Utilización de NoSQL para resolución de problemas al trabajar con cantidades masivas de datos

Giovanni Daián Róttoli[1, 2], Marcelo López Nocera[1], Ma. Florencia Pollo-Cattaneo[1,2]

[1] Grupo de Estudio en Metodologías de Ingeniería de Software (GEMIS). Facultad Regional Buenos Aires. Universidad Tecnológica Nacional. Argentina.
[2] Ingeniería en Sistemas de Información, Facultad Regional Concepción del Uruguay. Universidad Tecnológica Nacional. Argentina
{flo.pollo, gd.rottoli}@gmail.com, zappapet@yahoo.com

Resumen

Las bases de datos relacionales (RDBMS) se han utilizado tradicionalmente para resolver una amplia variedad de problemas asociados a datos de diversa naturaleza. Sin embargo, con el advenimiento de Big Data, se han presentado nuevos desafíos que esta arquitectura no ha podido resolver eficientemente. Dicha situación dio origen a nuevas tecnologías que no utilizan SQL como lenguaje de consulta y además plantean nuevas arquitecturas para el modelado de los datos. Son conocidas genéricamente bajo el nombre "NoSQL" y son alternativas que podrían resolver estas cuestiones asociadas con cantidades masivas de datos. El presente trabajo lleva adelante un estudio de campo para ver en qué casos se obtiene mejor resultado utilizando cada una de estas innovadoras tecnologías.

Palabras clave: SQL, Big Data, NoSQL, Persistencia Políglota.

Contexto

El proyecto planteado articula líneas incipientes de trabajo del Grupo de Estudio en Metodologías de Ingeniería de Software (GEMIS) de la Facultad Regional Buenos Aires (FRBA) y Concepción del Uruguay (FRCU) de la Universidad Tecnológica Nacional (UTN)

Introducción

Las bases de datos tradicionales, de arquitectura relacional, que usan el lenguaje SQL, y frecuentemente englobadas bajo la abreviatura RDBMS, parecen no ofrecer soluciones eficientes para un variado universo de nuevos problemas relacionados con el tratamiento de datos masivos, conocidos genéricamente como Big Data (por caso, el análisis en línea de los datos recabados de las redes sociales). Esto provocó, entre otras cosas, el advenimiento de NoSQL [1].

El concepto NoSQL tiene su auge en el año 2009, y se refiere a todas aquellas tecnologías de bases de datos que no utilizan el lenguaje ANSI SQL para sus consultas [1]. Se trata generalmente de proyectos de código abierto, que corren en grupos de servidores, utilizan una arquitectura de procesamiento distribuido, tienen modelos de datos distintos del relacional tradicional, permitiendo el uso

de agregados (listas, registros, etc.), y operan sin esquemas, permitiendo agregar o quitar elementos a la base de datos de manera sencilla, sin que esto represente un problema [1]. Estas características hacen posible palear las principales problemáticas aportadas por la masividad de los nuevos conjuntos de datos, entre las cuales se encuentra:

1. El bajo rendimiento para grandes volúmenes de datos [1][6][7][8].
2. La discordancia de la impedancia (los datos en memoria tienen una estructura distinta a la que se almacena en la base de datos física) [1][5].
3. La necesidad de escalado del almacenamiento físico, ya sea aumentando la capacidad de los servidores, lo cual es caro y limitado, o utilizando servidores colaborativos, lo cual no es soportado por las bases de datos tradicionales [2][3][6][7][8][10].
4. La imposibilidad de las bases de datos relacionales de utilizar estructuras de datos complejas anidadas, lo cual surge a partir de la necesidad de modelar datos de estructuras poco usuales que, si se modelaran de una manera clásica, por un lado no resultaría en un modelo fiel de la realidad en cuestión, y por otro disminuiría la eficiencia total de las consultas de manera considerable [1][4][6][7][8].

NoSQL engloba una gran cantidad de alternativas que operan bajo paradigmas completamente distintos y lenguajes de consulta muy variados. Ante tal gran abanico de posibilidades, las grandes empresas, como Google[1] o Amazon[2] , optan por utilizar estas tecnologías, que inclusive desarrollan por su cuenta para adaptarlas exclusivamente a sus necesidades particulares.

Entre las opciones NoSQL, se pueden destacar cuatro grupos o tipos principales, los cuales se diferencian entre sí por el paradigma de modelado de datos que utilizan. Estos son: las bases de datos de Clave-Valor, las de Familia de Columnas, las Documentales y las Gráficas, cada una con sus propias particularidades, ventajas y desventajas a considerar a la hora de elegir por alguna de ellas. [2][4][5][6][7][8].

