

Comparativa de Gestores de Base de Datos No Relacionales

Belizario Mamani, Loja Mamani, Lizárraga Paolo, Kenyi Chino

July 08, 2020

Abstract

There are two types of databases, relational and non-relational. In this article, we will see the comparison between two non-relational database managers (NoSQL), their differences, advantages, disadvantages and characteristics of each of the two managers.

1. Introduccion

En este artículo veremos los cuadrantes del Agile Testing, que está formado por 4, el lado izquierdo es sobre guiar el desarrollo y realizar pruebas en forma temprana, el lado derecho es sobre la evaluación del producto y aprender lo que aún falta.

2. RESUMEN

Existen dos tipos de bases de datos, las relacionales y no relacionales. En este artículo veremos la comparativa entre dos gestores de bases de datos NO relacionales (NoSQL), sus diferencias, ventajas, desventajas y características de cada uno de los dos gestores.

3. TITULO

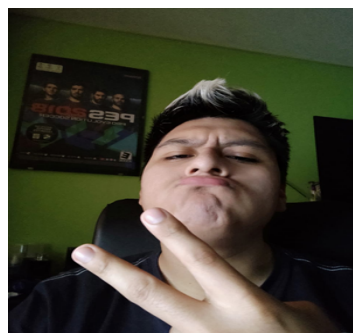
Cuadrantes de pruebas ágiles.

4. AUTORES

Los autores de este trabajo final de unidad son:



- Paolo Lizárraga



- Anthony Belizario

- Daniel Loja



- Kenyi chino



V.DESARROLLO

En nuestro mundo actual, donde la información es primordial en cada una de las etapas de almacenamiento, recuperación y visualización se nos presenta una necesidad muy grande que consiste en el procesamiento de la información de una forma eficaz y eficiente.

Una base de datos no relacional es una alternativa a que se nos presenta actualmente para este procesamiento de la información donde consiste en guardar la información en un almacenamiento de datos mucho más rápido, sin ningún tipo de estándar convencional y sobre todo escalable.

Tipo		Características
Almacenes (Clave - Valor)	Key-Value	Cassandra, Membase, Redis o Riak. Almacenan los datos en pares clave-valor (un valor podría ser un objeto).
Orientadas a Columnas		Vertica, Microsoft, Qd Technology, Sybase, ParAccel, Infobright. Mejor rendimiento. Eficiente en el almacenamiento. Cambios en el esquema tiene menor impacto y por lo tanto el coste de realizarlo es menor.
Basada en Documentos		MongoDB y CouchDB. Almacenan los datos en formatos estructurados (registros) como JSON.
Orientadas a Grafos		Neo4J y FlockDB. Almacenan los elementos y las relaciones entre ellos con un estilo de grafo (para redes sociales, redes de transporte, mapas de carreteras, topologías de red).

Beneficios

Al enfocarnos en base de datos no relacionales una pregunta primordial sería: el porqué del cambio de que muchas empresas reconocidas en el mundo lo está haciendo es migrar sus bases de datos tradicionales relacionales a un almacenamiento de datos en la mayoría libres, sin llegar a tener una estructura convencional.

- En la velocidad en el tiempo de respuesta ya que los usuarios necesitan que sus necesidades sean resueltas en el menor tiempo posible
- La cantidad de información que se pueda almacenar

Ventajas de la base de datos no relacionales

- Hay bases de datos no relacionados de código abierto

Los productos de código abierto aportan beneficios a los desarrolladores como en precio, no se necesita de un servidor con grandes recursos, no tiene una estructura de datos definida, los datos pueden ser diversos existiendo heterogeneidad

- Es de escalamiento sencillo

Las bases de datos no relacionales buscan una manera de añadir más servidores para manejar más cargas de datos, permitiendo a las empresas una distribución de los equipos dependiendo de las actividades a realizar.

- Economía

Las bases de datos no relacionales utilizan servidores de bajo costo para la administración de los datos y el volumen de las transacciones que realicen. El costo por gigabyte por segundo para estas bases puede ser mucho menor que el costo de los RDBMS, lo que le permite almacenar y procesar más datos a un precio mucho más bajo, pudiendo así añadir máquinas según sean las necesidades de la empresa

- No generan cuellos de botella

Las base de datos relacionales tienen este problema ya que estos tienen que transcribir cada sentencia para ser ejecutadas incluyendo las sentencias complejas, además de un nivel de ejecución más preciso para llevar a cabo por lo que constituye un punto de entrada común, único, y conflictivo en base a rendimiento.

Descripción	No Relacional	Relacionales
Estructura de Datos	Es una estructura flexible, no es necesario definir una estructura de datos	Asumen una estructura bien definida de datos tienen que ser uniformes y las propiedades de estos datos pueden definirse por adelantado.
Relaciones entre tablas	No existen relaciones entre colecciones, no obstante puede depender del modelado de datos.	Tienen que estar muy bien establecidas y ser referenciadas de forma sistemática.
Transaccionalidad	Se pierde integridad en las transacciones.	Utilizan ACID
Consultas e Índices	Disminuye el uso de indexación y el poder de las consultas.	

