



Facultad de Ingeniería de Sistemas e Electrónica

Carrera Profesional de Ingeniería de Sistemas e Informática

“BASE DE DATOS NO RELACIONALES”

ALUMNOS:

CHAPOÑAN SANDOVAL, Jair

MIMBELA NUÑEZ, Gianpier

Chiclayo, Perú 2018

Contenido

Introducción	3
Contextualización	4
Base de Datos NoSQL	5
Ventajas de los sistemas NoSQL.....	5
Desventajas de los sistemas NoSQL.....	6
Principales diferencias con las bases de datos SQL.....	6
Tipos de bases de datos NoSQL	7
Bases de datos clave – valor.....	7
Bases de datos documentales.....	8
Bases de datos en grafo	9
Bases de datos orientadas a objetos.....	9
Conclusión	11

Ilustraciones

Ilustración 1:Cambio de Requisitos en las aplicaciones.....	4
Ilustración 2:Comparación de Características entre NoSQL y SQL	7
Ilustración 3:Base de datos clave-valor.....	8
Ilustración 4:Base de datos documentales	8
Ilustración 5:Base de datos en grafo	9
Ilustración 6:Cassandra	10
Ilustración 7:MongoDB.....	10

Introducción

Son muchas las aplicaciones web que utilizan algún tipo de bases de datos para funcionar. Hasta ahora estábamos acostumbrados a utilizar bases de datos SQL como son MySQL, Oracle o MS SQL, pero desde hace ya algún tiempo han aparecido otras que reciben el nombre de NoSQL (Not only SQL – No sólo SQL) y que han llegado con la intención de hacer frente a las bases relacionales utilizadas por la mayoría de los usuarios.

En los últimos años, una gran variedad de bases de datos NoSQL han salido a la luz, creadas por compañías principalmente para cubrir sus propias necesidades. Temas como escalabilidad, rendimiento, mantenimiento, etc. que no encontraban en ninguna solución que existía en el mercado. Debido a la variedad de enfoques que existe entre requisitos y funcionalidades que debe cumplir una base de datos NoSQL, es bastante difícil mantener una visión general de la situación actual de las bases de datos no relacionales. Se puede decir que las bases de datos NoSQL son una categoría independiente dentro del conjunto de bases de datos¹.

En particular, las bases de datos NoSQL permiten una replicación entre nodos más simple que la de las bases de datos relacionales, utilizando principalmente dos técnicas (replicación y sharding), de las que derivan otros modelos de distribución. La replicación duplica los datos en múltiples nodos, utilizando el modelo maestro-esclavo o el modelo peer-to-peer, mientras que el sharding divide(fragmenta) la información en varios nodos. Estas técnicas, con sus ventajas y desventajas, también pueden combinarse entre sí, generando un esquema más complejo².

¹ Raúl Herranz (2014). *Base de datos NOSQL: Arquitectura y ejemplos de aplicación*. Leganés. Pág. 34

² http://sedici.unlp.edu.ar/bitstream/handle/10915/52954/Documento_completo.pdf-PDFA.pdf pág. 3

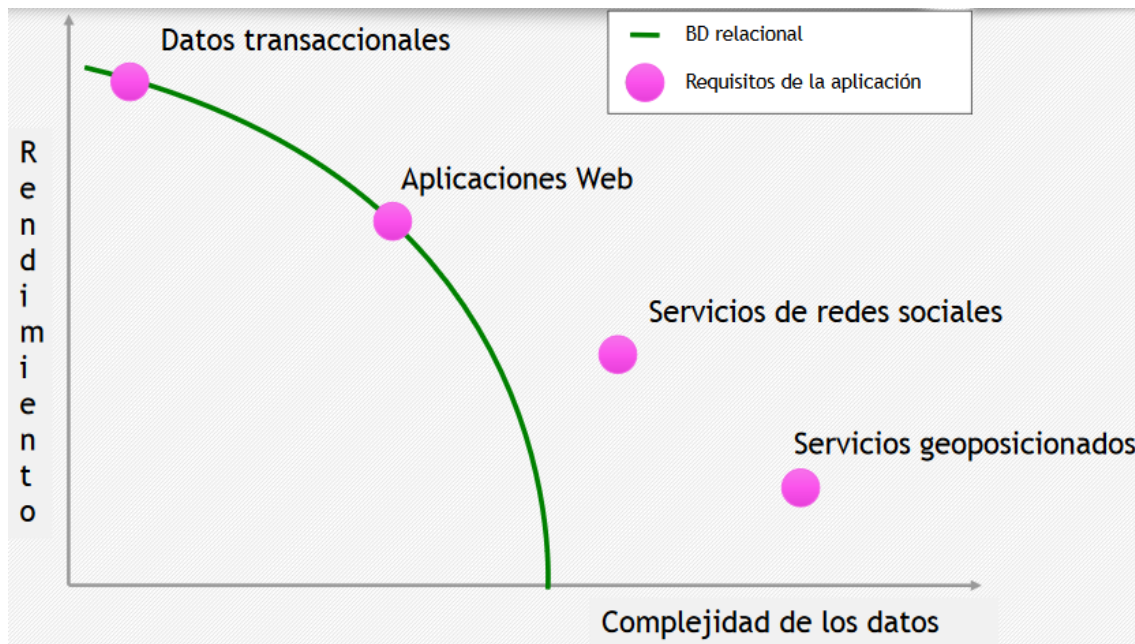


Ilustración 1: Cambio de Requisitos en las aplicaciones.