Por otro lado, se debe tener en cuenta que la encapsulación de servicios puede ayudar al cambio de las tecnologías de almacenamiento de datos a medida que las necesidades y evolucionan. La separación por capas de las partes de las aplicaciones permite introducir NoSQL en una aplicación preexistente, pudiendo además coexistir arquitecturas SQL y NoSQL, aprovechando las ventajas de cada una de ellas. Esto último se conoce como persistencia políglota [5] y [9], es decir, el uso de diferentes almacenamientos de datos en distintas circunstancias. Como ejemplo, podemos mencionar la utilización de una base de datos NoSQL Gráfica para mantener las relaciones de compras entre usuarios y productos, y una SQL para mantener los datos de los usuarios. Otro ejemplo, consistiría en la utilización de una base de datos Documental para guardar historiales médicos, debido a su falta de esquemas, y una base de datos Clave-Valor para vincular los pacientes con datos sobre su habitación, médico a cargo, u otros datos, debido a su rapidez de consulta y sencillez en el manejo de datos simples.

Entre las principales características de la persistencia políglota, podemos enumerar [9]:

• La implicación de diferentes tecnologías de datos para manejar las diversas necesidades de almacenamiento de los mismos.

[1] Google - www.google.com
[2] Amazon - www.amazon.com

• La aplicación de dicha arquitectura en la totalidad de los datos de una empresa o para un subconjunto de ellos.
• La reducción del impacto de los cambios en la totalidad del sistema, al encapsular los distintos servicios de bases de datos.
• El aumento de la complejidad de las aplicaciones al necesitar manipular diversidad de lenguajes de consulta y particularidades de los motores de bases de datos.

Por todos estos motivos y para muchas situaciones que lo requieran, comenzar a utilizar motores de bases de datos NoSQL, o bien, persistencia políglota parecen ser opciones más que satisfactorias para las organizaciones. Sin embargo, el impacto de realizar estos cambios puede resultar en costos elevados, problemas de rendimiento y otros relacionados a la seguridad de los datos, debido a la necesidad de adaptar el modelo de datos actual a los diferentes paradigmas NoSQL[2][7][8].

El presente proyecto, propone el estudio del comportamiento de las bases de datos NoSQL al ser utilizadas con un modelo de datos diferente, para conocer el impacto de ello, esperando encontrar evidencias que indiquen que un modelo políglota permitiría mitigar dicho impacto de una mejor manera.

Líneas de Investigación, Desarrollo e Innovación

En el último tiempo, con la llegada de las nuevas tecnologías de Bases de Datos y la tendencia NoSQL, muchas empresas quieren migrar sus datos a estas plataformas por diferentes motivos [9].

Muchas veces, diseñar una estructura de bases de datos NoSQL que se adapte a la estructura de los datos actual, implica sacrificar ciertas características como el rendimiento, la normalización de las tablas, etc., para que ambas arquitecturas sean totalmente compatibles [5] y [6].

Surgen así interrogantes como ¿cuál es el rendimiento entre las bases de datos NoSQL si se mantiene la estructura de los datos entre una y otra? y ¿qué tan beneficioso es mantener un modelo "genérico" entre las distintas bases de datos y aprovechar solamente las características de los motores? Una organización que se encuentra en vías de crecimiento, se plantea estas preguntas, siendo la incertidumbre sobre el futuro lo que le dificulta tomar una decisión sobre la estructura de sus datos.

Por ello, se propone realizar una serie de pruebas con datos de distinta naturaleza, a los que se los modelará de manera relacional y según los distintos paradigmas NoSQL (Documentales, Gráficas, Clave-Valor y Familia de Columna), y cada uno de dichos modelos se implantará en motores de bases de datos tanto relacional, como es PostgreSQL[3], y NoSQL, como son MongoDB[4] (Documentos), Cassandra[5] (Familia de Columnas), Redis[6] (Clave-Valor) y Neo4J[7] (Gráfos).

De esta manera, se realizará un modelo adecuado para las bases de datos documentales y, si es posible, se trasladará esa estructura de datos hacia las demás tecnologías. Así mismo, este procedimiento se repetirá con todos los tipos de bases de datos nombrados anteriormente.

Para llevar a cabo el proyecto propuesto, se plantean las siguientes actividades:

[3] PostgreSQL – http://www.postgresql.org/
[4] MongoDB – http://www.mongodb.org/
[5] Cassandra – http://cassandra.apache.org/
[6] Redis – http://redis.io/
[7] Neo4J – http://neo4j.com/

1. Obtención de datos característicos: por lo menos 3 juegos de datos de distinta naturaleza. Por ejemplo, datos correspondientes a compras realizadas por personas, datos médicos de pacientes, sus historiales y relaciones con médicos, y datos de redes sociales.

