

Evaluación del Impacto de Migración al Cloud

María A. Murazzo, Nelson R. Rodríguez

Departamento e Instituto de Informática – UNSJ – Facultad de Ciencias Exactas,
Físicas y Naturales

CUIM (Complejo Universitario Islas Malvinas) - Cereceto y Meglioli – C.P. 5400 –
Rivadavia, San Juan, Argentina

marite@unsj-cuim.edu.ar, nelson@iinfo.unsj.edu.ar

Resumen

Cloud Computing en un modelo de computación que permite que los recursos computacionales, tales como infraestructura, aplicaciones, software o procesamiento puedan ser ofrecidos y consumido bajo demanda como un servicio más en Internet. Contar con una infraestructura de cloud computing, permite a los usuarios migrar sus datos, aplicaciones y cómputo de un entorno privado con fuertes requisitos de recursos físicos a otro con gran cantidad de recursos virtualizados. Sin embargo, uno de los aspectos que se debe tener en cuenta a la hora de realizar la migración es el costo en términos de degradación de performance. En este sentido se debe analizar cómo impacta en el cloud el proceso de la virtualización de los recursos, por ello el objetivo de este trabajo es analizar cuánto se degrada la performance de algoritmos cuando se corren en el cloud.

Abstract

Cloud Computing is a computer model which allows computing, such as infrastructure, applications, software or processing resources can be offered and consumed on demand as a service over the Internet. Having a cloud computing infrastructure, enables users to migrate their data, applications and computing a private environment with strong physical requirements to another with a lot of resources virtualized. However, one aspect that should be taken into account when migrating is the cost in terms of performance degradation. In this sense we must analyze how the cloud impacts the process of virtualization of resources, so the aim of this paper is to analyze how the algorithms degrades performance when running on the cloud.

1 Introducción

Cloud Computing se ha convertido en un enorme repositorio de recursos computacionales, lo cual es una buena posibilidad para construir una plataforma para las aplicaciones que necesitan una gran cantidad de recursos. Esta capacidad del cloud se debe principalmente a la habilidad de escalado elástico de recursos en función de las necesidades de las aplicaciones y el presupuesto del usuario. Esta es una tecnología centrada en ofrecer cómputo bajo demanda como cualquier otro servicio. Uno de los aspectos clave que afecta la performance de las aplicaciones al montarlas sobre arquitectura cloud, es la capacidad de virtualización de cualquier recurso (bases de datos, red, procesador, etc.) y ofrecerlo como un servicio (AaaS, Anything as a Service) [Schubert, Jeffery and Neidecker-Lutz, 2010].

Sin embargo y a pesar de estas ventajas, el principal problema al que se debe hacer frente es la escasa performance en el acceso a Internet, la cual puede generar retardos, que impactaran de manera negativa en la ejecución de cierto tipo de aplicaciones (por ejemplo, las de cálculo intensivo con restricciones de calidad, donde es necesario mucha transferencia de información).

Para solucionar este problema una posibilidad es montar una infraestructura cloud privada con el objetivo de sacar el máximo provecho a las bondades de la virtualización de recursos, sin pagar el precio de la degradación de la calidad por tener que acceder a Internet.

2 Fundamentos del Cloud Computing

Cloud Computing es un modelo para permitir el acceso en red omnipresente, conveniente y bajo demanda a un conjunto compartido de recursos informáticos configurables tales como, redes, servidores, almacenamiento, aplicaciones y servicios que pueden ser rápidamente aprovisionados y liberados con una mínima gestión esfuerzo o interacción con el proveedor o administrador de servicio. Por ejemplo, en lugar de comprar un servidor, switches, firewalls, cables, etc. y alojarlos físicamente, se puede contratar capacidad de procesamiento, memoria, disco y ancho de banda a un proveedor de cloud hosting; esto es Infraestructura Cloud.

