

CLOUD COMPUTING COMO UNA RED DE SERVICIOS

Luis Fernando Espino Barrios
Instituto Tecnológico de Costa Rica
luisespino@yahoo.com

Reporte Técnico MC-6006-200869282
Noviembre 2009

Resumen: En este reporte técnico se presenta *Cloud Computing* como una red de servicios, desde una perspectiva que ayude a comprender de mejor manera los conceptos asociados. *Cloud Computing* ha evolucionado en los últimos años debido a la heterogeneidad del hardware y del software, y no se debe confundir con *Grid Computing* que es una combinación de recursos o con *Utility Computing* que está orientado a un servicio específico, aunque ambos pueden complementarse con *Cloud Computing*. Por lo anterior en este reporte se contrastan dichos conceptos para notar las diferencias. También, se presentan la arquitectura, la infraestructura y las actuales tendencias de *Cloud Computing*, expuestos desde el punto de vista de red.

Palabras claves: *Cloud Computing*, *Cloud Networking*, *Grid Computing*, *Datacenter*, *Utility Computing*, *Inter Cloud*, Virtualización.

0. Introducción

Cloud Computing provee servicios por un tercero a través de internet, de allí la metáfora de *Cloud*, dichos servicios son proveídos en cuanto sean requeridos en ambientes distribuidos, disponibles, con menor complejidad y a gran escala. *Cloud Computing* proporciona herramientas de software, plataformas e infraestructura como servicios, a saber, SaaS, PaaS, IaaS. El enfoque de este reporte técnico es estudiar el concepto de *Cloud Computing* como una red servicios desde una perspectiva comprensiva, además de definir cuál sería la manera idónea de implementarla, teniendo en cuenta aspectos como el diseño, los modelos de red y los protocolos de comunicación. Además, se tratan algunos modelos propuestos de arquitectura que varían de acuerdo al proveedor, generando con esto ambientes heterogéneos que hacen compleja la interconexión. La infraestructura física se implementa generalmente a través de *Datacenters* que según el tipo podrían ser *Public* o *Private Cloud*, también se utiliza en gran parte la virtualización debido a la escalabilidad y disponibilidad que presenta. Además de estos beneficios la virtualización provee una propiedad importante que es el “aislamiento”, sin embargo, no hay que dejar atrás aspectos por un lado como la seguridad y por otro lado la definición de estándares para interconectar *Clouds*. Hay dos enfoques derivados de *Cloud Computing*, uno es el *Cloud Networking* que forma parte de la base de infraestructura, también está el enfoque *Inter Cloud* que básicamente trata temas relacionados a la interconexión de *Clouds*.

0.1. ¿Qué es *Cloud Computing*?

Cloud Computing [1] se refiere por una parte a las aplicaciones entregadas como servicios sobre internet y por otra parte al hardware en *Datacenters* que proveen estos servicios. También en [2] se menciona que *Cloud Computing* es la convergencia y evolución de muchos conceptos de virtualización, de aplicaciones distribuidas, de *Grids* que habilitan un enfoque flexible para el despliegue y del escalado de aplicaciones [3]. *Cloud Computing* es un término utilizado para describir una plataforma y el tipo de aplicación [4], siendo una plataforma que dinámicamente aprovisiona, configura y reconfigura servidores como sea necesario.

0.2. Evolución de *Cloud Computing*

El término *Cloud Computing* no es nuevo, ni revolucionario, sino que es un término que ha venido evolucionando con el tiempo [5]. Empezó en los años 1980's bajo conceptos de *Grid Computing*, aunque con ciertas diferencias y enfatizado a servidores virtuales; luego en los años 1990's se expandió el concepto de virtualización elevando el nivel de abstracción de los servidores virtuales, primero como plataforma virtual y luego como aplicaciones virtuales; más adelante se conoció el término *Utility Computing*, que ofrece *clusters* como plataformas virtuales; recientemente el término software como servicio (SaaS) elevando el nivel de virtualización a las aplicaciones, con un modelo de negocio no recargado en recursos consumidos.

Debido a esta evolución el concepto de *Cloud Computing* combina los términos anteriores de *Grid*, *Utility* y SaaS, siendo un modelo emergente en donde los usuarios pueden tener acceso a las aplicaciones desde cualquier lugar a través de dispositivos conectados.

0.3. Beneficios y riesgos

En [6] se mencionan los siguientes beneficios de *Cloud Computing*:

- Reduce el costo total de propiedad: debido a que en toda organización ese es uno de los principales objetivos, minimización de costos en hardware y software.
- Incrementa la escalabilidad y fiabilidad: debido que a través de diferentes infraestructuras se puede habilitar lo escalable y fiable que pueden llegar a ser los sistemas bajo conceptos de *Cloud Computing*.
- Además, existen varios beneficios que también puede actuar como riesgos dependiendo si la implementación se realizó bajo los pasos sugeridos, entre estos se pueden mencionar la disponibilidad, el rendimiento, la capacidad, etc.

Por otra parte en [7] se mencionan beneficios referentes al desarrollo y despliegue:

- Reduce el tiempo de ejecución y el tiempo de respuesta.
- Minimiza el riesgo de adquirir nueva infraestructura.
- Menor costo de inversión inicial.
- Aumenta el ritmo de la innovación.

1. Implementación

En [8] y [9] se mencionan tres escenarios para la formación de *Cloud Computing*, a saber, *Private Cloud*, *Public Cloud* e *Hybrid Cloud*. Estos escenarios se han convertido en medios atractivos para el intercambio computacional [10], de almacenamiento y de recursos de red entre desarrolladores de servicios múltiples y de aplicaciones de prestación de servicios. No hay que olvidar la capacidad de reasignar dinámicamente los recursos utilizando tecnologías de virtualización, ayudando a mitigar la necesidad de inversiones adicionales en infraestructura en tiempos de alta demanda.

Estos tres escenarios son implementados en los *Datacenters* y su disposición hace que su categorización dependa del enfoque, ya sea interno, externo o combinado.

1.1. *Private Cloud*

Private Clouds [8] son escenarios donde las compañías realizan sus operaciones fuera de línea, ejecutando aplicaciones seguras en *Datacenters*. De modo que, este escenario también se puede llamar *Internal Cloud*.

En [9] se menciona que *Internal Cloud* aplica los conceptos de *Cloud Computing* a recursos propios de la empresa que consume el servicio, proveyendo la capacidad de manejar aplicaciones web nuevas y existentes, mientras se provee de seguridad y regulación. También se menciona que *Private Cloud* trae consigo ciertas ventajas:

- Disponible en demanda.
- Rápido aprovisionamiento de servicios de negocio.
- Reducción del costo a través de economías a escala.
- Flexibilidad y libertad de selección,
- Basado en el uso.
- Controlado y asegurado por corporación de IT.