2. Modelado

2.1. Modelado de los datos de forma relacional, documental, gráfica, clave-valor, y familia-columnas.

2.2. Traslado de cada modelo a los diferentes motores de base de datos SQL y NoSQL.

2.3. Pruebas mediante consultas complejas (que requieran la utilización de operaciones como Inner Join por ejemplo), y en cada uno de los motores para cada uno de los modelos desarrollados. Obtención de tiempos de ejecución de las consultas.

3. Análisis de Datos

3.1. Determinar si existen casos donde una estructura de tal característica no influye en la eficiencia de las consultas bajo un determinado paradigma.

3.2. Determinar si las estructuras de datos resultantes resultan comprensibles y la utilización de los datos no ocasionaría problemas futuros.

Resultados y Objetivos

Mediante la ejecución del procedimiento descripto en el apartado anterior, se obtendrán los tiempos de consulta correspondientes a cada uno de los modelos de datos en los distintos motores de bases de datos.

A partir del análisis de los mismos, se podrá determinar si la migración de los datos desde una base de datos SQL o NoSQL a otra, puede realizarse (aunque sea en una primera instancia) sin mayores modificaciones de la estructura de los mismos.

Se pretende además, confirmar que un modelo políglota sería la mejor alternativa a adoptar ante un escenario plural, para aprovechar las características de los distintos motores, o bien para realizar las migraciones de datos de manera modular.

Formación de Recursos Humanos

Este proyecto busca tanto la obtención de nuevos conocimientos como la motivación de los implicados para que asciendan dentro del escalafón de la carrera de investigadores. El grupo de trabajo se encuentra integrado por dos investigadores formados y un investigador en formación. Además se encuentra en desarrollo un Trabajo Final de Especialidad.

Se pretende formar especialistas en el análisis de adopción de procesos vinculados con la Ingeniería de Software.

Finalmente, en el marco de este proyecto de investigación se prevé la radicación de una Tesis de Maestría en Ingeniería en Sistemas de Información.

Referencias

[1]. P. Sadalage, M. Fowler. NoSQL Distilled, A Brief Guide 1to the Emerging World of Polyglote Persistence. Addison-Wesley, Boston, USA, 1st. Edition, 2013

[2]. R. Hecht. NoSQL Evaluation. International Conference on Cloud and Service Computing. ISBN:978-1-4577-1637-9.P.336-341.2011.
URL:
http://rogerking.me/wp-content/uploads/2012/03/DatabaseSystemsPaper.pdf (verificado el 23/02/2015)

[3]. D. López. “Análisis de las posibilidades de uso de Big Data en las organizaciones”. Universidad de Cantabria, Santander, España, 2012. URL: http://repositorio.unican.es/xmlui/bitstream/handle/10902/4528/TFM%20-%20David%20L%C3%B3pez%20Garc%C3%ADaS.pdf?sequence=1 (Verificado el 27/02/2015)

[4.] A. Ramírez, H. Helio, C. Herrera, J. Francined. Un viaje a través de bases de datos espaciales. NoSQL: Redes de ingeniería, Univ. Distrital Francisco J de Caldas, Bogotá, Colombia, vol.4, no2, págs. 35-47, agosto-diciembre 2013 URL: http://revistas.udistrital.edu.co/ojs/index.php/REDES/article/download/5923/7426 (verificado el 21-02-2015)

[5]. A. Nayak, A. Poriya, D. Poojary. “Types of NOSQL Databases and its Comparison with Relational Databases”. International Journal of Applied Information Systems (IJAIS) – ISSN: 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 5– No.4, March 2013 URL: http://research.ijais.org/volume5/number4/ijais12-450888.pdf (verificado el 21-02-2015)

[6]. C. Strauch, W. Kriha. “NoSQL databases”. 2011. URL: http://www.christof-strauch.de/nosqldbs.pdf (verificado el 21-02-2015)

[7]. H. Del Busto, G. Hansel, O. Enríquez. “Bases de datos NoSQL”. Revista Telemática, vol. 11, no 3, 2013 URL: http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/74/74 (verificado el 21-02-2015)

[8]. F. Bugiotti, L. Cabibbo. “A Comparison of Data Models and APIs of NoSQL Datastores”. Dipartamento di Ingegneria della Università di Roma, 2013 URL: http://www.bugiotti.it/downloads/publications/noamSEBD13.pdf (verificado el 21-02-2015)

[9]. C. Nance, T. Losser, R. Iype, G. Harmon. “NoSQL vs RDBMS - why there is room for both”, Proceedings of the Southern Association for Information Systems Conference, Savannah, GA, USA March 8th–9th, 2013 URL: http://sais.aisnet.org/2013/Nance.pdf (verificado el 21-02-2015)

[10]. M. Mannino. Administración de Base de Datos. ISBN 9789701061091. MCGRAW-HILL / Interamericana de México.3ra Edición. 2007