

**Monografía de estudio sobre la caracterización de NoSQL como sistema de gestión de
bases de datos**

Carlos Enrique Sánchez Pardo

Universidad Nacional Abierta y a Distancia - UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería

Ingeniería de Sistemas

Santiago de Cali – 2021

**Monografía de estudio sobre la caracterización de NoSQL como sistema de gestión de
bases de datos**

Carlos Enrique Sánchez Pardo

Director de Monografía

Ing. JULIÁN ANDRÉS RUIZ AYALA

Universidad Nacional Abierta y a Distancia - UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería

Ingeniería de Sistemas

Santiago de Cali – 2021

Tabla de contenido

Resumen.....	4
Justificación	5
Objetivos.....	7
Metodología de la investigación.....	8
Marco conceptual y teórico.....	9
1. Qué es NoSQL.....	10
2. Tipos de bases de datos NoSQL	13
2.1 Clave-valor.....	13
2.2 Documentos	14
2.3 Columnas	15
2.4 Grafos.....	17
3. Motores para bases de datos NoSQL.....	18
3.1 Redis	18
3.2 MongoDB	19
3.3 HBase.....	20
3.4 Neo4j.....	22
4. Aplicación de bases de datos NoSQL en Colombia.	23
Conclusiones.....	25
Referencias bibliográficas.....	26

Resumen

Las bases de datos son un componente esencial en un sistema de información, los sistemas de gestión más conocidos son las bases de datos relacionales, son los que con mayor frecuencia se enseñan durante la formación profesional, en gran parte se debe a que este sistema está basado en el modelo relacional, el cual proporciona consistencia e integridad a los datos que se guardan. No obstante, en las bases de datos relacionales se han presentado dificultades cuando es relevante la disponibilidad y la rapidez de los datos, ante esto surge NoSQL como una alternativa a las bases de datos relacionales, en la actualidad el concepto de NoSQL representa el modelo para bases de datos no relacionales. La ausencia de NoSQL como temática en el pregrado de Ingeniería de Sistemas de la UNAD representa una oportunidad para recopilar los rasgos principales de esta temática, tales como la explicación de su definición y origen, la comparación de los principales tipos de bases de datos NoSQL, la distinción de los motores que se emplean según su tipo y su aplicación en Colombia.

Palabras clave: NoSQL, bases de datos relacionales, datos semiestructurados, escalabilidad horizontal, teorema CAP, tabla Hash, JSON, Cypher.

Justificación

La base de datos es un elemento primordial en un producto de software, siendo las bases de datos relacionales el modelo preferido durante la toma de decisiones en la planificación y desarrollo de un proyecto de TI, dicha popularidad se debe en gran parte a su amplio conocimiento entre desarrolladores y programadores, su estado del arte se ha documentado ampliamente, manipula de forma eficiente datos estructurados, su diseño se define a través del modelo relacional, el cual considera tanto las propiedades como las relaciones entre varios conjuntos de datos para determinar la distribución de entidades o tablas que conformará la base de datos, proporcionando consistencia e integridad a los datos que se almacenan.

No obstante, en las bases de datos relacionales suelen presentarse inconvenientes cuando es relevante la interconectividad de los datos, debido a que la manipulación de los datos en una base de datos relacional puede ser más compleja y consumir más tiempo (Cáceres, Tolaba, Pérez, Quispe, Rodríguez y Sandoval, 2018), por lo tanto, cuando es relevante la disponibilidad y la rapidez de los datos, principalmente cuando su tamaño es importante y su homogeneidad no es uniforme, en términos de simplicidad y eficiencia las bases de datos relacionales no son lo suficientemente confiables, a pesar de lo pragmático y aplicable que es su modelamiento.

Por tal motivo, surge NoSQL como propuesta alternativa en sistemas de gestión de bases de datos, el cual se ha usado de forma expedita para el almacenamiento de datos en redes sociales, donde se genera una gran cantidad de datos masivos semiestructurados, los cuales son necesarios para el usuario final. Para Rossel, Manna y Bugni no existe una definición precisa de NoSQL, el

término actúa como un paraguas para un conjunto de bases de datos y tecnologías de almacenamiento que no siguen los principios de los sistemas de bases de datos relacionales (2016), por lo tanto, NoSQL representa una serie de propiedades favorables cuando se requiere de un almacenamiento de datos diferencial en cuanto a la cantidad y a la forma de los datos que se pretenden guardar.

Cabe anotar que NoSQL no depende directamente del lenguaje SQL como tal, incluso, existe una discusión en cuanto al nombre asignado, ya que suele malinterpretarse como una contrariedad al lenguaje SQL, cuando su intención era indicar que en este sistema de gestión de base de datos no solamente se utiliza SQL como lenguaje de programación.

