemas
Facultad de ingeniería
Universidad de San Carlos de Guatemala

En atención a la designación de que fuera objeto para asesorar el trabajo de tesis con título **“PROCESO PARA LA IMPLEMENTACION DE INFRAESTRUCTURA DE “CLOUD COMPUTING” Y SUS VENTAJAS SOBRE LA INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES”** del estudiante Luis Fernando Juárez Avila con número de carné 200511691, tengo el agrado de comunicar que procedí a efectuar dicha asesoría, encontrándose el trabajo concluido satisfactoriamente mediante la culminación de cada una de las actividades planificadas.

Sin otro particular me suscribo de usted,

Atentamente,



Ing. Mario Enrique Sosa

Ing. Mario Enrique SOSA CASTILLO
Colegiado No. 2421



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 27 de Abril de 2011

Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **LUIS FERNANDO JUÀREZ AVILA**, carné **2005-11691**, titulado: **"PROCESO PARA LA IMPLEMENTACION DE INFRAESTRUCTURA DE CLOUD COMPUTING Y SUS VENTAJAS SOBRE LA INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES"**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado **“PROCESO PARA LA IMPLEMENTACIÓN DE INFRAESTRUCTURA DE “CLOUD COMPUTING” Y SUS VENTAJAS SOBRE LA INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES.”**, presentado por el estudiante LUIS FERNANDO JUÁREZ AVILA, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Ing. Marlon Antonio Pérez Turk

Director, Escuela de Ingeniería Ciencias y Sistemas

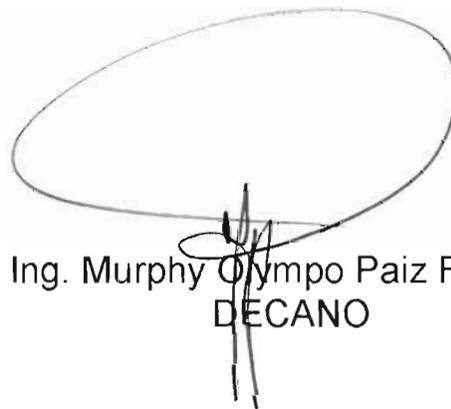


Guatemala, 08 de agosto 2011



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **PROCESO PARA LA IMPLEMENTACIÓN DE INFRAESTRUCTURA DE “CLOUD COMPUTING” Y SUS VENTAJAS SOBRE LA INFRAESTRUCTURA DE SERVIDORES DEDICADOS TRADICIONALES**, presentado por el estudiante universitario **Luis Fernando Juárez Avila**, procede a la autorización para la impresión del mismo.

IMPRÍMASE.



Ing. Murphy Olympo Paiz Recinos
DECANO



Guatemala, agosto de 2011

AGRADECIMIENTOS A:

Dios	Por darme la vida y las bendiciones para alcanzar esta meta profesional.
Mis padres	Por formarme como persona y por su apoyo para alcanzar esta meta, la cual constituye la herencia más valiosa que he podido recibir.
Mi esposa e hijo	Por su amor, son la inspiración para seguir adelante.
Ing. Mario Sosa	Por su ayuda en la asesoría, revisión y corrección de este trabajo.
La Facultad de Ingeniería de Universidad de San Carlos de Guatemala	Por albergarme en mi formación profesional.
Todos mis familiares y amigos	Que de una u otra manera estuvieron pendientes a lo largo de este proceso, brindando su apoyo incondicional.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	I
GLOSARIO	III
RESUMEN.....	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN	XV
1. <i>CLOUD COMPUTING</i>	1
1.1. Características.....	2
1.1.1. Infraestructura dinámica.....	3
1.1.2. Se consume o provee como servicio	3
1.1.3. Reducción de costos en infraestructura	4
1.1.4. Auto administración de la plataforma	4
1.1.5. Se paga por lo que se consume	5
1.1.6. Independencia de la ubicación.....	5
1.1.7. Fácil mantenimiento	5
1.1.8. Fiabilidad.....	6
1.1.9. Seguridad.....	6
1.1.10. Medible	6
1.1.11. Recuperación y tolerancia a fallos	7
1.1.12. Virtualizado	7
1.1.13. Multipropósito.....	8
1.2. Evolución	8
1.2.1. <i>Mainframes</i>	9
1.2.2. Cliente – servidor	10
1.2.3. <i>Service Oriented Architecture (SOA)</i>	11

1.2.4.	<i>Cloud computing</i>	11
1.3.	Elementos del <i>cloud computing</i>	11
1.3.1.	Infraestructura	12
1.3.2.	<i>Software</i> para administrar la nube.....	12
1.3.3.	Máquinas virtuales.....	13
1.4.	Niveles del <i>cloud computing</i>	14
1.4.1.	<i>Infrastructure as a service</i> (IAAS).....	15
1.4.2.	<i>Platform as a service</i> (PAAS)	16
1.4.3.	<i>Software as a service</i> (SAAS)	17
1.5.	Modelos utilizados en el <i>cloud computing</i>	18
1.5.1.	Modelo de adquisición – basado en servicio	18
1.5.2.	Modelo de negocio – basado en uso	19
1.5.3.	Modelo de acceso – <i>Internet</i> o <i>intranet</i>	20
1.5.4.	Modelo técnico – dinámico y flexible	20
2.	INFRAESTRUCTURA DEL <i>CLOUD COMPUTING</i>	21
2.1.	Servidores	21
2.1.1.	Procesadores	22
2.1.2.	Memoria RAM.....	23
2.1.3.	Espacio en disco duro	24
2.1.4.	Interfaz de red	24
2.2.	Sistema operativo.....	25
2.3.	Computación distribuida	26
2.3.1.	Computación distribuida	27
2.3.2.	<i>Clustering</i>	27
2.3.3.	<i>Grid</i>	27
2.4.	<i>Data Center</i>	28
2.5.	Virtualización	30
2.6.	<i>Middleware</i> o <i>software</i> para <i>cloud computing</i>	33

2.6.1.	<i>Node Controller</i> (Controlador de nodos)	34
2.6.2.	<i>Cluster Controller</i> (Controlador de <i>cluster</i>).....	34
2.6.3.	<i>Walrus Storage Controller</i> (Controlador de almacenamiento de máquinas virtuales).....	35
2.6.4.	<i>Storage Controller</i> (Controlador de almacenamiento).....	35
2.6.5.	<i>Cloud Controller</i> (Controlador de la nube)	36
3.	TIPOS DE NUBES Y SUS COSTOS.....	39
3.1.	Nubes públicas	39
3.1.1.	Principales proveedores.....	40
3.1.2.	Acuerdo de nivel de servicio	41
3.1.3.	Costos de nubes públicas	42
3.1.4.	Ventajas	45
3.1.5.	Desventajas	48
3.2.	Nubes privadas	49
3.2.1.	Proveedores de tecnología para nubes privadas.....	50
3.2.2.	Costos de nubes privadas.....	51
3.2.3.	Ventajas	53
3.2.4.	Desventajas	54
3.3.	Nubes híbridas.....	54
4.	PROCESO PARA LA CREACIÓN DE INFRAESTRUCTURA <i>CLOUD COMPUTING</i> PRIVADA.....	57
4.1.	Metodología	57
4.2.	Capacitación	58
4.3.	Análisis	58
4.4.	Diseño.....	60
4.5.	Instalación y configuración.....	61
4.5.1.	Instalando los controladores de la nube.....	62

4.5.2.	Instalando los nodos.....	68
4.6.	Administración.....	70
4.7.	Creación de imágenes.....	75
5.	VENTAJAS Y DESVENTAJAS DEL <i>CLOUD COMPUTING</i> ANTE INFRAESTRUCTURA DEDICADA TRADICIONAL	81
5.1.	Infraestructura tradicional de servidores dedicados	81
5.2.	El Cloud Computing sobre la infraestructura tradicional.....	82
5.2.1.	Desarrollo	83
5.2.2.	Pruebas	84
5.2.3.	Instalación en el ambiente de producción.....	85
5.2.4.	Reversión	85
5.2.5.	Temporadas altas para el negocio.....	86
5.2.6.	Adquisición de nuevo <i>hardware</i>	87
5.2.7.	Averías en <i>Hardware</i>	88
5.2.8.	Un nuevo <i>software</i> o producto.....	89
5.2.9.	La caída de un servicio.....	90
5.3.	Ventajas sobre la infraestructura tradicional.....	90
5.4.	Desventajas sobre la infraestructura tradicional	91
5.5.	Cuándo utilizar cloud computing público	92
5.6.	Cuándo utilizar cloud computing híbrido.....	93
5.7.	Cuándo utilizar cloud computing privado	94
5.8.	Cuándo utilizar servidores dedicados tradicionales.....	94
	CONCLUSIONES.....	95
	RECOMENDACIONES	97
	BIBLIOGRAFÍA.....	99

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Evolución de servicios y tecnología	9
2.	Niveles del <i>cloud computing</i>	15
3.	Virtualización de <i>hardware</i>	31
4.	Virtualización completa	31
5.	Paravirtualización.....	32
6.	Virtualización con KVM	33
7.	Esquema de los componentes de UEC	37
8.	<i>Cloud computing</i> híbrido	55
9.	Elementos de Ubuntu Enterprise Cloud	60
10.	Diseño para una nube privada con 3 equipos utilizando UEC	61
11.	Pantalla de inicio del instalador de UEC	63
12.	Pantalla de selección de lenguaje para UEC	63
13.	Pantalla de configuración de red para un equipo en UEC	64
14.	Pantalla para seleccionar el controlador de la nube en UEC	65
15.	Pantalla para seleccionar los componentes de UEC a instalar.....	65
16.	Pantalla para el particionado de los discos	66
17.	Pantalla para asignarle un nombre a un <i>cluster</i> en UEC	67
18.	Pantalla para definir un rango de direcciones para las instancias virtuales	67
19.	Página de inicio de sesión para el sitio de administración de UEC.....	71
20.	Página de la tienda de imágenes de Canonical en el administrador de UEC.....	72
21.	Ciclo de vida de una instancia virtual.....	73

22.	Pantalla para agregar el <i>cluster</i> a HybridFox.....	74
23.	Pantalla para definir credenciales y llaves en HybridFox.....	74
24.	HybridFox en la administración de imágenes para UEC	75

TABLAS

I.	Precios mensuales por recursos Amazon Web Services	44
II.	Guía para los costos en un centro de datos	52
III.	Ventajas de <i>cloud computing</i> privado sobre público.....	53
IV.	Equipos disponibles para crear una plataforma <i>cloud computing</i>	59

GLOSARIO

AoE	Protocolo de red desarrollado por la compañía Brantley Coile y diseñado para acceder a dispositivos de almacenamiento masivo mediante redes <i>Ethernet</i> .
API	Siglas del inglés <i>Application Programming Interface</i> , que se refiere a una interfaz de comunicación entre componentes de <i>software</i>
Aplicación	Programa de cómputo diseñado como herramienta y preparado para una utilización específica.
Cache	Es un conjunto de datos copiados de otros originales, con la propiedad de que estos datos son de fácil acceso, a diferencia de los datos originales que son costosos de acceder en cuanto a tiempo.
Cluster	Conjuntos de computadoras interconectadas entre sí, que se comportan como si fuesen una única computadora con más poder de procesamiento.
CRM	Siglas del inglés <i>Customer Relationship Management</i> que se refiere a un tipo de <i>software</i> de apoyo a la gestión de las relaciones con los clientes, ventas y mercadeo.

Curl	Herramienta de <i>software</i> que automatiza las transferencias de archivos y operaciones no supervisadas.
Data Center	Un lugar especial donde se encuentra el <i>hardware</i> , se supervisa, y se asegura su buen funcionamiento.
ERP	Siglas del inglés <i>Enterprise Resource Planning</i> que se refiere a sistemas de información, que integran y manejan muchos de los negocios asociados con las operaciones de producción y de los aspectos de distribución de una compañía comprometida en la producción de bienes o servicios.
Ethernet	Es un estándar de redes de área local para computadores con acceso al medio por contienda.
Firewall	Es un <i>software</i> o dispositivo, parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas. Configurados para permitir, limitar, cifrar y descifrar el tráfico entre los diferentes ámbitos sobre la base de un conjunto de normas y otros criterios.
Front-end	Es la parte del <i>software</i> que interactúa con el o los usuarios.
HTTP	Protocolo de transferencia de hipertexto es el protocolo utilizado para el intercambio de información en la web.

HTTPS	Es la versión segura del HTTP, toda la información viaja cifrada y de este modo se consigue que la información no sea interceptada por un tercero.
IAAS	Siglas de <i>Infrastructure As A Service</i> , nombre que se le da a uno de los niveles de <i>cloud computing</i> en el cual se ofrece infraestructura computacional y de comunicación como un servicio.
Infraestructura	Conjunto de elementos o servicios que se consideran necesarios para la creación y funcionamiento de una organización cualquiera.
Interfaz	Conexión física y funcional entre dos aparatos o sistemas independientes.
Intranet	Es una red de computadoras privadas conectadas para compartir dentro de una organización parte de su información y sistemas operacionales.
ISP	Del inglés <i>Internet Service Provider</i> que se refiere a los proveedores del servicio de <i>Internet</i> .
Kernel	Software que permite el manejo del hardware por el sistema operativo y sus aplicaciones.
KVM	Modulo del <i>kernel</i> Linux que permite crear máquinas virtuales sobre una máquina física.

LAN	Siglas del inglés <i>Local Area Network</i> que se refiere a una red de computadoras, cuya extensión física está limitada a un edificio o a una distancia de 1 km como máximo utilizando repetidores.
Linux	Es el núcleo de un sistema operativo libre, iniciado por Linus Torvalds y que está desarrollado por colaboradores de todo el mundo.
Middleware	Es un <i>software</i> de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.
Multi hilo	Capacidad que tienen los procesadores para ejecutar eficientemente múltiples hilos de ejecución, es decir manejar varios programas o procesos en paralelo.
Navegador	Es un programa que permite ver la información que contiene una página web.
Nodo	Es un punto de intersección o unión de varios elementos que confluyen en el mismo lugar. Por ejemplo: en una red de ordenadores cada una de las máquinas es un nodo, y si la red es <i>Internet</i> , cada servidor constituye también un nodo.
Núcleo	En un procesador es la parte que en realidad hace la lectura y la ejecución de las instrucciones.

PAAS	Siglas de <i>Platform As A Service</i> , nombre que se le da a uno de los niveles de <i>cloud computing</i> en el cual se ofrece ambientes de desarrollo como un servicio.
Paravirtualización	Modo de virtualización en la que los sistemas operativos de las máquinas virtuales están levemente modificados para que sepan que están virtualizados y de esta manera interactúen con el hipervisor de una manera óptima.
Plataforma	Es un sistema que sirve como base para hacer funcionar determinados módulos de <i>hardware</i> o de <i>software</i> con los que es compatible.
Proxy	Es un programa o dispositivo que realiza una acción en representación de otro, su finalidad más habitual es permitir el acceso a <i>Internet</i> a todos los equipos de una red cuando sólo se dispone de un equipo conectado.
Rack	Es un bastidor destinado a alojar equipamiento electrónico, informático y de comunicaciones. Las medidas para la anchura están normalizadas para que sea compatible con equipamiento de cualquier fabricante, siendo la medida más normalizada el de 19".
RAM	Siglas del inglés <i>Random Acces Memory</i> , es la memoria desde donde el procesador recibe las instrucciones y guarda los resultados, es de acceso aleatorio pues se puede acceder a cualquier byte de memoria sin acceder a los bytes precedentes.

REST	Cualquier interfaz web simple que utiliza XML y HTTP.
SAAS	Siglas de <i>Software As A Service</i> , nombre que se le da a uno de los niveles de <i>cloud computing</i> en el cual se ofrece software como un servicio.
SCSI	Es una interfaz estándar para la transferencia de datos entre distintos dispositivos del bus de la computadora.
Servidor	Es una computadora que formando parte de una red, provee servicios a otras computadoras denominadas clientes.
SOA	Siglas del inglés <i>Service Oriented Architecture</i> es un concepto de arquitectura de <i>software</i> que define la utilización de servicios para dar soporte a los requisitos del negocio.
SOAP	Protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.
Terminal	Es un dispositivo electrónico o electromecánico de <i>hardware</i> , usado para introducir o mostrar datos de una computadora o de un sistema de computación.

- UEC** Siglas de Ubuntu Enterprise Cloud, nombre que se le da a un conjunto de *software* que permite crear ambientes *cloud computing* de carácter privado.
- WAN** Siglas del inglés *Wide Area Network* que se refiere a una red de computadoras que se extienden sobre una área geográfica extensa, a veces un país o un continente, y su función fundamental está orientada a la interconexión de redes o equipos terminales que se encuentran ubicados a grandes distancias entre sí.
- XEN** Es un monitor de máquina virtual de código abierto, desarrollado por la Universidad de Cambridge y que soporta la ejecución de varios sistemas virtualizados sobre uno físico.

RESUMEN

Cada cierto tiempo se desarrollan nuevas tecnologías que vienen a mejorar cualquier aspecto de la vida de las personas. En el área de sistemas informáticos también se generan nuevas ideas para mejorar los procesos, sistemas y plataformas. El problema es que muchas veces por falta de interés o falta de información, éstas nuevas tecnologías se pasan por alto, dejando a un lado todas las ventajas que éstas pueden dar.

El *cloud computing* es sin duda la nueva tendencia para el manejo de recursos y servicios informáticos. Ha nacido como una mejora a lo hasta ahora existente, por tal razón muchas empresas han ido migrando a esta nueva forma de almacenar y administrar sus sistemas informáticos. Todos podemos sacar provecho de ello, sin embargo por la poca información y el poco interés de algunos responsables de tecnología, no se está usando de forma masiva, perdiendo así todas las ventajas que el *cloud computing* nos puede dar.

Este trabajo trata de dar a conocer qué es *cloud computing*, su evolución, características, cómo se puede implementar y las ventajas que nos puede ofrecer. De esta manera un encargado de tecnología de cualquier empresa, podrá evaluar si realmente le conviene utilizar dicha tecnología en función de los beneficios que esta le va a traer.

En el primer capítulo se define que es *cloud computing*, sus características, evolución, elementos, niveles y todo lo concerniente con el concepto.

En los siguientes dos capítulos se explica qué es lo que se necesita para construir o utilizar cada uno de los tipos de nubes, los equipos, los costos, las ventajas, desventajas y los proveedores de cada uno de los tipos de *cloud computing* . De esta manera se puede dar la idea de qué es lo que el negocio necesita y si está preparado para implementarlo.

En la última parte se describe el proceso para crear una plataforma *cloud computing*, dentro de la empresa haciendo uso de Ubuntu Enterprise Cloud y se hace un análisis de las ventajas que podemos obtener de cualquier tipo de *cloud computing* sobre la infraestructura de servidores tradicionales.

OBJETIVOS

General

Mostrar el proceso de la implementación de *cloud computing* y definir ventajas y desventajas de su utilización, ante las infraestructuras tradicionales de servidores individuales.

Específicos

1. Explicar en qué consiste el *cloud computing* y señalar las tecnologías que intervienen en él.
2. Estudiar los niveles, modelos y evolución del *cloud computing*
3. Explicar la diferencia entre *cloud computing* público y *cloud computing* privado.
4. Mostrar el proceso y las herramientas necesarias para implementar *cloud computing* privado.
5. Dar las bases y conocimientos para poder distinguir y escoger la infraestructura o método para implementar servicios según los requerimientos
6. Dar a conocer las ventajas y desventajas de usar *cloud computing*, tanto público como privado.

INTRODUCCIÓN

Las instituciones, negocios y empresas con operaciones a mediana y gran escala, han requerido de la tecnología para que todos sus procesos puedan realizarse lo más sencillo y eficientemente posible. Podemos comprobar que la mayoría de las empresas necesitan y usan una serie de aplicaciones tecnológicas para facilitar sus operaciones, estas aplicaciones son ofrecidas ya sea a sus clientes o a sus empleados en forma de servicios como bases de datos, correo electrónico, mensajería, sistemas ERP, sistemas CRM, gestores de contenidos, repositorios de archivos, sitios o aplicaciones web en general, *firewalls* y *proxies*.

Hasta ahora la forma tradicional para poder implementar y ofrecer cada uno de estos servicios es comprar el *hardware* para un servidor, configurarlo, instalar el *software* necesario, agregarlo a la infraestructura de red y darle mantenimiento. Aparte de ser un proceso relativamente complicado conlleva a ciertas limitaciones y problemas. Comprar *hardware* puede resultar en costos muy altos y en desperdicio del mismo pues a la hora de comprar un determinado *hardware*, en un principio puede ser mucho más de lo que realmente se necesita y entonces se van a desperdiciar recursos, luego conforme el sistema o el número de usuarios vayan creciendo puede llegar al punto en que el *hardware* no sea suficiente.

Otra limitación es el tiempo en el que el servidor se compra, configura, instala y se agrega a la red ya que desde la cotización de los componentes, pasando por la compra, la instalación, la configuración, y la implementación pueden pasar mucho tiempo antes de que esté totalmente en producción.

Para optimizar estos procesos de despliegue de infraestructura ha aparecido el *cloud computing*, el cual es una nueva forma de manejar la infraestructura de servidores, en la cual se pretende dar los servicios a través de máquinas virtualizadas, que están en un conjunto menor de máquinas físicas conectadas entre sí, de esta manera se pretende que se usen solo los recursos necesarios. Esta nueva plataforma ofrece nuevas formas de manejar la seguridad, disponibilidad, escalabilidad y flexibilidad, además del nuevo tipo de negocio que ha surgido, en donde un proveedor de *cloud computing* vende el servicio ya sea de *software*, plataforma o recursos computacionales a precios bajos.

