

# HOJA DE TRABAJO - REVISIÓN DE SISTEMAS NOSQL PARA GESTIÓN DE BIG DATA

Nicolas Jiménez, Iván Parra, Ricardo Moncaleano

Validación conocimientos

Universidad de los Andes, Bogotá, Colombia

{en.jimenez, id.parra, a.moncaleano}@uniandes.edu.co

Fecha de presentación: septiembre 16 de 2020

## 1 Introducción

Esta tercera entrega tiene como objetivo entender en un mejor detalle cómo trabajan las bases de datos no relacionales, que si bien ya existen hace bastantes años, últimamente se han popularizado más dada su versatilidad y especialidad dependiendo de la necesidad del usuario. En esta oportunidad se hará una detallada revisión de dos tipos puntuales: Key Value y Document Store, realizando una comparativa entre ellas, revisando fortalezas y debilidades, su integración y capacidad transaccional, entre otros aspectos.

## 2 Objetivos

### 2.1 Objetivo Principal.

- Comprender los conceptos y sus aplicaciones de las Base de Datos NoSQL.

### 2.2 Objetivos Específicos.

- Revisar el estado de productos representativos de las tendencias actuales del mercado no relacional de gestión de Big Data.
- Análisis de los productos con respecto a las características esperadas de un SMBD
- Análisis de criterios de selección de tecnologías

## 3 Reporte

A nuestro grupo le corresponde el estudio de las familias de repositorios Document store y Key value, vamos a abordar la primera de ellas, para la familia de repositorios Document store se puede afirmar que nace desde el programa “Plato Notes” de la universidad Illinois en el año 1973, este proyecto tenía como fin permitir compartir a los usuarios una comunicación segura, tres años después se lanza “Plato Group Notes” que entre sus principales características y las que más nos importan para este reporte fueron organizar notas por asunto y poder leer las que fueron escritas en una fecha puntual, en 1980 uno de sus fundadores empezó a trabajar en una solución más accesible y compatible para la época, de esta idea nace “Lotus Notes” que lanza su primera versión en 1987, la cual buscaba que los clientes crearan sus propias aplicaciones, esta versión fue muy acogida e impulsa en la idea de la colaboración y comunicación grupal mediante computadora, a este producto nos podemos referir como el primer motor de base de datos orientado a documentos; Continuando con la historia de “Lotus notes” en 1996 lanzan su versión 4.0 basada en la integración con internet mediante su producto “Domino”, posteriormente esta idea es comprada por IBM y se convierte en “IBM notes”.

En la actualidad podemos ver que MongoDB tiene el protagonismo para las bases de datos documentales, nace en 2007 buscando ser competencia de Google App Engine, su idea no fue muy aceptada en el mercado pero teniendo serios problemas de escalabilidad con bases de datos relacionales, los desarrolladores trabajan en un producto que no tenga esas falencias, de lo deciden desprenderse de su proyecto PASS, dando paso en 2009 a MongoDB como una herramienta open source y heredando su nombre de la palabra “Humongous” que significa enorme.

Otros productos importantes para esta familia de sistemas NoSQL pueden ser Apache CouchDB, OrientDB, Mark Logic, Amazon DocumentDB, BaseX, eXist, Elasticsearch y SimpleDB

Muchos de estos productos se ejecutan en una máquina virtual JAVA, por lo tanto, puede utilizarse con independencia del sistema operativo. Otros solo pueden ser ejecutados en entornos Unix como Ubuntu y SimpleDB es un entorno diseñado por Amazon el cual es exclusivo para prestación de servicio por cobro de tarifa de uso.

Estos productos están diseñados para soportar infraestructuras en la nube (Cloud Computing), sin embargo, pueden funcionar con los requisitos mínimos de un servidor, servidores de replicación, infraestructuras Maestro / Esclavo, Tuning de bases de datos e infraestructuras basadas en respaldo y recuperación.

Hay muchas herramientas que acompañan a este tipo de bases de datos, por ejemplo, para el caso de monitoreo, bases de datos como DynamoDB y Azure Cosmos, están cubiertas bajo las herramientas de monitoreo de las nubes de AWS y Azure respectivamente. Otras herramientas importantes para el desarrollo con este tipo de bases de datos son los framework basados en ORMs para el manejo de bases de datos como objetos, dentro de los complementos más reconocidos están: mongoose (para MongoDB) y DynamoDB ODM (para DynamoDB).