A pesar de la aparente prelación de NoSQL frente a los sistemas de gestión de bases de datos relacionales, principalmente para el almacenamiento de datos masivos, la ausencia de NoSQL como temática en el pregrado de Ingeniería de Sistemas de la UNAD representa una oportunidad para recopilar los rasgos principales de esta temática, tales como la explicación de su definición y origen, la comparación de los principales tipos de bases de datos NoSQL, la distinción de los motores se emplean en la actualidad y los casos en que se ha aplicado esta tecnología en Colombia.

Objetivos

Objetivo general

Describir las características de NoSQL como sistema de gestión de bases de datos, a través de un ensayo argumentativo, recopilando los principales hallazgos expuestos en artículos y libros recientes, con el fin de señalar los rasgos sustanciales de esta temática.

Objetivos específicos

Explicar tanto el origen como la definición de NoSQL, abordando las diferentes discusiones existentes y formulando la necesidad por la cual surge esta tecnología.

Comparar los principales tipos de bases de datos NoSQL, relacionando las cualidades de cada una y señalando en qué sistemas de información se utilizan en la actualidad.

Distinguir los motores para bases de datos NoSQL que se emplean en la actualidad según su tipo, los tipos de datos que soporta y una interpretación simple de su código.

Examinar la aplicación de bases de datos NoSQL en Colombia, verificando tanto la necesidad como su eficacia.

Metodología de la investigación

El desarrollo del presente informe inicia tratando de explicar el concepto de sistema de gestión de bases de datos y la forma en que derivó NoSQL a partir de las bases de datos relacionales, sin dejar de explicar los principales términos que rodean las bases de datos no relacionales, posteriormente, se describirán los tipos de bases de datos NoSQL existentes junto con los motores que se emplean en mayor medida, finalizando con la exposición de algunos casos de éxito en Colombia y una corta discusión sobre la importancia de formar profesionales capaces de crear soluciones donde se aplique este tipo de tecnología.

Marco conceptual y teórico

Los sistemas de gestión de bases de datos nacen por la necesidad de desarrollar sistemas de información que respondan a la competitividad en la globalización de la información, la cual requiere ser estructurada, organizada y consultada (Capacho Portilla y Nieto Bernal, 2017). Por lo tanto, la base de datos es un elemento fundamental en un producto de software, sea este una aplicación de escritorio, un sitio web, una aplicación móvil, entre otros; es el mecanismo que permite almacenar tanto los datos como la información que se gestiona en el producto de software.

Existen diferentes tipos de sistemas de gestión, el más conocido es el sistema de gestión de bases de datos relacional, ya que este tipo define su estructura a través del modelo relacional, el cual está basado en la teoría de conjuntos. Básicamente, en el modelamiento de una base de datos relacional se identifican inicialmente los conjuntos de datos que conformarán las entidades, posteriormente se caracterizan los atributos de cada entidad y sus relaciones, las cuales definirán el grado de asociación o dependencia entre estas, posteriormente, se finaliza el modelamiento con la normalización de la base de datos, por lo general hasta la tercera forma normal. Ha sido uno de los modelos más usados en proyectos de TI, ya que proporciona integridad a la base de datos, evita la redundancia de los datos y el proceso de normalización es bastante conocido entre desarrolladores y programadores de software.

Sin embargo, en las bases de datos relacionales se han presentado dificultades cuando es relevante la disponibilidad y la rapidez de los datos, ante esto surge NoSQL como una alternativa a las bases de datos relacionales.

1. Qué es NoSQL.

Se le atribuye el origen del término como tal a Carlo Strozzi, quien lo utilizó por primera vez con el fin de referirse de forma específica a una base de datos relacional con código abierto que prescindía del lenguaje SQL (1998). Para Venkatraman, Fahd, Kaspi y Venkatraman las bases de datos NoSQL se caracterizan por ser no relacionales, distribuidas, de código abierto y escalables horizontalmente (2016), además, Camargo-Vega, Camargo-Ortega y Joyanes-Aguilar afirman que los principales aspectos que diferencian a NoSQL del modelo relacional son: no tienen esquemas, no usan SQL ni Joins, no almacenan datos en tablas de manera uniforme, usan la memoria principal del computador y su objetivo es gestionar grandes volúmenes de información (2015). En la actualidad el concepto de NoSQL representa el modelo para bases de datos no relacionales.