El propósito de este trabajo es investigar, si la implementación de *cloud computing* en entornos privados, es más eficiente que los métodos de alquilar o comprar servidores individuales para infraestructura hasta ahora tradicional.

1. CLOUD COMPUTING

Cloud computing es un paradigma de la computación, ofrece información y servicios computacionales por medio de un conjunto o red de computadores que simulan la ya conocida metáfora del *Internet* como una nube (dada la complejidad de la infraestructura del *Internet*, en los diagramas de red se usa el dibujo de una nube para su representación). Haciendo énfasis en el aprovechamiento, extensibilidad y distribución de los recursos. El *cloud computing* no es una tecnología nueva, si no la integración de tecnologías y conceptos preexistentes para crear una manera de almacenar, distribuir y ejecutar aplicaciones y la información que éstas manejan.

De una manera simple y técnica puede verse como una red de computadoras físicas, que se convierten en un solo conjunto de recursos computacionales, que pueden distribuirse en computadoras virtuales según las necesidades, para poder ofrecer en forma de servicio ya sea una aplicación, una plataforma o los mismos recursos de *hardware*.

De la misma forma que para hacer un diagrama de red se utiliza una nube para abstraer toda la infraestructura que está en *Internet*, el cloud computing trata de ocultar todo lo concerniente al *hardware* e infraestructura compleja y dejar enfocarse a tareas de más alto nivel, tareas que no tengan que ver con mantenimiento de *hardware* y *software* para su administración.

De aquí en adelante le llamaremos “nube” a una implementación de *cloud computing*, la cual consiste en un conjunto variable de componentes de *hardware* y *software* para proveer un servicio.

El *cloud computing* permite enfocarse en los recursos computacionales, dejando por un lado la arquitectura técnica y física como servidores, infraestructura, configuración de redes, dispositivos de almacenamiento, etc., permitiendo a los usuarios el acceso a los servicios ofrecidos, según sean requeridos; también permite compartir estos recursos o tenerlos disponibles de una manera rápida y escalable con un mínimo de esfuerzo de los proveedores de dicho servicio.

Una de las principales ideas del *cloud computing*, es que el usuario pueda acceder a cualquier servicio a través del navegador u otra aplicación, mientras que la información está en algún punto de la nube, el cual no interesa al usuario, además puede hacerlo desde cualquier parte.

Aunque uno de los principales puntos del *cloud computing* es que el usuario no tenga nada que ver con el *hardware* y la infraestructura, en este trabajo y dado que nos ponemos en el lugar de proveedores y no de usuarios finales, si se hace énfasis en cómo crear una nube privada, y esto obviamente conlleva a tratar temas de *hardware*, redes, infraestructura, sistemas operativos, etc. El trabajo mismo tiene como fin mostrar el procedimiento para crear una infraestructura, en la cual se pueda implementar una nube para proveer servicios computacionales.

1.1. Características

El *cloud computing* es la integración o uso de tecnologías preexistentes, que le dan un nuevo enfoque a la manera en la cual podemos usar y adquirir infraestructura y lógica computacional, por esta razón muchas veces no se puede explicar lo que realmente es el *cloud computing*, y suele confundirse con

una de las tecnologías que utiliza. El *cloud computing* tiene varias características que lo definen como tal, entre ellas están las siguientes:

1.1.1. Infraestructura dinámica

El *cloud computing* se basa en la capacidad que tiene para brindar infraestructura de una manera dinámica, conforme el usuario o las necesidades lo vayan solicitando, a esto lo podemos llamar escalabilidad, otras veces también se le llama computación elástica, pues se puede incrementar o rara vez disminuir como sea necesario. Un usuario de *cloud computing* en cualquier momento puede aumentar la cantidad de recursos que está utilizando, así mismo el proveedor debe estar en la capacidad de proporcionar dichos recursos de una forma fácil e inmediata.

Para poder crear una nube que pueda ir creciendo dinámicamente, es necesaria una infraestructura también dinámica, esto se logra con una serie de *hardware* estandarizado, escalable y seguro, en el que se pueda implementar redundancia y que desde un principio esté diseñado para aceptar el cambio y la expansión. También es importante la virtualización que es la clave principal para una plataforma dinámica, pues ésta permitirá mover y expandir los datos entre distintos tipos de *hardware* sin modificar los entornos creados de los clientes.

1.1.2. Se consume o provee como servicio

Otra de las características principales del *cloud computing* es que el proveedor ofrezca un servicio y el usuario lo consuma como tal. El servicio puede ser el uso de una aplicación, el uso de una plataforma o el uso de

recursos de *hardware*. El usuario no se preocupa de como se le brinda el servicio, solamente lo consume y paga por dicho uso.

1.1.3. Reducción de costos en infraestructura

Desde el punto de vista de un usuario que renta servicios relacionados al *cloud computing*, no tiene que comprar ningún tipo de *hardware* y de igual manera no tiene que contratar personal para la instalación y el mantenimiento. Y desde el punto de vista del proveedor la compra de *hardware*, es menor y más eficiente para las operaciones, todo el *hardware* y *software* es adquirido a gran escala y por ende más barato. Se necesita menos personal para la administración, pues todo el *hardware* esta centralizado con instalaciones y configuraciones bien definidas y estandarizadas.

1.1.4. Auto administración de la plataforma

Otra de las optimizaciones que se pretende con el *cloud computing*, es el de evitar invertir tiempo en la administración de la nube. La idea es que los mismos usuarios la administren de una manera sencilla la parte que les corresponde. El proveedor ni siquiera tiene que estar enterado de como el usuario utiliza su servicio. Para ello se debe crear inicialmente una interfaz, por la cual el usuario podrá administrar ya sean recursos, una plataforma o un *software* específico.

El trabajo por parte del proveedor debe estar limitado a tareas específicas bien definidas y calendarizadas de mantenimiento, al final la nube debe auto-administrarse con la ayuda de a sus usuarios.

1.1.5. Se paga por lo que se consume

Una de las ventajas que ofrece el *cloud computing* a los usuarios, es la de pagar únicamente por lo que consumen. No es rentable para una empresa que tiene pocos usuarios pagar una gran cantidad de dinero por un licenciamiento o por el servicio, el *cloud computing* da la ventaja de pagar ya sean según los usuarios, el tiempo de utilización, la cantidad de recursos, etc. Cualquier medida previamente acordada tendrá un precio de acuerdo a como se consuma.

1.1.6. Independencia de la ubicación

Los datos pueden estar en cualquier lugar al igual que el usuario que los está solicitando. Incluso los datos no deben estar necesariamente en un solo *data center*, se puede dar el caso que la información de una misma persona o empresa está distribuida a través de varios *data center* en todo el mundo, además la información puede tener réplicas en distintos lugares para mayor seguridad. Recordemos que el *cloud computing* se basa en *clusters* de computadoras que pueden estar en cualquier parte, usando *Internet* o redes dedicadas para mantener la comunicación y esto debe ser totalmente transparente para el usuario.

1.1.7. Fácil mantenimiento

Tanto el *hardware* como del *software* pueden ser administrados de una manera sencilla, todo el equipo y los datos están centralizados, hay un conjunto de personas que se dedican a esto y hay muchas herramientas para su administración que hacen el trabajo aún más sencillo. Al crear una nube se busca la estandarización del *hardware* y *software*, por lo que los procesos para mantenerlos son sencillos, calendarizados y bien definidos.

1.1.8. Fiabilidad

Dado que los datos ya no estarán en las instalaciones del cliente y ahora van a estar en la nube, la cual está en poder del proveedor, debe existir un acuerdo de servicio y confianza entre clientes y proveedores para el manejo de la información y la disponibilidad de los sistemas, estos contratos son parte fundamental del *cloud computing*, pues una de las desventajas es tener la información fuera de la empresa y en manos de terceros. En los contratos hechos entre proveedores y clientes se debe definir las necesidades del cliente y cómo el proveedor según sus capacidades las va a solventar.

1.1.9. Seguridad

Dado que los datos están centralizados en grandes *data center*, los proveedores pueden invertir en seguridad y proporcionarla en paralelo a todos sus clientes. Los clientes no podrían gastar tanto si fuera para asegurar solo sus datos y a los proveedores al final gastan poco y aseguran mucho.

La seguridad es un tema vital en el *cloud computing* pues le da la confianza al cliente para que quiera usar una nube.

1.1.10. Medible

En *cloud computing* todos los recursos deben ser medibles, pues como recordamos, aquí se le brinda al cliente solo lo que él quiera utilizar, no más y no menos. Puede ser tiempo de procesador, memoria, número de usuarios que utilizan *software*, espacio de disco, etc. Cualquiera que sea el servicio que se le esté brindando al cliente debe ser medible.

1.1.11. Recuperación y tolerancia a fallos

Los proveedores en el *cloud computing* tienen una gran responsabilidad ya que les están pagando para que los recursos computacionales siempre estén disponibles, los acuerdos de servicio con el cliente se manejan con porcentajes muy altos en cuanto a disponibilidad y muchas veces el no cumplir con éstos resultan en penalizaciones económicas, mala reputación, pérdidas de dinero a los clientes, pérdida de clientes, etc. El *cloud computing* ha desarrollado mecanismo para responder a los fallos que puedan ocurrir, entre estos mecanismos están: copias de seguridad y restauraciones tan sencillas que gracias a la virtualización en el *cloud computing* han llegado a ser tan fácil, como copiar y pegar un archivo, y gracias a la manera en que se usa el *hardware* sí uno de los servidores se cae, los demás responden por él.

1.1.12. Virtualizado

Todo el *software* dentro de la nube será independiente del *hardware*, esto quiere decir que la aplicación de un cliente puede correr sobre varias máquinas físicas o varias aplicaciones de varios clientes pueden correr sobre una máquina física. El *cloud computing* usa la virtualización para poder crear plataformas extensibles, movibles y fáciles de mantener. Gracias a la virtualización se pueden mantener varias máquinas virtuales con distintos sistemas operativos, distintas configuraciones, diferente *software* pero conviviendo en la misma máquina física.

1.1.13. Multipropósito

La virtualización mencionada anteriormente aporta otra característica importante del *cloud computing*, todos los clientes pueden tener o necesitar distintas plataformas compartiendo la misma infraestructura, entonces el mismo *hardware* tiene que ser capaz de soportar cualquier propósito que el cliente necesite, por esto se dice que una nube debe ser multipropósito.

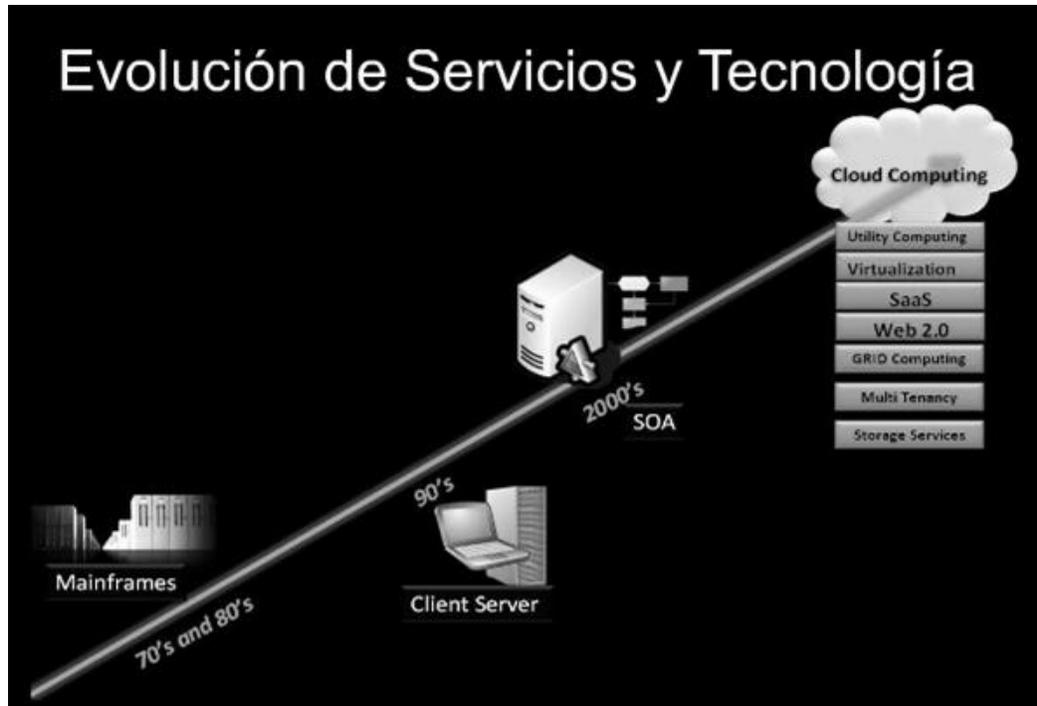
1.2. Evolución

A lo largo de los años se ha visto como distintos modelos de computación han logrado posicionarse como estándares de su época, ofrecen características que son aprovechadas y que hacen la distribución, almacenamiento y procesamiento de la información mucho más fácil y conveniente.

El *cloud computing* es un eslabón más de esta evolución, la cual a cada paso que da, busca innovación con tal de obtener mayores beneficios. Desde los *mainframes* hasta el *cloud computing* la computación sigue buscando auxiliar al ser humano en sus negocios, tareas y vidas cotidianas.

Aunque ésta clasificación de los modelos de arquitectura en la computación pueden ser variados, los más marcados son los *mainframes*, el cliente servidor, SOA y el *cloud computing*, vea figura 1.

Figura 1. Evolución de servicios y tecnología



Fuente: Saavedra E. <http://www.slideshare.net/estebansaavedra/introduccion-al-cloud-computing>.

1.2.1. *Mainframes*

Los *mainframes* son computadoras de gran tamaño y con gran capacidad de cómputo a las cuales se conectan terminales remotas, de esta manera varias personas podrán manejar la información de forma paralela.

Este fue uno de los primeros modelos computacionales para acceder y manejar información. Aun es usado en ambientes empresariales como bancos y empresas telefónicas, pero ya es un modelo bastante antiguo. Las terminales necesitaban una conexión directa con el *mainframe* y el *Internet* no había tomado la madurez de estos tiempos, por lo que el tema de conexión era un tanto difícil.

Los *mainframes* son grandes computadoras difíciles de mantener, de gran tamaño y muy caras. Las empresas que los usan deben tener personas especializadas para poder administrarlos y mantenerlos.

1.2.2. Cliente – servidor

Las computadoras personales fueron aumentando sus capacidades y se convirtieron en servidores, evolucionaron hasta el punto que también podían manejar grandes cantidades de información. De esta manera aparecen la arquitectura cliente servidor que en concepto se parece mucho a los *mainframes* siendo accedidos por medio de las terminales “tontas”, con la diferencia de que en el cliente servidor, un cliente también puede tener cierta participación en el procesamiento de información, además que la conectividad ya se ha hecho mucho más fácil gracias a la estandarización de los protocolos de comunicación, las redes y el *Internet*.

Aunque se puede decir que el *cloud computing* utiliza esta arquitectura, en la época cuando dio inicio, se trataba de uno o varios servidores físicos y autónomos, la mayoría de aplicaciones, tanto la parte cliente como la parte servidor, eran de la misma entidad, es decir una empresa tenía uno o varios servidores físicos, en los cuales instalaban el sistema y lo hacían disponible en la red de sus trabajadores para que estos accedieran como clientes.

Si bien las computadoras que se usan como servidores se han estandarizado y la diferencia con las computadoras personales son la cantidad y calidad de los recursos computacionales con los que cuentan, no dejan de ser costosos y requieren de personal para administrarlos. Además requiere inversión en cuanto a crear la infraestructura de red por la cual el servidor atenderá a los clientes si se trata de una *intranet*.

1.2.3. Service Oriented Architecture (SOA)

Aunque SOA es una arquitectura a nivel *software*, se puede mencionar como otro hito en la forma de cómo ha cambiado la computación ya que aporta características para llegar al *cloud computing*.

Luego de la arquitectura cliente - servidor y gracias al *Internet* y los protocolos que se fueron estandarizando, las aplicaciones se ofrecieron como un conjunto de servicios a través de la web. En SOA no importa sobre que plataforma se ejecuten los servicios pues las aplicaciones se comunican con un lenguaje estándar, se ofrecen como servicio y no importa la localización pues pueden ser accedidas desde cualquier lugar a través de *Internet*.

1.2.4. Cloud computing

Cuando se comprende el modelo del *cloud computing*, se aprecian las características de los modelos citados y como con el tiempo se han tomado las mejoras para ir logrando crear modelos, cada vez más funcionales y que se adapten al entorno de las personas que los usan.

1.3. Elementos del *cloud computing*

Utilizando un enfoque técnico de lo que es *cloud computing*, podemos de una manera muy global mencionar ciertos elementos que deben estar presentes para poder crear una nube.

1.3.1. Infraestructura

Cuando se habla de infraestructura, se refiere al *hardware*, el cual es el corazón de la nube, pues como sabemos el *hardware* es toda la maquinaria mecánico – eléctrica en la cual se almacena y se procesa cada bit de información. Todos los *data center* están conformados por *racks* llenos de servidores en los cuales se maneja y se guarda la información. Una de las tareas principales al crear una nube es adquirir el *hardware* necesario sobre el cual correrá todas las máquinas virtuales, este *hardware* físico deberá estar en un lugar especializado, en el cual se pueda posicionar ordenadamente y de una manera óptima para ahorrar espacio, con la suficiente alimentación eléctrica y un sistema de enfriamiento.

El *hardware* que se utiliza para crear una nube, no debe ser necesariamente el mismo, las nubes por lo general tienen la capacidad de estar construidas bajo *hardware* heterogéneo, aunque una buena práctica sería estandarizar el *hardware* para facilitar las tareas de mantenimiento.

El *cloud computing* está sonando cada vez más y tal es el caso que las empresas de *hardware* están creando y vendiendo *hardware* especializado y optimizado para el *cloud computing*, servidores con bajo consumo de energía, mucha memoria, almacenamiento amplio e incluso servidores con varios nodos internos.

1.3.2. Software para administrar la nube

Cuando ya contamos con el *hardware* para crear la infraestructura, necesitamos de un *software* con el cual podamos unirlo y administrarlo para que se comporte como un *cluster* de computadoras; nodos que puedan compartir

recursos e interactuar para poder formar esa nube donde cada computadora por sí misma no interesa, si no que es importante el conjunto de recursos que estas pueden proveer y la administración de las máquinas virtuales que correrán sobre esta plataforma.

Puede ser que entre un *software* y otro varíe la forma de administrar la nube, por lo general en este tipo de *software* pueden ir incluidos los siguientes elementos: un controlador de *cluster*, un controlador para los nodos, el controlador de la nube, un controlador de almacenamiento, etc. La idea es que este *software* pueda manejar el conjunto de máquinas físicas o nodos y poder hacer que trabajen juntos para poder brindar una única fuente de recursos.

Este *software* también nos da una serie de interfaces para poder crear máquinas virtuales, administrarlas, asignarles recursos y hacer que se expandan por el *cluster* de máquinas físicas para que aprovechen los recursos de todas éstas. Existen varios *software* de este tipo que están especializados en el *cloud computing* y que veremos más adelante como Eucalyptus, Open Nebula, Abicloud, Enomaly y Nimbus.

Las interfaces que nos brindan pueden ser por medio de comandos en una consola, una aplicación para el escritorio y la más comúnmente usada una interfaz web.

1.3.3. Máquinas virtuales

La virtualización es el elemento que hace la tarea más significativa en el *cloud computing*, permite la elasticidad de los recursos. En la nube cohabitarán todas las máquinas virtuales que necesitemos, utilizando los recursos disponibles pero con independencia total una de otra.

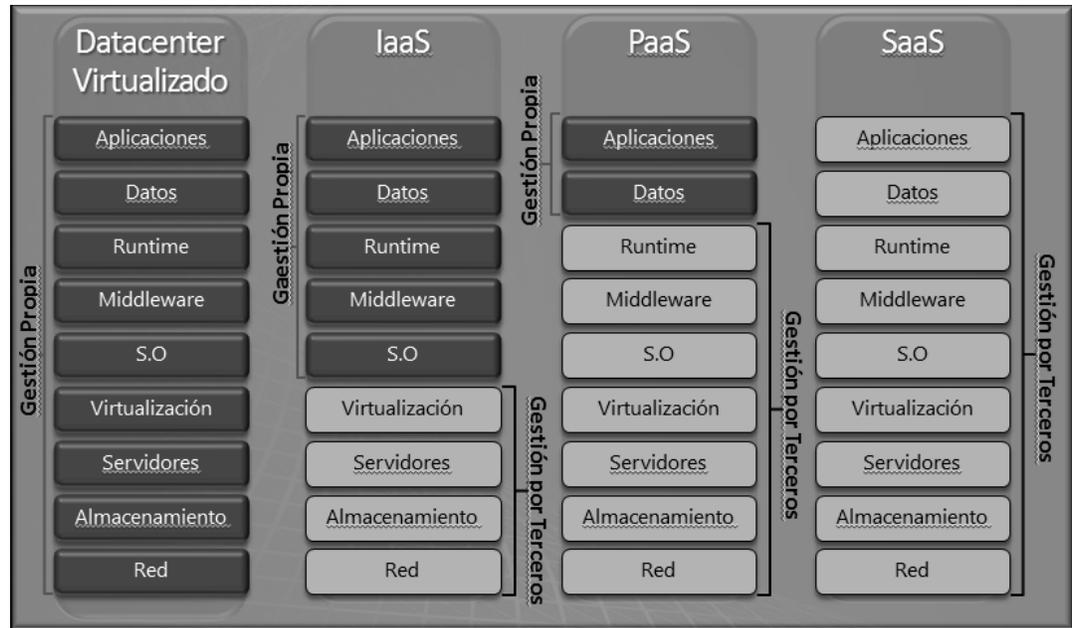
Las implementaciones de virtualización más utilizadas son KVM y XEN (KVM y Xen son hipervisores que permite la virtualizar máquinas sobre una máquina física) ya que son óptimas por utilizar paravirtualización y las extensiones de los procesadores. Sobre éstas máquinas virtuales están los recursos, las plataformas y/o las aplicaciones que luego podrán ser proveídos como un servicio a los clientes.