1.2. *Public Cloud*

Public Clouds [8] son escenarios donde las compañías necesitan mover datos o aplicaciones desde su interior al exterior, ambos escenarios utilizan la misma arquitectura, con la diferencia que el escenario público se conecta con otros escenarios. De modo que, este escenario también se puede llamar *External Cloud*.

External Cloud involucra recursos y servicios IT que son vendidos [9], tales como auto servicio, aprovisionamiento en demanda y pago por utilización, todos estos servicios accedidos a través de navegadores web o a través de API's.

Bajo el escenario de *External Cloud* se ejecutan diferentes tipos servicios que se detallarán en la siguiente sección, que son software, plataformas e infraestructura como servicios.

1.3. *Hybrid Cloud*

Por último, se hace referencia a una mezcla entre los dos escenarios anteriores, llamada *Hybrid Cloud*, en [8] se le llama *Inter Cloud*, que es un tipo de escenario semi público, el cual se comporta como un *Private Cloud* con la particularidad que ciertas empresas pueden compartir su información con ciertos niveles de permiso, por ello el término *semipúblico*.

El control de *Public Cloud* lo hace el proveedor, mientras que el control de *Private Cloud* lo hace la empresa, y la finalidad es que a través de ambos enfoques satisfacer las necesidades de un sistema de aplicación.

Inter Cloud como escenario [9], brindaría la capacidad de elegir los proveedores de servicio, y los proveedores de servicio federados serían capaces de compartir las cargas de servicio, siendo una relación más flexible.

2. Niveles de servicio

2.1. Niveles de servicio

Los diferentes niveles de servicio que componen a *Cloud Computing* son IaaS, PaaS y SaaS. Sin embargo, en [11] se menciona otro nivel de servicio, aunque ciertos autores lo prefieren alojar en entre los mismo tres niveles de servicio y no crear uno nuevo, es el llamado ITaaS, IT como servicio, siendo un modelo de servicio donde una organización o individuo contrata con un proveedor de servicios para obtener conectividad de red y cualquier otro servicio incluido, como *backup* de red, recuperación de desastres, VPN, conferencias web, etc. Aunque este nivel es muy general y puede abarcar a los tres anteriores pero de una forma unificada.

2.1.1. IaaS

Infrastructure as a Service o infraestructura como servicio [2] es entregar tanto hardware como software como un servicio. El ejemplo más común es el *hosting*, el cual, nos provee de hardware como un servidor y de software como un *webserver*, sin embargo, este concepto ha evolucionado a infraestructura como EC2 y S3.

Otra manera de ver IaaS [12] es como la manera de compartir recursos para ejecutar servicios, típicamente utilizando tecnología de virtualización, en donde múltiples usuarios utilizan dichos recursos. Los recursos pueden fácilmente escalar cuando la demanda se incrementa, y generalmente se utiliza métodos como pago por uso.

Uno de los servicios que toma importancia concerniente a la infraestructura es el *Cloud Storage* que se menciona a continuación:

2.1.1.1. Cloud Storage

Es almacenamiento localizado ya sea en *Datacenters* públicos o privados, separados del almacenamiento primario. Pertenece al nivel de infraestructura como servicio, la manera de implementación es a través [13] del *Service Oriented Architecture* (SOA), y la localización tiene diferentes variaciones, *Cloud Storage* puede ubicarse en:

- Un *Datacenter* público,
- Un *Datacenter* privado, o
- Separado del almacenamiento primario.

La manera de acceso puede ser de dos formas: Directamente como bloques o archivos; o indirectamente a través de aplicaciones que están ubicadas en el mismo lugar del almacenamiento.

Hay dos tecnologías [14] que proporcionan métodos de almacenamiento: La primera es *Storage Area Network* (SAN) que son *switches* de redes de alta velocidad que permiten que múltiples computadoras tengan acceso compartido a varios dispositivos de almacenamiento; y la segunda es *Network-Attached Storage* (NAS) que vienen como aplicaciones NAS o *Gateways* NAS, son servidores de archivos virtuales que tienen soporte a protocolos como NFS, siendo un dispositivo que directamente concede a la red y que tiene capacidades de compartir archivos.

Los protocolos utilizados para *Cloud Storage* son SOAP o REST, más adelante se detallarán dichos protocolos. También existen ciertas barreras para su adopción, tales como, el ahorro de costos no es significativo, se pone en riesgo la privacidad de los datos, cuestiones de migración, disponibilidad de datos y contratos como SLAs.

2.1.2. PaaS

Platform as a Service o plataforma como servicio [2], es entregar una plataforma de desarrollo de aplicaciones como un servicio para desarrolladores en la web. Generalmente se provee de herramientas tipo *middleware*, por ejemplo, Google AppEngine. Además de dicha entrega, también se ofrece un ambiente de ejecución como el servidor de aplicaciones.

2.1.3. SaaS

Software as a Service o software como servicio [2] provee la administración y *hosting* de aplicaciones con sus propios *Datacenters*, se maneja el término de múltiples inquilinos, por ejemplo Oracle CRM On Demand o Salesforce.

Es interesante lo que dice [15] acerca de SaaS, lo compara con la controversia que generó la computación cliente-servidor para reemplazar la computación de *mainframes*, y para el usuario final SaaS es un simple concepto, el usuario solamente ingresa a una aplicación a través del navegador web sin saber en donde se aloja o como está siendo servida, solamente sabe que es rápida y segura.

2.2. Tendencias

Entre las tendencias principales se pueden mencionar: para IaaS están EC2 y S3; para PaaS está Google AppEngine; y para SaaS están Oracle CRM On Demand, Salesforce y Netsuite.

En [16] se menciona que el departamento de IT de Intel ha ganado experiencia en SaaS e IaaS migrando muchas aplicaciones a *External Clouds*, y ha utilizado escenarios IaaS para alojar algunos servicios experimentales como *proof-of-concept* (PoC). SaaS principalmente se ha utilizado para conferencias web y se han explorado aplicaciones de oficina y *Custom Relationship Management* (CRM).

En [14] se menciona un aspecto importante de la tendencia de los niveles de servicio, especialmente de SaaS, y es que la demanda de los clientes y la competencia en la industria del software ha evolucionado los modelos de precios para software empresariales y este ha cambiado los términos de licencia, se ha migrado de ambientes de términos fijos a proveedores de servicios en términos variables, el término común que se ha utilizado es *on-demand*.

3. Tecnologías similares

3.1. Datacenter

Los *Datacenters* son ambientes especializados [17], que salvaguardan muchos de los valiosos equipos de las compañías y propiedad intelectual.