Se hizo el uso de base de datos no relacional en un proyecto de investigación relacionada al estudio científico del espacio para esa investigación se requirió de una base de datos no relacional

Diseño e implementación de un prototipo de una base de datos no relacional con manejo de flujo eficiente en el contexto de alertas astronómicas

Base de datos NoSql

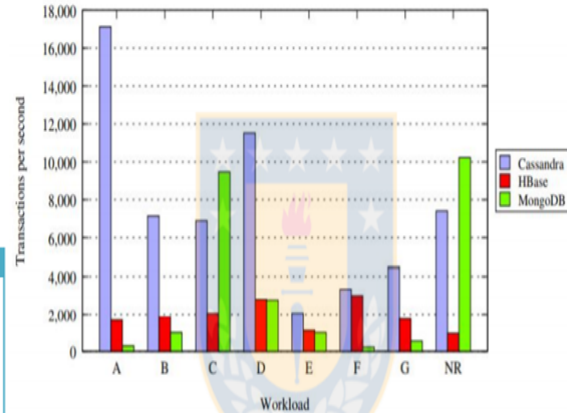
Debido a la cantidad de datos que se generan en la actualidad, ya sea en las redes sociales, en investigaciones científicas, en los telescopios, etc., los sistemas de gestión de bases de datos relacionales o SGBD comenzaron a presentar dificultades en los aspectos de escalabilidad y rendimiento, es por esto que surgieron las bases de datos no relacionales o NoSQL (Non Structured Query Language). Estas difieren del modelo clásico de base de datos relacionales, ya que poseen una forma de almacenamiento no estructurada, son flexibles y generalmente no poseen atomicidad ni operaciones JOIN. Además, normalmente las bases de datos NoSQL son sistemas distribuidos, por ende, se debe mencionar el teorema CAP (Consistency, Availability and Partition Tolerance), que, en resumen, enuncia que un sistema distribuido sólo puede garantizar 2 de las siguientes 3 características:

Consistencia: Cuando se realice una consulta, el sistema retorne la escritura más reciente de un registro dado.

Disponibilidad: Cuando se realice una consulta, el sistema debe retornar una respuesta en un tiempo razonable, es decir, que no retorne error ni timeout.

Tolerancia de particiones o tolerancia a fallos: El sistema puede seguir funcionando, aunque fallen nodos o instancias del sistema, se produzcan problemas en la comunicación entre los nodos, etc

es diseñar e implementar una base de datos no relacional, para posteriormente, realizar una simulación del streaming de las alertas astronómicas y un análisis experimental de la eficiencia de la base de datos en tiempo real.

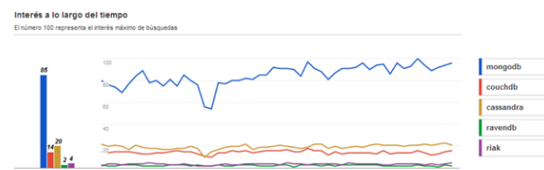


realizó una evaluación de bases de datos NoSQL, en específico Cassandra, HBase y MongoDB, para aplicaciones científicas, utilizando datos tanto de bioinformática como de astronomía.

NoSQL: futuro de almacenamiento de datos

Tendencia actual de las NoSQL DBMS | Google Trends

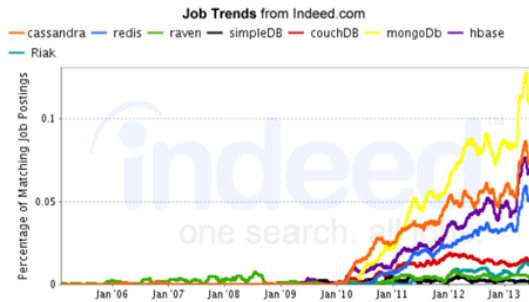
En el 2011, se estimó que los proveedores de la tecnología, entre todos ellos, generaron ingresos de US\$ 20 millones, donde, aproximadamente, la mitad de estos fue logrado por 10gen (nombre del proveedor del producto MongoDB, ahora nombrado MongoDB Inc.³³), pronosticando, además, que el mercado crecerá a una tasa compuesta anual del 82 % hasta alcanzar los US\$ 215 millones . Con la anterior afirmación, no solo podemos apreciar que las NoSQL en un futuro tendrán un alto crecimiento a nivel de ingresos y posicionamiento, sino, también, que MongoDB Inc con su producto MongoDB han logrado asentarse una gran base de clientes a través de su enfoque en la facilidad de adopción para el desarrollador, y volviendo su atención al tipo de necesidades actuales requeridas por las empresas.



