



UNIVERSIDAD DE BUENOS AIRES

Departamento de Computación

Facultad de Ciencias Exactas y Naturales

Bases de Datos

Trabajo Práctico 2

29 de junio de 2015

Integrante	LU	Correo electrónico
Chapresto, Matías	/	
Garassino, Agustín	394/12	ajgarassino@gmail.com
Vileriño, Silvio	/	

Índice

1. Introducción	1
2. Problemática de las bases de datos relacionales	1
3. Bases de datos NoSQL	1
4. Apache Cassandra	1
5. ¿Por qué Apache Cassandra?	1

1. Introducción

Hoy en día el vertiginoso avance de la tecnología hace que las herramientas que una vez nos fueron útiles deban adaptarse constantemente, para poder seguir cumpliendo su función ante la inevitable concepción de las nuevas problemáticas que estos avances conllevan. Por ejemplo en el ámbito de la informática el crecimiento exponencial del internet vino acompañado del desafío de almacenar grandes volúmenes de información, mucho mayores a los que se venían manejando en décadas pasadas. Si bien ya existían herramientas con este fin, su capacidad no logró cubrir todas las necesidades que se presentaron junto con esta revolución. El objetivo de este documento es plantear, bajo este contexto, los inconvenientes de las bases de datos tradicionales y las ventajas que ofrecen las nuevas para la resolución de este dilema.

2. Problemática de las bases de datos relacionales

Las bases de datos relacionales son aquellas que cumplen con el modelo relacional ¹. Este fue uno de los primeros modelos en surgir y es aún el más difundido en la actualidad. Este tipo de bases se desempeñan muy bien a la hora de enfrentar problemas como modificar los datos manteniendo en todo momento la consistencia de los mismos. Las más utilizadas proveen soluciones inteligentes para respetar las llamadas propiedades ACID ². Si bien estas propiedades son deseables, no son un requisito excluyente para el funcionamiento de una base de datos bajo determinados contextos. Cuando surgió la necesidad de manejar grandes cantidades de información se comenzaron a utilizar sistemas distribuidos. Es decir, una gran cantidad de computadoras trabajando en conjunto para proveer una interfaz única para el acceso a los datos. Así salió a la luz el hecho de que las herramientas más utilizadas hasta el momento (MySQL, PostgreSQL, etc...) no lograban desarrollar todo su potencial bajo esta nueva arquitectura. Esto se formalizó en el teorema de Brewer también llamado teorema CAP ³. El teorema establece la imposibilidad de obtener consistencia, disponibilidad y tolerancia a la partición al mismo tiempo en una base de datos. La consistencia está relacionada con la misma C que en las propiedades ACID antes mencionadas. La disponibilidad está relacionada con la capacidad del sistema de devolver una respuesta en todo momento, ya sea con la información solicitada o con un error, pero el servicio debe estar disponible. Finalmente la tolerancia a la partición tiene que ver con la misma problemática que nombramos en las bases de datos relacionales: la capacidad de funcionar en múltiples nodos de manera eficiente, no necesariamente en una única computadora.

3. Bases de datos NoSQL

Las bases de datos NoSQL surgen como una solución al problema de escalar ante los volúmenes masivos de información que surgieron junto con la popularización de la internet. Sacrifican algunas de las propiedades tradicionales como consistencia, a favor de la capacidad de funcionar en múltiples ordenadores. Esto abre la posibilidad de escalar agregando nuevos nodos al sistema distribuido en el que se encuentra funcionando. El término NoSQL es utilizado para todas aquellas bases de datos que no siguen el modelo relacional antes mencionado. Estas nuevas bases por lo general proveen una característica denominada consistencia eventual. La consistencia eventual nos asegura que a pesar de que la base de datos sea momentáneamente inconsistente, el estado del sistema se modificará de manera autónoma hasta ser consistente.

El término SQL se refiere a un lenguaje de programación utilizado para interactuar con las bases de datos. Las bases de datos NoSQL no necesariamente proveen la capacidad de interactuar con ellas a través de este lenguaje e incluso pueden tampoco ofrecer alternativas análogas para todas las operaciones permitidas típicamente.

¹ "A relational model of data for large shared data banks" Codd, E. F. (1970).

² "The Transaction Concept: Virtues and Limitations" Gray, Jim (September 1981).

³ Brewer's Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Gilbert y Lynch (2002)

4. Apache Cassandra

Apache Cassandra es una base de datos NoSQL distribuida. El proyecto es de código abierto y está originalmente escrito en Java. En lugar del sistema típico de tablas utilizado en las bases de datos relacionales, se basa en un modelo de *clave-valor*

5. ¿Por qué Apache Cassandra?