Entre las principales tareas que un *Datacenter* maneja están:

- Procesar las transacciones del negocio
- *Hosting*
- Procesar y almacenar la propiedad intelectual
- Mantener los registros financieros
- Manejar correos electrónicos

3.1.1. Diseño de *Datacenters*

En esta sección se trata el diseño de red para la implementación y puesta en práctica de *Datacenters*, existen varios enfoques. La mayoría de enfoques se concentra en el diseño de infraestructura, el cual, se tratará en la siguiente sección.

En [18] se sugieren ciertas indicaciones en el diseño de un *Datacenter*, entre las cuales, se menciona el proceso de diseño que incluye el diseño arquitectónico y el diseño de las capacidades del *Datacenter*; también los trazos estructurales.

Además, se debe tomar en cuenta los sistemas de soporte; una de las principales indicaciones se refieren a la seguridad lógica y física del *Datacenter*; también es importante los sistemas de monitoreo para tener un buen control; otra indicación importante son los sistemas de administración remota; y por último, es importante tener en cuenta cierta planificación para posibles expansiones futuras.

3.1.2. Infraestructura de red

En [17] se planteó un enfoque para el diseño de la infraestructura de *Datacenters*, se define de la siguiente manera:

- Enfocando el proyecto de *Datacenter*: El *Datacenter* refugia y protege los servidores que realizan operaciones importantes en una organización. Para diseñar la infraestructura se necesita de un buen diseño y luego de una buena mantenimiento. Hay cinco estrategias de diseño que ayudan a un *Datacenter* a ser efectivo:
 - Hacerlo robusto incluyendo infraestructura redundante.
 - Hacerlo modular por simplicidad y consistencia.
 - Seleccionar componentes de infraestructura que sean intercambiables y movibles.
 - Estandarizar el etiquetado y procedimientos.
 - Diseñar el espacio con detalles intuitivos para la usabilidad del usuario.
- Seleccionando el sitio perfecto: Desde que ocurre la adquisición de un sitio para una organización hay que planificar la puesta en marcha de un *Datacenter*, dicho lugar debe ser seguro, poseer características para escalar y cambiar, además, es necesario que cubra los factores de riesgos ya sea climáticos como sociales, también que tenga infraestructura energética suficiente, así como conectividad.
- Cuantificando el espacio del *Datacenter*: Esta es una tarea compleja, esto porque si se diseña un *Datacenter* pequeño, rápidamente se alcanzará su capacidad; mientras que si se diseña demasiado grande, es posible que se gasten recursos no necesarios. Hay cierta formula aunque no necesariamente exacta que dice que por cada trabajador es necesario un metro cuadrado para el *Datacenter*.

Lo que hay que tomar en cuenta es que el diseño debe incluir el crecimiento por lo menos para varios años, se sugieren espacios cuadrados o rectangulares para no ocupar espacio de más. Áreas que se deben tomar por aparte al *Datacenter* son cuartos destinados para desempaque y armado de equipos, mientras que otros cuartos como los eléctricos, cuartos con equipo de sistemas de baterías, cuartos de conectividad de redes se toman como parte del *Datacenter*.

- Trazar el *Datacenter*: Es importante trazar de una manera clara el espacio físico del *Datacenter*, tomando en cuenta la posición cercana de fusibles eléctricos, aire acondicionado, y extintores, también hay que tomar en cuenta el espacio de pasillos.

- Seleccionando la instalación bajo el piso o en el techo: La instalación de cables de red, aire acondicionado y cables de energía es más barata y sencilla hacerla en el techo, esto porque debajo del piso hay que montar cierta plataforma de soporte, además de dejar alrededor de 50 centímetros de alto para el subsuelo. Sin embargo, ambas maneras tienen ventajas y desventajas, lo mejor es analizarlas y decidir cual utilizar.
- Creando un sistema robusto de energía: Es imprescindible proteger el sistema eléctrico para evitar caídas innecesarias de los sistemas debido a problemas eléctricos. Hay que considerar circuitos amplios, para que en un futuro soporten la demanda energética de todos los dispositivos. Es muy recomendable tener proveedores duales de energía, además de sistemas de baterías, tomando en cuenta la disposición de los sistemas eléctricos de cada país, debido a cierta varianza. El sistema de baterías o sistema dual, debe ser exclusivo para el uso del *Datacenter*, asimismo se recomienda el *Grid* de referencia de señal para reducir el ruido eléctrico de alta frecuencia. Tomar en cuenta sistemas de etiquetado y la identificación de luces indicadoras.
- Diseñando una infraestructura de red escalable: Existen dos enfoques: tener los cables de red conectados directamente a las ubicaciones de gabinetes de servidores; y la otra es tener los cables de red conectados a una subestación, siendo esta más manejable, acortando la longitud de los cables, además de brindar más facilidad de crecimiento.
- Mantener frío el *Datacenter*: Es indispensable un sistema de enfriamiento, ayudando a disolver los *hot spots* creados por la extenuación y calentamiento de servidores, dentro del diseño se debe tomar en cuenta una buena circulación de la ventilación para mantener homogénea la temperatura en todo el *Datacenter*, manteniendo un balance para no variar los niveles de humedad, controlando constantemente que el lugar se mantenga seco y no se produzca condensación.
- Removiendo esqueletos del closet del servidor: No todo el diseño de un *Datacenter* requiere de un espacio nuevo, sino que hay veces en que se desea mejorar el diseño actual. Hay ciertas mejoras que se deben tomar en cuenta, por ejemplo, utilizar monitores compartidos, utilizar estanterías que se coloquen directamente en el piso, si el problema es el espacio, hay que considerar el diseño de un segundo *Datacenter*. Si el *Datacenter* posee seguridad de acceso es recomendable extraer puertas de los gabinetes, para que circulen mejor la ventilación y de esta manera ahorrar espacio.

3.1.3. Recomendaciones técnicas

Con base a los anteriores puntos, y a lo expuesto en [18] se presenta una serie de recomendaciones técnicas para la exitosa implementación de los *Datacenters*.

- Planificar
- Mantener la simplicidad
- Ser flexible
- Pensar modularmente
- Utilizar *racks*
- Preocuparse por el peso

- Utilizar piso de aluminio
- Etiquetar todo
- Mantener cubierto todo
- Esperar lo mejor, planear para lo peor

Las recomendaciones técnicas respecto al cableado de red contempla la creación de infraestructura tomando en cuenta los requerimientos de conectividad, el diseño modular y la estructura jerárquica de la red; también son importantes los puntos de distribución, estandarización de conectores; evitar a toda costa el *spaghetti* de cables; codificación y etiquetado por colores y verificación de conectividad.

3.2. Grid Computing

Grid Computing sugiere un paradigma [19], siendo un ambiente con un conjunto de recursos compartidos (procesadores, memoria y almacenamiento) creados para muchos consumidores que acceden como la vayan necesitando.