1.4. Niveles del *cloud computing*

Se ha hablado ya bastante sobre que el *cloud computing* tiene la idea de brindar un servicio a sus clientes; éste servicio puede ser recursos computacionales, una plataforma sobre la cual desarrollar o una aplicación. Estas tres clases de servicio se les conoce como niveles del *cloud computing*, y desde el nivel más bajo que es el *hardware*, pasando por *framework*, hasta el nivel más alto que es el *software*, estos niveles pueden ser dados al cliente y usar alguna medida para poder cuantificar el uso de este servicio.

Estos tres niveles suelen ser mencionados comúnmente por las siglas de su nombre en inglés que son IAAS, PAAS y SAAS.

Figura 2. Niveles del cloud computing



Fuente: <http://ad-hoc.net/blogs/2010/12/la-nube-iaas-paas-saas-ita-a-un-breve-repaso-de-conceptos>.

1.4.1. *Infrastructure as a service (IAAS)*

Este nivel del *cloud computing* es el más bajo pues se refiere a proveer infraestructura como un servicio, es decir proveer de recursos de *hardware*, tales como procesador, memoria y espacio de almacenamiento. Como veremos más adelante estos recursos de *hardware* son virtualizados, de esta manera podemos darle al cliente lo que necesita y poder cuantificar su uso y cobrar por ello. Obviamente este método es de mucha conveniencia al cliente pues paga sólo por lo que utiliza y no tiene la necesidad de comprar *hardware* que puede ser más caro y que debe estar actualizando.

La característica más interesante de usar este servicio es el de poder manejar el *hardware* con completa libertad y poder extender capacidades en

cualquier momento a tal punto de percibir una expansibilidad casi ilimitada, siempre y cuando paguen por ello.

En este nivel el cliente desea tener cierta capacidad de almacenamiento, cierta cantidad de memoria RAM, cierto poder de procesamiento, cierta tasa de transferencia de datos, etc. El cliente puede usar estos recursos a su antojo, instalarle el sistema operativo que más se adapte a sus necesidades, la plataforma que utilicen y cualquier *software* que necesiten, el proveedor únicamente le brinda los recursos virtuales.

Entre los actuales proveedores y tecnologías de *cloud computing* a nivel infraestructura podemos mencionar a Amazon EC2, Eucalyptus, FlexiScale, GoGrid, Nimbus, Rackspace Cloud y VPC de Amazon.

1.4.2. Platform as a service (PAAS)

Como su nombre en inglés lo dice, este nivel del *cloud computing* brinda una plataforma lista para que el cliente pueda implementar sus aplicaciones, el proveedor ofrece un set de *software* para que el cliente pueda diseñar, desarrollar, probar, implementar y publicar sus aplicaciones.

En este nivel el usuario no se preocupa de infraestructura, pues ésta ya va implícita cuando el proveedor ofrece la plataforma, está compuesta de varias capas de *software* especializado además de servicios y aplicaciones de utilidad para el cliente. Las plataformas se actualizan automáticamente por lo que el cliente solo se debe enfocar en el desarrollo de sus aplicaciones

En este nivel el cliente desarrolla sus aplicaciones y el proveedor le ofrece el *software* y herramientas sobre las cuales puede desarrollar. Al referirnos a

plataforma, nos referimos a un set de *software* y un claro ejemplo de este conjunto de *software* sería un sistema operativo, una máquina virtual de java, un manejador de base de datos, un manejador de tareas; en otro caso podría ser Php, Mysql, Apache, etc. Este conjunto de aplicaciones permitirían al cliente crear sus propias aplicaciones, a la vez que se olvida de tareas de bajo nivel como instalar los *framework*, actualizarlos, mantener el *hardware* entre otras.

Algunos de los actuales proveedores y tecnologías de *cloud computing* a nivel plataforma que podemos mencionar son App Engine, AppScale, Azure, Engine Yard, Force.com, Heroku, MTurk, RightScale, S3, SimpleDB y SQS, estos últimos 3 servicio de Amazon.

1.4.3. Software as a service (SAAS)

En este nivel el proveedor ofrece al cliente el uso de una aplicación completa, diseñada para tareas específicas, el cliente no se debe preocupar de *hardware* ni de un *framework*, ni del funcionamiento y disponibilidad de la aplicación, únicamente en el uso de ésta, ingresar información y obtener los resultados esperados. Mientras los niveles anteriores, IAAS y PAAS van dirigidos a programadores y empresas que desarrollan, el nivel SAAS está dirigido a los usuarios finales.

El *software* dado como servicio al cliente es un producto terminado, el cual está instalado en la nube del proveedor para estar disponible en cualquier momento y desde cualquier lugar. El cliente, en la mayoría de los casos, no deberá instalar ninguna aplicación en los equipos, lo más común es que se necesite un navegador para poder acceder a la aplicación que el proveedor le brinda, pero como se aclara es la mayoría de los casos y no la totalidad ya que existen otro tipo de clientes que no son necesariamente navegadores, por

ejemplo; en los dispositivos móviles es muy común instalar aplicaciones que acceden a servicios en la nube, pero aún así se pretende que el usuario pueda acceder a esta aplicación desde cualquier parte en la que se encuentre utilizando solo una conexión a *Internet*.

La modalidad de pago sigue siendo la misma que en los niveles anteriores, se paga por lo que se consume, ya sea por usuarios, por almacenamiento, por tiempo o una combinación de varios, la idea es que siempre sea más rentable que comprar una licencia de uso tradicional, las cuales tienen precios elevados en comparación con las cuotas periódicas que se pagan al usar *cloud computing*.

Entre los actuales proveedores y tecnologías de *cloud computing* a nivel *software* podemos mencionar a Google Apps, Intacct, Microsoft Online, NetSuite y Salesforce.

1.5. Modelos utilizados en el *cloud computing*

El *cloud computing* como ya se mencionó, se basa en tecnologías preexistentes, pero para que estas tecnologías funcionen y brinden las ventajas implícitas del *cloud computing*, existen modelos en los cuales se fundamenta y por medio de los cuales ofrece todas sus ventajas.

1.5.1. Modelo de adquisición – basado en servicio

La forma en la que se adquiere el producto o el bien en cuestión, es brindándolo como un servicio. Lo que ofrece el *cloud computing* no es un producto que el cliente tiene que comprar para poseerlo, es prestado y el usuario hace uso de él para poder suplir alguna necesidad de su negocio.

El usuario en ningún momento es el dueño del producto, necesita del proveedor para que éste lo ponga a su disponibilidad para utilizarlo, con la ventaja de hacerlo disponible desde cualquier lugar, en cualquier momento y con la garantía de siempre tenerlo listo y funcionando. El cliente no se preocupa del equipo, ni del *software* y de las tareas que deba hacer el proveedor para brindarle el servicio, simplemente lo consume y disfruta de sus beneficios.

1.5.2. Modelo de negocio – basado en uso

Tanto el proveedor como el cliente hacen uso de la tecnología para generar ganancias y harán uso de *cloud computing* para reducir gastos y mejorar sus rentas. Para esto el modelo de negocios más atractivo para el cliente es pagar solo por lo que consume, así no tiene que realizar grandes inversiones de capital en *hardware* ni de personal y puede ir pagando y escalando conforme el negocio lo vaya requiriendo.

El comprar licencias caras de *software* requiere inversión, además de tener que gastar en la instalación y el mantenimiento, muchas empresas pequeñas y medianas no pueden hacer esta clase de inversión, de esta manera los proveedores de *software* también pierden clientela que no pueden pagar sus altos precios. Al poder pagar solo por el uso que le den a los recursos, las empresas pueden optar por probar las herramientas que los proveedores ponen a sus servicios y verificar si es rentable de acuerdo al negocio. De esta forma, tanto el cliente que paga solo lo que necesita, como el proveedor que tiene muchos clientes, obtienen mayor beneficio de este modelo de negocio basado en el uso.

1.5.3. Modelo de acceso – *Internet* o *intranet*

Para poder acceder a los servicios del *cloud computing* se utiliza el *Internet* si es una nube pública ó una *intranet* si es privada, esto se hace generalmente por medio de un navegador, aunque existen aplicaciones que también consumen servicios de *cloud computing*. Tanto si se quiere utilizar la aplicación del proveedor a nivel SAAS o administrar los recursos a nivel IAAS se puede hacer por medio de un navegador web.

Este modelo permite que los recursos estén disponibles para el cliente o usuario desde cualquier lugar y desde cualquier dispositivo, siempre y cuando tenga conexión a la nube. Como se comprenderá el usuario no tiene control físico sobre los datos o recursos, solo los puede acceder de manera remota a través de una red, por lo que la misma se convierte en un punto crítico para el acceso al servicio. Es muy aconsejable que se tengan varias conexiones por si alguna falla cuando el acceso a los recursos es crítico.

1.5.4. Modelo técnico – dinámico y flexible

El *cloud computing* tiene la ventaja de proveer recursos de una manera especial, en la cual es muy fácil escalar en los recursos que el cliente pida, esto es posible gracias a que usa un modelo dinámico y flexible que como ya se mencionó, se fundamenta en la agrupación de máquinas físicas y la repartición de los recursos por medio de la virtualización. En cualquier momento el cliente puede solicitar que se aumenten o disminuyan los recursos que utiliza, y el proveedor gracias a este modelo no requiere de mucho trabajo para hacerlo. Se puede hacer de manera automática y sin complicación alguna.

2. INFRAESTRUCTURA DEL *CLOUD COMPUTING*

Hasta este punto en el trabajo ya se ha explicado a grandes rasgos qué es el *cloud computing*, sus características, elementos, niveles, etc. En este capítulo se desarrollarán los elementos necesarios para poder construir la nube.

Cuando nos referimos a infraestructura vamos a mencionar todos aquellos elementos por los cuales se puede crear una plataforma *cloud computing*, elementos que son la estructura de una nube.

2.1. Servidores

Un servidor no es más que una computadora con características especiales para poder proveer de servicios a otras. Un servidor debe tener los recursos suficientes para poder responder a las peticiones de muchas computadoras conocidas como clientes.

Un servidor también se puede referir a una aplicación especial que brinda servicios, un servidor puede ser cualquier máquina o *software* que brinde un servicio a un cliente que lo consuma, sin embargo en el contexto de este trabajo se tomará como el *hardware* de una computadora, que nos brindara los recursos para poder crear y mantener las máquinas virtuales de la nube.

Entonces cuando se habla del conjunto de máquinas físicas se refiere al *hardware* que se usará como base de la nube, o sea un conjunto de servidores. Para crear una nube necesitamos que estos servidores estén interconectados y que puedan ofrecer y compartir sus recursos. A este nivel, donde lo que se

quiere es construir la base de la nube, los recursos que un servidor puede brindar son procesamiento, memoria y almacenamiento.

Una de las ventajas del *cloud computing* es que no necesita de servidores especializados, cualquier computadora podría formar parte de una nube ya que al final lo que importa son los recursos que ésta aporte. Sin embargo debido al gran auge que el *cloud computing* ha tenido, los fabricantes de servidores han optimizado estos para que aporten gran cantidad de recursos y puedan ser fácilmente acoplados y conectados para formar nubes de una manera más sencilla en espacios más pequeños, con ahorro en el consumo de energía y un enfriamiento más eficiente.

2.1.1. Procesadores

El procesador es la parte más importante de una computadora, ya que este le brinda la capacidad de cálculo y procesamiento. Para el *cloud computing* es importante contar con un buen poder de procesamiento, pues es uno de los recursos que se venden y comparten entre los clientes. Gracias a los avances de la tecnología los procesadores son cada vez más pequeños y poderosos, además los fabricantes les agregan características especiales para un buen desempeño en ambientes específicos como una nube.

Una elección óptima para poder utilizar en una nube es un procesador veloz, con varios núcleos y con soporte multi-hilo, de esta manera puede manejar varios procesos en paralelo.

La mayoría de los procesadores destinados a servidores tienen varios núcleos, en este momento hasta 12 comercialmente, lo que se convertiría en un excelente manejo de procesos simultáneos.

También existen los procesadores con extensiones de virtualización, lo cual es muy útil en un ambiente de *cloud computing* ya que los procesadores con esta tecnología pueden manejar la virtualización sin ningún tipo de penalización en el rendimiento, tanto Intel como AMD han desarrollado estas extensiones en la mayoría de sus procesadores recientes.

2.1.2. Memoria RAM

Este es otro recurso importante pues también las máquinas virtuales que forman la nube necesitan de memoria para poder funcionar, responder y manejar todas las conexiones, procesos, aplicaciones y demás actividades que sean necesarias.

Tanto los componentes que manejan la nube como los sistemas operativos que mantendrán las máquinas virtuales necesitarán suficiente memoria RAM. El controlador de la nube se beneficia si tiene suficiente memoria pues puede usarla como *cache* para dar y recibir instrucciones más rápidamente entre los nodos, de igual manera el manejador de almacenamiento se ayuda de memoria como *cache* para poder servir archivos de una manera más rápida y los nodos que son las máquinas en las cuales corren las máquinas virtuales, también necesitan de memoria para poder mantener muchas y/o grandes máquinas virtuales.

Dependiendo del *software* que se use para crear la nube, los requerimientos de memoria para cada uno de los componentes pueden variar, por lo general para los componentes administradores 1 o 2 *gigabytes* de memoria RAM estarán bien, en cambio para los nodos que servirán para correr las máquinas virtuales será según lo demanden las operaciones, la carga de trabajo y los mismos clientes.

2.1.3. Espacio en disco duro

Este recurso es muy necesario ya que se necesita un lugar donde poder guardar los datos de los cliente, donde guardar la imágenes de las máquinas virtuales, además que el cliente paga por espacio de almacenamiento.

Los fabricantes de discos duros, cada día sacan a la venta discos duros con más espacio, capaces de almacenar grandes cantidades de información, por lo que el almacenamiento cada vez es mejor y más barato. Para el *cloud computing* también se necesitan discos duros, que sean rápidos en la transmisión de la información, ya que la nube conlleva mucho tráfico de datos que obviamente se encuentran almacenados en estos discos.

De igual manera que la memoria RAM, el espacio en disco para algunos componentes de la nube depende del *software* que utilizamos para crearla y administrarla, pero necesitamos el suficiente espacio para almacenar todas las imágenes de las máquinas virtuales que se utilizaran y según los datos y la demanda de los clientes.

2.1.4. Interfaz de red

Queda mencionar en cuanto al *hardware* de los servidores, que se necesitan interfaces de red de gran velocidad, ya que como se ha dicho una nube es un conjunto de computadoras unidas por medio de una red, sí queremos que la nube procese, administre y comparta información como si fueran un único ente de procesamiento, las interfaces de red deben ser capaces de transmitir esta información tan rápido, que no se perciba distancia alguna entre los nodos que conforman esta nube.

A través de estas interfaces de red se transmitirán varios Gigabytes de información y es necesario que se realice lo más rápido posible. Para poder crear una nube con estas características es necesario usar redes con fibra óptica o tecnología cableada de velocidades altas como Ethernet Gigabit o Ethernet 10Gigabit. Para un entorno privado dependiendo de las necesidades se puede utilizar Ethernet megabit.

2.2. Sistema operativo

El *hardware* por sí solo no serviría de mucho pues se necesita un sistema operativo para poder manejar estos recursos. El sistema operativo es la interfaz para que tanto los usuarios como las aplicaciones puedan hacer uso del *hardware*.

Para crear la nube, se necesita que todos los servidores que se convertirán en nodos del *cluster* y los que funcionaran como *front-end*, tengan un sistema operativo sobre el cual van a funcionar los componentes que sirven para crear y administrar la nube, así mismo es necesario un sistema operativo, sobre el cual van a correr todas las máquinas virtuales.

Se debe usar un sistema operativo para el cual exista *software* con el que podamos crear un *cluster*, virtualizar y que puedan manejar los recursos de una manera óptima para poder servir las peticiones del cliente. Existen varios sistemas operativos que incluso han sido modificados o especializados para esta tarea, entre los más populares hasta este momento están Redhat, Windows Server 2008 R2 Hyper-V con System Center y Ubuntu. Redhat cuenta con *software* para implementar computación distribuida y también soporta virtualización. Microsoft ofrece fácil integración con Windows Azure y Ubuntu por su parte ofrece un sistema operativo especializado para crear nubes

privadas y aun así tener compatibilidad con nubes públicas que se llama eucalyptus. Estas serían plataformas listas para usar en el *cloud computing* ahorrando así mucho trabajo para el implementador de la nube.

2.3. Computación distribuida

Se ha hablado mucho sobre nodos, y un nodo no es más que una computadora en un conjunto de las mismas que trabajan de forma conjunta. La computación distribuida es un modelo computacional donde varias computadoras, a las que se les llama nodos están separadas físicamente pero unidas por medio de una red y trabajan en conjunto para resolver un problema, que sería difícil de resolver por una sola máquina.

La computación distribuida ofrece ciertas características que la hacen atractiva para ciertas tareas como en el *cloud computing*, entre estas características están la tolerancia a fallos, la escalabilidad, la multitarea y el acceso remoto entre otras.

El objetivo principal de la computación distribuida es compartir recursos. Ya que una máquina por si sola tiene recursos limitados, la computación distribuida pretende ir enlazando varias máquinas e ir sumando sus recursos, para conformar un ente de procesamiento y almacenamiento, el cual es la suma de sus partes, que idealmente no tiene límites según se vayan agregando mas máquinas con recursos.

El *cloud computing* cuya principal característica es la extensibilidad de los recursos computacionales, utiliza computación distribuida para crear una plataforma donde pueden correr las máquinas virtuales, de las cuales los clientes hacen uso.

La computación distribuida puede tomar varios nombres, según la forma en que se use, y es aquí donde existe confusión, pues este modelo de varias computadoras conectadas y trabajando juntas es muy general, por lo que se va a explicar las pequeñas variaciones y como se usa para crear nubes.

2.3.1. Computación distribuida

En un principio el modelo de computación distribuida no delimita distancia entre las computadoras, aún así suele llamarse computación distribuida cuando las computadoras están a grandes distancias y están unidas por redes Wan, además que trabajan en paralelo dividiéndose el trabajo en pequeñas tareas.

2.3.2. *Clustering*

El *clustering* se refiere a un conjunto de computadoras físicamente cercanas unas a otras, con una tarea específica, trabajando todas al mismo tiempo para resolverla y para hacerlo se dividen el trabajo en tareas más pequeñas que resuelven en paralelo.

2.3.3. *Grid*

Cuando se habla de un *grid*, las computadoras pueden estar físicamente cerca o lejos unas de otras, pero la diferencia principal es que la actividad que realizan no son tareas o procesamientos en paralelo, si no trata de balancear los recursos disponibles de todo el conjunto de computadoras y ofrecerlos a los clientes como recursos bajo demanda.

En conclusión el *cloud computing* usa un *grid* como modelo de computación distribuida, ya que no es el procesamiento paralelo sino los recursos los que más interesan, para brindar servicios al cliente. Entonces para crear una nube, se necesita de varios servidores conectados entre sí para poder crear un *grid*, el cual funcionará como una fuente escalable de recursos computacionales.

Al final lo que se obtiene, son grupos de computadoras trabajando en conjunto, no importando la distancia física que haya entre éstas, por esta razón, esta variedad de nombres puede ser intercambiable, *grid*, *cluster*, o computación distribuida, éstos no tendrán diferencia alguna a lo largo de este documento, incluso en los siguientes capítulos de este mismo trabajo se utiliza más la palabra *cluster* debido a las tecnologías que se mencionan.

2.4. Data Center

Un *data center* es el lugar, habitación, sótano, piso o edificio completo, dispuesto específicamente para albergar equipo electrónico y computacional, de manera que el equipo esté en un ambiente óptimo y seguro para su funcionamiento continuo.

Dado que cuando hablamos de la infraestructura de *cloud computing* nos referimos a muchos servidores conectados entre sí, que manejan información de clientes, cuyos negocios pueden ser críticos, debemos imaginar que se necesita poner todos estos servidores en un lugar seguro, donde se pueda garantizar la continuidad del servicio, la seguridad de los datos y un óptimo desempeño. Esto quiere decir que los proveedores de *cloud computing* necesitan construir o hacer uso de *data centers*. Los aspectos más significativos

para crear un *data center* son la seguridad, el consumo de energía eléctrica, el enfriamiento y el espacio.

Cuando se habla de seguridad, se refiere a resguardar los datos almacenados en las computadoras, para que ninguna persona sin autorización los pueda acceder o destruir, seguridad también se refiere al resguardo de todo el equipo de cualquier eventualidad o catástrofe que puedan resultar en daños o desperfectos, como un incendio o un terremoto.