Cabe anotar que la escalabilidad es la capacidad de adaptación de un software cuando se requiere incrementar su capacidad, conjuntamente, una base de datos distribuida es aquella que está dividida en diferentes espacios tanto lógicos como físicos. Una base de datos NoSQL suele estar distribuida entre varios equipos que trabajan como si fuesen un único servidor, a este grupo de equipos se les llama clúster, por ende, la escalabilidad horizontal es la capacidad de añadir más equipos al clúster para mejorar su rendimiento. En consecuencia, NoSQL es una base de

datos distribuida, por lo tanto, es necesario señalar que según el teorema CAP, en una base de datos NoSQL se garantiza el cumplimiento simultáneo de máximo dos de las siguientes características: consistencia, disponibilidad y tolerancia al particionado.

Según Venkatraman et al. el objetivo principal de NoSQL es permitir un fácil almacenamiento y recuperación de datos, independientemente de su estructura y contenido, lo que es posible debido a la inexistencia de una estructura de datos rígida en bases de datos no relacionales (2016), así mismo, Pérez Román afirma que NoSQL emplea modelos de datos muy específicos usando esquemas flexibles, sus tablas no poseen una estructura fija, permite almacenar grandes volúmenes de datos y proporciona baja latencia (2020). Estas características hacen que una base de datos NoSQL sea más liviana y veloz que una base de datos relacional, por dicha razón las redes sociales utilizan este tipo de bases de datos precisamente por la cantidad de usuarios que están conectados al mismo tiempo y que frecuentemente realizan consultas, registran y comparten información, alterando constantemente la base de datos de la red social, incluso donde gran parte de esta es semiestructurada.

Por ejemplo, en la Tabla 1 se definen los atributos de la entidad “Estudiantes” de forma estructurada, pensando en el diseño de una base de datos relacional, se puede observar que las restricciones asignadas a cada atributo proporcionan una estructura definida para los datos por guardar, en este caso, todos los atributos de la entidad al ser no nulos deben contener datos y tienen un tamaño máximo definido:

Tabla 1

Atributos de la entidad “Estudiantes” para una base de datos relacional.

Atributos	Tipo de Dato	Tamaño	Lista de Valores	Rango de Valores	Restricción de Control
IdUsuario	Número	6	--	1 - 999999	PRIMARY KEY, IDENTITY
TipoDocumento	Texto	2	CC, CE, TI	--	NOT NULL, CHECK
NumeroDocumento	Número	10	--	1 - 9999999999	NOT NULL, UNIQUE
Nombres	Texto	50	--	--	NOT NULL
Apellidos	Texto	50	--	--	NOT NULL
Sexo	Texto	1	F, M	--	NOT NULL, CHECK
Email	Texto	50	--	--	NOT NULL
Celular	Número	10	--	1 - 9999999999	NOT NULL
Usuario	Texto	10	--	--	NOT NULL, UNIQUE

Nota: Autoría Propia.

Por el contrario, NoSQL permite almacenar datos semiestructurados, es decir, se puede crear una colección llamada “Estudiantes” donde el primer valor u objeto a guardar podría tener solo los siguientes datos: tipo de documento, nombres, apellidos, email y celular. Posteriormente, el segundo valor u objeto a guardar podría tener: nombre, email y usuario. Además, se podría preestablecer o no el tipo y tamaño de los datos. En consecuencia, la información de dicha colección se puede guardar sin seguir una estructura predefinida.

Dentro de la bibliografía consultada para el desarrollo del presente informe se encontró que los tipos de bases de datos NoSQL con mayor aplicación en la actualidad son: clave-valor, documentos y columnas, no obstante, vale la pena resaltar a las bases de datos en grafos, las cuales están basadas en la teoría de grafos y podrían ser unas de las bases de datos NoSQL con

mayor potencial de desarrollo. Así mismo, los motores de bases de datos que más se emplean según su tipo son respectivamente: Redis, MongoDB, Hbase y Neo4j.

2. Tipos de bases de datos NoSQL

2.1 Clave-valor

Según Lombardi esta categoría soporta disponibilidad continua y cargas extremas, los datos se almacenan con un modelo de datos sin esquema en un elemento que contiene una clave junto con los datos, siendo bastante similar a una tabla Hash (2018).

Cabe resaltar que una tabla Hash es un tipo de estructura de datos donde se relaciona una clave con un valor a partir de una función Hash que arroja un índice, por lo general alfanumérico. Un índice puede relacionarse con varias claves, es decir colisionan al ser almacenados, por ende, la tabla se fragmenta a partir de los índices, optimizando la búsqueda de dentro de la tabla Hash. En la Ilustración 1 se muestra un ejemplo de una tabla Hash.

