MongoDB en ambiente Cloud Híbrido con OpenStack

Adriana E. Martín[1], Susana Chávez[3], María Murazzo[2], Nelson Rodríguez[4], Adriana Valenzuela[5],

Departamento e Instituto de Informática - F.C.E.F. y N. - U.N.S.J.

Complejo Universitario Islas Malvinas, Av. I. de la Roza 590 (O), Rivadavia, SJ CP: 5402

[1]arianamartinsj@gmail.com [2]schvz@.gmail.com [3]marite@unsj-cuim.edu.ar [4]nelson@iinfo.unsj.edu.ar

[5]franciscaadriana.valenzuela@gmail.com

Resumen

Debido a los grandes cambios que vienen sufriendo las Tecnologías de la Información (TI) por el gran flujo de datos y operaciones que deben manejarse dentro de las organizaciones, es que surge Cloud Computing (CC) como un modelo que tiende a proveer servicios que utilizan eficientemente estos recursos.

Con el surgimiento de los sistemas distribuidos en la web, el software como servicio (SaaS), los servicios en el Cloud y los constantes requerimientos de procesamiento y análisis a gran escala de enormes cantidades de datos, los sistemas tradicionales de base de datos son insuficientes. Las Bases de Datos NoSQL llenan una importante carencia de las bases de datos relacionales en cuanto a la capacidad que estas tienen en escalabilidad, distribución y manejo de datos no estructurados. Estas 3 características son cada día más relevantes debido precisamente al avance de CC, a los múltiples y diversos servicios cuyo crecimiento y replicación distribuida son extremadamente necesarios.

Palabras claves: Cloud Computing, Open Stack, NoSQL,

Contexto

El presente trabajo se encuadra dentro del área de Base de Datos y Minería de Datos, y se enmarca dentro del proyecto de investigación "Cloud Computing con herramientas libres para evaluación de modelos de despliegue híbrido" que tiene como unidades ejecutoras al Departamento e Instituto de Informática de la FCEFyN de la UNSJ. Los trabajos que se realizan en el citado proyecto tienden a analizar, desarrollar y/o probar aplicaciones con herramientas libres sobre Clouds híbridos.

Introducción

Cloud Computing

Permite aumentar el número de servicios basados en la red, generando beneficios tanto para los proveedores, que pueden ofrecer de forma más rápida y eficiente un mayor número de servicios, como para los usuarios que tienen la posibilidad de acceder a ellos, disfrutando de la 'transparencia' e inmediatez del sistema y de un modelo de pago por consumo. De este modo el consumidor ahorra los costes salariales o los costes en inversión económica (locales, material especializado, etc.).

Cloud Computing, se apoya sobre una infraestructura tecnológica dinámica que se destaca por una gran automatización, movilización de los recursos, una capacidad de adaptación para atender a una demanda variable, como así también virtualización avanzada y un precio flexible en función del consumo realizado, evitando la piratería y software fraudulento lo cual es bastante ventajoso[1]

Como consecuencia de esta expansión de Internet, el uso de los distintos servicios que los usuarios disponen en el cloud ha provocado la generación de grandes cantidades de datos, con características

diferentes a lo que se estaba acostumbrado. Los datos en línea generados a través de foros, wikis, redes sociales, originados por usuarios on line pueden dar lugar a contenidos con sólo navegar por una página. De hecho Facebook mantiene estos datos no transaccionales, para la determinación de los patrones de comportamiento de los usuarios. Con el fin de gestionar estas grandes cantidades de datos, que no tienen en general una estructura determinada, surgen las Bases de Datos NoSQL.

Líneas de Investigación, Desarrollo e Innovación

CC involucra una gran variedad de tecnologías, además el modelo de despliegue híbrido incluye al modelo público y al privado. Esto determina que los investigadores del grupo tengan que resolver múltiples problemas.
Aunque afortunadamente en la actualidad se cuenta con Cloud Computing públicos que permite desplegar las aplicaciones PaaS a un costo bajo, e incluso, sin costo alguno para el periodo de aprendizaje y pruebas.

Bases de Datos Nosql.