La energía eléctrica es indispensable para un *data center*, se necesita de una buena fuente o generador de electricidad para poder alimentar a todos los servidores y demás equipo que se tengan funcionando. Además se debe tener un generador alterno en caso de que el principal falle, por lo general en cualquier *data center* aparte de la acometida principal, se cuenta con una serie de baterías que mantendrán encendidos los servidores en caso que la línea de alimentación principal falle; como las baterías no son suficientes, se cuenta con un generador de electricidad a diesel que funcionará como respaldo cuando la fuente de energía eléctrica principal falle.

Otro de los aspectos a tomar en cuenta en el *data center* es el enfriamiento de los equipos, equipos que funcionan con energía eléctrica y que tienden a sobre calentarse requieren mantenerse a una temperatura adecuada para que sus componentes no sufran daños y puedan mantener un funcionamiento constante y eficiente. Los *data center* cuentan con sistemas de enfriamiento para monitorizar y definir la temperatura del ambiente en donde se encuentra el equipo; en *data center* pequeños basta con el aire acondicionado, y en otros más grandes hay que diseñar complejos sistemas de enfriamiento.

El espacio es un parámetro más que hay que tomar en cuenta, la manera en que se distribuya el equipo por todo el *data center* tiene ciertas

consecuencias tanto en el funcionamiento de todo el equipo como en la seguridad, ya que para poder darle un buen mantenimiento necesitamos que todo este ordenado, visible, identificado y accesible. Así mismo para seguridad del lugar y del mismo equipo, cables, sistemas de electricidad y enfriamiento deben tener espacios asignados separados unos de otros y que no se conviertan en obstáculo para desplazarse por el *data center*.

Por lo general los servidores se colocan sobre estructuras llamadas *racks* que pueden apilar varios servidores uno sobre otro, permitiendo hacer las conexiones de una manera sencilla, estos *racks* se disponen en filas dentro del *data center* de manera que queden pasillos por donde los administradores puedan desplazarse y acceder a los servidores. Los cables tanto de red y electricidad así como los artefactos para el enfriamiento se encuentran apartados por medio de cielos y pisos falsos, las baterías también tendrán un lugar específico para poder funcionar sin problemas.

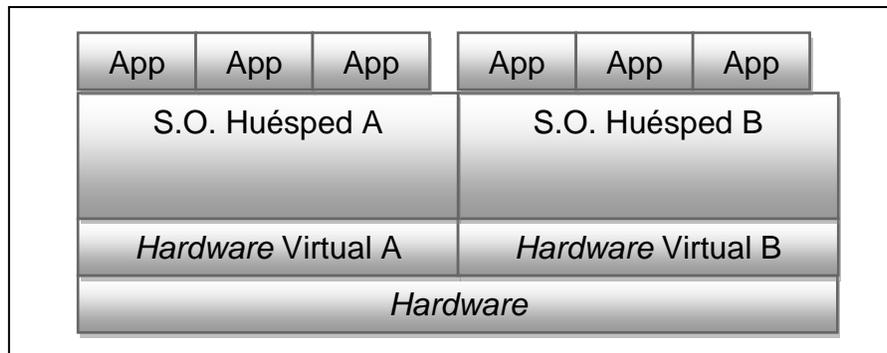
2.5. Virtualización

La prestación de los servicios en el *cloud computing* se basan en la virtualización, tecnología que permite simular los recursos de varias computadoras dentro de una misma máquina física, permitiendo que los recursos de *hardware* reales puedan ser administrados y distribuidos para crear versiones virtuales y simular más equipos en funcionamiento. Así mismo permite un despliegue de aplicaciones más sencillo y una manera más fácil de manejar los respaldos de información y la tolerancia a fallos.

Existen varias maneras de poder virtualizar, entre ellas la virtualización de *hardware* que trata de emular los dispositivos de *hardware* sobre los recursos

de la máquina física, este tipo de virtualización puede resultar lenta ya que cada instrucción a *hardware* debe ser simulada por el *hardware* virtual (figura 3).

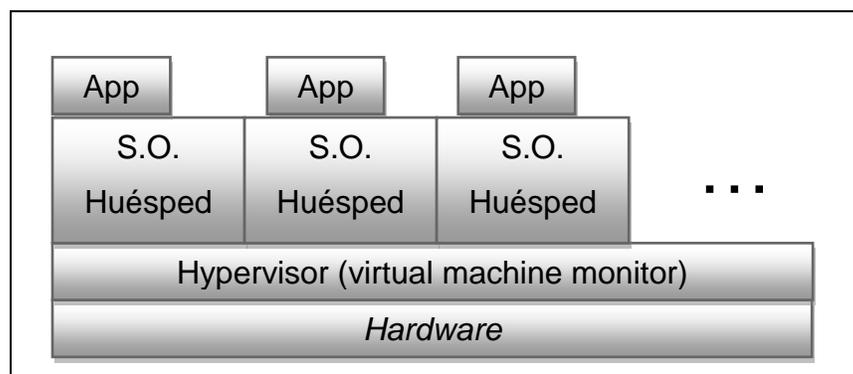
Figura 3. **Virtualización de *hardware***



Fuente: <http://www.ibm.com/developerworks/linux/library/l-linuxvirt/>.

Otra de las formas para virtualizar es la conocida como virtualización completa, esta consiste en utilizar un componente intermedio entre el *hardware* físico y el sistema operativo llamado hipervisor o máquina virtual el cual se encarga de mediar y conceder recursos para la partes virtualizadas. Tiene un mejor rendimiento que la virtualización de *hardware* (figura 4).

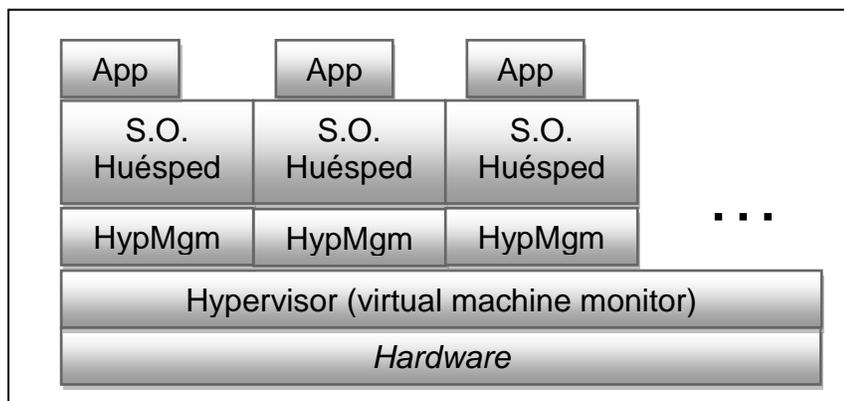
Figura 4. **Virtualización completa**



Fuente: <http://www.ibm.com/developerworks/linux/library/l-linuxvirt/>.

Para mejorar el rendimiento de la virtualización existe una técnica llamada paravirtualización que es parecida a la virtualización completa, con la diferencia que los sistemas operativos de las máquinas están levemente modificados para que sepan que están virtualizados y de esta manera interactúen con el hipervisor de una manera optima (Figura 5).

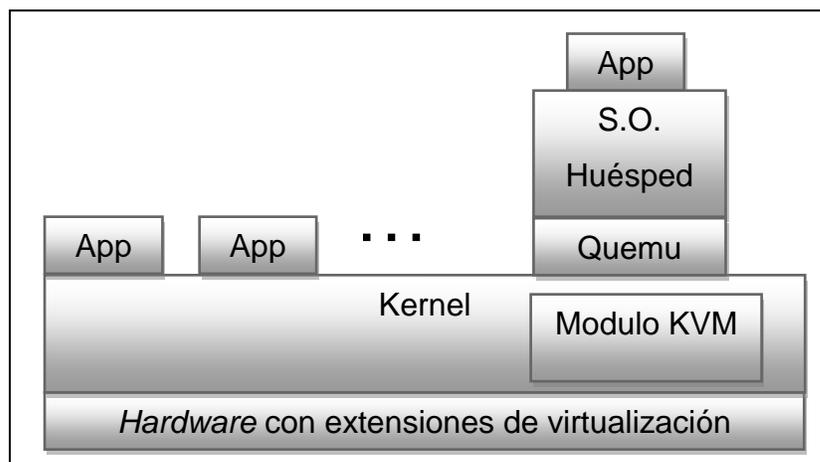
Figura 5. **Paravirtualización**



Fuente: <http://www.ibm.com/developerworks/linux/library/l-linuxvirt/>.

Por último revisamos la técnica que utilizaremos en este trabajo para virtualizar nuestros recursos, la virtualización con Kernel Virtual Machine la cual hace que el núcleo de Linux se convierta en un hipervisor utilizando un módulo de su núcleo (figura 6).

Figura 6. **Virtualización con KVM**



Fuente: <http://www.ibm.com/developerworks/linux/library/l-linuxvirt/>.

2.6. **Middleware o software para cloud computing**

En el primer capítulo de este trabajo, se mencionó en los elementos del *cloud computing* el *middleware* o *software* necesario para crear una nube de una manera sencilla. Para tener una visión de lo que realmente es este tipo de *software* se va a ahondar en uno en específico y que probablemente es uno de los más usados actualmente, debido a que es uno de los pocos proyectos de este tipo de código abierto, eucalyptus es un *software* que tiene la capacidad de crear y manejar entornos *cloud computing* (eucalyptus es una empresa dedicada al cloud computing habiendo liberado el proyecto con el mismo nombre para implementar nubes privadas), implementa virtualización por medio de KVM y además de poder crear una nube privada en cualquier *data center*, tiene interfaces que le permiten ser compatible con Amazon EC2 que es la plataforma de *cloud computing* público más utilizada hasta este momento.

2.6.1. Node Controller (controlador de nodos)

Los nodos en una plataforma de *cloud computing*, son servidores con procesadores que cuentan con extensiones de virtualización y en donde estarán ejecutándose las máquinas virtuales a las que se les llaman instancias, que proveerán los recursos computacionales. El *node controller* se encarga de ejecutar y controlar el ciclo de vida de las instancias, así mismo supervisa los recursos físicos de los servidores nodos y se comunicará con el *cluster controller* para intercambiar información y distribuir las instancias por la nube.

Las funciones principales del *node controller* son:

- Recopilación de datos relacionados con disponibilidad y recursos del nodo.
- Informar al controlador de la nube.
- Gestionar el ciclo de vida de las instancias.

2.6.2. Cluster Controller (controlador de cluster)

Este componente se encarga del manejo del conjunto de nodos y el despliegue de las instancias sobre ellos, se encarga de los aspectos de comunicación entre los nodos y sirve como puente de comunicación entre los nodos y sus instancias con el *cloud controller*.

Las funciones principales del *cluster controller* son:

- Manejar las peticiones de *cloud controller* para implementar instancias sobre los nodos.

- Decidir en base a recursos qué nodo utilizar para implementar una instancia.
- Controlar la red virtual que se crea entre las instancias y nodos.
- Recolectar información sobre los nodos y enviarla al *cloud controller*.

2.6.3. Walrus Storage Controller (controlador de almacenamiento de máquinas virtuales)

Este componente provee, a toda la plataforma *cloud computing* en la que se encuentre, un servicio de almacenamiento simple el cual es accesible por medio de un API de tipo SOAP o REST.

Las funciones principales del *walrus storage controller* son:

- Almacenar las imágenes de las máquinas virtuales.
- Almacenar las copias de seguridad de las máquinas virtuales.
- Almacenar archivos en general y usando el api de Amazon Web Services para accederlos.

2.6.4. Storage Controller (controlador de almacenamiento)

Mientras el *Walrus Storage Controller* provee de almacenamiento general de mantenimiento para toda la nube, el *Storage controller* sirve para proveer de almacenamiento persistente a los datos de las instancias, es decir que se pueden crear bloques que simulan unidades de disco para que puedan ser usadas por las instancias que estén funcionando.

Las funciones principales del *storage controller* son:

- Creación de dispositivos de almacenamiento compatibles con el estándar de Amazon Web Services.
- Proporciona almacenamiento para las instancias a través de protocolos AoE o SCSI.
- Permite la creación de volúmenes de almacenamiento para las copias de seguridad de las instancias.

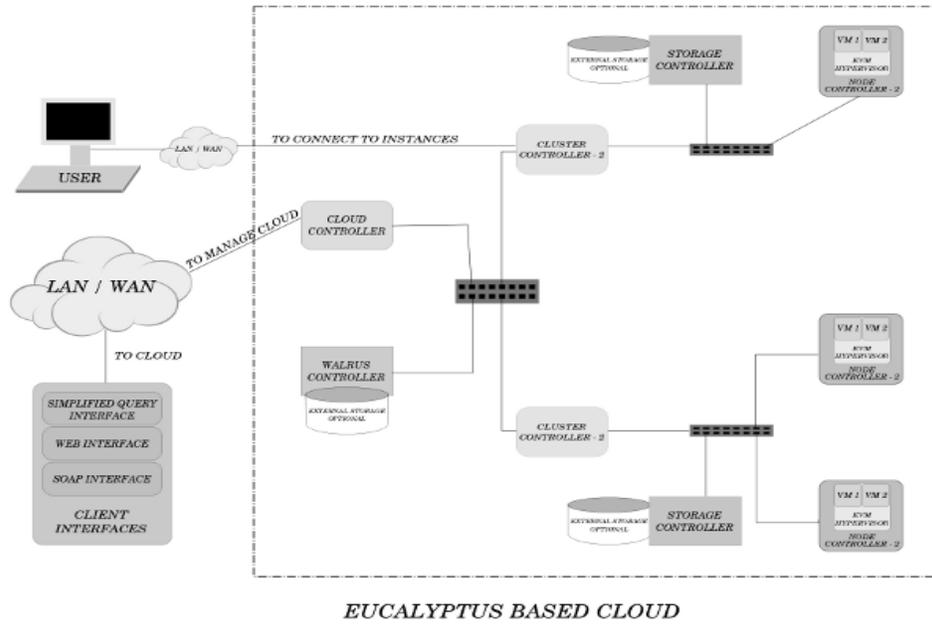
2.6.5. *Cloud Controller* (controlador de la nube)

El *cloud controller* es la interfaz entre el usuario y toda la plataforma de *cloud computing*. Por un lado tiene un conjunto de web services que pueden interactuar con el usuario o cliente y por el otro lado puede interactuar con los componentes internos de la plataforma como los controladores de *cluster*, de esta manera un usuario a través del *cloud controller* podrá configurar ciertos aspectos de la infraestructura.

Las funciones principales del *cloud controller* son:

- Monitorear la disponibilidad de los recursos de varios componentes como el hipervisor de cada nodo que brindan los recursos y el *cluster controller* que maneja los hipervisores de estos nodos.
- Decidir que *cluster* se utilizará para proveer de recursos.
- Monitorear las instancias que se están ejecutando.

Figura 7. Esquema de los componentes de UEC



Fuente: Eucalyptus Beginner's Guide – UEC edition, p.5.

3. TIPOS DE NUBES Y SUS COSTOS

A lo largo de este trabajo se le ha llamado nube a toda aquella plataforma computacional, construida con todos los principios y características previamente mencionados sobre *cloud computing*. Existen 3 tipos de nubes definidas por su ubicación o por quien las implementa, estos tipos son las nubes privadas, nubes públicas y nubes híbridas.

Cada uno de estos tipos conlleva a diferentes prácticas, procesos, operaciones y costos de parte de quien lo implemente. Cada negocio puede utilizar un tipo diferente de nube según sus necesidades.

3.1. Nubes públicas

Las nubes públicas son infraestructura *cloud computing* a gran escala, creadas por empresas que buscan rentabilidad, a través de ofrecer a sus clientes los recursos computacionales como un servicio. Estos proveedores de infraestructura *cloud computing* cuentan con grandes *data centers*, en los cuales tienen grandes cantidades de recursos que pueden ofrecer a sus clientes.

Se les llama públicas porque cualquier persona puede optar por usar este servicio y estará accesible a través de *Internet* en cualquier parte. Al utilizar esta clase de servicio, los clientes se olvidan de comprar *hardware*, del mantenimiento del mismo, de las actualizaciones y de los recursos. El proveedor será el encargado de manejar todas las demás tareas de bajo nivel.

3.1.1. Principales proveedores

Existen grandes compañías, pioneras en las nubes públicas y que por ende se encuentran en la cima del negocio, ofreciendo servicios que tienen mucha rentabilidad, para ellos y para sus clientes. Existen proveedores de infraestructura, proveedores de plataformas y proveedores de aplicaciones.

Cabe mencionar que existen proveedores que ofrecen servicios de *cloud computing* sin cobrar ni un centavo, ya sea como parte de algún proyecto libre al cual se deba contribuir, como parte de alguna promoción para probar el servicio o porque crean algún otro modo indirecto de percibir ganancias como publicidad. En este caso no hay ningún acuerdo que garantice el servicio pues éste es gratis, sin embargo se puede usar para escenario de prueba o aspectos que no son críticos para el negocio.

Entre los proveedores más destacados se encuentra Amazon Web Services, que desde que inició en el año 2002 ha estado entre los mejores, cuenta con una serie de servicios que van desde servidores virtuales, bloques de almacenamiento y balanceadores de carga hasta base de datos relacionales y aplicaciones de comercio electrónico. Amazon Web Services principalmente se dedica a ofrecer infraestructura como un servicio (IAAS). Otros proveedores de IAAS que pueden mencionarse son Akamai, Joyent, GoGrid, Rackspace, 3Tera y FlexiScale. Al igual que Amazon Web Services estos son proveedores que ofrecen soluciones muy rentables de recursos computacionales, cada uno ofrece ciertas características distintas a los demás y también distintos precios de acuerdo a sus servicios.

Entre los proveedores de plataformas para poder desarrollar soluciones personalizadas están Google App Engine, Salesforce AppExchange, LongJump, Azure y Scalr entre otros que ofrecen distintas plataformas en las cuales se pueden desplegar aplicaciones creadas por el cliente.

3.1.2. Acuerdo de nivel de servicio

El acuerdo de nivel de servicio abreviado como SLA por sus siglas en inglés (*Service Level Agreement*), es un contrato en el que se especifican una serie de condiciones y responsabilidades entre el proveedor de un servicio y su cliente, ya que mientras los clientes estén pagando por usar el servicio, el proveedor debe mantenerlo disponible según las cláusulas que se definan en este acuerdo.

Por lo general se especifican aspectos de tiempo, calidad y seguridad, basados en ciertos parámetros definidos por ambas partes que tiene incluso penalizaciones económicas por la caída del servicio.

Los principales puntos a tratar en estos acuerdos son:

- Tipo de servicio
- Soporte a cliente y asistencia
- Provisiones para seguridad y datos
- Garantías del sistema y tiempos de respuesta
- Disponibilidad de sistema
- Conectividad
- Multas por caída del sistema

En la práctica estos acuerdos son muy importantes ya que existen empresas cuyo negocio depende del servicio, por lo tanto un mal servicio o la caída del mismo puede repercutir en pérdidas muy grandes. Este documento es la única forma de asegurarse de que el servicio estará disponible y podemos contar con él para realizar nuestras operaciones de buena manera.

3.1.3. Costos de nubes públicas

Como se dijo a principio de este capítulo, una nube pública es *hardware* a gran escala para ofrecerla como un tipo de servicio a los clientes, quiere decir que los grandes proveedores propietarios de estas nubes gastan cantidades muy grandes de dinero con el fin de crear *data center*, capaces de proveer un buen servicio a todos los clientes que deseen rentarlo.

Esta gran cantidad de dinero es vista como una inversión que se recuperará conforme los clientes usen y paguen el servicio. Y gracias a que esta infraestructura, plataforma y/o *software* puede ser dado a muchos usuarios a la vez, se convierte en un tipo de economía de escala, en donde entre más clientes hay, más económico se puede prestar el servicio y los clientes al final pagan cuotas relativamente pequeñas.

Para un proveedor que quiera crear una nube pública del tipo de Amazon Web Services requieren inversiones de millones de dólares, en cambio para un cliente los precios son muy accesibles y ventajosos sobre crear su propia infraestructura. Al referirnos a cliente, estamos hablando tanto de personas individuales que requieren el servicio para sitios y blog personales como de grandes empresas que requieren el servicio para correr extensos sistemas y almacenar grandes cantidades de información.

Si se va a escoger una nube pública como plataforma se pueden omitir costos de *hardware*, renta o espacio donde estaría el equipo, los agregados para que el *data center* este seguro, refrigeración, energía eléctrica, baterías, plantas secundarias de energía que funcionan con diesel, no deben contratarse administradores, licencias de sistemas operativos, etc. Simplemente se paga por lo que se consume del servicio.

Los costos en que se incurren cuando se usa nube pública van a depender de lo que se necesita. Si lo que se quiere es montar el sistema personalizado de una empresa que incluye aplicación, sistema de archivos, base de datos, procesos y servicios, se debe pensar en pagar por infraestructura en la cual se puede hospedar todo el sistema y sus dependencias, se deben pensar en costos de memoria RAM, CPU, almacenamiento y transferencia. Si lo que se quiere es simplemente hospedar un sistema desarrollado o en desarrollo, que sea autosuficiente para el manejo de la información, puede decidirse por un servicio que provea plataforma para hospedar el sistema, pagando simplemente por espacio y transferencia. Y si lo que se quiere es usar un sistema prefabricado, específico para ciertas tareas que bien puede ser un CRM o un ERP y cuyo proveedor lo venda como servicio se debe pensar en los costos de uso que por lo regular son por usuario.