El NIST (National Institute of Standards and Technology), ha presentado una de las definiciones de Cloud más clara y comprensible. La define como *“un modelo tecnológico que permite el acceso ubicuo, adaptado y bajo demanda en red a un conjunto compartido de recursos de computación configurables (por ejemplo: redes, servidores, equipos de almacenamiento, aplicaciones y servicios), que pueden ser rápidamente aprovisionados y liberados con un esfuerzo de gestión reducido o interacción mínima con el proveedor del servicio”* [Mell and Grance, 2011].

Según dicha definición, Cloud Computing se entiende como un modelo de prestación de servicios informáticos cuya principal orientación es la escalabilidad. Desde el punto de vista de los usuarios, los servicios son elásticos, es decir, que pueden crecer o recuperar su tamaño original de manera rápida y sencilla. Esta orientación permite que los usuarios que acceden a los servicios, perciban que todo funciona de manera simple y rápida, dando como resultado una experiencia más gratificante.

Para lograr esto, cloud está organizada en una arquitectura de capas llamada Modelo de Servicio, donde el *SaaS (Software as a Service)* se encuentra en el tope del stack, es el nivel más visible y permite a los usuarios tener acceso a las aplicaciones alojadas en el cloud (por ejemplo: Google App). El *PaaS (Platform as a Service)* es una capa de abstracción o middleware que se encuentra entre el SaaS y la infraestructura virtualizada, y su objetivo es proveer a los desarrolladores un framework y un conjunto de API's que puedan usar para construir sus aplicaciones (por ejemplo: Google App Engine). Por último, el *IaaS (Infrastructure as a Service)* ofrece una infraestructura virtualizada de recursos (procesador, red, base de dato, disco, etc.) como servicio, de forma de ofrecer a los usuarios un ambiente con recursos escalables de manera elástica y pagando por su uso.

Un aspecto muy importante de esta arquitectura es la capa de *Virtualización* que se ubica como un middleware sobre el IaaS y es la encargada del aislamiento y la flexibilidad en el manejo de recursos. Gracias a esta capa de virtualización, es posible la compartición de recursos permitiendo la asignación, reasignación y reasignación de “instancias” de cada recurso. Para lograr esto es necesario contar con mecanismos de “reservación de recursos” y de “administración de recursos” para permitir el manejo consistente de todas las instancias de los recursos virtualizados con el objeto de lograr su máxima performance.

Por otro lado, se encuentra el Modelo de Despliegue, que se refieren a la posición (localización) y administración (gestión) de la infraestructura de cloud y pueden ser *públicos* (manejadas por terceras partes), *comunitarios* (operada por varias organizaciones para uso compartido) o *privados* (propiedad de una organización), como una cuarta opción aparece el modelo *híbrido* como la combinación de cualquiera de los tres modelos anteriores.

Hay que aclarar, que cuando se decide contar con una solución basada en cloud es de suma importancia analizar cual despliegue se usará, pues esta elección impacta directamente en la performance de la solución. Hay que recordar que cualquier elección que implique el uso intensivo de comunicaciones degrada el desempeño por los retardos incurridos en el proceso de comunicación en si [Rodríguez, Chávez, Martín, Murazzo and Valenzuela, 2011].

Cloud Computing provee una plataforma en la cual los usuarios interactúan con aplicaciones de forma remota sobre Internet, bajo la premisa de la compartición de recursos; permitiendo acceder a una plataforma donde se provee “cualquier cosa”: poder de cómputo, infraestructura, almacenamiento, comunicaciones y servicios en cualquier lugar y en cualquier momento.

3 Migración al Cloud Computing

El principal objetivo que se persigue al instalar una infraestructura cloud, es permitir a los usuarios la migración de sus datos y aplicaciones sin que se genere un impacto en la performance. Esta migración ha sido impulsada en gran medida por el bajo costo que representa contar con una solución de commodity frente a los altos costos que puede generar hosting. En definitiva, migrar al cloud puede generar a los usuarios ventajas de escalabilidad, calidad de servicio garantizada, personalización y flexibilidad en la provisión de recurso. Sin embargo, el menor costo no siempre es

una buena razón debido a que el usuario puede tener otros requerimientos en términos de fiabilidad y disponibilidad del servicio.