Más adelante se expondrán diversos conceptos y características comunes de las bases de datos NoSQL.

CONTEXTUALIZACIÓN

Cuando hablamos de entornos de base de datos o aplicaciones web, el termino NoSQL se nos hace más conocido, pero nos preguntamos en que se diferencia esta del base de datos SQL. Hablar de bases de datos NoSQL es hablar de estructuras que nos permiten almacenar información en aquellas situaciones en las que las bases de datos relacionales generan ciertos problemas debido principalmente a problemas de escalabilidad y rendimiento de las bases de datos relacionales donde se dan cita miles de usuarios concurrentes y con millones de consultas diarias.³

Además de lo comentado anteriormente, 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 como clave-valor, mapeo de columnas.

³ <https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf> pág. 2

En un principio, para solucionar estos problemas de accesibilidad, las empresas optaron por utilizar un mayor número de máquinas, pero pronto se dieron cuenta de que esto no solucionaba el problema, además de ser una solución muy cara. La otra solución era la creación de sistemas pensados para un uso específico que con el paso del tiempo han dado lugar a soluciones robustas, apareciendo así el movimiento NoSQL.

Pero, hoy en día los servicios de redes sociales como Facebook, Twitter o Amazon aparecieron, estos requieren dar servicios a miles de usuarios concurrentes y responder diariamente a millones de personas. Estas compañías se dieron cuenta de que el rendimiento y sus necesidades de tiempo real eran más importantes que la consistencia de los datos, aspecto este último al que las bases de datos relacionales tradicionales dedicaban una gran cantidad de tiempo de proceso. Los avances en virtualización condujeron a la construcción de nodos de computación en la nube (*cloud computing*), que a su vez requería nodos de almacenamiento (*cloud storage*).⁴

BASE DE DATOS NOSQL

VENTAJAS DE LOS SISTEMAS NOSQL

Esta forma de almacenar la información ofrece ciertas ventajas sobre los modelos relacionales. Entre las ventajas más significativas podemos destacar:

- **Se ejecutan en máquinas con pocos recursos:** Estos sistemas, a diferencia de los sistemas basados en SQL, no requieren de apenas computación, por lo que se pueden montar en máquinas de un coste más reducido.
- **Escalabilidad horizontal:** Para mejorar el rendimiento de estos sistemas simplemente se consigue añadiendo más nodos, con la única operación de indicar al sistema cuáles son los nodos que están disponibles.
- **Pueden manejar gran cantidad de datos:** Esto es debido a 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

⁴ Zorrilla, García (2017). *Base de datos NoSQL. Introducción*. Universidad de Cantabria. pág. 6

más complejo, lo que constituye un punto de entrada en común, que ante muchas peticiones puede ralentizar el sistema.

DESVENTAJAS DE LOS SISTEMAS NOSQL

Menos apoyo de la comunidad: aunque el NoSQL se ha expandido a un ritmo increíble, el apoyo de la comunidad es relativamente menor ya que es nuevo.

Estandarización: carece de una plataforma estandarizada como SQL, lo que impide que siga expandiéndose.

Esto ha estado creando preocupaciones durante la migración. La estandarización es lo que ayuda a la industria de bases de datos a unificarse.

Las interfaces y la interoperabilidad es otra preocupación a la que se enfrenta NoSQL, que necesita una solución inmediata.

Principales diferencias con las bases de datos SQL

Algunas de las diferencias más destacables que nos podemos encontrar entre los sistemas NoSQL y los sistemas SQL están:

- **No utilizan SQL como lenguaje de consultas.** La mayoría de las bases de datos NoSQL evitan utilizar este tipo de lenguaje o lo utilizan como un lenguaje de apoyo. Por poner algunos ejemplos, Cassandra utiliza el lenguaje CQL, MongoDB utiliza JSON o BigTable hace uso de GQL.
- **No utilizan estructuras fijas como tablas para el almacenamiento de los datos.** Permiten hacer uso de otros tipos de modelos de almacenamiento de información como sistemas de clave–valor, objetos o grafos.
- **No suelen permitir operaciones JOIN.** Al disponer de un volumen de datos tan extremadamente grande suele resultar deseable evitar los JOIN. Esto se debe a que, cuando la operación no es la búsqueda de una clave, la sobrecarga puede llegar a ser muy costosa. Las soluciones más directas consisten en desnormalizar los datos, o bien realizar el JOIN mediante software, en la capa de aplicación.
- **Arquitectura distribuida.** Las bases de datos relacionales suelen estar centralizadas en una única máquina o bien en una estructura máster–

esclavo, sin embargo, en los casos NoSQL la información puede estar compartida en varias máquinas mediante mecanismos de tablas Hash distribuidas.

