

## Rendimiento de Cloud Computing para HPC en IaaS privados y públicos.

Brian Galarza<sup>1</sup>, Gonzalo Zaccardi<sup>1</sup>, Román Bond<sup>1</sup>, Federico Montes de Oca<sup>1</sup>, Eduardo Maxit, Jorge Osio<sup>1</sup>, David Duarte<sup>1</sup>, Martín Morales<sup>1,2</sup>, Diego Encinas<sup>1,3</sup>

<sup>1</sup>Proyecto de Investigación SimHPC - Programa TICAPPS - Instituto de Ingeniería y Agronomía - UNAJ

<sup>2</sup>Unidad CodApli - Facultad Regional La Plata - UTN

<sup>3</sup>Instituto de Investigación en Informática LIDI (III-LIDI) - Facultad de Informática - UNLP – Centro Asociado CIC

bgalarza@unaj.edu.ar, gzaccardi@unaj.edu.ar, rbond@unaj.edu.ar,  
federicomdo97@gmail.com, eduardomaxit5@gmail.com, josio@unaj.edu.ar,  
davito.duarte.22@gmail.com, martin.morales@unaj.edu.ar, dencinas@unaj.edu.ar

### Resumen

El objetivo de esta línea de investigación es el estudio del performance de las arquitecturas tipo cloud a través del despliegue de IaaS y utilización de IaaS públicos, en particular en el área de cómputo paralelo de altas prestaciones (HPC). Enfocando a la obtención de herramientas que permitan predecir la eficiencia del sistema ante posibles escenarios. Analizando los diferentes componentes del sistema que pueden influir en las prestaciones significativamente y pueden llegar a modelarse y/o configurarse.

**Palabras clave:** *Arquitecturas Multiprocesador. Cloud Computing. OpenStack. Sistemas de Archivos en clústers. Redes definidas por Software.*

### Contexto

Se presenta una línea de Investigación que es parte del Proyecto de Investigación “Simulación y tecnología en Cómputo de Altas Prestaciones (High Performance Computing, HPC) para aplicaciones de interés social” - SimHPC de la Universidad Nacional Arturo Jauretche (UNAJ), acreditado por resolución interna 148/18. Además, el proyecto aporta al Programa “Tecnologías de la información y la comunicación (TIC) en aplicaciones de interés social” – TICAPPS de la UNAJ.

### Introducción

Cloud Computing es un paradigma que ha estado en constante crecimiento, cada vez más compañías y grupos de investigación trabajan en conjunto con el fin de explotar las oportunidades ofrecidas por el mismo [1]. Dicho paradigma ofrece muchas ventajas, tales como el bajo costo de implementación, ya que no se necesitan computadoras de

última tecnología debido a que éstas trabajan conjuntamente (Clustering) con la posibilidad de escalar horizontalmente de manera sencilla. Además, hay software Open Source disponible para los nodos en el clúster como las infraestructuras Eucalyptus, OpenNebula, CloudStack u OpenStack integradas con GNU/Linux y compatibles, por ejemplo, con Amazon WebServices.

### **Despliegue de IaaS**

Las comunicaciones en Cloud Computing son una parte fundamental del paradigma que consisten en utilizar distintos nodos y lograr hacerlos funcionar conjuntamente.

Para lograr una comunicación sincronizada entre estos nodos se propone utilizar OpenStack [2] como también OpenNebula [3].

OpenStack es una plataforma de tecnología open source que utiliza recursos virtuales agrupados para diseñar y gestionar nubes privadas y públicas a través de múltiples servicios que, de manera coordinada, cumplen diferentes propósitos para lograr el correcto funcionamiento de, por ejemplo, una "Infraestructure as a service" (IaaS). Algunos de los servicios ofrecidos por OpenStack son: hypervisor (Nova), autenticación (Keystone), Imágenes (Glance), Dashboard (Horizon), Networking (Neutron) y block storage (Cinder). Según las necesidades se pueden requerir de ciertos servicios u otros. La Arquitectura básicamente consiste en dos tipos de nodos: "Compute Node" y "Controller Node". Se llaman Compute Node a todos aquellos que se encargan del procesamiento de servicios específicos mientras que Controller Node es aquel que comunica a cada uno de los anteriores [4] [5] [6].

Fuel es una herramienta open source desarrollada por Mirantis en la cual se ejecuta un script que permite configurar, de manera más amigable respecto a OpenStack, los recursos que se desean otorgar a la infraestructura, como la cantidad de nodos, los núcleos de procesador, la memoria RAM, entre otros [7].