Para tener una idea clara sobre los costes del *cloud computing* se mostrará un ejemplo de la vida real, sacado del blog de Ricardo Galli, uno de los fundadores del sitio *mename.com*, el cual es un sitio de agregación y valoración de noticias que recibe más de 120,000 visitas diarias. En este ejemplo se puede mostrar cómo es posible subcontratar infraestructura para poder alojar un sitio comercial, de aportación y con bastante tráfico, en los servidores de Amazon Web Services. A continuación se detalla los productos y

los precios mensuales para poder alojar un sitio de este tipo en la nube pública de Amazon Web Services.

Tabla I. Precios mensuales por recursos computacionales en Amazon Web Services

Producto	Descripción	Precio (dólares)
Instancia principal	Servidor principal, Base de datos, DNS, NIS y NFS, memcache, SVN, SSH y SMTP. 7.5 GB de memoria RAM 4 CPU 850 GB de almacenamiento Plataforma de 64 bits	244,80
Instancia secundaria	Servidor secundario y de <i>backups</i> , con el DNS secundario y la réplica de la base de datos 1.7 GB de memoria RAM 1 CPU 160 GB de almacenamiento Plataforma de 32 bits	61,20
Instancias de auto escalado	Estas instancias iniciaran automáticamente solo cuando haya necesidad de escalar. 1.7 GB de memoria RAM 5 CPU 350 GB de almacenamiento Plataforma de 32 bits	164,16 (máximo)
2TB de transferencia de datos	La transferencia entre las instancias y demás servicios de Amazon es gratis, solo cobran por la transferencia hacia usuarios.	300,00 (Aprox.)
50 GB de almacenamiento	Este almacenamiento es para realizar los <i>backups</i> cada cierto tiempo.	5,00
CludWatch	Monitor de las instancias que permite iniciar las instancias de auto escalado cuando la carga llega a un límite determinado.	20,40
2 Balanceadores de carga	Estos para balancear el trafico que va dirigido a los servidores auto escalables	56,32

Fuente: <http://aws.amazon.com/es/ec2/#pricing>.

Los precios mostrados en la tabla I son cálculos por el uso aproximado durante un mes en dólares, por lo que si sumamos los valores obtenemos un total de 851,88 dólares mensuales. Quiere decir que se obtiene una plataforma de *hardware* óptima, escalable y redundante de tipo empresarial, pagando menos de lo que le deberíamos pagar a un administrador de sistemas para monitorear esta misma plataforma, sin tener que comprar todo ese costoso *hardware* y teniendo la ventaja de aumentar o disminuir su capacidad en cualquier momento, sin tener que alquilar un espacio para almacenar el *hardware*, ni pagar electricidad, ni comprar sistema de enfriamiento, etc. Incluso estos costos pueden disminuir si el servicio se contrata para tiempos definidos de un año en adelante, ya que los precios mostrados anteriormente se basan en el precio de uso por hora, lo que permite alquilar un servicio por solo unas horas si así se necesitará.

3.1.4. Ventajas

Existen muchas ventajas de utilizar *cloud computing* público, siendo las más marcadas, el ahorro y la disminución de trabajo por parte de quien lo contrata. A continuación se describen varias ventajas detalladas de manera que se comprenda la variedad de beneficios que se obtienen.

Ahorro en *hardware*, ya que en ningún momento hay que entrar en la difícil tarea de analizar, definir, cotizar, presupuestar y comprar el *hardware* necesario para nuestra situación. Con el *cloud computing* solamente pagamos por lo que necesitamos y en cualquier momento podemos pedir más o menos según nuestras necesidades.

El mismo hecho de no comprar *hardware* conlleva al ahorro en energía eléctrica, generadores de respaldo, baterías, alquiler de espacio físico y

sistemas de refrigeración, costos que pueden llegar a ser muy altos y muy complicados de mantener.

No se compran caras licencias de *software*, si se está contratando *software* como servicio, se está comprando el uso del *software* por determinado tiempo y según los acuerdos de ambas partes, en cualquier momento el cliente lo usa y lo puede dejar de usar. Así mismo en plataforma como servicio o incluso en infraestructura como servicio no hay que comprar las licencias de los sistemas operativos ya que son parte de la plataforma que estamos alquilando o lo incluye en el precio.

Si se contrata *software* jamás habrá que descargar e instalar actualizaciones, porque el *software* está instalado con el proveedor, no hay necesidad de velar por el buen funcionamiento de dicho *software*, la instalación de todas las mejoras, actualizaciones y parches son transparentes al usuario.

Según sea el negocio que se está manejando, existen temporadas en las cuales hay más ventas, más comunicación, mas carga de trabajo, y estas temporadas se dan cada cierto tiempo, ya sea una vez al mes, una vez al año o varias veces al año. Los sistemas de información de estos negocios sufren sobrecarga y por ende caídas en estas temporadas pues no soportan el exceso de trabajo, esto hace que el negocio pierda tiempo y dinero. La solución a este problema sería invertir más dinero en más *hardware* y mantenimiento, aunque luego de las temporadas altas todo el *hardware* y el personal se mantenga ocioso.

Cuando se usa Cloud computing público se tiene la flexibilidad y escalabilidad que se requiera, en cualquier momento se puede aumentar al doble o triple de la capacidad y luego regresar a la capacidad normal pagando

únicamente por lo que se ha consumido, es un gran ahorro y además le da confianza o capacidad al negocio o cliente de poder cubrir cualquier situación en cualquier momento.

Cuando se está pagando por un servicio de *cloud computing* no hay necesidad de darle mantenimiento a los sistemas, ni estar buscando el 100 % de la disponibilidad de los sistemas y por ende tampoco la necesidad de contratar más recurso humano para que haga este trabajo, los proveedores mantienen sus servidores funcionando y la disponibilidad es un tema crítico, pues si su sistema se cae muchos negocios lo harían también, esto representaría penalizaciones para el proveedor, penalizaciones económicas, pérdida de clientes, pérdida de reputación y confianza, etc. Por esto es que los proveedores mantendrán la disponibilidad a porcentajes muy cercanos al 100% invirtiendo mucho más en este tema que lo que el cliente podría invertir.

Al igual que la disponibilidad, la seguridad es un aspecto de gran importancia para todo aquel que maneje sistemas de información. En un negocio los datos pueden costar miles y hasta millones de dólares, por lo que invertir en seguridad es muy necesario. Cuando se utiliza *cloud computing* en cualquiera de sus niveles la seguridad es asunto de los proveedores que obviamente harán mucho más esfuerzo, tanto en conocimiento como en dinero, del que podría hacer uno de sus clientes para resguardar los datos.

Los contratos en el *cloud computing*, por lo general, no son comprometedores para con el cliente, casi siempre el cliente puede usar y dejar de usar el servicio cuando lo requiera, únicamente pagando por lo que consume. Si un cliente necesita un servidor por solo dos horas Amazon Web Services lo puede ofrecer por menos de 25 centavos de dólar.

Por último, el utilizar estos *data center* gigantes de los proveedores de *cloud computing* propician una filosofía en pro del medio ambiente pues se optimizan los recursos, menor consumo de electricidad, no se desperdicia equipo, mejores métodos de enfriamiento y se utilizan energías renovables. El *cloud computing* está siendo muy bien recibido en muchas organizaciones y negocios pues presenta todas las ventajas mencionadas y otras más que no se han considerado en este trabajo.

3.1.5. Desventajas

Las desventajas de usar una plataforma de *cloud computing*, aunque son pocas, pueden llegar a tener mucho peso dependiendo del negocio, ya que la desventaja más marcada es que los datos no están dentro del dominio de la empresa y esto no es conveniente para muchas empresas en las cuales la seguridad de los datos es un aspecto importante.

Al usar *cloud computing* los datos se encuentran físicamente en los servidores del proveedor del servicio y aunque la empresa tiene el control sobre ellos, no hay garantía que estos datos no puedan perderse o que alguien tenga acceso a ellos, por lo que por más seguro que el proveedor diga ser, siempre existirá desconfianza al usar estas plataformas debido a la posibilidad de que algo salga mal.

La seguridad sigue siendo uno de los conceptos mas mencionados en las desventajas del cloud computing, aunque la mayoría de los que ya están usando las nubes dicen lo contrario, la seguridad es uno de los principales temores para no usar las nubes públicas.

Los proveedores más grandes de estos servicios pueden sufrir de caídas en sus sistemas y servidores, por ende los sistemas alojados en estos servidores no estarán disponibles en un determinado lapso de tiempo, ésta caída no está en las manos de los clientes por lo que no pueden hacer nada más que esperar que el proveedor solucione la situación. Esta falta de disponibilidad de los sistemas en los negocios puede representar pérdidas de dinero.

Debido a que tanto los datos como las aplicaciones se encuentran en poder de los proveedores, se depende de una conexión a *Internet* para poder tener acceso a ellos. Esto se convierte en otro potencial punto de falla, debido a que estas conexiones también sufren caídas inesperadas en su servicio. Para tener una mejor disponibilidad, la empresa que quiere acceder a sus datos deberá contratar dos conexiones a *Internet* de distintos ISP para que en caso de que una de las dos no funcione, la otra servirá de respaldo permitiendo que el acceso a los datos continúe sin problemas.

Los proveedores de una plataforma de *cloud computing* por lo general publican una API para poder acceder a sus servicios y que el cliente pueda controlar de una mejor manera la información. Esta manera de operar requiere que del lado del cliente, hayan personas capacitadas para publicar los sistemas y la información en dicha plataforma por lo que existe una curva de aprendizaje media en el tiempo de implementación.

3.2. Nubes privadas

Contrario al estándar de las nubes públicas que representan un negocio en que cualquier cliente puede optar por comprar el servicio, una nube privada es la infraestructura propia de una empresa, creada con tecnología de *cloud*

computing, que es privada, está protegida por *firewalls* y con la cual provee de los servicios informáticos a sus usuarios finales y departamentos internos.

Una de las grandes desventajas que se han visto del *cloud computing* público, es que los datos del negocio están en poder del proveedor de *cloud computing*, lo cual genera un poco de desconfianza si estos datos tiene mucho valor. Un banco por ejemplo, aún no tiene la suficiente confianza en un tercero para que le almacene datos de las cuentas de sus clientes ya que pondría en riesgo todo el capital propio y el de sus clientes.

Cuando los datos del negocio representan grandes cantidades de dinero, deben ser privados ó simplemente queremos tener los datos físicamente dentro de nuestra empresa y aún así disfrutar de todas las ventajas que nos ofrece el *cloud computing*, entonces lo que se debe implementar es una nube privada, de esta manera podremos tener la flexibilidad, escalabilidad, seguridad, fácil mantenimiento y todas las demás características del *cloud computing* con la diferencia de que toda la infraestructura y los datos están dentro de la empresa y bajo nuestra responsabilidad.

3.2.1. Proveedores de tecnología para nubes privadas

La mayoría de empresas que trabaja con sistemas de información tienen infraestructura sobre la cual corren sus sistemas, y lo que las diferencian de tener una nube privada es la tecnología y el *software* para que el *hardware* con el cual ya cuentan, pueda proveer los servicios típicos del *cloud computing*.

Anteriormente se dijo en este trabajo que para crear una nube, debemos utilizar virtualización sobre un conjunto escalable de recursos de *hardware*, por lo que se necesita *software* para crear un *cluster* a partir de los servidores,

tecnología de virtualización y algún *middleware* que pueda administrar los recursos. Aunque estos requerimientos parecen sencillos, sería difícil encontrar las tres tecnologías específicas y que tuvieran una buena integración. Sin embargo, dadas las ventajas del *cloud computing*, muchas empresas y organizaciones se han especializado y han sacado productos específicos listos para implementar *cloud computing*, algunos para ser comercializados y otros como proyectos de código abierto.

Entre los que se han convertido en proveedores de tecnología *cloud computing* para ambientes privados están Eucalyptus, Open Nebula, VMWare, Citrix, Abicloud, Enomaly, etc., cada uno de estos proveedores tiene distintas tecnologías y distintas maneras de operar, pero al final proveen una manera de implementar una plataforma de *cloud computing* administrable y fácil de instalar. Eucalyptus, Open Nebula y Abicloud son proyectos de código abierto mientras que Citrix, Enomaly y VMWare son productos comerciales.

3.2.2. Costos de nubes privadas

Crear una nube privada no es más que crear un *data center* con tecnología de virtualización, automatizado, auto escalable y propenso a fallos. Esto requiere de inversión, tanto en *hardware* como en personal y en las tecnologías que se necesitan para construir este tipo de ambientes.

Los servidores representan el mayor gasto seguido de la infraestructura del *data center*, la electricidad, los enlaces y redes (ver tabla II).

Tabla II. **Guía para los costos en un centro de datos**

Costo	Componente	Sub Componentes
~45%	Servidores	CPU, Memoria, Almacenamiento
~25%	Infraestructura	Distribución de la electricidad, sistemas de enfriamiento, etc.
~15%	Consumo de energía	Costos de la electricidad
~15%	Red	Enlaces, equipo

Fuente: Greenberg, Hamilton, Maltz y Patel, *The Cost of a Cloud*, pág.68.

La tabla anterior no incluye los gastos del personal a cargo del *data center*, ya que al ser un *data center cloud computing* el cual va a estar automatizado, los gastos de personal son mínimos comparados con el gasto inicial según el artículo mencionado. Los precios de los servidores, de los enlaces, de la electricidad, etc., van a variar mucho entre cada país, temporada, época y situación de la empresa que lo quiera implementar, razón por la cual se omiten en este trabajo.

El gasto en una nube privada es mayor al de contratar los servicios de una nube pública, sin embargo existirán casos en los que se necesita un mayor control sobre la información y los sistemas que la manejan, entonces se deberá crear una plataforma segura y personalizada. Aun así una infraestructura de *cloud computing* privada resulta más barata o más productiva que un *data center* empresarial con infraestructura tradicional de servidores autónomos e independientes.

3.2.3. Ventajas

La principal ventaja del *cloud computing* privado sobre el público, es que los datos y sistemas están dentro de la empresa y no en el *data center* de un proveedor, por lo que existe el control total sobre los equipos y los datos. También se puede mencionar, el fácil despliegue de los sistemas, flexibilidad, independencia, no existe ninguna atadura a un proveedor, se puede reutilizar el *hardware* con el que cuenta la empresa y se puede prescindir de *Internet* pues la *intranet* de la empresa es suficiente.

Podemos observar la siguiente tabla comparativa sobre las ventajas de *cloud computing* privado sobre público extraída de un artículo en Saasmania.com

Tabla III. **Ventajas de *cloud computing* privado sobre el *cloud computing* público**

	Nube Privada	Nube Publica
Localización de los datos	Dentro de la empresa	Con el proveedor sin conocer la ubicación exacta
Percepción de inseguridad sobre los datos	No	Si. Los datos y la lógica están fuera de las instalaciones
Dificultada para integrar con otros sistemas propietarios	Baja	Alta
Puntos de fallo	El propio sistema	El proveedor de <i>cloud computing</i> y el proveedor de comunicaciones
Paradas por mantenimiento	El propietario decide	El proveedor de <i>cloud computing</i> impone los lapsos de tiempo

Fuente: <http://www.saasmania.com/2009/06/20/¿puede-ser-privado-el-cloud-computing.>

3.2.4. Desventajas

El mayor inconveniente con el *cloud computing* privado comparándolo con el público es la inversión en la que se debe incurrir para comprar el equipo y el *software* necesario, además del gasto de mantenimiento que se menciona antes, es mínimo comparado con el tradicional, pero aún así representa un gasto fijo que se debe cubrir para que los sistemas estén funcionando bien en la infraestructura privada.

El retorno de la inversión es más lento debido a la inversión inicial. Ahora la disponibilidad, la escalabilidad y todo el mantenimiento de la infraestructura es responsabilidad de la empresa que la usa en beneficio de su negocio.

Los *backups* y todo lo relacionado con seguridad también es responsabilidad interna de los empleados encargados, de esta manera hay que instruir a estas personas y muchas veces adoptar nuevas tecnologías lo que ocasionan gastos y riesgos.

3.3. Nubes híbridas

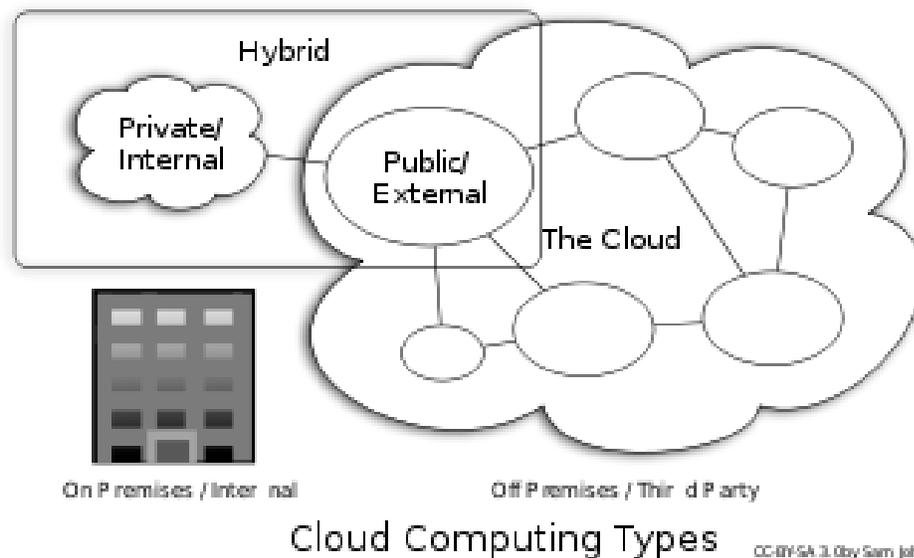
Para aprovechar lo económico de las nubes públicas y la seguridad de la nubes privadas, se han creado las nubes híbridas, que constan de una parte de la información y los sistemas en ambiente público como Amazon Web Services e información que debe ser privada o sistemas de operación crítica adentro de la empresa en una infraestructura privada, conectados por medio de un enlace estándar, comúnmente utilizando *Internet*.

Este tipo de infraestructura puede volverse muy compleja, debido a la necesidad de manejar y distribuir la información en ambos ambientes. Aún así

este modelo se está volviendo tan común que los proveedores están orientando sus plataformas para que soporten la infraestructura híbrida ofreciendo flexibilidad a los clientes.

Al utilizar este tipo de tecnología, debe existir el conocimiento para decidir qué es lo que se va a colocar en la parte privada y que se va a delegar en una plataforma pública. Obviamente en la parte privada estará la información más importante que requiere mucha seguridad así como los procesos críticos de los sistemas. La información restante y los sistemas de menor importancia pueden estar en los servidores de terceros.

Figura 8. **Cloud computing híbrido**



Fuente: <http://lingonlife.blogspot.com/2010/11/xcerion-icloud-hybrid-cloud-computing.html>.

4. PROCESO PARA LA CREACIÓN DE INFRAESTRUCTURA *CLOUD COMPUTING* PRIVADA

A través de los capítulos anteriores se ha dado a conocer lo que es *cloud computing*, sus características, evolución, elementos, niveles, el tipo de *hardware* utilizado, los tipos de infraestructura, etc. Con esta información ya será más fácil poder crear una infraestructura de tipo *cloud computing* privado, el cual es un punto importante de este trabajo a analizar para poder enumerar las ventajas que esta infraestructura nos brinda y compararla con la infraestructura tradicional de servidores dedicados.

4.1. Metodología

Para poder realizar este proceso de una manera ordenada se propone una metodología un tanto empírica, basada únicamente en las experiencias previas con estas tecnologías. En esta metodología se propone una serie de pasos secuenciales divididos en etapas, en donde se deberán realizar una o varias tareas para que al final, la creación de la infraestructura sea satisfactoria. Las etapas serán la capacitación e información en tecnología, análisis (incluye selección de herramientas y equipo), diseño, instalaciones y configuraciones, pruebas y Administración que incluye la creación de las máquinas virtuales o imágenes y el ciclo de vida de las instancias virtuales.

A continuación se explicará cada una de las etapas y a la vez se irá realizando el ejemplo práctico para explicar detalladamente la construcción de la plataforma privada.

4.2. Capacitación

Antes de crear una infraestructura virtualizada del tipo *cloud computing* se debe tener los conocimientos necesarios, conceptos, técnicas, procesos, proveedores, herramientas, todo lo concerniente al tema, pues de esta manera podremos crear algo funcional, óptimo y que realmente cumpla con los requerimientos dispuestos. Si no hay personal capacitado el intento por construir la plataforma puede verse fallido, ocasionando gastos innecesarios y no recibir los beneficios que se esperaban obtener.

Para el ejemplo ya hemos estudiado bastante sobre *cloud computing* en los capítulos anteriores, por lo que estamos en un nivel intermedio de conocimiento sobre el tema. Se sabe que se necesita *hardware*, cierto tipo de *software*, la manera en que el *hardware* se comunicara, los proveedores del *software*, etc. Habrá mucho más que aprender pero conforme se avanza y se hacen pruebas se obtendrá mucho más conocimiento.

4.3. Análisis

Generalmente el análisis en una metodología se encarga de analizar los requerimientos y proponer una solución, en esta metodología no hay toma de requerimientos pues el requerimiento es claro, construir una nube privada, sobre la cual, cuando ya esté terminada habrá flexibilidad para poder satisfacer cualquier futuro requerimiento de infraestructura.