También llamados *Computational Grids* [20], son agregados a gran escala de recursos de comunicaciones y computación, habilitando nuevos tipos de aplicaciones y conllevan beneficios de economías a escala, inicialmente establecidos de manera académica en los 1990's, y de manera subsiguiente *Cloud Computing* comparte la misma visión original de *Grid Computing*. Son sistemas computacionales a gran escala, distribuidos y virtualizados. La clave de los *Grids* es la capacidad de escalar ya sea de expansión o de reducción.

A lo largo de los últimos años la computación distribuida se ha caracterizado por el despliegue de *Grids* a gran escala [21]. Generalmente estos ambientes han sido probados en escenarios científicos y se han esforzado para que haya interoperaciones entre *Grids*, sin embargo, la heterogeneidad de estos sistemas han incrementado la complejidad de despliegue en este tipo de infraestructuras.

Principalmente la relación de *Grid Computing* es por los avances en tecnologías de virtualización que han hecho emerger proveedores de infraestructura y este nuevo enfoque se ha conocido como *Cloud Computing*.

3.3. Utility Computing

Comúnmente *Utility Computing* se confunde con cierta forma de *outsourcing*, esto se puede aclarar recordando que el *outsourcing* se refiere al lugar donde los recursos residen y quien lo maneja, mientras que *Utility Computing* se refiere a como los recursos son manejados, consumidos y utilizados. *Utility Computing* es un modelo [14] que puede ser utilizado en un *Datacenter* corporativo o en un *Datacenter* externo, ayudando a romper la infraestructura monolítica en piezas separadas.

La introducción de la tecnología y el modelo de *Utility Computing* transforman la manera en que se gestionan los recursos y la manera en que se consumen, tiene como finalidad ayudar a las empresas a reducir la complejidad de IT y de aumentar la flexibilidad.

El concepto detrás de *Utility Computing* es simple [22], la idea principal es minimizar los costos mientras se aumenta la eficacia y eficiencia, siendo más que procesos y principios acerca de la tecnología.

Utility Computing transforma la manera en que las organizaciones de IT entregan recursos de IT formando esta nueva infraestructura de utilidad que tiene como propósito proveer una infraestructura de servicios como sean necesario, puede ser implementado de manera interna o externa.

3.4. Cloud Networking

Cloud Networking es un término emergente debido a los requerimientos de los nuevos modelos de virtualización y ambientes *cloud* [23], se requiere un alto nivel de disponibilidad de red, integración y administración. Por lo que emerge un concepto de interconexión de componentes para satisfacer requerimientos de *Cloud Computing*.

Por lo que *Cloud Networking* es una infraestructura de red requerida para soportar *Cloud Computing*, que requiere mejoras fundamentales en la escalabilidad, fiabilidad y retardo de las redes más allá que las redes tradicionales ofrecen, en cada una de estas dimensiones las necesidades de red de *Cloud Computing* son al menos un orden de magnitud más que las redes tradicionales.

La principal plataforma de *Cloud Networking* es Arista, tiene soporte para *Cloud Computing* en *Datacenters* con alta escalabilidad, robustez y costos efectivos de infraestructura de red, utiliza una combinación de alto rendimiento y tiene una arquitectura robusta para alcanzar los requerimientos.

4. Arquitectura

4.1. Consideraciones de arquitectura

Es el conjunto de capas que se encuentran acopladas entre sí para brindar la funcionalidad del sistema, en este caso la arquitectura de *Cloud Computing* es similar a la arquitectura de red, desde un nivel físico hasta un nivel de aplicación. Esto debido a que *Cloud Computing* utiliza protocolos similares a los que se usan en Internet como medio de comunicación, ya sea basado en web o no basado en web. En [24] se menciona una arquitectura genérica para *Cloud Computing*, que tienen las siguientes capas mencionadas de abajo hacia arriba:

- Recursos físicos: incluyen elementos como servidores, almacenamiento y red.
- Virtualización: incluye infraestructura virtual como un servicio.
- Infraestructura: incluye software de plataforma como servicio.
- Plataforma: incluye componentes de aplicación como servicio.
- Aplicación: incluye servicios basados en web y software como servicio.

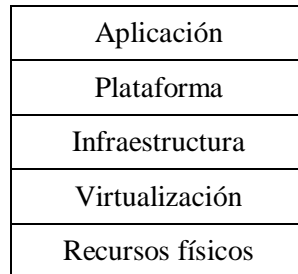


Figura 1: Arquitectura genérica de Cloud Computing

4.2. Enfoques propuestos

En [24] se extiende la arquitectura genérica, con una primer capa como base en la que interactúan y sirven a la segunda capa llamada interfaces de medición inteligentes, que realizan mediciones basadas en tiempos de consumo de los usuarios, consiste también de una subcapa de comunicación de consumo, luego viene la tercer capa compuesta de interfaces de administración y por último una capa que es de aplicación orientada a la organización.

Otro enfoque se presentó en [25], en donde se propone una arquitectura tipo plataforma, compuesta por tres capas y es llamada TPlatform. La primera capa representa los clúster de computadoras que soportan la segunda capa compuesta de la infraestructura en sí, esta capa tiene tres componentes: el MapReduce que es un marco de programación distribuida, el BigTable que es el almacenamiento de datos estructurados y el TFS que es el almacenamiento de alta fiabilidad y escalabilidad. La tercer capa son las aplicaciones de procesamiento de datos que soporta varios elementos como análisis de enlaces, indexadores, clasificadores, etc.

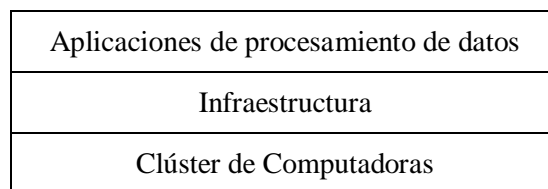


Figura 2: Arquitectura TPlatform

En [7] se reutiliza la arquitectura genérica extendiendo algunas capas, siempre como base tiene los servidores físicos, luego los servidores virtuales, luego el sistema operativo, luego el *middleware*, luego las aplicaciones y por último los servicios. Esta definición de capas sirve para acoplar de forma traslapada los diferentes servicios dentro de esta arquitectura tradicional.

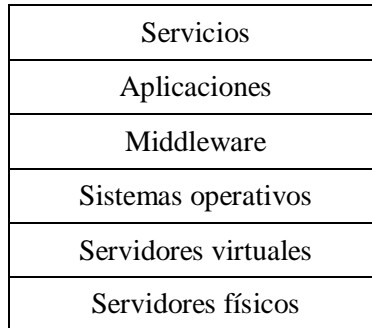


Figura 3: Arquitectura tradicional utilizada por Sun

En la próxima sección se tratará el tema de acoplar los diferentes niveles de esta arquitectura con los servicios de *Cloud Computing* como infraestructura.