El termino NoSQL hace referencia a sistemas de Base de Datos que no son DMBS tradicionales. Los conceptos fundamentales de este tipo de Bases de Datos NoSQL son la escalabilidad, la distribución y el manejo de datos no estructurados. Características que son primordiales frente al gran avance de CC, y a los múltiples servicios que requieren replicación distribuida. De este modo se puede asegurar que la información en el Cloud siempre esté disponible, y además gestionar el gran crecimiento de información y su variabilidad de formatos.
Las bases de datos NoSQL son sistemas de almacenamiento de información que no cumplen con el esquema entidad–relación. Tampoco utilizan una estructura de datos en forma de tabla donde se van almacenando los datos sino que para el almacenamiento hacen uso de otros formatos.
Entre las ventajas de las NoSql podemos encontrar:
- No admite consultas complejas o join por lo cual las operaciones son simples
- Se ejecutan en máquinas con pocos recursos, a diferencia de los sistemas basados en SQL, ya que requieren de poco cómputo por lo tanto se pueden montar en máquinas de un coste más reducido.
- Escalabilidad horizontal: es decir tiene una arquitectura que posee la capacidad de distribuir y cargar los datos lo más uniformemente posible, en tantos servidores como sea viable, con una arquitectura shared-nothing.
- Pueden manejar gran cantidad de datos: ya que utiliza una estructura distribuida, en muchos casos mediante tablas Hash.
- No genera cuellos de botella: el principal problema de los sistemas SQL es que necesitan transcribir cada sentencia para poder ser ejecutada, y cada sentencia compleja requiere además de un nivel de ejecución aún más complejo, lo que constituye un punto de entrada en común, que ante muchas peticiones puede hacer al sistema cada vez más lento. [2] [3]
De entre las muchas posibilidades de Bases de Datos NoSql disponibles se ha optado por MongoDb, elegida por ser una tecnología de código abierto muy bien aceptada, de uso generalizado, y cuyo futuro es bastante prometedor.

MongoDB.

Desarrollado bajo el concepto de código abierto por la empresa de software 10gen en el 2007 y lanzado en 2009 como un producto independiente publicado bajo la licencia de código abierto AGPL(Affero General Public License), desde entonces se han lanzado varias versiones.

MongoDB guarda la estructura de los datos en documentos tipo JSON con un esquema dinámico llamado BSON, no existe un esquema predefinido. Los elementos de los datos son llamados documentos y se guardan en colecciones, las cuales pueden tener un número indeterminado de documentos. Comparando con una base de datos relacional, se puede decir que las colecciones son como tablas y los documentos son archivos. La diferencia es que en una base de datos relacional cada archivo en una tabla tiene la misma cantidad de campos, mientras que en MongoDB cada documento en una colección puede tener diferentes campos. En un documento, se pueden agregar, eliminar, modificar o renombrar nuevos campos en cualquier momento, ya que no hay un esquema predefinido. MongoDB soporta la búsqueda por campos, consultas de rangos y expresiones regulares. El sistema permite al usuario personalizar una consulta en tiempo real y pueden devolver un campo específico del documento pero también puede ser una función definida por el usuario. Trabaja bajo un sistema de indexación similar a los de las base de datos relacionales. Soporta el tipo de replicación maestro-esclavo. El maestro puede ejecutar comandos de lectura y escritura. El esclavo puede copiar los datos del maestro y sólo se puede usar para lectura o para copia de seguridad, pero no se pueden realizar escrituras. El esclavo tiene la habilidad de poder elegir un nuevo maestro en caso de que se caiga el servicio con el maestro actual. MongoDB se ejecuta en múltiples servidores, balanceando la carga y/o duplicando los datos para poder mantener el sistema funcionando en el caso que exista un fallo de hardware. Utiliza la función MapReduce para procesamiento de datos por lotes y operaciones de agregación. Incluso permite la utilización de índices geoespaciales, que si bien no es necesario para las operaciones normales o de desarrollo de aplicaciones, puede ser útil para solucionar problemas y para una mayor comprensión. [4]

Incluso MongoDB se puede utilizar con un sistema de archivos, por su capacidad para el balanceo de carga y replicación de datos utilizando múltiples servidores para el almacenamiento de archivos, mediante el uso de GridFS. Cuando los archivos exceden el tamaño límite del documento BSON que es de 16 MB, divide el archivo y almacena cada uno de esos trozos en un documento aparte. [5][6]

MongoDB tiene drivers oficiales para los siguientes lenguajes de programación: C, C++, C#, Erlang, Haskell, Java, Java Script, Lisp, node.JS, Perl, PHP, Python, Ruby, Scala. El código binario está disponible para los sistemas operativos Windows, Linux, OS X y Solaris.