En función de esto, se puede decir que la principal motivación de mover aplicaciones al cloud es reducir los costos de mantenimiento y abastecimiento, al tiempo que se aumenta la escalabilidad y la disponibilidad. Cloud Computing ofrece una forma de eliminar la necesidad de mantener el hardware caro permitiendo a los usuarios y desarrolladores centrarse en las aplicaciones [Murazzo, Rodríguez, Medel, Chávez, Martín and Valenzuela, 2015].

Un aspecto muy importante a tener en cuenta es que, si bien migrar al cloud tiene muchas ventajas, también tiene desventajas, entre las cuales se puede mencionar como la más relevante las demoras incurridas en usar una arquitectura distribuida donde no existe un sincronismo global y las comunicaciones degradan la performance de la infraestructura.

En función de esto, se debe tratar de minimizar los efectos provocados por la migración a un entorno distribuido mediante la elección adecuada de herramientas que permitan minimizar los impactos negativos sobre la performance de las aplicaciones.

3.1 Parámetros para la migración

Una de las características más relevantes del Cloud Computing es su elasticidad. Los entornos Cloud son escalables, es decir, capaces de ajustarse a la demanda de los usuarios. Ante un pico elevado de requisitos de cómputo, un usuario puede solicitar mayor capacidad de cálculo, que provocará el despliegue automático de nuevas instancias sobre la infraestructura física existente. Por el contrario, si lo que se detecta es una baja de la demanda de cómputo, el usuario puede decidir prescindir de parte de su infraestructura, pudiéndole ajustar a su uso de forma dinámica. Esta es una gran ventaja, puesto que uno de los problemas que existían hasta la aparición del Cloud Computing era la creación de una infraestructura ad hoc preparada para soportar demandas no previsibles de cómputo, ya que esto suponía una elevada inversión y además no se podía asegurar que fuese capaz de responder correctamente a estos incrementos repentinos en la demanda de los recursos de cómputo.

Desde el punto de vista de la arquitectura de Cloud Computing la capa más involucrada en la migración hacia el cloud es el IaaS, pues es la que dará el soporte de hardware (físico o virtualizado) para concretar la migración. El IaaS brinda la capacidad de cómputo, procesamiento y almacenamiento para que el usuario pueda desplegar de forma transparente sistemas operativos y aplicaciones. Para lograr esta transparencia la infraestructura se monta sobre una capa de virtualización que permite contar con un ambiente optimizado independientemente del hardware físico mediante la generación de instancias abstractas de recursos llamadas máquinas virtuales, las cuales se encuentran administradas por los hipervisores [Rahma, Adji and Widyan, 2013].

Los productos disponibles en el mercado para instalar infraestructuras cloud cuentan con diferentes hipervisores, los cuales se encargan de orquestar los recursos virtuales de cada máquina virtual con el objeto de sacar el máximo beneficio a los recursos físicos.

En función de esto, aparecen dos problemas que deben ser paliados con el objetivo que la migración sea exitosa. En primer lugar, hay que tratar de *minimizar los retardos* incurridos por las comunicaciones. Si este aspecto no es tratado de forma adecuada, puede degradar la performance de las soluciones que tengan fuertes restricciones de tiempo; tales como el cómputo en tiempo real.