Teniendo claro el requerimiento se debe realizar el análisis. El análisis no es más que definir que se necesita para poder crear la plataforma teniendo en cuenta los recursos con los que se cuentan. Definir lo que se debe comprar, lo que se puede usar, cómo se va a usar y plantear una solución. Se puede decir

que el análisis es el estudio del escenario en el que nos encontramos para proponer los puntos principales de una posible solución.

Para el ejemplo práctico vamos a definir un escenario ficticio. Un escenario pequeño y sencillo para poder mostrar de una manera fácil el proceso de construcción. Se va a suponer que es una pequeña empresa que quiere probar la nueva tecnología *cloud computing* que hasta hace poco había escuchado. No se quiere gastar dinero y se cuenta con 3 equipos de características medias.

Tabla IV. **Equipos disponibles para una plataforma *cloud computing***

Equipo	CPU	Memoria	Disco Duro	Extensiones de virtualización
Dell Inspiron	Intel core 2 duo a 1.5 Ghz	2 Gb	80 Gb	No
HP Pavillion	Intel i3 a 2.2 Ghz	4 Gb	120 Gb	Si
Compaq	AMD Athlon Dual Core a 2.1 Ghz	3 Gb	120 Gb	Si

Fuente: elaboración propia.

Haciendo el análisis nos damos cuenta que tenemos equipos que aunque son de uso personal, se pueden utilizar, además sabemos que existen plataformas gratuitas de código libre para crear entornos virtualizados.

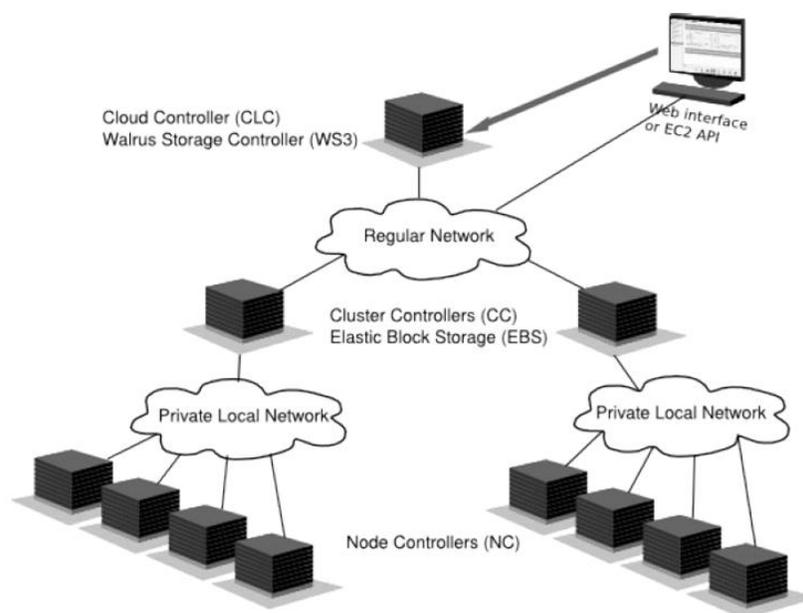
Por tales razones se propone crear una infraestructura virtualizada del tipo *cloud computing* con los 3 equipos a disposición utilizando el *software* desarrollado por eucalyptus e implementado en Ubuntu Enterprise cloud. Se sabe que debemos crear un *cluster* con los equipos disponibles e instalar los componentes de Ubuntu Enterprise Cloud que mencionamos antes.

4.4. Diseño

Sabiendo lo que podemos utilizar, hay que definir cómo lo vamos a utilizar. En la etapa de diseño se bosqueja y planifica la construcción.

Para el ejemplo debemos definir la manera en que utilizaremos los recursos que tenemos, sabiendo cómo funcionan los equipos y herramientas que vamos a utilizar, realizamos un diseño de lo que se busca construir.

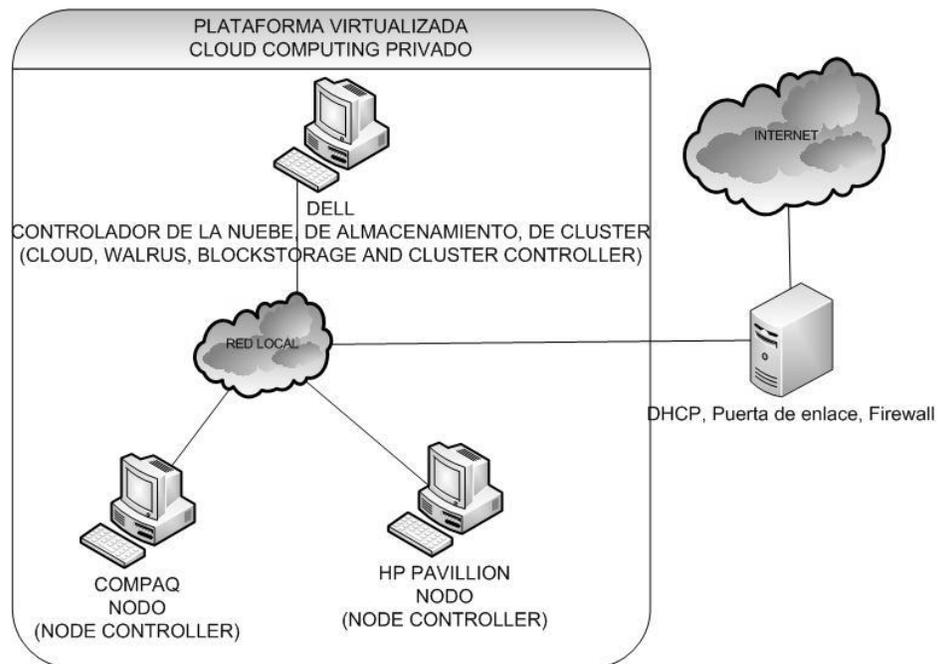
Figura 9. Elementos de Ubuntu Enterprise Cloud



Fuente: Technical White Paper Ubuntu Enterprise Cloud Architecture, p.9.

El esquema anterior es un diseño de nube privada que muestra los componentes de UEC, podemos utilizarlo como diseño de lo que vamos a construir, con la salvedad que en lugar de tener dos *cluster*, únicamente haremos uno conformado por un controlador y dos nodos. Nuestro diseño real sería el siguiente.

Figura 10. **Diseño para una nube privada con 3 equipos utilizando UEC**



Fuente: elaboración propia.

En cuanto a las configuraciones de red, lo único que se necesita es que las máquinas que van a formar parte del *cluster* estén en la misma red. Para el ejemplo utilizaremos 192.168.1.0 y se va a utilizar un servidor DHCP disponible para que la asignación dirección a las máquinas sea más sencilla.

4.5. Instalación y configuración

Vamos a utilizar una máquina como controlador de la nube y las otras dos máquinas como nodos. En la máquina Dell, debido a que no cuenta con extensiones de virtualización, instalaremos los componentes controladores y en las otras dos máquinas instalaremos controladores de nodo.

Hay varias maneras de hacer la instalación, recordemos que vamos a utilizar el *software* eucalyptus, el cual fue implementado en UEC de Ubuntu por lo que podemos utilizar los paquetes originales de eucalyptus e instalarlos sobre gran variedad de distribuciones Linux, o podemos utilizar los paquetes UEC específicos para instalarlos sobre Ubuntu Server, o aún más, fácil utilizar el disco de instalación de UEC que viene con Ubuntu Server a partir de Ubuntu Server 9.10. Por facilidad utilizaremos esta última opción ya que evitará unos cuantos pasos en las configuraciones.

A continuación se trata de describir el proceso de instalación y configuración de los equipos con ayuda de algunas imágenes obtenidas durante el proceso, de todas formas existe documentación específica para la instalación de cualquiera de los componentes; lo descrito a continuación es a modo de resumen.

4.5.1. Instalación de los controladores de la nube

En los controladores de la nube incluiremos el *Cloud controller*, *Walrus*, *Cluster controller* y el *Storage controller*, componentes que ya hemos especificado son propios de Ubuntu Enterprise cloud y eucalyptus.

Para esto se ha descargado la imagen de Ubuntu Server 10.04 desde el sitio oficial y a continuación se detalla la instalación.

Se arranca el equipo desde el disco de UEC (ver figura 11).

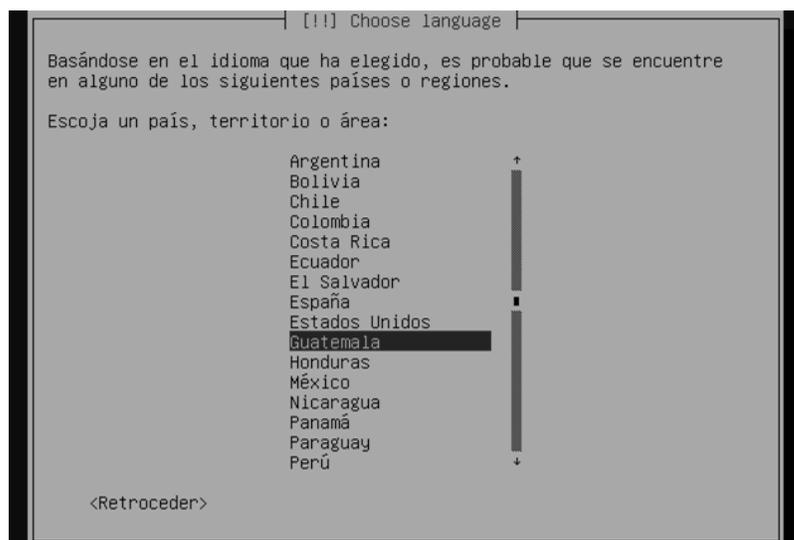
Figura 11. **Pantalla de inicio del instalador de UEC**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

Seleccionar la opción “Instalar Ubuntu Enterprise Cloud” y a continuación seleccionar la región para el lenguaje (ver figura 12).

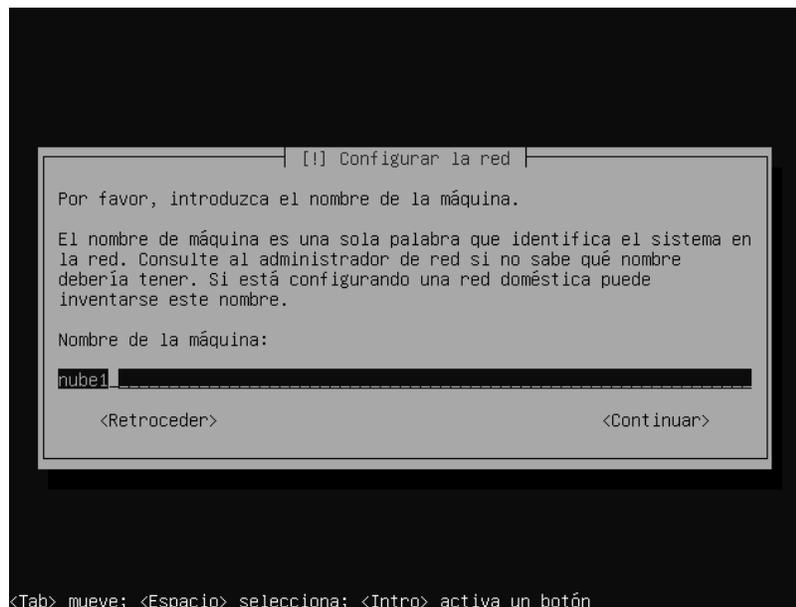
Figura 12. **Pantalla de selección de lenguaje para UEC**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

El siguiente paso será asignarle un nombre a la máquina. En este caso hemos usado “nube1” para reconocerlo como el equipo controlador de la nube (ver figura 13).

Figura 13. **Pantalla de configuración de red para un equipo en UEC**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

Luego se selecciona el tipo de instalación, en ese momento el instalador realiza una búsqueda en la red para ver si existe algún *Cloud Controller* previo. Como recordamos, este equipo será el controlador de la nube, entonces el espacio debe quedar vacío como se muestra a continuación (ver figura 14).

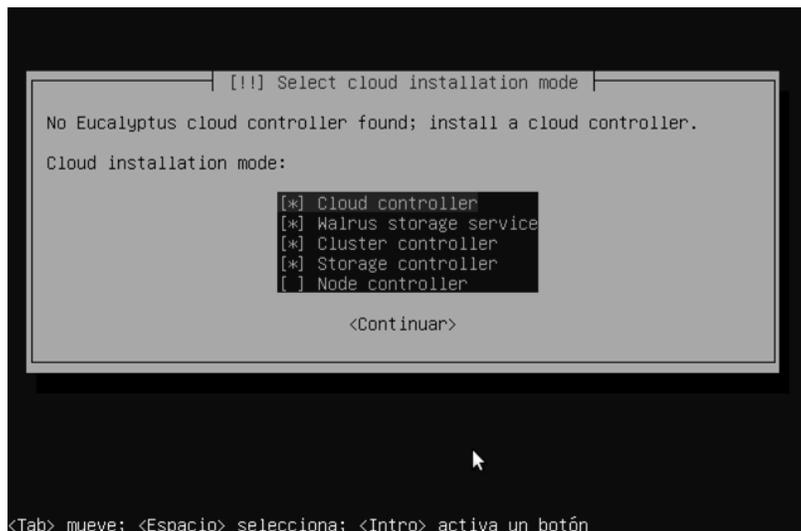
Figura 14. **Pantalla para seleccionar el controlador de la nube en UEC**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

Se seleccionan los componentes a instalar, los controladores de la nube, el *cluster*, el almacenamiento y el servicio *Walrus* (ver figura 15).

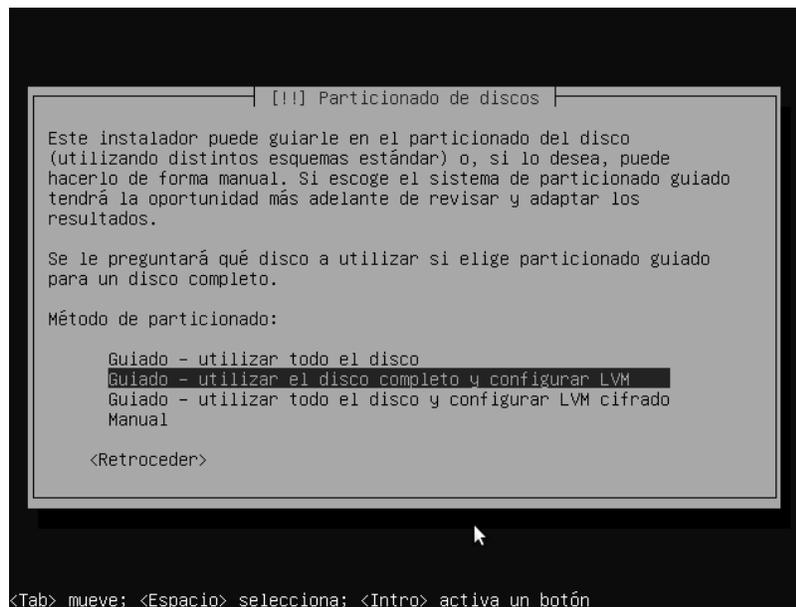
Figura 15. **Pantalla para seleccionar los componentes de UEC a instalar**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

Lo que sigue a continuación es el particionado y formato de los discos, lo cual también se hace con el instalador por medio de sus distintas guías (ver figura 16).

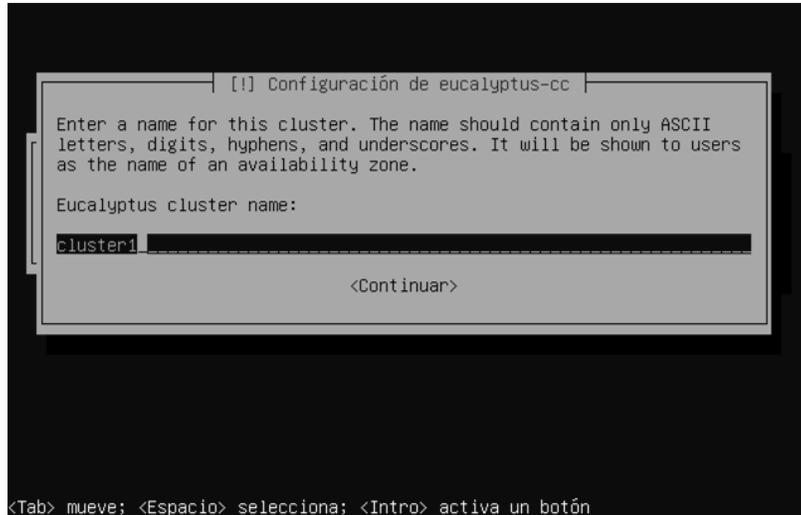
Figura 16. **Pantalla para el particionado de los discos**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

El siguiente paso es definir nombre y contraseña para un usuario administrador como en cualquier distribución Linux. Luego se deberá definir el nombre del *cluster* o lo que se conoce en Amazon Web Services como zona de disponibilidad, en el caso del ejemplo le llamaremos “*cluster 1*” (ver figura 17).

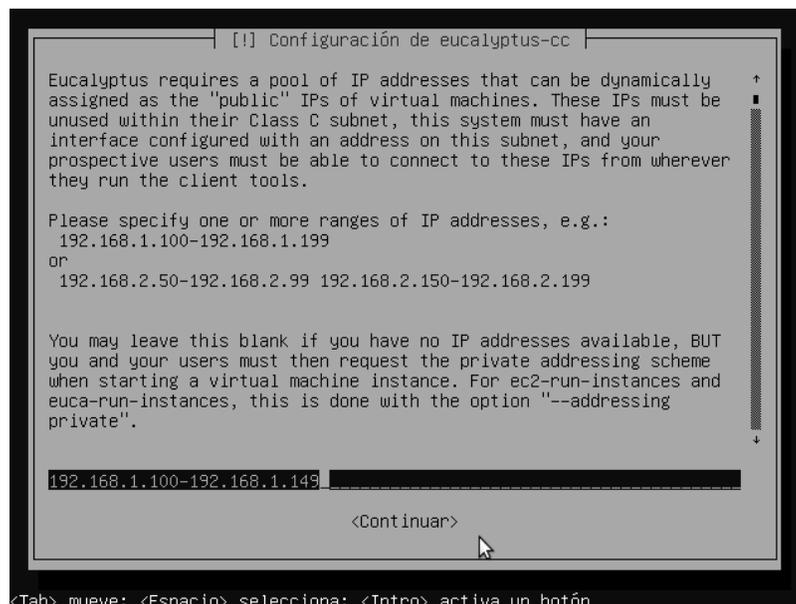
Figura 17. **Pantalla para asignarle un nombre a un *cluster* en UEC**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

Luego se podrá definir un rango para las direcciones IP que se van a asignar a las máquinas virtuales conforme se vayan instanciando (figura 18).

Figura 18. **Pantalla para definir un rango de direcciones para las instancias virtuales**



Fuente: pantalla del programa de Instalación de Ubuntu Server.

Luego de realizar los pasos anteriores, se instalará el gestor de arranque y el sistema operativo junto con los componentes controladores.

4.5.2. Instalación de los nodos

El componente a instalar es el Node *Controller*, se debe instalar en cada uno de los equipos que se utilicen como nodos. Se debe repetir el proceso de instalación en cada uno de los equipos cambiando únicamente el nombre del equipo.

El proceso de instalación de los nodos es aún más sencillo que el de los controladores, pues se omiten pasos como el nombre de *cluster* y definir las direcciones IP. Lo único que hay que verificar es que el nodo este en la misma red que el controlador. Instalar el controlador de nodos es sencillo gracias al CD de instalación de Ubuntu Server.

Por simplicidad y sabiendo que es el mismo proceso que describimos en el punto anterior se omite dicho procedimiento, pasando a explicar el registro de los nodos y el controlador.

Después que todos los equipos tienen los componentes instalados, el registro debería ser automático si se está usando el instalador de UEC, sin embargo, no siempre ocurre y es mejor conocer el proceso. Se debe intercambiar una llave pública entre los equipos para poder tener comunicación y que puedan realizar operaciones con un usuario definido en cada equipo, este usuario es "eucalyptus". Como en este caso los controladores están en un solo equipo debemos asegurarnos de la comunicación entre dicho equipo y todos los equipos nodos. Esto se hace asignándole un *password* temporal al usuario eucalyptus en cada nodo, enviarle la llave pública del controlador, para que

luego el controlador pueda acceder y tenga el control del nodo sin la necesidad de un *password*. Esto se hace con las siguientes instrucciones.

En cada nodo

```
| sudo passwd eucalyptus
```

y se define un *password* temporal.

En el controlador

```
| sudo -u eucalyptus ssh-copy-id -i  
/var/lib/eucalyptus/.ssh/id_rsa.pub eucalyptus@<ip_del_nodo>
```

se debe hacer para cada uno de los nodos.

Desactivar el *password* para el usuario “eucalyptus” en cada nodo

```
| sudo passwd -d eucalyptus
```

Y es de esta manera que los equipos podrán darle acceso a el controlador sin ningún problema.

Por último se deben iniciar, o reiniciar si ya estuvieran activos, los servicios de todos los componentes, Walrus, Cloud, *Storage* y Node. Los tres primeros están en un solo equipo, por lo tanto ejecutamos en el equipo controlador

```
| sudo start eucalyptus-walrus-publication  
| sudo start eucalyptus-cc-publication
```

```
| sudo start eucalyptus-sc-publication
```

Luego en todos los equipos nodos ejecutamos

```
| sudo start eucalyptus-nc-publication
```

y por ultimo en el equipo controlador se ejecuta la instrucción

```
| sudo start uec-component-listener
```

En este punto tanto nodos como el controlador ya están instalados, lo siguiente será empezar a administrar la plataforma.