Fuel trabaja con un nodo master el cual es el encargado de controlar a los nodos slaves que contendrán la infraestructura OpenStack. Es decir, desde el nodo Fuel Master se indican qué paquetes se van a instalar en cada nodo slave (Glance, Nova-Compute, Keystone, etc.) para luego en los slaves tener armados los nodos compute y controller, sin necesidad de realizar configuraciones manuales en cada uno de los mismos.

OpenNebula es un software de código abierto que permite el despliegue de IaaS. Busca reducir la complejidad generada por OpenStack y ofrece soporte con hipervisores tales como KVM y VMware vCenter.

OpenNebula clasifica a los nodos en dos tipos, Front – end los cuales entran en contacto con los usuarios y a su vez se comunican con los nodos de la infraestructura en los cuales se lanzarán las instancias y los nodos virtualizados los que a su vez deben contar con los paquetes correspondientes de storage, autenticación y networking para poder funcionar correctamente.

La implementación de estas infraestructuras ofrece ventajas en las cuales los clústeres virtualizados trabajan en conjunto ofreciendo un buen rendimiento a bajos costos y con posibilidad de escalabilidad al poder agregar mayor cantidad de nodos para procesamiento de manera sencilla.

## **Sistemas de Archivos Paralelos en clústeres**

Amazon Webservices [8], mediante el servicio EC2 (Amazon Elastic Compute Cloud) permite desplegar clústeres virtuales mediante instancias de VMs y almacenamiento para las mismas. Mediante este servicio y el correspondiente clúster conformado, se utiliza un sistema de archivos paralelo (PVFS2) que permite la gestión de datos particionados y distribuidos en los distintos nodos, mediante múltiples tareas de una aplicación ejecutada sobre el clúster.

PVFS2 utiliza una estructura cliente-servidor. En función de su rol, existen tres tipos de nodos dentro de un clúster con este sistema de archivos: servidores de datos, servidores de metadatos y clientes, en donde cada uno de los nodos puede cumplir los tres roles.

PVFS2 [9] contiene una herramienta interna que permite volcar información en logs durante la ejecución de tareas dentro del sistema de archivos. Por cada uno de los nodos, el administrador del clúster puede obtener información de depuración (GOSSIP) conformado por registros de debug, de acceso, contadores de rendimiento y errores producidos durante la ejecución. Referido a los contadores de rendimiento, se obtienen mediante el software de monitoreo Atop. Esta herramienta permite obtener reportes de la actividad de los procesos y la utilización de los diferentes recursos del sistema (memoria, disco, CPU, red, etc) [10].

Asimismo, es posible especificar los datos a obtener en función de las distintas capas de PVFS2, del rol específico de cada nodo (cliente-servidor), así como también de otros factores relacionados con el funcionamiento interno y

operaciones asociadas a la gestión de archivos.

Teniendo conocimiento acerca del funcionamiento del código fuente correspondiente a PVFS2 y del modo en el que realiza la escritura de registros en los logs, resulta factible realizar modificaciones que permitan obtener otro tipo de parámetros de interés a partir de esta herramienta.

Finalmente se propone el análisis, uso y configuración de distintas herramientas no invasivas para determinar la performance del sistema de archivos en clústeres virtuales.

## **Redes definidas por Software**

Las redes definidas por software (SDN) son un paradigma de gestión y administración de redes por medio de software que permiten tener un control más flexible con respecto al control del tráfico de datos por medio de hardware ya que permite cambiar en tiempo real las normas y políticas establecidas en la red.

Mininet[11] es un emulador de redes SDN open source que permite generar tráfico artificial entre nodos virtuales de la red.

Se ha llevado a cabo el montaje de un laboratorio SDN utilizando Mininet junto con Amazon WebServices con el fin de analizar el comportamiento de este nuevo paradigma de redes de computadoras en un entorno de Cloud Computing.[12]

## **Líneas de Investigación, Desarrollo e Innovación**

### **Temas de Estudio e Investigación**

- Arquitecturas multiprocesador para procesamiento paralelo: multiprocesador de memoria

compartida, multiprocesador on-chip de memoria distribuida. Multicore, Clusters, Clusters de multicore. Grid. Cloud.

- Plataformas de software para implementar y administrar Clouds públicos, privados e híbridos.
- Sistemas de Archivos Paralelos.