En segundo lugar, hay que tratar de *maximizar la escalabilidad* de la infraestructura en la medida que exista un aumento en la demanda de recursos virtualizados. La escalabilidad de una infraestructura cloud está fuertemente ligada con la tecnología de virtualización o hipervisores que se está usando. Por esta razón es tan importante la elección adecuada del hipervisor. Hay que destacar que si bien la virtualización aísla al usuario / desarrollador de los detalles de orquestación de recursos, un aspecto que se debe analizar es el overhead producido por los hipervisores. Estos, consumen recursos para la administración de los recursos virtualizados, lo que lleva a que mientras más se virtualice, mas overhead haya y por consiguiente mas sobrecarga y menos rendimiento [Huber, von Quast, Hauck, and Kounev, 2011].

4 Infraestructura Cloud Propuesta

Con el objetivo de medir el desempeño de de la capa de virtualización, se ha decidido realizar la instalación de un cloud privado open source.

En el mercado existen numerosos productos de este tipo, tales como: Eucalyptus, Nimbus, OpenStack, OpenNebula, CloudStack, entre los que poseen mayor aceptación. De todas ellas se han seleccionado aquellas que tienen mayor aceptación en la comunidad académica y mayor soporte de la comunidad de usuarios: OpenStack, CloudStack y OpenNebula.

- *OpenStack*: Es una herramienta de colaboración global, libre y de código abierto distribuido bajo los términos de la licencia Apache. es una de las iniciativas de cloud que cuenta con más apoyos, entre los que están Dell, Cisco y HP, así como RackSpace.
- *CloudStack*: software código abierto para crear, controlar y desplegar infraestructuras cloud. Originalmente fue desarrollado por cloud.com y posteriormente por Citrix, actualmente es un proyecto de la Apache Software Foundation (ASF). Implementa Amazon EC2, S3 APIs, vCloud API y su propia API CloudStack, escrita en Java. Soporta Xen, KVM, VMware y acceso a Amazon EC2s.
- *OpenNebula*: conjunto de herramientas código abierto que permiten manejar la infraestructura de virtualización de un centro de datos para construir cloud privadas, públicas e híbridas. Esta escrita en C++, Ruby y Shell. Soporta Xen Server, KVM, vSphere, y acceso a Amazon EC2s.

De estos productos se ha seleccionado OpenStack, debido a que es el proyecto que mayor aceptación y mayores contribuciones ha tenido en los últimos años [Jiang, 2015]. Además, es el producto con mayor número de hipervisores soportados, lo que lo hace adecuado para poder realizar una evaluación más amplia de su desempeño.

OpenStack [OpenStack, 2015] es un proyecto global para crear una plataforma de cómputo open source en el cloud, que cumpla con las necesidades de

los proveedores de servicios en clouds públicos, privados e híbridos, independientemente de su tamaño y cuyas premisas son que sea fácil de implementar y escalable.

El proyecto fue fundado en octubre de 2010 por la empresa Rackspace Cloud y por la agencia espacial estadounidense, NASA. Actualmente es gestionado por la Fundación OpenStack y cuenta con el apoyo de más de 178 empresas, entre las que destacan Rackspace, Oracle, AMD, Cisco, Canonical, IBM, HP, Dell Red Hat, Suse Linux, VMware, Yahoo y KIO Networks en América Latina.

Las características más importantes de OpenStack, se pueden resumir en:

- *Es escalable*: OpenStack es usado por compañías multinacionales cuyo volúmenes de datos se mide en petabytes, con arquitecturas distribuidas en 1 millón de nodos, con 60 millones de máquinas virtuales y billones de objetos almacenados.
- *Es compatible y flexible*: debido al soporte de múltiples hipervisores: ESX, Hyper-V, KVM, LXC, QEMU, UML, Xen y XenServer.
- *Es abierto*: todo su código se puede modificar y adaptarse a cualquier necesidad. Además, el proyecto tiene un proceso de validación para el desarrollo y adopción de nuevos estándares.

La arquitectura de OpenStack es totalmente modular y en la actualidad engloba dos proyectos principales: *OpenStack Compute* y *OpenStack Object Storage*. El primero, denominado NOVA, aportado por la NASA, consiste en software dedicado al aprovisionamiento y administración de un gran número de servidores privados virtuales. El segundo proyecto, denominado SWIFT, aportado por Rackspace, permite la creación de almacenamiento redundante, escalable y de alta disponibilidad e integridad para terabytes o petabytes de datos.