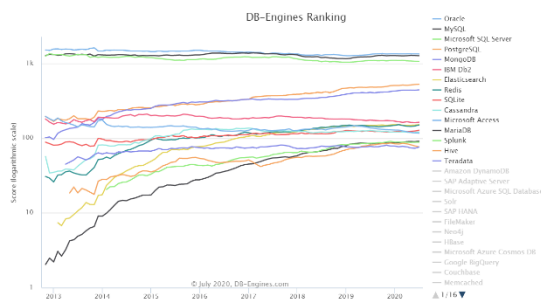
Tendencia actual de las NoSQL DBMS | Indeed

En este punto realizamos un análisis a las tendencias de empleo NoSQL en setiembre de 2013. La lista incluye a CassandraDB, Redis, RavenDB, SimpleDB, CouchDB, MongoDB, HBase y Riak

Como se puede evaluar, la demanda de MongoDB continúa superando a los otros productos de base de datos NoSQL, y aumentando su ventaja en los últimos meses. Cassandra continúa con un crecimiento sólido pero no puede superar a MongoDB. Esta tendencia refleja no solo el crecimiento de determinados productos, sino el hecho de que las NoSQL están logrando implementarse en los negocios e impactar, agitantamente, en las necesidades actuales de estos, puesto que requieren de los servicios de mucho más personal a lo largo del 2010.



Tendencia actual de las NoSQL DBMS | DB-ENGINES



- **MongoDB:**

Estamos ante el sistema gestor de bases de datos no relacionales más **popular** y utilizado actualmente. MongoDB es un SGBD NoSQL orientado a ficheros que almacena la información en estructuras BSON con un esquema dinámico que permite su facilidad de integración. Empresas como Google, Facebook, eBay, Cisco o Adobe utilizan MongoDB como sistema gestor de bases de datos. Las **principales** características de MongoDB son:

- Indexación y replicación
- Balanceo de carga.
- Almacenamiento en ficheros
- Consultas ad hoc
- Escalabilidad horizontal
- Open Source

Como **desventaja** principal, Mongo DB no es un SGBD adecuado para realizar transacciones complejas

- **Apache Cassandra:**

Al igual que Redis (Otro SGBD conocido) Cassandra tambien utiliza almacenamiento clave-valor. Es un SGBD NoSQL distribuido y masivamente escalable.

Empresas como Facebook, Twitter, Instagram, Spotify o Netflix utilizan Cassandra.

Dispone de un lenguaje propio para las consultas denominado CQL (Cassandra Query Language).

Las principales características de Cassandra son:

- Multiplataforma
- Propio lenguaje de consultas (CQL)
- Escalado lineal y horizontal
- Es un SGBD distribuido
- Utiliza una arquitectura peer-to-peer.

Características de MongoDB y Cassandra.

Característica	MongoDB	Cassandra
Tipo	Orientado a Documentos	Orientado a Columnas
Lenguaje de desarrollo	C++	Java
Lenguaje de Consulta	Objetos y métodos personalizados, basado en JavaScript	CQL
Tipo de almacenamiento	Documentos BSON	Columnas

Protocolo	TCP/IP	TCP/IP
Locks	Si	Si
Triggers	No	Si
Concurrencia	Actualización instantánea	MVCC
Sistemas Operativos	Linux/ MAC OS/ Windows	Linux/ MAC OS/ Windows
Replicación	Master-Slave	Multi-Master
Tolerancia a Fallos	Alto	Excepcionalmente Alto
Características mas importantes	Conserva algunas propiedades SQL, como la consulta y el índice	Escala linealmente, Es distribuida, Escala de forma horizontal.
Áreas de uso	Almacenamiento y registro de eventos, Sistemas gestión de contenidos (CMS)	Sistemas bancarios, Sistemas financieros, Publicidad
Código Abierto	Si	Si

Mediante el análisis de las propiedades básicas, es posible concluir que existen similitudes cuando se trata de tipos de archivos utilizados, consultas, transacciones, locks, almacenamiento de datos, código abierto y sistemas operativos. En cuanto a términos de uso, MongoDB tiene un mejor uso para Sistemas de Gestión de Contenidos (CMS), mientras tiene consultas dinámicas con datos escritos frecuentemente. Cassandra está optimizado para almacenar e interactuar con grandes cantidades de datos que se pueden utilizar en diferentes áreas, tales como, finanzas o publicidad.

VI. CONCLUSIONES

- Existen muchos gestores de bases de datos en el mercado, pero los más usados son estos.

- Hay que entender bien que para elegir un SGBD más adecuado a nuestras necesidades hay que comenzar por un estudio del tipo de datos que vamos a almacenar y como lo vamos a administrar.
- Se determinó que al usar una base de datos no relacional en una aplicación web no transaccional al momento de realizar las consultas su tiempo de respuesta en la visualización de los resultados disminuye considerablemente por la gran cantidad de información que tiene la base
- El sistema de gestión de bases de datos Apache Cassandra es usado ampliamente en la actualidad, debido a sus características y su funcionamiento. Lo que significa que Cassandra puede soportar tanto la ingestión, como las consultas en una situación real, sin ningún inconveniente, y en un tiempo válido para procesos.

5. RECOMENDACIONES

- Entre los SGBD citados encontraremos alguno que se adapte a nuestras necesidades de acuerdo a la inversión a realizar, el volumen de información a almacenar, el tipo de consulta a realizar, etc.

VIII. BIBLIOGRAFIA