5. Infraestructura

5.1. Consideraciones de infraestructura

Es importante destacar la administración de infraestructura [5], siendo la virtualización la tecnología fundamental habilitadora para la infraestructura como servicio en los nuevos *Datacenters*, la capa de administración actúa como el controlador central para hacer eficiente la administración de recursos en todo el ambiente. Esta administración es importante por la automatización del aprovisionamiento, la calendarización y otros aspectos importantes que se mencionarán en la siguiente sección.

5.2. Frameworks de infraestructura

En [5] se propone un modelo de infraestructura basado en la arquitectura del mismo, el cual se muestra en la siguiente Figura:



Figura 4: Framework de servicios de infraestructura por IBM

Básicamente este *Framework* contiene como base el hardware físico, en el cual incluye servidores, almacenamiento y red; luego la virtualización que a su vez puede ser servidores virtuales, almacenamiento virtual y red virtual; continúa con la administración que contiene elementos como el aprovisionamiento dinámico, la calendarización dinámica, portales de autoservicio, monitoreo y contratos SLAs; por último llega a la carga de trabajo habilitando la innovación, el desarrollo de software, clases virtuales, procesamiento intensivo de datos de la web 2.0 y el procesamiento transaccional escalable.

Otro *Framework* es el llamado CloudSim [26], siendo un *Framework* para el modelado y simulación de infraestructura y servicios de *Cloud Computing*. Este *Framework* tiene características novedosas como: el soporte para el modelado e instanciación de grandes infraestructuras de *Cloud Computing*, plataformas autocontenidas para el modelado de *Datacenters*, la disponibilidad de un motor de virtualización y la flexibilidad para seleccionar entre un espacio compartido y un tiempo compartido, ubicando los núcleos de procesamiento en servicios virtualizados.

También en [27], se propone un modelo de infraestructura en tres capas, a saber, infraestructura, plataforma y aplicación. Proporciona un ambiente básico de alta escalabilidad con servicios virtualizados de conexiones de red de manera física para unidades de computación y almacenamiento.

5.3. Servicios de infraestructura

En [7], tal como se mencionó en la sección 1 de implementación, se proponen tres tipos de servicios de infraestructura: *Private*, *Public* e *Hybrid Cloud*.

Además es importante destacar la funcionalidad de *Utility Computing* en [22], donde se propone una infraestructura de capas muy detallada, que va de la mano con la arquitectura en sí, empieza con la capa de hardware y la de software, luego hace una referencia a las capas de proceso, de organización y de servicio.

6. Diseño de red

6.1. Cuestiones de diseño

Independiente de la infraestructura bajo *Cloud Computing*, se debe tomar en cuenta aspectos en [17], para lograr un alto rendimiento. Hay dos enfoques para estructurar un *Datacenter*, una es conectar cables directos de la red al servidor y otra es conectar los cables de la red a una subestación y luego hacia el servidor. Hay ciertos consejos acerca del diseño, primero es construir la estructura de cableado completa durante la construcción inicial, otra es utilizar la menor cantidad de cable posible en cuanto a longitud, y seleccionar la tecnología adecuada a la necesidades.

Otras cuestiones en cuanto a diseño [18] mencionadas anteriormente son: crear una infraestructura de cableado de red, seleccionar los puntos de distribución, seleccionar los conectores correctos, evitar desordenar los cables, tratar de utilizar sistemas de etiquetado y por último la verificación.

En la sección 3.1 se trató el tema de diseño de red a nivel de *Datacenters*, lo cubre bastante del diseño en general de red para *Cloud Computing*, debido a que los *Datacenters* son una las bases de construcción de *Cloud Computing*.

Siempre en la sección 3 se trató el tema de *Cloud Networking*, un tema muy relacionado con las redes que son la base fundamental para el funcionamiento de *Cloud Computing*, en [28] se detalla un modelo de despliegue para servicios de *Cloud Networking*, el cual, tiene dos componentes:

- *Cloud Network*
- *Service Network*

6.2. Enfoques y modelos

Existen diferentes enfoques y modelos, cada uno presenta ciertas diferencias, también tienen diferentes puntos de vista, pero convergen en que ayudan al desarrollo de *Cloud Computing*, entre algunas de estas están:

- En cuanto al nivel de almacenamiento se encuentra el *Storage Area Network* [4], llamado SAN, utilizado para beneficios de virtualización, tiene como finalidad tener acceso a los datos desde la red comportándose como un nodo de la red.
- Un enfoque que soporta servicios de *Cloud Computing* brindando un API es NetInf[12], siendo un nuevo mecanismo de enrutamiento basado en la construcción de localización utilizando un enrutamiento de objeto a objeto más que un enrutamiento de host a host. Este mecanismo puede funcionar en una topología de red altamente dinámica y escalable, permitiendo el manejo de un número grande de objetos.
- A nivel de red se encuentra *Cloud Networking* [23], mencionado en la sección 3.4 que es relevante para habilitar el despliegue y aprovisionamiento de aplicaciones dinámicas. El diseño de red propuesto por Arista está construido utilizando una arquitectura de dos capas, llamadas columna y hoja, que mantiene el ancho de banda cruzado de manera uniforme, utilizando un flujo de trabajo sencillo:
 - o Un usuario inicializa una solicitud web
 - o El motor de transmisión recibe el flujo
 - o El motor trasmite el flujo hacia el componente ESA (que provee los servicios)
 - o ESA aplica el servicio y devuelve el flujo al motor de transmisión.
 - o El motor de transmisión transmite el flujo al servidor web

- El modelo VBSF [29] significa *Virtual Business Service Fabrics*, desde el punto de vista de red, este modelo puede extender los límites empresariales, geográficos y tecnológicos, facilitando la manejabilidad, incluyendo seguridad, aportando puntos importantes para *Cloud Computing*. Este modelo pertenece a una generación emergente de ambientes computacionales, que consiste en *Clouds* públicas y privadas, así como la infraestructura, proveyendo recursos virtuales bajo demanda.
- En [29] se menciona FCAPS, que se encarga de la administración de fallos, configuración, cuentas, rendimiento y seguridad introducida en redes de administración de telecomunicaciones (TMN). Las capacidades de FCAPS son realizadas a través de la administración de políticas y la colaboración entre agentes.

6.3. Seguridad

Uno de los principales desafíos de *Cloud Computing* es la seguridad, debido a que los datos son los activos más valiosos de las organizaciones

Es muy importante seguir algunos pasos de seguridad [7]:

- Cifrar los datos para evitar posibles penetraciones de intrusos en el sistema.
- Cifrar los datos en tránsito, asumiendo que los datos pasarán por una red pública.
- Requerir autenticación fuerte entre aplicaciones.
- Poner atención a la criptografía y estar actualizados en algoritmos de cifrado.
- Manejar de una manera segura los accesos de los usuarios.