Con base en estos dos proyectos, OpenStack, posee tres módulos principales: Compute, Networking y Storage, como muestra la figura 1, a partir de los cuales se han desarrollado un conjunto de componentes que permiten la administración y gestión del cloud, de los cuales los principales son: Nova, Glance, Swift, Horizon, Keystone, Neutron. Además, dependiendo de la versión de la que se trate se cuentan con otros componentes [Rodríguez, Murazzo, Chávez and Guevara, 2014]. Cada uno de estos componentes se comunican mediante APIs provistas por el componente.

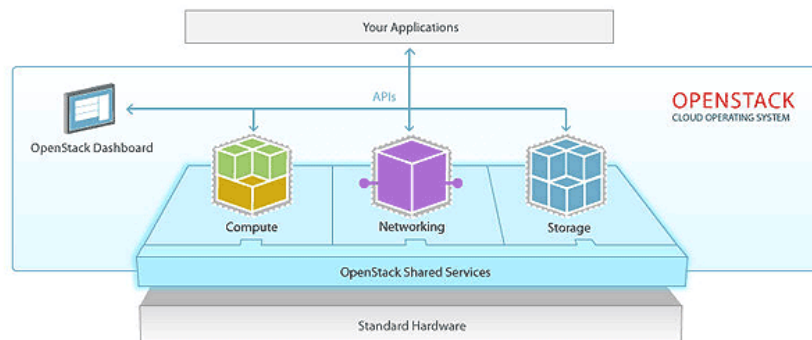


Figura 1: Arquitectura de OpenStack

4.1 Pruebas realizadas

En la actualidad, se cuenta con una infraestructura instalada OpenStack Juno, la cual es la penúltima versión estable de la infraestructura que fue liberada en octubre de 2014. Esta instalación se ha realizado con una topología monolítica nativa, esto es que todos los componentes se han montado en un solo nodo [Murazzo, Tinetti and Rodríguez, 2015]. La característica de esta topología es la inexistencia de comunicaciones fuera de la infraestructura, puesto a que todos los componentes se comunican en un ambiente de memoria compartida, por lo tanto se eliminan los retardos de las comunicación, sin embargo es necesario analizar las comunicación incurridas entre la infraestructura y el sistema operativo.

La principal desventaja de esta topología es la manera en la cual maneja la virtualización, pues no soporta hipervisores que manejen virtualización nativa. El hipervisor que se instala por defecto es QEMU [Qemu, 2015], por lo que el manejo de las instancias virtualizadas de los recursos no posee la mejor performance.

Para medir la performance de la virtualización en el cloud se han generado tres escenarios de prueba:

- *Escenario 1:* Ejecución Nativa – consiste en correr el algoritmo sobre la PC con Ubuntu 14.04 de 64 bits. Para ello se han usado tres arquitecturas diferentes:
 - ✓ Intel Dual Core con 4 Gb de RAM
 - ✓ Intel Core i3 con 8 Gb de RAM
 - ✓ Intel Core i5 con 8 Gb de RAM
- *Escenario 2:* Ejecución Virtualizada – consiste en correr el algoritmo sobre una maquina virtual Qemu con Ubuntu 14.04 de 64 bits como sistema operativo guest y virtualizado. En este caso se ha instalado la maquina virtual sobre las tres arquitecturas mencionadas en el escenario 1.
- *Escenario 3:* Ejecución en el Cloud – en este caso el algoritmo se corre sobre el cloud instalado. Para ello se a lanzado una instancia basada en un sabor con 8 Gb de RAM y 6 cores; la imagen que se usó fue una mini iso de Ubuntu 14.04 de 64 bits. En este caso por defecto se usa Qemu para manejar las instancias dentro del cloud.