## Resultados y Objetivos

### Investigación experimental

- Implementación de un IaaS encargado de realizar operaciones en procesamiento paralelo aumentando la eficiencia y reduciendo los costes generados.
- Implementación de OpenStackDashboard y de un sistema desarrollado para poder controlar/administrar de manera visual (web) y más básica cada uno de los servicios.
- Implementación de OpenNebula en un sistema con las mismas características que el implementado por OpenStack con el fin de poder realizar pruebas en entornos similares.
- Utilización de Fuel para administrar OpenStack como sistema de administración de nube (Cloud Computing) a partir de la infraestructura de 2 nodos compute y el controller [13].
- Lograr escalabilidad agregando nuevos nodos compute a la infraestructura obteniendo un mayor performance en el sistema.

- Análisis del rendimiento de un Cloud privado en la ejecución de instancias personalizadas.
- Ejecutar sobre el modelo de nodos implementado en OpenStack la distribución GNU/Linux Hetnux, desarrollada en la UNAJ en el marco del programa “Universidad, Diseño y Desarrollo Productivo 2014”.
- Ejecutar diferentes benchmarks en la infraestructura desplegada sobre OpenStack y OpenNebula para así comparar los resultados obtenidos de ambas infraestructuras y poder realizar un análisis del rendimiento en cada caso.
- Análisis y configuración de clústeres virtuales.
- Análisis y configuración de herramientas no invasivas para la obtención de métricas en las distintas capas de software de los sistemas de archivos paralelos.

### Formación de Recursos Humanos

Dentro de la temática de la línea de I/D se participa en el dictado de la carrera de Ingeniería Informática de la UNAJ. También aportan trabajos de alumnos de las materias Sistemas Operativos 1, Redes de Computadoras 2, Programación en Tiempo Real y Organización y Arquitecturas de Computadoras.

Durante 2019 se han realizado 6 publicaciones nacionales. Además, se encuentran en desarrollo y concluidas varias Prácticas Profesionales Supervisadas (PPS) con las que concluyen sus estudios los alumnos de Ingeniería en Informática.

En esta línea de I/D existe cooperación a nivel nacional. Hay 2 investigadores realizando carreras de postgrado, 1 becario de grado (EVC CIN) y 2 alumnos avanzados de grado colaborando en las tareas.

## Referencias

1. Kondo, D., Javadi, B., Malecot, P., Cappello, F., Anderson, D. P.: "Cost-benefit analysis of Cloud Computing versus desktop grids". In: IPDPS '09 Proceedings. IEEE International Symposium on Parallel and Distributed Processing. Washington, USA (2009).
2. OpenStack Cloud Software: Open source software for building private and public clouds. <http://www.openstack.org>. Febrero 2015.
3. OpenNebula. <https://opennebula.org/>. Febrero 2019
4. Galarza, B.; Tuamá, C.; Zaccardi, G.; Encinas, D.; Morales, M. "Implementaciones de Cloud Computing y aplicaciones en el ámbito universitario". 1° Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2013). Ciudad de Córdoba, Argentina.
5. Zaccardi, G.; Galarza, B.; Encinas, D.; Morales, M. "Implementación de Cloud Computing utilizando OpenStack". 2° Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2014). Ciudad de San Luis, Argentina.
6. Galarza, B.; Zaccardi, G.; Encinas, D.; Morales, M. "Análisis de despliegue de una IaaS utilizando Openstack". XXI Congreso Argentino de Ciencias de la Computación (CACIC 2015). Ciudad de Junín, Argentina.
7. OpenStack Deployment Fuel. <https://www.mirantis.com/products/mirantis-openstack-software/openstack-deployment-fuel/>. Febrero 2016
8. Amazon Web Services (AWS)-Cloud Computing Services. <https://aws.amazon.com> Marzo 2019
9. T. PVFS2, "PVFS 2 File System Semantics Document," tech. rep., PVFS Development Team, 2015
10. Atop Tool. <https://www.atoptool.nl/index.php>
11. Mininet. <http://mininet.org>
12. Montes de Oca, F.; Galarza, B.; Morales, M.; Encinas, D. "Redes Definidas por Software en Entorno de Cloud Computing". 6° Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2018). Mar del Plata, Argentina.
13. Zaccardi, G.; Galarza, B.; Morales, M.; Encinas, D. "Despliegue y ejecución de un cloud privado". 4° Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2016). Ciudad de Salta, Argentina.