En [30] se trata el problema de la seguridad, en especial de almacenamiento que es esencial en sistemas distribuidos y en *Cloud Computing*, y donde se propone un esquema distribuido que es flexible y efectivo, que tiene soporte dinámico de datos incluyendo agregado, actualizado y borrado. Se centra en la verificación de datos a través de un integrado sistema de almacenamiento garantizando la identificación del servidor que este causando problemas de comportamiento.

7. Protocolos de comunicación y cuestiones de red

En esta sección se mencionará de manera funcional diferentes protocolos que son utilizados en tecnología de *Cloud Computing* y similares.

7.1. Protocolos utilizados en *Cloud Computing*

- REST: *Representation State Transfer* [13], es un protocolo que define las operaciones en recursos y en formatos de datos. Basado en principios o reglas de arquitectura de red, los estados y la funcionalidad de la aplicación se representa mediante recursos, utiliza HTTP para transferencia de información.

- SOAP: *Simple Object Access Protocol* [14], es un protocolo basado en XML para aplicaciones que envían o reciben mensajes en internet, siendo una recomendación de la W3C. SOAP fue diseñado para ser simple, extensible e independiente de cualquier plataforma o modelo de programación. Utiliza HTTP como protocolo de transferencia, aunque puede ser utilizado también en RPC.
- WSDL: *Web Services Description Language* [14], es una especificación basada en XML que provee un método para describir características de servicios web, entre las cuales se menciona el nombre, dirección, protocolo, funciones, parámetros y tipos de datos utilizados.
- UDDI: *Universal Description, Discovery, and Integration* [14], es una especificación de registro de negocio que forma una fundación técnica para el soporte de la descripción y descubrimiento de servicios web que otros proveedores brindan.
- SNMP: *Simple Network Monitoring Protocol* [18], protocolo importante para monitorear el sistema. Se monitorea el estado del sistema, carga de trabajo, en general la “salud” del sistema. Varios sistemas pueden acoplarse con un monitor para obtener información y dar soporte al sistema.
- CIFS: *Common Internet File System* [31], basado en el protocolo *Server Message Block* (SMB) originalmente inventado por IBM, define las operaciones de transferencias, se considera muy estable. Otro protocolo basado en SMB es SAMBA.
- Otros protocolos utilizados en *Cloud Computing* son los ya conocidos HTTP, FTP, RPC, TCP, IP, DNS, entre otros.

7.2. Comparación entre protocolos

En [32] se mencionan ciertas características que diferencian a protocolos similares, por ejemplo, SOAP y REST, haciendo una comparación. Por ejemplo menciona que Amazon poseen ambos protocolos en sus implementaciones de servicios web, sin embargo, hay un 85% de sus clientes que utilizan REST a comparación de SOAP, siendo REST un enfoque más sencillo y por ello que los desarrolladores lo prefieren.

Básicamente la diferencia radica en que REST se representa mediante estados, definiendo operaciones básicas, mientras que SOAP se basa en mensajes. Aunque se utilice un protocolo más que otro no interviene en que la comunicación usuario-computadora ya esté resuelta, pero la comunicación computadora-computadora aún sigue siendo un desafío.

Con respecto a los demás protocolos, más que compararlos hay que hacer una reseña de complementación, debido a que cada protocolo ayuda a implementar servicios de *Cloud Computing*. Por ejemplo, para SOAP es complementario utilizar los protocolos de WSDL y UDDI. Además, estos protocolos utilizan definitivamente HTTP entre otros.

8. Virtualización

8.1. Virtualización de redes

La virtualización separa dispositivos y canales brindando seguridad, pero se requiere que se resuelvan cuestiones acerca de los recursos físicos que se pueden compartir, acerca de la capa de red que se manejará, la manera de ofrecer aislamiento, rendimiento, escalabilidad y la flexibilidad [14]. También se debe analizar la carga de trabajo para hacer redes eficientes y estudiar los mecanismos de resolución de nombres y ruteo.

La virtualización de redes se realiza por medio de *switches* de virtualización [14], se divide el ancho de banda disponible en canales seguros, esto permite crear zonas seguras internas y consolidar la seguridad externa.

Tal como se describe en [33], los sistemas virtuales y la tecnología de virtualización están en estos días en su auge para *Datacenters* y modelos de infraestructura de IT, el análisis de rendimiento en estos sistemas es algo muy importante, un análisis de la carga de trabajo cuantificando la sobrecarga en una máquina virtual es muy útil, pero es importante capturar el comportamiento a través de diferentes plataformas a través de un modelo común de carga de trabajo para tener resultados útiles.

En [34] se menciona que la virtualización de redes puede atenuar las fuerzas de internet y estimular la innovación al permitir diversas arquitecturas de red convivir en un sustrato físico compartido.

La utilización de máquinas virtuales trae consigo varios beneficios[21]:

- Consolidación de servidores
- Habilidad de creación de máquinas virtuales sin interferir con otras aplicaciones
- Mejora la seguridad
- Proporciona aislamiento

8.2. Modelos y enfoques

Entre algunos modelos se destacan:

- En [21] se propone el modelo *Virtualized Distributed Computing* que ayuda a proveer un escenario de ejecución para aplicaciones en el tope de infraestructuras interconectadas. Este sistema distribuido utiliza máquinas virtuales como bloques para la construcción de escenarios de ejecución que extiende múltiples sitios para computación. Además, el ambiente de ejecución es una red de máquinas virtuales creadas para satisfacer los requerimientos de cierta aplicación, de esta forma se ejecuta aisladamente de otros ambientes de ejecución.

- Desde el punto de vista de virtualización se puede mencionar VBSF [35], *Virtual Business Service Fabric* como un modelo para virtualizaciones heterogéneas y abstracción de servicios, de políticas, de capacidades, de recursos y de infraestructura. Este modelo es aplicable a servicios y ambientes de infraestructura que trascienden entre *Private* y *Public Clouds*, servicios, aplicaciones e infraestructuras.
- Existe otro modelo basado en contratos, Virtual Machine Contracts (VMC) [36], que propone una plataforma independiente que automatiza las comunicaciones y la administración de ciertos requerimientos. Consiste en una extensión del *Open Virtual Machine Format* (OVF), el cual, es un *hypervisor* estándar para describir, empaquetar y distribuir aplicaciones virtuales.
- En [37] se propone otro modelo llamado Diverter, el cual, es un modelo basado en red para infraestructuras virtualizadas. Este modelo busca alta flexibilidad, larga escalabilidad con ambientes múltiples de arrendamiento, eficiencia, sistemas de ruteo virtualizados distribuidos, estos sistemas permiten que la comunicación *end-to-end* sea a través de un salto.
- Otro modelo se llama SnowFlock [38], es un modelo de clonación de máquinas virtuales para *Cloud Computing*. Está basado en una implementación de Xen, es un proyecto *open-source* que provee cierta reducción drástica acerca del tiempo que toma la clonación y la búsqueda eficiente de la memoria, reduciendo el tráfico de red por la eliminación de la transferencia de páginas que serán sobrescritas. Obteniendo preferiblemente rendimiento a fiabilidad y quedando reducida la barrea de entrada de *Cloud Computing*.