Un aspecto importante cuando se decide instalar un ambiente cloud es contar con infraestructura de almacenamiento solida y escalable de almacenamiento que ofrezca disponibilidad, eficiencia y protección de los datos. Con este objetivo OpenStack cuenta con la capacidad de brindar Database as a Service (DaaS).

OpenStack

Se comenzó a desarrollar en 2010 y consiste de un conjunto de herramientas de software que permiten construir y administrar cloud privadas, públicas e híbridas mediante la colaboración de proyectos OpenSource para lograr obtener un IaaS escalable y robusto. [7]

Trove [8]es una API que permite acceder a Database as a Service para OpenStack. Este servicio de Base de datos proporciona una funcionalidad de aprovisionamiento en el cloud escalable y seguro tanto para motores de las Base de datos relacionales como las NoSql como es el caso de MongoDB. Los usuarios pueden utilizar rápida y fácilmente las características de la base de datos sin la carga que supone el manejo de tareas

administrativas complejas. Los usuarios del cloud y administradores de la Base de datos pueden aprovisionar y gestionar tantas instancias de Base de datos como se necesiten.

El DaaS proporciona recursos de aislamiento a un alto nivel de rendimiento, y automatización de tareas administrativas complejas como son el despliegue, configuración, aplicación de parches, copias de seguridad, restauraciones y monitorización. Y además el DaaS incluye los siguientes componentes, los cuales se pueden ver gráficamente en la figura 1:

- Python-troveclient. Una línea de comandos que se comunica con el componente trove-api.
- Trove-api. Proporciona una API RESTful nativa de OpenStack, que soporta JSON, para aprovisionar y gestionar las instancias de Tove.
- Trove-conductor. Corre en el host, y recibe mensajes de las instancias de los huéspedes que quieren actualizar información en el host.
- Trove-taskmanager. Instrumentos del sistema para de flujos que ayuda al aprovisionamiento de instancias, gestión y manejo del ciclo de vida de las instancias, y rendimiento de operaciones sobre las instancias.
- Trove-guestagent. Corre dentro del huesped de la instancia. Gestiona y realiza operaciones en la misma Base de Datos.

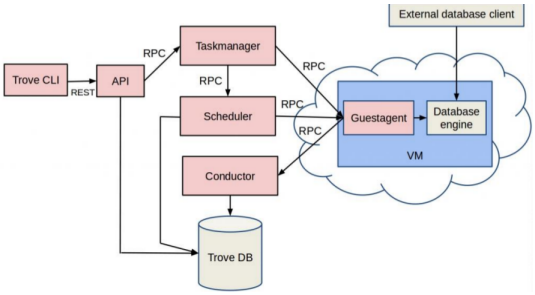

Figura 1: Diagrama de Trove

La API de Trove viene nativa en la decima distribución de OpenStack llamada Juno y que fue liberada en Octubre de 2014. [8]

Resultados y Objetivos

Resultados Obtenidos

El grupo de investigación viene trabajando sobre Cloud Computing desde hace cuatro años y los dos últimos en el proyecto de investigación marco de este trabajo. Se han realizado varias publicaciones en el área y en áreas afines al proyecto entre las que encontramos:

Cinco (5) trabajos de investigación en diferentes Congresos Seminarios y Jornadas: [9][10][11][12][13].

Un (1) artículo en la Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología (TE&ET). [14]

Dos (2) artículos en Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação. http://revistas.setrem.com.br/index.php/reabtic/issue/view/1. [15][16]

Objetivos

El objetivo del grupo de investigación es realizar el análisis de la escalabilidad, eficiencia, interoperabilidad de la tecnología de Cloud Computing híbrido utilizando herramientas libres.