Comparación de características entre NoSQL y bases de datos relacionales:

Característica	Bases de datos NoSQL	Bases de datos relacionales
Actuación	Alto	Bajo
Confiabilidad	Pobre	Bueno
Disponibilidad	Bueno	Bueno
Consistencia	Pobre	Bueno
Almacenamiento de datos	Optimizado para datos enormes	De tamaño mediano a grande
Escalabilidad	Alto	Alto (pero más caro)

Ilustración 2: Comparación de Características entre NoSQL y SQL

Tipos de bases de datos NoSQL

Dependiendo de la forma en la que almacenen la información, nos podemos encontrar varios tipos distintos de bases de datos NoSQL. Veamos los tipos más utilizados.

Bases de datos clave – valor

Son el modelo de base de datos NoSQL más popular, además de ser la más sencilla en cuanto a funcionalidad. En este tipo de sistema, cada elemento está identificado por una llave única, lo que permite la recuperación de la información de forma muy rápida, información que habitualmente está almacenada como un objeto binario (BLOB). Se caracterizan por ser muy eficientes tanto para las lecturas como para las escrituras.

Algunos ejemplos de este tipo son Cassandra, BigTable o HBase.

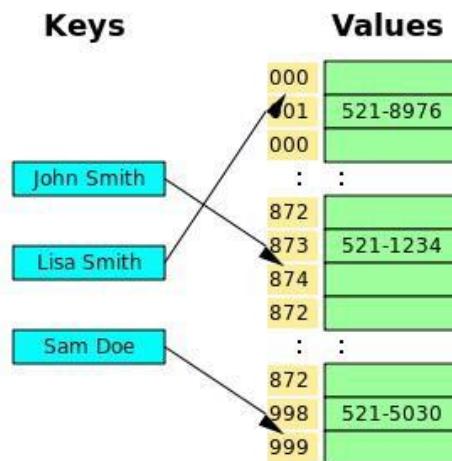


Ilustración 3: Base de datos clave-valor

Bases de datos documentales

Este tipo almacena la información como un documento, generalmente utilizando para ello una estructura simple como JSON o XML y donde se utiliza una clave única para cada registro. Este tipo de implementación permite, además de realizar búsquedas por clave-valor, realizar consultas más avanzadas sobre el contenido del documento.

Son las bases de datos NoSQL más versátiles. Se pueden utilizar en gran cantidad de proyectos, incluyendo muchos que tradicionalmente funcionarían sobre bases de datos relacionales.

Algunos ejemplos de este tipo son MongoDB o CouchDB.

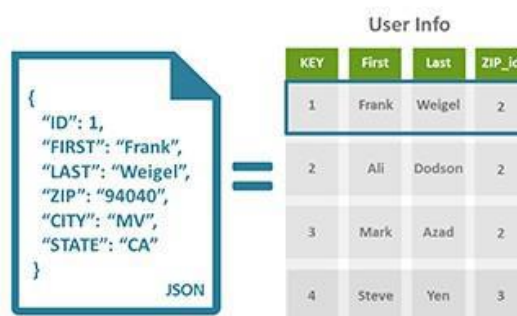


Ilustración 4: Base de datos documentales

Bases de datos en grafo

En este tipo de bases de datos, la información se representa como nodos de un grafo y sus relaciones con las aristas del mismo, de manera que se puede hacer uso de la teoría de grafos para recorrerla. Para sacar el máximo rendimiento a este tipo de bases de datos, su estructura debe estar totalmente normalizada, de forma que cada tabla tenga una sola columna y cada relación dos.

Este tipo de bases de datos ofrece una navegación más eficiente entre relaciones que en un modelo relacional.

Algunos ejemplos de este tipo son Neo4j, InfoGrid o Virtuoso.