9. Tendencias y ejemplos de *Cloud Computing*

9.1. Descripción de tendencias y ejemplos

Algunas tendencias que de facto se han utilizado se mencionan a continuación, sin embargo, cada tendencia sigue una línea separada de desarrollo, y la selección de las mismas se dará dependiendo de las necesidades específicas de cada organización:

- *Cloud Computing Center* [39] en Wuxi, China. Wuxi es una ciudad a 100 millas de Shanghai, China, tienen un desafío de inversión en tecnología para aceptar negocios de clientes empresariales, por lo que la municipalidad de Wuxi ha trabajado con IBM para construir un *Cloud Computing Center*, basado en un nuevo modelo de *Datacenter* empresarial.
- Crossbow [40], es una tecnología de Solaris que utiliza NICs virtuales, *switches* virtuales y máquinas de red virtuales, y es utilizada como una fundación para construir redes virtuales aisladas para arquitecturas de *Cloud Computing*.

- Eucalyptus [41], es un marco de software *open source* para *Cloud Computing* que implementa IaaS, siendo sistemas que proporcionan la capacidad de ejecución y control a través de máquinas virtuales desplegadas a través de recursos físicos.
- Google App Engine [42], es una plataforma de ejecución basada en *Python* que provee *hosting* de aplicaciones web, almacenamiento de datos y redes de alta velocidad ejecutando a nivel de aplicación la infraestructura de Google.
- El núcleo de Hadoop [43] provee un marco *open source* para *Cloud Computing* así como un sistema de archivos distribuido.
- Nimbus [44], es otra tendencia que permite por un lado a los proveedores construir un ambiente para *Cloud Computing*, por otro lado permite a los usuarios utilizar dicho ambiente y también permite a los desarrolladores experimentar con Nimbus.
- El S3 de Amazon [45], o *Simple Storage Service*, es un servicio de almacenamiento simple que permite almacenar y recuperar datos a través de un servicio de *host*. Los desarrolladores pueden tener servidores virtuales para el desarrollo de sus aplicaciones.
- Variando un poco los conceptos de servicios empresariales de marcos para *Cloud Computing* aparece Seattle [46], como una plataforma para *Cloud Computing* educacional, es manejada por una comunidad universitaria, es muy flexible y soporta una variedad de usos pedagógicos, también es portable y la finalidad es crear una red distribuida alrededor del mundo basada en comunidades educativas.
- TCCP [47], o *Trusted Cloud Computing Platform*, es una plataforma confiable de *Cloud Computing*, nace por la falta de confidencialidad e integridad de los datos en redes tipo *Cloud*, por lo que fue diseñado para habilitar a los proveedores de infraestructura como servicio IaaS para proveer un ambiente de ejecución cerrado de máquinas virtuales invitadas, permitiendo a los usuarios confiar en los proveedores y determinar qué servicios son o no seguros antes de utilizarlos.
- Una plataforma muy utilizada es VM vSphere [48], habilita robustez en la infraestructura de IT basada en *Cloud Computing*, siendo una plataforma de virtualización que utiliza ya sea *Private Cloud* o *Public Cloud*, asimismo la federación y normas para interconectar infraestructuras *Cloud*, creando una estructura híbrida de *Cloud* que pueda responder a las necesidades empresariales que comúnmente evolucionan. Reducen los costos operativos, aumentan el control sobre las infraestructuras de IT manteniendo al mismo tiempo la flexibilidad para elegir cualquier sistema operativo, aplicaciones y hardware.

9.2. Comparación entre tendencias y ejemplos

En general, las comparaciones son realizadas mediante el nivel de infraestructura que se utiliza, basado si son servicios de infraestructura, de plataforma o de software.

Como se mencionó en la sección anterior, la selección de una plataforma o marco de desarrollo dependerá de las necesidades de cada organización, tal es el caso para organizaciones orientadas a la educación, pueden estudiar la utilización de Seattle.

Y no se puede decir de forma objetiva cuál es el mejor, sino lo que se buscará es cuál es el más apropiado para cierta empresa. Los tres enfoques con mayor número de usuarios y con más desarrollo es S3 de Amazon, Google App Engine y Hadoop.

10. Conclusiones

Con base a la anterior investigación se llegó a la siguiente conclusión: “*Cloud Computing es una tecnología que ha venido evolucionando y que provee infraestructura, plataforma y software como servicio, generalmente proporcionado por un tercero y bajo demanda, asimismo es la convergencia de muchas tecnologías orientadas a la interconexión a través de medios de comunicación*”, debido a esto se presenta en este reporte técnico a *Cloud Computing* como una red de servicios.

Además, en este reporte técnico se elaboró un estado del arte de los conceptos y tecnologías asociados a *Cloud Computing*, para describir de una manera comprensiva este concepto que ha tomado auge en los últimos años, asimismo se enfocó la perspectiva de análisis hacia ver a *Cloud Computing* como una red de servicios. También es importante mencionar que hay tres niveles de servicio: de infraestructura, de plataforma y de software, a saber: IaaS, PaaS, SaaS, que corresponden de manera traslapada a las distintas capas de la arquitectura de *Cloud Computing*, creando un enlace entre los diferentes niveles de servicio. Dentro de la infraestructura existen tres tipos de redes, las que corresponden a *Private Cloud*, a *Public Cloud* e *Hybrid Cloud*. La base ideal para la implementación de *Cloud Computing* son los *Datacenter* por todos los beneficios que estos conllevan. Dependiendo de la orientación de cualquier organización, así será el estudio que debe hacer para seleccionar alguna plataforma existente que soporte *Cloud Computing*.

11. Recomendaciones

Entre las recomendaciones figuran las que se refieren a seguridad de redes, mostradas en la sección 6.3, además de utilizar cualquier componente como servicios sea requerido.

Además, es necesario que los desarrolladores puedan realizar pruebas en la plataforma antes de ponerlas en producción, tal como lo provee Nimbus.

También es necesario realizar un estudio previo para luego seleccionar al proveedor idóneo que nos permita alcanzar nuestros requerimientos en pro del rendimiento, escalabilidad, seguridad y disponibilidad de nuestros servicios.