El algoritmo que se usó para evaluar la performance de los escenarios fue un algoritmo secuencial de multiplicación de matrices de grado n , para este trabajo se ha trabajado con $n= 200, 500$ y 1000 . En la tabla 1 se pueden ver los resultados obtenidos de la ejecución realizada en estos tres escenarios.

| | Nativo | | | Virtualizado | | | Cloud | | |
|-----------|--------|------|-------|--------------|------|-------|-------|------|-------|
| | 200 | 500 | 1000 | 200 | 500 | 1000 | 200 | 500 | 1000 |
| Dual Core | 0.114 | 2.91 | 23.37 | 2.28 | 6.04 | 57.94 | 0.145 | 3.21 | 28.32 |
| Core i3 | 0.097 | 1.66 | 16.96 | 1.23 | 5.56 | 26.54 | 0.110 | 1.58 | 16.32 |
| Core i5 | 0.078 | 0.83 | 8.092 | 1.65 | 3.25 | 13.45 | 0.067 | 0.81 | 7.65 |

Tabla 1: Resultados de la ejecución (en seg.)

4.2 Análisis de Resultados

Como se puede ver en la tabla 1, la primera columna “Nativo” muestra los resultados de la ejecución del algoritmo en las tres arquitecturas, como era de esperar a medida que las características de arquitectura aumentan, los resultados de la ejecución mejora. En todos los casos resulta mas rápida la ejecución sobre un i5 por sobre las demás arquitecturas.

En la segunda columna “Virtualizado”, se muestran los resultados de la ejecución en el segundo escenario, esto es sobre la maquina virtual Qemu, como era de esperarse para todas las arquitecturas, la ejecución resulta más lenta que en forma nativa. Esto se debe fundamentalmente a que se ha usado un hipervisor que emula, o sea que es de tipo 2, esto genera una demora aun mayor por sobre los hipervisores tipo 1 pues trabajan en modo usuario.

En la tercera columna “Cloud”, se pueden ver los resultados obtenidos al correr los algoritmos sobre una instancia de OpenStack. Cabe aclarar que cuando se trabaja con una topología monolítica nativa, OpenStack corre como un servicio que se debe levantar, cuando esto sucede el sistema operativo pasa a ser OpenStack y el virtualizador gestiona de forma optimizada las instancias virtualizadas. Es por ello que aunque se esté usando el mismo hipervisor, que en este caso es Qemu, la administración de la virtualización es más rápida.

Como se puede ver en la tabla 1, los tiempos de ejecución del algoritmo cuando se usa un dual core son mayores cuando se usa el cloud que cuando se ejecuta en forma nativa, sin embargo en este caso la opción virtualizada tiene un muy pobre desempeño. Para el caso del procesador i3, se puede ver que los tiempos de ejecución mejoran en cuanto el valor de n aumenta. Y por último, para el procesador i5, los tiempos de ejecución son mejores sobre OpenStack.

Como una conclusión fruto de este análisis, se puede decir que para una topología monolítica de OpenStack los tiempos de ejecución mejoran, conforme mejora la arquitectura y la cantidad de cómputo que se realiza. Esto último se debe a que cuando la cantidad de procesamiento es poco, como en el caso de un n=200, los tiempos se ven afectados por la contextualización de la virtualización sobre OpenStack.

5 Conclusiones y Futuros Trabajos

El cloud computing es un nuevo paradigma que está cambiando la ubicación de las infraestructuras informáticas hacia Internet. Con ello se consigue que los usuarios no tengan que gestionar ni sus propios servidores ni tampoco su software, ahorrando de este modo energía, espacio físico y personal técnico. Además, los sistemas cloud ofrecen un gran rendimiento en términos de escalabilidad, mantenibilidad y procesamiento masivo de datos en entornos dinámicos y de necesidades cambiantes.