4.6. Administración

Existen dos entornos que se deben administrar, el *cluster* de las máquinas físicas y el ambiente o plataforma virtual sobre el que van a correr las instancias que creemos.

El primero es casi autoadministrable, no se requiere mayor esfuerzo en su mantenimiento y cuenta con un entorno web para su administración. Para acceder debemos hacerlo por medio de un navegador web con la dirección IP donde está instalado el controlador y por el puerto 8443.

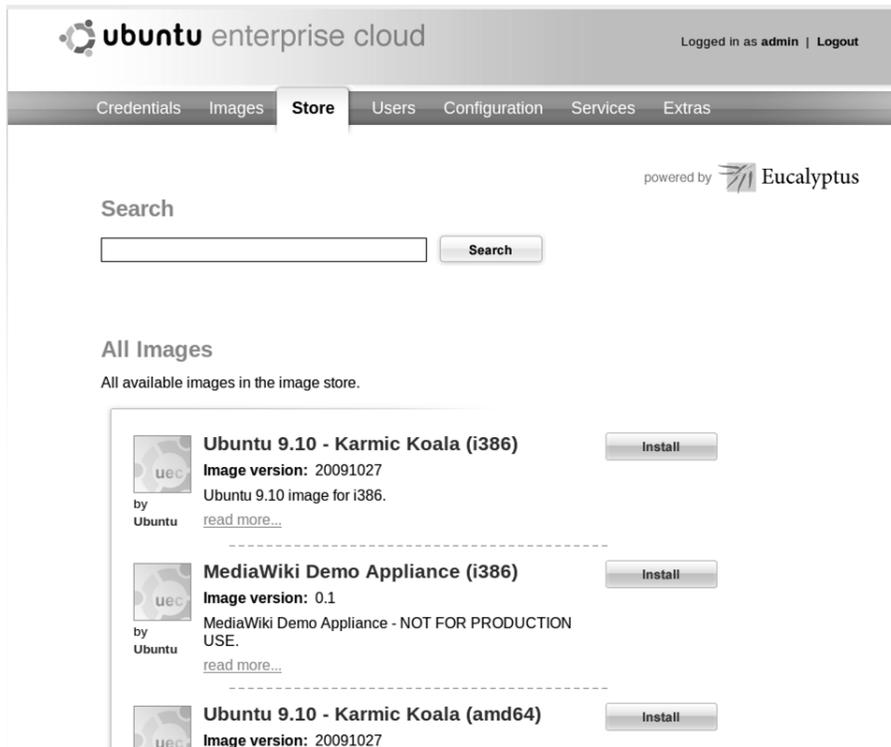
Figura 19. **Página de inicio de sesión para el sitio de administración de UEC**



Fuente: pantalla del programa de administración de Ubuntu Server.

La primera vez de uso por *default* tiene como usuario y contraseña la palabra “admin” luego ya se podrá cambiar a unas credenciales más seguras. Por medio de este sitio se podrá configurar los controladores, sus direcciones, los servicios, usuarios, imágenes y las credenciales para controlar el entorno virtual. Lo más importante será descargar las credenciales para poder acceder al controlador, para administrar el entorno virtual por medio de Euca2ool o alguna otra aplicación, como Landscape, Elasticfox, Hibridfox y otros. Así mismo por medio de este administrador se pueden adquirir imágenes en la tienda de canonical, se descargarán y estarán disponibles para poder correrlas como instancias virtuales.

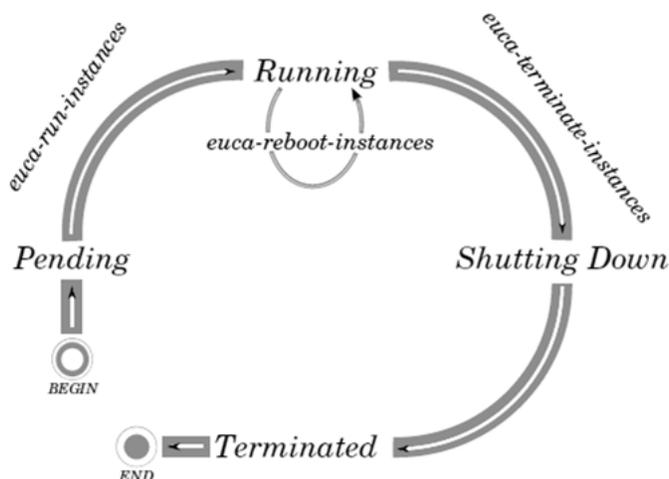
Figura 20. **Página de la tienda de imágenes de Canonical en el administrador de UEC**



Fuente: pantalla del programa de administración de Ubuntu Server.

EL segundo elemento administrable son las instancias virtuales, esto se hace con la ayuda de alguna herramienta de administración compatible con UEC como Landscape, ElasticFox, HybridFox o Euca2ools. Con la ayuda de estas herramientas se podrá controlar el ciclo de vida de las instancias virtuales.

Figura 21. **Ciclo de vida de una instancia virtual**



Fuente: Eucalyptus Beginner's Guide – UEC edition, p.5.

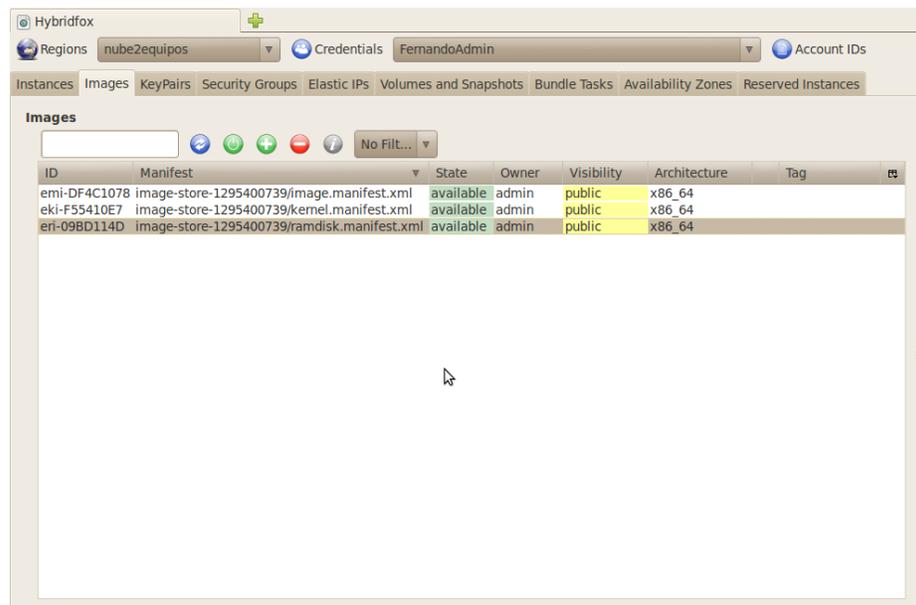
Euca2ool es la herramienta que provee eucalyptus de manera gratuita para la administración de entornos *cloud computing*, son utilidades ejecutadas desde cualquier consola Linux. ElasticFox e HybridFox son extensiones del navegador FireFox también de uso gratuito. Landscape es la herramienta provista por Canonical para el manejo de servidores y entornos *cloud computing*.

Para el ejemplo utilizaremos HybridFox ya que es la herramienta gratis, nueva y sencilla de utilizar. A través de esta extensión podremos seleccionar un *cluster* para administrar, ver la lista de imágenes disponibles, crear una instancia a partir de una imagen, asignarle recursos, etc.

Para poder utilizar Hybridfox se debe especificar la región a administrar, en este caso será la dirección IP del controlador del *cluster* por el puerto 8773 y se le asigna un nombre para un manejo más sencillo.

Luego de una autenticación exitosa se dispone de un panel con todo lo necesario para administrar las máquinas virtuales, imágenes, llaves, grupos de seguridad, etc.

Figura 24. **HybridFox en la administración de imágenes para UEC**



Fuente: pantalla del programa de administración de Ubuntu Server.

Desde esta consola es fácil registrar una imagen de una máquina virtual, instanciarla, asignarle recursos a las instancias, detenerlas, reiniciarlas, etc., todo esto de una manera simple e intuitiva.

4.7. Creación de imágenes

Ciertamente tener disponibles máquinas virtuales con distintos sistemas operativos listas para instanciar en cualquier momento es muy útil, pero es aún más útil, tener una máquina virtual personalizada para el negocio según los requerimientos de la empresa.

Como último punto se mostrará cómo crear una imagen personalizada con las aplicaciones y configuraciones necesarias para un determinado propósito. Esto quiere decir que se puede tener una máquina lista para base de datos, otra máquina servidora de correos, otra máquina lista con un ERP instalado, en fin puede haber una máquina lista para cada situación y lista para instanciarse una o varias veces, pudiendo tener en cuestión de segundos servidores listos para utilizarse.

Para crear una imagen personalizada partimos de la creación de un disco virtual, como sabemos UEC utiliza KVM como el *software* encargado de la virtualización por lo que se creará un disco virtual para KVM, debemos definir el formato que tendrá el disco y el tamaño. Qcow es un formato expandible que va ocupando solo el espacio que va necesitando, entonces se puede definir una imagen llamada image.img con 5 *gigabytes* de espacio y formato Qcow.

```
| kvm-img create -f qcow2 image.img 5G
```

El siguiente paso es instalar el sistema operativo en el disco que se acaba de crear. Para el ejemplo se puede utilizar la imagen de Ubuntu 9.04 la cual podemos descargar desde el sitio oficial de Ubuntu. Para instalarlo podemos ejecutar el siguiente comando para correr la instalación y conectarnos a ella con cualquier cliente de escritorio remoto.

```
| sudo kvm -m 256 -cdrom ubuntu-9.04-server-amd64.iso -drive  
file=image.img,if=scsi,index=0 -boot d \ -net nic -net user -  
nographic -vnc :0
```

De este modo se puede hacer la instalación del sistema operativo sobre el disco que se creó con las opciones que se requieran. Del mismo modo y después de reiniciar la máquina virtual, se le puede instalar cualquier *software*

que se necesite, ya sea por medio de un repositorio o transfiriéndole archivos de alguna aplicación propia. Se aprovecha a crear todos los scripts y configuraciones que también sean necesarias hasta que la máquina virtual quede según las necesidades del negocio.

El siguiente paso será integrar la máquina virtual que se acaba de crear con eucalyptus, el *software* manejador de la nube que ha sido implementado en UEC. Para realizar esta integración es necesario crear un script en la máquina virtual para consumir un servicio web que publica el controlador de la nube, además de copiar el archivo del *kernel* y el de disco RAM inicial que se encuentra en la carpeta *boot* de la máquina virtual y ponerla en un lugar fuera de ella, ya que servirá para crear la imagen completa.

Dentro de la máquina virtual debemos instalar Curl la cual es una herramienta para transferencia de datos que soporta http y https entre otros y agregar el siguiente script al archivo `/etc/rc.local`

```
| depmod -a
| modprobe acpihp
| mkdir -p /home/user/.ssh
| echo &gt;&gt; /home/user/.ssh/authorized_keys
| curl -m 10 -s http://169.254.169.254/latest/meta-data/public-
| keys/0/openssh-key | grep 'ssh-rsa' &gt;&gt; \
|     o /home/user/.ssh/authorized_keys
| echo &quot;AUTHORIZED_KEYS:&quot;;
| echo &quot;*****&quot;;
| cat /home/user/.ssh/authorized_keys
| echo &quot;*****&quot;;
```

Esto sirve para que cuando arranque la instancia, se copien automáticamente las llaves para el acceso autorizado. También hay que borrar un archivo que se encarga de guardar los nombres de las interfaces de red.

```
| sudo rm -rf /etc/udev/rules.d/70-persistent-net.rules
```

Y antes de apagar la máquina virtual se debe copiar los archivos del *kernel* y la RAM de arranque que se mencionaron anteriormente.

```
| scp /boot/initrd.img-2.6.28-11-server usuario@A.B.C.D:  
| scp /boot/vmlinuz-2.6.28-11-server usuario@A.B.C.D:
```

Ya en una máquina cliente se podrán crear los archivos para empaquetar, subir y registrar en la nube. Este proceso se debe hacer para el archivo del *kernel*, el archivo de la RAM inicial y la imagen del disco que creamos.

Para el kernel

```
| euca-bundle-image -i vmlinuz-2.6.28-11-server --kernel true  
| euca-upload-bundle -b mybucket -m /tmp/vmlinuz-2.6.28-11-  
server.manifest.xml  
| euca-register mybucket/vmlinuz-2.6.28-11-server.manifest.xml
```

Este comando produce un texto de salida que se necesita para registrar la máquina virtual. El texto tendrá un formato “eki-XXXXXXXX” donde cada X representa un número generado por el programa para identificar al nuevo kernel.

Para el disco RAM

```
| euca-bundle-image -i initrd.img-2.6.28-11-server  
| euca-upload-bundle -b mybucket -m /tmp/initrd.img-2.6.28-11-  
server.manifest.xml  
| euca-register mybucket/initrd.img-2.6.28-11-server.manifest.xml
```

Este comando también produce un texto de salida que se necesita para registrar la máquina virtual. El texto tendrá un formato “eri-XXXXXXXX” donde cada X representa un número generado por el programa para identificar la nueva imagen del RAM.

Para registrar la imagen del disco

```
| euca-bundle-image -i image.img --kernel eki-XXXXXXXX --ramdisk  
eri-XXXXXXXX  
| euca-upload-bundle -b mybucket -m /tmp/image.img.manifest.xml  
| euca-register mybucket/image.img.manifest.xml
```

Al realizar los pasos anteriores ya tendremos una máquina virtual almacenada en el *Walrus* de la nube lista para poder instanciarla.

5. VENTAJAS Y DESVENTAJAS DEL *CLOUD COMPUTING* ANTE INFRAESTRUCTURA DEDICADA TRADICIONAL

5.1. Infraestructura tradicional de servidores dedicados

Desde hace muchos años se ha utilizado el esquema cliente servidor, y desde entonces se han usado máquinas con recursos suficientes para poder servir aplicaciones y datos. Las empresas han creado centros de datos para almacenar estos servidores que albergan sus datos y sus aplicaciones. Cada servidor es un equipo independiente que cuenta con cierta cantidad de memoria RAM, una cantidad de espacio para almacenamiento y la potencia de sus procesadores.

Por lo general se escoge el equipo y sus características según el tipo de *software* que va a residir en él, el uso que se le va a dar y la cantidad de usuario que vayan a utilizar sus recursos. En grandes empresas se requieren de servidores potentes con muchos recursos computacionales, muchas veces especializados para esta tarea, ya que esperan un número alto de usuarios de forma simultánea. En cambio una empresa pequeña, con menos usuarios puede usar un equipo común y de uso cotidiano pues contará con un número menor de usuarios concurrentes.

Al tratar de mejorar la calidad de un servicio con este tipo de infraestructura, se ha llegado hasta el punto de hacer réplicas para hacerlas de alta disponibilidad, pero esto implica comprar dos equipos con las mismas o casi las mismas características.

Conforme las necesidades de las empresas van requiriendo, se compran nuevos equipos con mayor capacidad o se trata de mejorar los ya existentes hasta donde se puede. De igual manera conforme se van creando nuevos sistemas de *software* útiles para la empresa se debe comprar nuevo equipo e integrarlo a la red de servidores.

Se deben hacer *backups* periódicos para que la información esté íntegra y segura. Se debe mantener los equipos en buen estado, con todas la bondades de un *data center* que ya hemos visto en capítulos anteriores. Estos equipos deben ser mantenidos y monitorizados por personal autorizado y capacitado.

Al usar esta infraestructura por lo general se cuenta con un servidor para cada sistema o aplicación de la empresa, un servidor para los correos electrónicos, un servidor para la base de datos, un servidor para cada aplicación como por ejemplo un CRM o un ERP o un Sistema de *software* a la medida de la empresa.

5.2. El Cloud Computing sobre la infraestructura tradicional

Para poder diferenciar la ventajas y desventajas vamos a tratar de verlo desde el punto de vista de varios ambientes empresariales simulados, de esta manera se podrá comparar los resultados en cuanto a rapidez, trabajo y dinero.

Supongamos que existen tres empresas que hacen uso de varios sistemas de información. Una de ellas tiene la infraestructura tradicional de servidores dedicados (Empresa 1), otra empresa ha contratado a un tercero como proveedor de infraestructura (Empresa 2) y la tercer empresa ha creado una plataforma de *cloud computing* privado (Empresa 3). Veamos cómo proceder en distintos escenarios.

5.2.1. Desarrollo

Supongamos que se está desarrollando un nuevo sistema o una nueva funcionalidad para un sistema existente. El proceso de desarrollo requiere de cierta infraestructura para que se realice de buena manera. Se necesita un ambiente de desarrollo completo, un servidor de base de datos, un repositorio para las fuentes y un servidor para el manejo de la configuración además de los entornos de trabajo para los desarrolladores. Como esta actividad será temporal basta con un nuevo servidor donde se instale la base de datos, el repositorio de las fuentes y las herramientas de manejo de configuración.

La empresa 1 deberá comprar un nuevo equipo, que aunque cuente con pocos recursos representa un gasto que posiblemente no será aprovechado al 100% pues el tiempo de desarrollo es temporal. El nuevo equipo deberá soportar un manejador de base de datos y las herramientas necesarias para el desarrollo, además, deberá ser instalado y configurado por personal calificado, esto requerirá un tiempo que va desde la cotización hasta la puesta del servidor en el *data center*.

La empresa 2 deberá alquilar a su proveedor un servidor con lo requerido por el tiempo previsto que dure el desarrollo, el servidor estará disponible inmediatamente y será necesario solo instalar las herramientas para el desarrollo.

La empresa 3 posiblemente tendrá entre sus imágenes una lista para desarrollo la cual podrá ser puesta en marcha inmediatamente. Utilizando los recursos de la nube completa y utilizará solo lo que se crea conveniente. Si se desarrolla en temporada baja se podrá utilizar recursos de otros sistemas.

5.2.2. Pruebas

Luego de cualquier desarrollo es necesario realizar pruebas para poder certificar que el *software* cumpla los propósitos para los que fue hecho, estas pruebas se deben realizar en un ambiente muy parecido a producción para lograr simular los escenarios reales durante el funcionamiento de la aplicación.

La empresa 1 deberá comprar o conseguir un servidor donde se pueda instalar la nueva aplicación y posiblemente una base de datos de tal manera que el nuevo ambiente se asemeje al ambiente de producción que se espera tener o que ya está en uso, esto representa dinero por la compra del *hardware*, tiempo en conseguir, instalar y configurar el nuevo servidor, meterlo a la red, restaurar *backup* de producción para que simule dicho ambiente. Después de hacer las pruebas y de hacer el gasto de tiempo, trabajo y dinero queda un servidor ocioso hasta que se vuelvan a necesitar hacer pruebas siempre y cuando este soporte los nuevos requerimientos.

La empresa 2 simplemente tendrá que alquilar recursos computacionales y cargar una imagen copia de producción si también tiene su ambiente de producción en la nube. Esto lo estará pagando únicamente por el tiempo que duren las pruebas y la puesta en marcha de dicho servidor será inmediata.

La empresa 3, si tiene recursos disponibles en su infraestructura, podrá hacer una copia de su máquina de producción y luego de unas mínimas configuraciones empezará con las pruebas. De no contar con recursos podrá prestar a alguna instancia que no sea crítica o que este ociosa únicamente por el tiempo que duren las pruebas.

5.2.3. Instalación en el ambiente de producción

Instalar o actualizar una aplicación en un ambiente que ya esté en uso por usuarios es una tarea un tanto crítica, máximo si el negocio requiere de alta disponibilidad.

En la empresa 1 se debe definir un lapso de tiempo en el cual el sistema no esté disponible y tomar en cuenta que si ocurre algún error, el tiempo de caída será mayor. Se tendrá que hacer un *backup* previo a la instalación/actualización el cual consumirá tiempo y recursos.

En la empresa 2 con poco dinero se podrá hacer una copia de la imagen en producción y asignarle recursos para modificarla sin que la original tenga que parar de ofrecer el servicio, el cambio se puede hacer casi de inmediato.

En la empresa 3 de igual manera se creará una instancia que es copia del servidor de producción y se harán los cambios en dicho servidor, cuando estos estén listos se hará el cambio de las máquinas virtuales y quedará la anterior instancia como un respaldo del sistema anterior.

5.2.4. Reversión

En el caso de que una actualización o instalación falle y el sistema se encuentre en un estado inconsistente, se debe regresar a su estado previo cuando todo funcionaba bien. Para realizar esto se debe contar con un proceso definido a realizar para que el tiempo de inconsistencia del sistema sea mínimo.

En el caso de la empresa 1, antes de hacer la actualización o instalación se debe hacer una copia de respaldo de todo el sistema o un punto de

restauración dependiendo del *software* con que cuente el equipo tratando de no afectar las operaciones, entonces por lo general se debe planificar en un horario donde no hayan usuarios o el nivel de trabajo sea mínimo. Si la instalación presenta problemas y se debe hacer la reversión, se deberán deshacer los cambios hechos y restaurar el sistema hacia el punto donde todo estaba bien.

En el caso de la empresa 2 puede simplemente apagar la instancia que está dando problemas y a partir de una imagen predefinida instanciar un servidor con el sistema funcionando bien.