12. Referencias

- [1] M. Ambrust, et al., "Above the Clouds: A Berkeley View of Cloud Computing," Electrical Engineering and Computer Sciences, University of California at Berkeley, California, Technical Report UCB/EECS-2009-28, 2009.
- [2] S. Bennett, M. Bhuller, and R. Covington, "Architectural Strategies for Cloud Computing," Oracle Corporation, 2009.
- [3] S. Kajeepeta, "Cloud Computing: From Metaphor to Mainstream," *Software Magazine*, vol. 27, no. 6, pp. 10-13, Nov. 2008.
- [4] G. Boss, P. Malladi, D. Quan, L. Legregni, and H. Hall, "Cloud Computing," IBM Corporation, 2007.
- [5] D. Quan, "From Cloud Computing to the New Enterprise Data Center," IBM Corporation, 2008.
- [6] D. Thomas, "Cloud Computing - Benefits and Challenges!," *Journal of Object Technology*, vol. 8, no. 3, pp. 37-41, 2009.
- [7] Sun Microsystems, "Introduction to Cloud Computing Architecture," Sun Microsystems White Paper, 2009.
- [8] J. D. Lasica, *Identity in the Age of Cloud Computing*. United States of America: The Aspen Institute, 2009.
- [9] Cisco, "Private Cloud Computing for Enterprises," Cisco White Paper, 2009.
- [10] R. Mikkilineni and V. Sarathy, "Cloud Computing and the Lessons from the Past," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 2009, pp. 57-62.
- [11] Cisco Systems, Inc., "The Cisco Powered Network Cloud: An Exciting Managed Services Opportunity," Cisco Systems, Inc., 2009.
- [12] B. Ohlman, A. Eriksson, and R. Rembarz, "What Networking of Information Can Do for Cloud Computing," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 2009, pp. 78-83.
- [13] A. Kamaraju and P. Nicolas, "Cloud Storage," Storage Networking Industry Association, 2009.
- [14] A. Mendoza, *Utility Computing Technologies, Standards, and Strategies*. United States of America: Artech House, Inc., 2007.
- [15] A. Dver, "Enemy of SaaS?," *Software Magazine*, vol. 27, no. 6, p. 24, Nov. 2008.
- [16] H. Li, et al., "Developing an Enterprise Cloud Computing Strategy," Intel Corporation, 2009.
- [17] D. Alger, *Build the Best Data Center Facility for Your Business*. Indianapolis, United States of America: Cisco Press, 2005.
- [18] R. Snevely, *Enterprise Data Center Design and Methodology*. California, United States of America: Prentice Hall, 2002.
- [19] V. Silva, *Grid Computing for Developers*. United States of America: Charles River Media, Inc., 2006.

- [20] K. Delic and M. Walker, "Emergence of The Academic Computing Clouds," *ACM Ubiquity*, vol. 9, no. 31, 2008.
- [21] A. di Constanzo, M. Assuncao, and R. Buyya, "Building a Virtualized Distributed Computing Infrastructure by Harnessing Grid and Cloud Technologies," The University of Melbourne, 2009.
- [22] G. Bunker and D. Thomson, *Delivering Utility Computing*. England: John Wiley & Sons Ltd., 2006.
- [23] Arista, "A Novel Network Approach for Cloud Computing Models," Arista White Paper, 2008.
- [24] T. Singh and P. Kumar Vara, "Smart Metering the Clouds," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, United States of America, 2009, pp. 66-71.
- [25] B. Peng, B. Cui, and X. Li, "Implementation Issues of A Cloud Computing Platform," in *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, Pekin, 2009.
- [26] R. Calheiros, R. Ranjan, C. De Rose, and R. Buyya, "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services," The University of Melbourne, 2009.
- [27] W. Ji, J. Ma, P. Nanjing, and X. Ji, "A Reference Model of Cloud Operating and Open Source Software Implementation Mapping," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 2009, pp. 63-65.
- [28] Arista - Citrix, "Cloud Networking Services," 2009.
- [29] P. Goyal, R. Mikkilineni, and M. Ganti, "FCAPS in the Business Services Fabric Model," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 2009, pp. 45-51.
- [30] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data Storage Security in Cloud Computing," Illinois Institute of Technology, 2009.
- [31] A. Leung, S. Pasupathy, G. Goodson, and E. Miller, "Measurement and Analysis of Large-Scale Network File System Workloads," in *Proceedings of the 2008 USENIX Annual Technical Conference*, Boston, MA, United States of America, 2008.
- [32] R. Navarro, "Rest vs Web Services," ELP-DSIC-UPV, 2007.
- [33] M. El-Rafaey and M. Abu, "Virtual Systems Workload Characterization," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 2009, pp. 72-77.
- [34] M. Kabir and R. Boutaba, "A Survey of Network Virtualization," School of Computer Science, University of Waterloo, Ontario, Canada, Technical Report CS-2008-25, 2008.
- [35] P. Goyal, "The Virtual Business Services Fabric," in *18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, 2009, pp. 33-38.
- [36] J. Matthews, T. Garfinkel, C. Hoff, and J. Wheeler, "Virtual Machine Contracts for Datacenter and Cloud Computing Environments," Clarkson University & Stanford University, 2009.
- [37] A. Edwards, A. Fischer, and A. Lain, "Diverter: A New Approach to Networking Within Virtualized Infrastructures," in *WREN*, Barcelona, Spain, 2009, pp. 103-110.

- [38] H. Lagar-Cavilla, et al., "SnowFlock: Rapid Virtual Machine Cloning for Cloud Computing," in *EuroSys*, Nuremberg, Germany., 2009.
- [39] D. Quan, "From Cloud Computing to the New Enterprise Data Center," IBM Corporation, 2008.
- [40] N. Droux, "Crossbow for Cloud Computing Architectures," Sun Microsystems, Inc., 2009.
- [41] D. Nurmi, et al., "The Eucalyptus Open-source Cloud-computing System," University of California, Santa Barbara, 2008.
- [42] E. Ciurana, *Developing with Google App Engine*. Germany: Springer-Verlag, 2009.
- [43] J. Venner, *Pro Hadoop*. United States of America: Apress, 2009.
- [44] T. Kielmann, "Cloud Computing with Nimbus," OGF25, 2009.
- [45] D. Robinson, *Amazon Web Services Made Simple*. Australia: Emereo Pty Ltd., 2008.
- [46] J. Cappos, I. Beschastnikh, A. Krishnamurthy, and T. Anderson, "Seattle: A Platform for Educational Cloud Computing," in *SIGCSE*, Tennessee, United States of America, 2009.
- [47] N. Santos, K. Gummadi, and R. Rodrigues, "Towards Trusted Cloud Computing," MPI-SWS, 2009.
- [48] VMware, "What's New in VMware vSphere™ 4: Virtual Networking," VMware, 2009.