Este tipo de motores de base de datos usan índices para asociar documentos a diferencia de los modelos relacionales donde tenemos una estructura de filas y columnas, lo que les permite tener un escalamiento horizontal, su funcionamiento se resume en tener una clave única que asocia a una estructura llamada documento, los cuales podemos asociarlos como un objeto en programación, ya que comparten muchas de sus características.

Esta tecnología es mayormente utilizada en soluciones para e-commerce, blogs y analítica, sus debilidades son el manejo de transacciones y el manejo de consultas complejas, entre las empresas más significativas que usan estas tecnologías, podemos encontrar a Facebook, Google, Ebay, SAP, PayPal.

Las Bases de datos tipo Document Store suelen usar lenguajes de consulta basados en: XQuery, XSLT, SPARQL, Java, Javascript, Python, para Key Value base de datos se usa comúnmente PERL.

Para la familia de Key Value podemos afirmar que Document Store es un subtipo de ellas, Tenemos un id como Key y un documento como value, pero antes de abordar más a fondo de ello veremos algo de la historia de este tipo de motores de bases de datos.

El primer indicio que tenemos de las bases de datos basadas en Key Value, nos remite a 1966 cuando el hospital general de Massachusetts y en esa época no existían muchas opciones para sistema operativo y software, el MIT entidad asociada al hospital diseñó desde cero el sistema “MUMPS”, para lo que nos

interesa, este sistema utilizo un sistema de almacenamiento Key Value el cual fue implementado a esta familia de repositorios NoSQL del cual hablamos.

El funcionamiento este sistema en síntesis es simple, funciona mediante un par de valor y llave, estos datos están vinculados, Key funciona como el identificador y value funciona como el contenido, se considera este el sistema más simple de repositorios NoSQL, esa característica le permite ser rápido y escalable, no se utiliza un esquema definido como en SQL.

Los usos más populares para estos repositorios son en el manejo de sesiones en sitios web y videojuegos, también en el almacenamiento de datos de usuario en perfiles como un ejemplo. Estos sistemas en lo general utilizan 2 técnicas para mejorar el rendimiento, algunas utilizan solo RAM y otros mezclan almacenamiento junto a la RAM, encontramos los motores más populares como: Amazon DynamoDB, Aerospike, Apache Cassandra, Memcached, Redis y Riak. Los casos más famosos para los usos de estos motores son, Apple, Cisco WebEx, Instagram, Netflix, Twitter.

Siempre se ha considerado que las bases de datos no relacionales carecen de integridad y su proceso transaccional poco eficiente al no ser relacional, perdiendo su explicación más importante ACID (Atomicity, Consistency, Isolation, Durability) , sin embargo ya existen diferentes herramientas que facilitan y hacen transparente las transacciones garantizando así la integridad de los datos, esto no implica que el proceso en las bases de datos no relacionales no sea complejo, en realidad es algo diferente, pues al inicio de la operación se define la cantidad de documentos donde se almaceno la información, luego se modifican y finalmente se guardan los cambios en todos, el inconveniente existe cuando no se puede guardar algún cambio, en ese momento la transacción podría fallar y la integridad de los datos verse comprometida.

## 4 Bibliografía

1. Drake, Mark. DigitalOcean. A Comparison of NoSQL Database Management Systems and Models. [En línea] <https://www.digitalocean.com/community/tutorials/a-comparison-of-nosql-database-management-systems-and-models#document-oriented-databases>.
2. PAT RESEARCH. TOP 12 NOSQL DOCUMENT DATABASES. [En línea] <https://www.predictiveanalyticstoday.com/top-nosql-document-databases/>.
3. Knut Haugen. *A Brief History of NoSQL*. [En línea] <http://blog.knuthaugen.no/2010/03/a-brief-history-of-nosql.html>.
4. BCC. *Top 6 Moments in IBM Notes and Domino History*. [En línea] <https://www.bcchub.com/bcc-blog/top-6-moments-in-ibm-notes-and-domino-history>.
5. MongoDB. *Successful innovators across the globe*. [En línea] <https://www.mongodb.com/who-uses-mongodb>.
6. Schaefer, Lauren. MongoDB. *MongoDB Q&A: What's the deal with data integrity in relational databases vs MongoDB?* [En línea] <https://www.mongodb.com/blog/post/mongodb-qa-whats-the-deal-with-data-integrity-in-relational-databases-vs-mongodb>.
7. Foote, Keith D. DATAVERSITY. *Understanding Key-Value Databases*. [En línea] <https://www.dataversity.net/understanding-key-value-databases>.
8. —. DATAVERSITY. *Understanding Key-Value Databases*. [En línea] <https://www.dataversity.net/understanding-key-value-databases/>.