En particular el objetivo de este trabajo es montar un DaaS NoSql en un Cloud Hibrido sobre Open Stack, que ofrezca funcionalidad distribuida y la posterior evaluación y análisis de los parámetros correspondientes.

Formación de Recursos Humanos

El equipo de trabajo está compuesto por los siete (6) docentes-investigadores, y cuatro (4) alumnos adscriptos.

A la fecha se ha aprobaron una tesina de grado sobre una sobre GoogleAppEngine, otra sobre SaaS sobre tecnologías PHP.

Se obtuvo una beca de estudiante avanzado, cuyo tema es: Instalación y puesta a punto de una arquitectura Cloud Open Source, Una tesis de Especialización en Redes cuyo tema es: CC híbrido.

Además se están realizando dos (2) tesinas de licenciatura una sobre Mobile Cloud Computing, otra sobre Hybrid Cloud Computing, y Una tesina sobre MongoDB en Python utilizando el framework Django

Se espera realizar también dos (2) tesis de maestría una sobre Metodologías de desarrollo aplicadas a SaaS y sobre bases de datos NoSQL y aumentar el número de publicaciones. También se prevé la divulgación de varios temas investigados por medio de cursos de postgrado y actualización o publicaciones de divulgación.

Referencias:

[1]http://www.3digits.es/sistemas/Cloud_Computing_-_Servicios_en_la_nube.html

[2]: Burtica, R., Mocanu, E.M., Andreica, M.I., Ţăpuş, N., Practical application and evaluation of no-SQL databases in Cloud Computing IEEE International Systems Conference (2012).

[3]http://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf

[4] https://github.com/uokesita/the-little-mongodb-book

[5]http://docs.mongodb.org/manual/reference/mongostat/

[6] Plugge, Eelco; Membrey , Peter; Hawkins,Tim (2010). The Definitive Guide to MongoDb The Nosql Database for Cloud and Desktop Computing. Apress. p. 306. ISBN 978-1-4302-3051-9.

[7] http://www.openstack.org/software

[8] http://www.openstack.org/software/juno

[9] Murazzo, Rodriguez, Chavez, Valenzuela, Martin, Guevara. "Despliegue de una arquitectura de Cloud Computing híbrida Open Source. Congreso Nacional de Ingeniería Informática/Sistemas de Información". UNSL. Congreso Nacional de Ingeniería Informática/Sistemas de Información. Nov. 2014. San Luis, Argentina. ISSN: 2346-9927.

[10] Murazzo, Rodriguez, Guevara. "Impacto de la implementación de 802.11e para tráfico heterogéneo en MANET".UNSL. Congreso Nacional de Ingeniería Informática/Sistemas de Información. Nov. 2014. San Luis, Argentina. ISSN: 2346-9927

[11] Rodriguez, Murazzo, Chavez, Guevara." Arquitectura de Cloud Computing híbrida basada en tecnología Open Source. VI Workshop Innovación en Sistemas de Software (WISS)." XX CACIC 2014. La Matanza. Bs As. Oct.2014. ISBN: 978-987-3806-05-6

[12] Murazzo, Rodriguez,Sheffer, "Guevara. Soporte de QoS con 802.11e para tráfico heterogéneo en ambientes MANET. Workshop Arquitectura, Redes y Sistemas Operativos (WARSO)." XX CACIC 2014. La Matanza. Bs As. Oct.2014. ISBN: 978-987-3806-05-6

[13] Rodriguez, Valenzuela, Murazzo, Martin, Chavez, Villafañe, González. "Cloud Computing con herramientas libres para evaluación de modelos de despliegue híbrido". XVI WICC 2014.Ushuaia. Tierra del Fuego.

[14] Rodriguez, Valenzuela, Villafañe, Murazzo, Chavez, Martin. "Una propuesta para la incorporación de Cloud Computing en la currícula de grado". TE & ET. UNLP. La Plata, 2014 ISSN1850-9959

[15] Rodriguez, Valenzuela, Murazzo, Chavez, Martin, Villafañe, Gonzalez. "Análisis de los parámetros de performance y escalabilidad para Clouds híbridos"

[16] Murazzo, Rodriguez, Scheffer, Guevara "Provisión de QoS a tráfico heterogéneo mediante 802.11e para MANET".