Lo mismo que se hizo con la empresa 2 se haría en la empresa 3, con la diferencia que la empresa 3 ya tiene sus recursos adquiridos y no debe pagar por su uso como haría la empresa 2.

5.2.5. Temporadas altas para el negocio

En los negocios existen temporadas en las que venden más, o sus cargas operacionales son más intensas, o simplemente el sistema será utilizado por más usuarios. Durante estas épocas los servidores se ven afectados y muchas veces no están capacitados para soportar dicho funcionamiento, por lo que los servicios se caen afectando al negocio. La solución sería aumentar la capacidad de estos servidores durante estas épocas.

La empresa 1 tendría que comprar nuevos servidores dependiendo de la carga esperada, los deben configurar y agregar al *data center*. Por lo que se requiere de inversión de tiempo y dinero. Luego de la temporada de carga los nuevos servidores quedarían ociosos, significando una pérdida de dinero y recursos a menos que se formatearan o se reconfiguraran para realizar otras tareas lo que significaría más tiempo y trabajo.

En la empresa 2 se debería alquilar más recursos en la época de carga alta, es sumamente rápido hacer la mejora de recursos, incluso existen procedimientos automáticos, se paga solo por lo que se consume por lo que representaría una solución muy eficiente y barata, luego de la época de carga simplemente se liberan los recursos y ya no se debe pagar por ellos.

En la empresa 3 aunque el procedimiento puede ser tan sencillo como el de la empresa 2, los recursos deben estar dentro de la empresa por lo que podría generar cierto gasto, la diferencia y ventaja sobre la empresa 1 es que la configuración es sumamente sencilla y luego de la temporada de carga los recursos se pueden asignar de una forma sencilla a cualquier otra tarea.

5.2.6. Adquisición de nuevo *hardware*

El *hardware* es un bien que se deprecia y se desactualiza de una manera muy rápida, cada cierto tiempo será necesario renovar parte de los equipos que están en el *data center*, ya sea por nuevas tecnologías o por la carga operacional.

En la empresa 1 siempre se compran nuevos equipos con más y mejores recursos haciendo a un lado los equipos viejos, puede ser que los donen o los tiren a la basura pero representan una pérdida. Los nuevos equipos se deberán configurar y agregar al *data center* lo cual lleva un proceso que va desde la cotización de los mismos hasta la puesta en producción.

En la empresa 2 no hay necesidad de comprar *hardware*, todos los sistemas están en una nube de terceros, por lo que son estos los que nos proveerán de los recursos computacionales que se necesiten, se va a pagar

únicamente por lo que se consume y en el momento que ya no se necesite se deja de pagar.

En la empresa 3 aunque también se necesitará comprar *hardware* se usará de una forma más inteligente, ya que se le podrá dar un uso generalizado para cualquier sistema y no para uno en específico. Además tenemos la ventaja de poder utilizar el *hardware* antiguo fusionándolo con el nuevo y de esta manera aprovechar todos los recursos disponibles; no se necesita de tiempo para hacer el cambio, simplemente se añade al *cluster* y los recursos están disponibles.

5.2.7. Averías en *Hardware*

El *hardware* puede fallar en cualquier momento y no importando la causa del fallo, la forma de cómo responder a este suceso debe ser un aspecto importante para los encargados de los sistemas.

Si el *hardware* tuviera una caída inesperada en la empresa 1, el servicio que ofrece dicho sistema no estará disponible hasta que logren arreglar el equipo afectado, ya que los sistemas corren sobre el propio *hardware* y si este no funciona, no se podrá prestar el servicio, en el mejor de los casos tendrán un equipo como respaldo y en el peor de los casos si el equipo no se puede arreglar deberán usar un nuevo equipo y restaurar el último backup.

En la empresa 2 no debe existir dicho problema, ya que no tienen responsabilidad alguna con *hardware*, el proveedor del servicio del *cloud computing* es el responsable de mantener la plataforma.

En la empresa 3 un equipo averiado no es indispensable si se cuenta con un *cluster* suficiente de equipos, bastará con retirar el equipo averiado y agregar uno que si funcione, los sistemas no sufren daños porque corren sobre la plataforma de virtualización.

5.2.8. Un nuevo *software* o producto

Cuando se adquiere un nuevo *software* se deberá instalar en un nuevo servidor, sin embargo las instalaciones dependiendo del proveedor y del tipo de sistema pueden ser un tanto difíciles de realizar, se puede requerir incluso soporte de parte del proveedor.

En la empresa 1 teniendo un nuevo servidor listo, se procederá a la instalación del nuevo sistema, teniendo el debido cuidado de documentar cada paso de la instalación, más si es parte del soporte que lo está realizando. Entonces existirá un script que puede guiar al personal para las instalaciones posteriores. Sin embargo en algún momento puede haber problemas y nuevamente habrá que llamar al soporte lo que representa pérdida en tiempo y dinero.

En la empresa 2 y 3 con su plataforma virtualizada se deberá instalar únicamente una vez, ya que se creará una máquina virtual que quedará almacenada y lista para crear instancias, si en algún momento se requiere un servidor con el sistema, ya no se tendrá que volver a realizar el proceso de instalación, simplemente crear una instancia de dicha máquina y listo.

5.2.9. La caída de un servicio

El *software* también es propenso a errores y puede dejar de ofrecer su servicio en cualquier momento.

La empresa 1 al darse cuenta que el *software* ha dejado de funcionar tendrá que revisar inmediatamente que está causando el error y hasta que no lo encuentre y lo resuelva, el servicio no estará disponible.

Tanto la empresa 2 como la empresa 3 van a tener imágenes personalizadas de su aplicación y en el momento que la instancia falle podrán cambiarla por una nueva instancia de la imagen previamente probada.

5.3. Ventajas sobre la infraestructura tradicional

El *cloud computing* es económico en casi todos los aspectos, el ahorro se percibe mejor cuando estamos en una situación empresarial con más de 3 servidores. Se ahorra en *hardware*, en electricidad, en enfriamiento, en mantenimiento, en actualizaciones, en personal, en espacio, en cables, etc. Máxime si hablamos de *cloud computing* público, pues los servicios son muy baratos y se paga solo por lo que realmente se usa. Aún el *cloud computing* privado representa ahorros y optimización sobre la infraestructura tradicional.

La flexibilidad es una de las ventajas más importantes pues el *cloud computing* permite manejar los recursos disponibles de la manera que sea necesaria, no hay límite ni restricciones. Se puede utilizar pocos servidores con muchos recursos o muchos servidores con pocos recursos, se define de acuerdo a las necesidades del negocio y mejor aún en *cloud computing* público pues da la idea de tener disponible una cantidad de recursos ilimitados.

Además brinda flexibilidad a la hora de duplicar, sustituir, borrar, respaldar y mover máquinas virtuales. Gracias al *cloud computing* se puede gozar de una plataforma elástica, podemos hacer crecer la plataforma al ritmo que el negocio lo requiera y de una forma dinámica.

Con el *cloud computing* cualquier tarea se hace mucho más rápido que con la infraestructura tradicional. Facilidad de implementación, tanto de los sistemas, como de la misma plataforma *cloud computing*. Existen cantidad de herramientas fáciles de instalar y administrar que proveen ambientes *cloud computing* optimizados y también están los proveedores de *cloud computing* público los cuales proveen una plataforma lista para utilizarse.

Con *cloud computing* se requiere menos *hardware* y en el *cloud computing* público no necesita *hardware* en lo absoluto por lo que aparte del ahorro también ofrece facilidad y rapidez para crear y mantener estas plataformas. Podemos decir entonces que no se necesita de una inversión inicial en el *cloud computing* público y una inversión mínima en el *cloud computing* privado.

Se puede acceder a la plataforma virtual de una manera más sencilla y desde cualquier lugar a través de *Internet*. El cloud computing impacta todo el negocio y no solo los costos de IT, todos los beneficios se reflejan en el negocio.

5.4. Desventajas sobre la infraestructura tradicional

Si hablamos de *cloud computing* público, las desventajas son como ya mencionamos en algún punto de este documento, la relación con el proveedor, la seguridad de los datos, la dependencia de una conexión de *Internet* y la

disponibilidad que pueda ofrecer el proveedor. Pero dadas estas desventajas del *cloud computing* público se ha pensado en el *cloud computing* privado, el cual viene a dar la solución a estas desventajas, sin embargo también hace que las ventajas del *cloud computing* público aminoren en cierto grado pero aún así trae mejores resultados que la infraestructura tradicional.

Las que son ventajas en el *cloud computing* público ahora en el privado puede que ya no lo sean, por ejemplo hay una inversión inicial, gasto de mantenimiento, el retorno de la inversión es más lento, la disponibilidad, los respaldos y la seguridad ya son responsabilidad de los trabajadores de la empresa, etc. Y por último se requiere que los administradores aprendan sobre estas nuevas tecnologías.

5.5. Cuándo utilizar cloud computing público

El *cloud computing* público es barato, eficiente, flexible y elástico, pero todavía crea cierta desconfianza por la seguridad de los datos y los sistemas, además del inconveniente de depender de una conexión fija a *Internet*.

El *cloud computing* público se debería usar cuando no existe el suficiente dinero para invertir en *hardware*, el negocio es nuevo por lo que todavía no se sabe la demanda que tendrán los sistemas ni los datos, los sistemas son críticos, no contamos con las instalaciones ni con el personal para mantenimiento de *hardware* y se requiere de rapidez y accesibilidad desde cualquier parte.

Lo anterior hace pensar que el *cloud computing* público es la elección perfecta para los emprendedores, creadores de nuevas empresas que se aventuran en el negocio sin mucho capital. También será de mucho beneficio

para cualquier negocio de cualquier tamaño que quiera ahorrar en tecnología siempre y cuando esté de acuerdo con que los datos estén en los servidores de un tercero. Los negocios basados en *Internet* son candidatos para el uso de *cloud computing* ya que aparte de accesibilidad, el ancho de banda y la escalabilidad se pueden manejar de inmediato dependiendo de la demanda. Y por último, también son candidatas las empresas que no necesitan de mucha integración, pues en la nube es más difícil manejar varios sistemas y tratar que se integren. Si la empresa utiliza sistemas autónomos el *cloud computing* es buena opción en sus distintos tipos.

5.6. Cuándo utilizar cloud computing híbrido

El *cloud computing* híbrido viene a resolver las desventajas que presenta el *cloud computing* público, ya que las partes críticas de los sistemas y los datos quedan detrás del *firewall* de la empresa, con la única desventaja que habrá que trabajar en integración entre la parte que estará en el lado público y la parte que quedará dentro de la empresa.

Es útil para empresas con personal capacitado que conozca el funcionamiento de sus sistemas y del negocio, que puedan dividir sus datos y sistemas para poder aprovechar las dos clases de *cloud computing*. Obviamente se debe estudiar si los sistemas y los datos se pueden dividir para estar en ambientes distintos y si esta división no representaría más gastos o problemas que los que se pretenden evitar con el *cloud computing*.

Se recomienda para empresas que necesitan ahorrar un poco de dinero pero que no pueden poner en riesgo la información propia o de sus clientes.

5.7. Cuándo utilizar cloud computing privado

Cuando los datos del negocio son realmente importantes y críticos, se cuenta con suficientes recursos y hay personal capacitado.

Existirán empresas en que sus sistemas y datos son de carácter confidencial, y en las que hacer una integración como en la nube híbridas no es suficiente o no es funcional, para estas empresas es bueno utilizar el *cloud computing* privado, ya que pueden reducir gastos y gozar de las ventajas de la plataforma virtual pero teniendo toda la información dentro de sus dominios.

5.8. Cuándo utilizar servidores dedicados tradicionales

Puede ocurrir que en algún momento una empresa pequeña cuente con uno o dos equipos listos para usarse y necesita iniciar operaciones con sistemas sencillos de ventas o para un sitio informativo sin mucha carga operacional, además que difícilmente contará con un grupo de personas encargadas de tecnología. Para estas situaciones no será necesario alquilar o crear una plataforma virtualizada ya que se cuenta con el *hardware* necesario y no se prevé aumento en la carga operacional, además el negocio no es crítico. Obviamente esta plataforma estará limitada a sus pocos recursos y no tendrá las ventajas que ya hemos visto del *cloud computing*.

CONCLUSIONES

1. Crear una plataforma *cloud computing* es muy sencillo, gracias a las herramientas y *software* especializado para este fin, se pueden crear plataformas virtuales que permiten flexibilidad y escalabilidad en tiempos muy cortos. Herramientas como Eucalyptus, Ubuntu Enterprise Cloud, Open Nebula, VMWare, Citrix y otros; hacen posible que personas con conocimientos intermedios puedan crear nubes privadas listas para soportar y administrar máquinas e instancias virtuales.
2. Las ventajas del cloud computing sobre la infraestructura de servidores tradicionales son numerosas. El ahorro económico, la flexibilidad, la escalabilidad y la fácil administración hacen que usar una plataforma *cloud computing* sea mucho mejor que la infraestructura tradicional, siempre y cuando el negocio tenga la visión de ir creciendo y mejorando.
3. El *cloud computing* consiste en combinar tecnologías como la computación distribuida, la virtualización, la arquitectura orientada a servicios y ofrecer un nuevo paradigma para la administración, almacenamiento y distribución de los datos y los sistemas de cómputo de las empresas, haciendo hincapié en el aprovechamiento, extensibilidad y distribución de los recursos.
4. El *cloud computing* público ahorra mucho dinero y presenta muchas ventajas, pero tiene ciertos puntos con los que las personas todavía no están conformes, como la seguridad de los datos, la dependencia de la conexión y la confianza del proveedor. Para suplir esas desventajas esta

el *cloud computing* privado, el cual permite crear una nube detrás del *firewall* de la empresa.

5. Según el capital con que se cuente, la seguridad de los datos y sistemas, el nivel de capacitación del personal, el número de servicios a utilizar, y la complejidad del negocio, se puede seleccionar ya sea *cloud computing* público, privado, híbrido o incluso quedarse con la infraestructura tradicional. Cada empresa deberá analizar las ventajas y desventajas de estas tecnologías según su situación y buscar la que más les convenga para reducir costos y aumentar las ganancias.
6. Esta tecnología está en su apogeo y existen muchos proveedores de *cloud computing* público y privado, cada uno ofrece distintos niveles de *cloud computing* con agregados extras que hacen aun más atractivo utilizar este servicio.
7. El *cloud computing* en Guatemala, todavía no tiene la demanda que ha tenido en otros países. Se está utilizando para uso personal con servicios como e-mail, almacenamiento, música, planeación, utilizando ampliamente el *software* como servicio, pero aún no se usa para las empresas y esto es debido a la desconfianza que se tiene sobre el tema y de las desventajas que en nuestro país pueden ser más marcadas, como la dependencia de una conexión a *Internet*, la disponibilidad de estas conexiones, los aspectos legales, los acuerdos de servicio y las velocidades de las conexiones.

RECOMENDACIONES

1. Investigar más esta nueva tecnología, hacer el ejemplo propuesto en este documento, ya que las herramientas son libres y fáciles de conseguir; practicar y experimentar para poder ahondar más en el tema y tener una mejor comprensión para saber si realmente conviene utilizar el *cloud computing*.
2. Al iniciar cualquier proyecto, negocio o emprendimiento donde se necesite de recursos computacionales y no se cuente con el capital necesario, se puede utilizar *cloud computing* ya que se puede iniciar con pocos recursos y conforme el proyecto crezca se podrá ir aumentando y se paga conforme el proyecto vaya generando los fondos.
3. Antes de migrar a una plataforma *cloud computing* se recomienda estudiar bastante sobre el tema, analizar y seleccionar las herramientas adecuadas, aprender a usarlas y crear un ambiente de pruebas para simular la nube y ya cuando se esté completamente seguro, entonces se podrá realizar la migración.
4. Probar el *cloud computing* público es barato, se paga solo por lo que se consume, se experimenta por un día o un par de horas y solo se paga por dicho uso, no se necesita contar con ningún tipo de *hardware* y será un excelente laboratorio para poder aprender más.
5. Cada vez que aparece una nueva tecnología es bueno investigar un poco ya que esta nueva tecnología nos podría ser de utilidad. El permanecer

estáticos en este mundo de avances tecnológicos no es buena idea pues nuevas tecnologías aparecen cada cierto tiempo con el fin de facilitar nuestras tareas.

BIBLIOGRAFÍA

1. BANERJEE, Prith, et al. "Everything as a service: powering the new information economy". *Computer - IEEE Computer Society*, 2011. No.44, p. 36-43.
2. BOTELHO, Bridget. *Hardware standardization benefits cloud computing, virtualization* [en línea]. Searchcloudcomputing.com. 29 de abril de 2009. [ref. de 08 de noviembre 2010]. Disponible en Web: <<http://searchcloudcomputing.techtarget.com/news/1355038/Hardware-standardization-benefits-cloud-computing-virtualization>>.
3. CALIDI, Yousef. "Building a cloud computing platform for new possibilities". *Computer - IEEE Computer Society*, 2011. No.44, p. 29-34.
4. CANO, Jeimy. *Cloud computing: evolución en los servicios de tecnología y grandes retos en seguridad* [en línea]. Channel planet. 14 de abril de 2010. [ref. de 12 de octubre 2010]. Disponible en Web: <<http://www.channelplanet.com/?idcategoria=23693>>.
5. Cisco Systems, Inc. *Cisco Data Center Infrastructure 2.5 Design Guide* [en línea]. 6 de diciembre de 2007. [ref. de 12 de enero 2011]. Disponible en Web: <http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c649/ccmigration_09186a008073377d.pdf>.

6. FALLA, Stephanie. *Cloud Computing: nueva era de desarrollo maestros del web* [en línea]. 12 de noviembre de 2008 [ref. de 02 de octubre de 2010]. Disponible en Web: <<http://www.maestrosdelweb.com/editorial/cloud-computing-nueva-era-de-desarrollo/>>.
7. FLOYER, David. *Private Cloud is more cost effective than Public Cloud for Organizations over \$1B*. Wikibon [en línea]. 13 de diciembre de 2010 [ref. de 20 de enero de 2011]. Disponible en Web: <[http://wikibon.org/wiki/v/Private_Cloud_is_more_Cost_Effective_than_Public_Cloud_for_Organizations_over_\\$1B](http://wikibon.org/wiki/v/Private_Cloud_is_more_Cost_Effective_than_Public_Cloud_for_Organizations_over_$1B)>.
8. GALLI, Ricardo. *Cómo montamos Menéame en Amazon EC2 - Ricardo Galli, de software libre* [en línea]. 30 de diciembre de 2009 [ref. de 11 de octubre de 2010]. Disponible en Web: <<http://gallir.wordpress.com/2009/12/30/como-montamos-meneame-en-amazon-ec2/>>.
9. GREENBERG, Albert, et al. *The cost of a cloud: research problems in data center networks*. ACM SIGCOMM Computer Communication Review, 2009. p. 68-73.
10. JANSEN, Wayne; GRANCE, Timothy. *Guidelines on security and privacy in public cloud computing* [en línea] enero de 2011. [ref. de 12 de febrero de 2010]. Disponible en Web: <https://wiki.ma.jackpinetech.com/sandbox/users/hopkins/weblog/c0833/attachments/1a5f5/Draft-SP-800-144_cloud-computing.pdf?sessionID=104f39e5b9affc6a93482a932a033f79896d9ac1>.

11. JOHNSON, D, et al. *Eucalyptus Beginner's Guide - UEC Edition* [en línea]. 25 de mayo de 2010. [ref. de 12 de enero 2011]. Disponible en Web: <http://cssoss.files.wordpress.com/2010/06/book_eucalyptus_beginners_guide_uec_edition1.pdf>
12. JONES, Tim. *Virtual Linux IBM* [en línea]. 29 de diciembre de 2006 [ref. de 02 de diciembre de 2010] . Disponible en Web: <<http://www.ibm.com/developerworks/linux/library/l-linuxvirt/>>
13. LEON, Ernest. *The five layers within Cloud Computing* [en línea]. 06 de diciembre de 2009 [ref. de 05 de septiembre de 2010]. Disponible en Web: <<http://cloudcomputing.sys-con.com/node/1200642>>.
14. MALCOLM, Dave. *The five defining characteristics of cloud computing* [en línea]. *ZDNet*. 09 de abril de 2009 [ref. de 02 de octubre de 2010]. Disponible en Web: <<http://www.zdnet.com/news/the-five-defining-characteristics-of-cloud-computing/287001>>.
15. NARASIMHAN, Balakrishna; NICHOLS, Ryan. "State of cloud applications and platforms: the cloud adopters' view". *Computer - IEEE Computer Society*, 2011. No.44, p. 24-28.
16. ROCHWERGER, Benny, et al. "Reservoir - When One Cloud Is Not Enough". *Computer - IEEE Computer Society*, 2011. No.44, p. 44-51.
17. SAAVEDRA, Esteban. *Cloud computing* [en línea]. Abril de 2009 [ref. de 02 de octubre de 2010]. Disponible en Web: <<http://www.slideshare.net/estebansaavedra/introduccion-al-cloud-computing>>.

18. STRICKLAND, Jonathan. *How cloud computing work HowStuffWorks.com* [en línea]. 08 de abril de 2008 [ref. de 12 de septiembre de 2010]. Disponible en Web: <<http://communication.howstuffworks.com/cloud-computing.htm/printable>>.
19. VASAN, Robin. "A Venture Perspective on Cloud Computing". *Computer - IEEE Computer Society*, 2011. No.44, p. 60-62.