Desde el punto de vista académico, el contar con una infraestructura cloud privada resulta un tema interesante de investigación y evaluación de cuan buena es esta solución para comenzar a migrar los datos, aplicaciones y cómputo a un entorno virtualizado. Sin embargo la migración, tiene ventajas y desventajas; entre las ventajas sobresale la posibilidad de contar con una plataforma de cómputo basada en instancias virtualizadas de recursos. La desventaja más relevante de migrar al cloud es la pérdida de performance producida por los procesos de comunicación y la virtualización de los recursos.

En este sentido, el grupo de investigación ha instalado y configurado una infraestructura cloud privada con OpenStack, dejándola en instancia de prueba para la evaluación del impacto de la migración del cómputo a esta plataforma. Las pruebas realizadas sobre la infraestructura cloud instalada a arrojado resultados positivos, pues ha permitido evaluar cual es el impacto que los procesos de virtualización (en este caso particular, el proceso de emulación) sobre la ejecución de algoritmos en instancias virtualizadas de recursos. La conclusión más importante a la que se ha podido llegar es probar que cuando existe un cómputo intensivo el uso de infraestructura cloud brinda una mejor performance aun contar la ejecución nativa, que a priori debería ser la de mejor performance.

Se prevé realizar pruebas con otro tipo de aplicaciones, por ejemplo con mezcla de acceso a disco y procesamiento, y aplicaciones con mucho acceso a disco que invalide la caché, para probar cuanto influye la virtualización en estos casos.

La topología que se ha usado para instalar la infraestructura tiene como principal inconveniente la imposibilidad de seleccionar un hipervisor, pues instala Qemu por defecto. Para solucionar este problema se está trabajando en la migración a una topología dual-node, que permite trabajar con KVM como hipervisor, esta migración supone que la performance del algoritmo deberá aumentar pues al ser un hipervisor de tipo 1, la gestión de los recursos se realiza de forma más dinámica.

Bibliografía

- Huber, N., von Quast, M., Hauck, N., Kounev, S.. "Evaluating and Modeling Virtualization Performance Overhead for Cloud Environments". In CLOSER (pp. 563-573) (2011).
- Jiang, Q. "Open source IaaS community analysis. OpenStack vs OpenNebula vs Eucalyptus vs CloudStack". School of Information Technologies (2015).

- Mell, P., Grance, T. "The NIST definition of cloud computing". NIST Special Publication 800 – 145 (2011).
- Murazzo, M., Rodríguez, N., Medel, D., Chávez, S., Martin, A., Valenzuela, F. "Análisis de mejora en la escalabilidad de las infraestructuras de cloud computing". III Jornadas de Cloud Computing (2015).
- Murazzo, M., Tinetti, F., Rodríguez, N. "Despliegue de una Infraestructura Cloud Privada de Código Abierto". III Jornadas de Cloud Computing (2015).
- OpenStack. "OpenStack: The Open Source Cloud Operating System". URL: <http://www.openstack.org/software/>.
- QEMU: "QEMU, Open Source Processor emulator". URL: <http://www.qemu.org>.
- Rahma, F., Adji, T., Widyawan, W. "Scalability Analysis of KVM-Based Private Cloud for IaaS". International Journal of Cloud Computing and Services Science", (IJ-CLOSER), 2(4), 288-295 (2013).
- Rodríguez, N., Chávez, S., Martin, A., Murazzo, M., Valenzuela, F. "Interoperabilidad en cloud computing". WICC 2011. Rosario Argentina (2011).
- Rodríguez, N., Murazzo, M., Chávez, S., Guevara, M. "Arquitectura de cloud computing híbrida basada en tecnología open source". XX Congreso Argentino de Ciencias de la Computación (2014).
- Schubert, L., Jeffery, K., Neidecker-Lutz, B. "The Future of Cloud Computing: Opportunities for European Cloud Computing Beyond 2010". Expert Group Report. European Commission, Information Society and Media (2010).