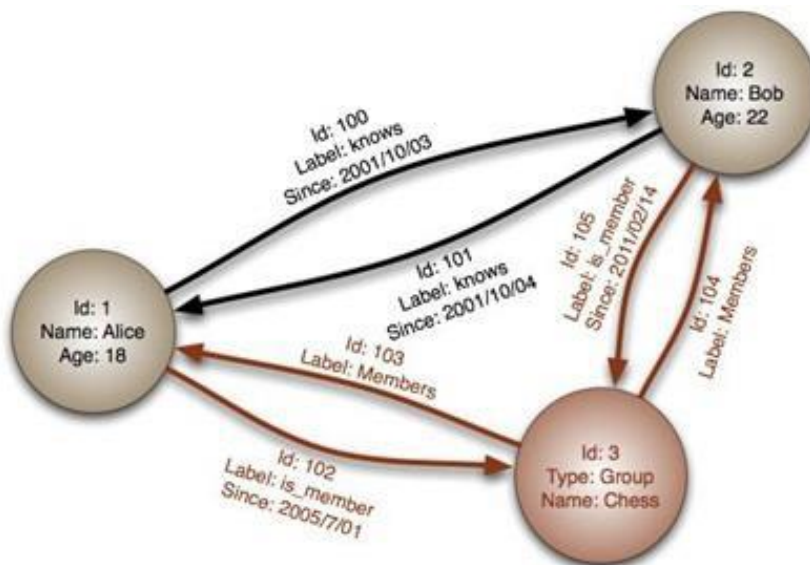


Ilustración 5: Base de datos en grafo

Bases de datos orientadas a objetos

En este tipo, la información se representa mediante objetos, de la misma forma que son representados en los lenguajes de programación orientada a objetos (POO) como ocurre en JAVA, C# o Visual Basic .NET.

Algunos ejemplos de este tipo de bases de datos son Zope, Gemstone o Db4o.

Ejemplos de Base de Datos NoSQL

Cassandra

Se trata de una base de datos creada por Apache del tipo clave – valor. Dispone de un lenguaje propio para realizar consultas CQL (Cassandra Query Language). Cassandra es una aplicación Java por lo que puede correr en cualquier plataforma que cuente con la JVM.



Ilustración 6:Cassandra

MongoDB

Se trata de una base de datos creada por 10gen del tipo orientada a documentos, de esquema libre, es decir, que cada entrada puede tener un esquema de datos diferente que nada tenga que ver con el resto de registros almacenados. Es bastante rápido a la hora de ejecutar sus operaciones ya que está escrito en lenguaje C++.

Para el almacenamiento de la información, utiliza un sistema propio de documento conocido con el nombre BSON, que es una evolución del conocido JSON pero con la peculiaridad de que puede almacenar datos binarios.

En poco tiempo, MongoDB se ha convertido en una de las bases de datos NoSQL favoritas por los desarrolladores.



Ilustración 7:MongoDB

CONCLUSIÓN

La información en la red se vuelve cada vez más compleja y de mayor capacidad. Toda esta información requiere ser almacenado en grandes dimensiones de base de datos. A todo esto, surge un problema, la lentitud de almacenar y hacer operaciones en las bases de datos relacionales. Las características que plantean las bases de datos NoSQL responden a las necesidades actuales de las diferentes organizaciones por lo que se vuelven más rápidos en los movimientos de software, debido a su capacidad y velocidad que son dos principios que concuerdan hoy en los sistemas actuales. Cuando se desea realizar un trabajo especializado lo más rápido y eficientemente posible es ahí donde se requiere o necesitas NoSQL.

Finalmente, es importante tener en cuenta que NoSQL es una buena adición a los estándares de bases de datos existentes, pero con algunas advertencias importantes. NoSQL intercambia características de fiabilidad y consistencia para un rendimiento y escalabilidad máximos. Esto lo convierte en una solución especializada, ya que el número de aplicaciones que pueden depender de las bases de datos NoSQL sigue siendo limitado.

Esta investigación se ha enfocado en aspectos generales de las bases de datos NoSQL, teniendo en cuenta como trabajos futuros la investigación en cada una de las taxonomías mencionadas y la comparación entre ellas, para determinar una guía de selección.

Bibliografía

(s.f.).

Acens. (02 de 09 de 2018). *Bases de DATos NoSQL. Qué son y tipos que nos podemos encontrar*.
Obtenido de Acens: www.acens.com

Carabio, A. L., Benedetto, M. G., & Falappa, M. A. (2016). *Comportamiento de Bases de Datos No Relacionales*. Concordia: Universidad Nacional de Entre Ríos.

Castro Romero, A., Gonzales Sanabria, J. S., & Callejas Cuervo, M. (2012). Utilidad y Funcionamiento de las Base de Datos NoSQL. *Revista de Ingenieria*, 21-32.

Herranz, R. (2014). *Base de datos NoSQL:Arquitectura y Ejemplos de Aplicacion*. Leganés: Universidad Carlos III de Madrid.

Martin, S. (10 de 01 de 2017). *Base de Datos NoSQL:Guía definitiva*. Obtenido de Toptal: <https://www.toptal.com/database/the-definitive-guide-to-nosql-databases>