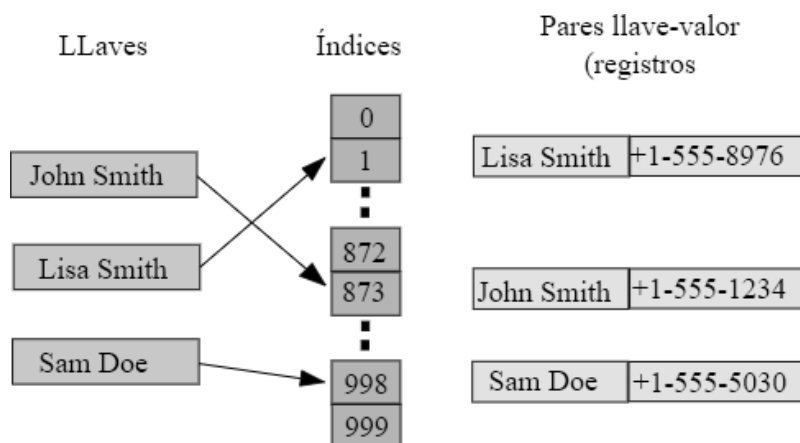


Ilustración 1: Ejemplo de una tabla Hash. Recuperado de: https://es.wikipedia.org/wiki/Tabla_hash

Las bases de datos de clave-valor se caracterizan por ser la categoría más fácil de implementar, también porque permite almacenar cualquier tipo de información sin tener en cuenta el tamaño de los datos, incluso, dentro de una clave se pueden asociar valores de tipo clave-valor, aunque para Armillas García presenta limitaciones en cuanto al modelo de datos, ya que estos se guardan en valores binarios, y no permite búsquedas complejas, solamente por medio de su clave (2015). Este tipo de base de datos es usado en el timeline de Twitter y en los tableros temáticos de Pinterest.

2.2 Documentos

Armillas García dice que las bases de datos de documentos son en esencia un clave-valor con la excepción de que el valor no se guarda sólo como un campo binario, sino con un formato definido de tal forma que el servidor puede entenderlo (2015). En la lustración 2 se muestra un ejemplo de una base de datos de documentos en formato JSON:

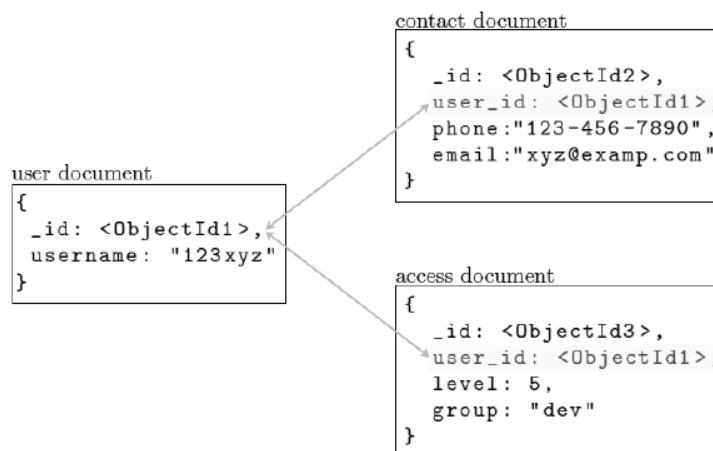


Ilustración 2: Ejemplo de una base de datos de documentos en formato JSON (Lombardi, 2018, p. 29).

Cabe anotar que JSON es un acrónimo de JavaScript Object Notation (Notación de Objetos de JavaScript), en su momento surgió como alternativa al formato XML, JSON es un formato de texto conocido entre programadores que se usa para el intercambio de datos, soporta los siguientes tipos de datos: numérico, cadenas, booleanos, nulos, arreglos y objetos.

En esta categoría los datos se pueden esquematizar según su tipo, cada segmento de los datos es un documento, en el ejemplo de la Ilustración 2 primero se crea un documento donde se guarda el valor del ID de un usuario en una clave, dicha clave se asocia a otro documento donde se guardan los datos de contacto del usuario y en otro documento se asocia el nivel de acceso de dicho usuario asociado con la misma clave. Este tipo de base de datos está implementado en los check-in de la red social Foursquare y en las estadísticas de tráfico de MTV Networks.

2.3 Columnas

En esta categoría los datos se guardan en columnas, Armillas García explica lo siguiente (2015):

“... se almacena más como una matriz tridimensional, donde una dimensión es el identificador de fila (una clave primaria), la segunda es la combinación de la familia de la columna y el identificador de la columna, la tercera dimensión consiste en la indicación del tiempo, de forma que para un mismo valor se guardan todos los cambios que ha tenido dicho valor”.

En la Ilustración 3 se muestra un ejemplo de la estructura de una base de datos de columnas:

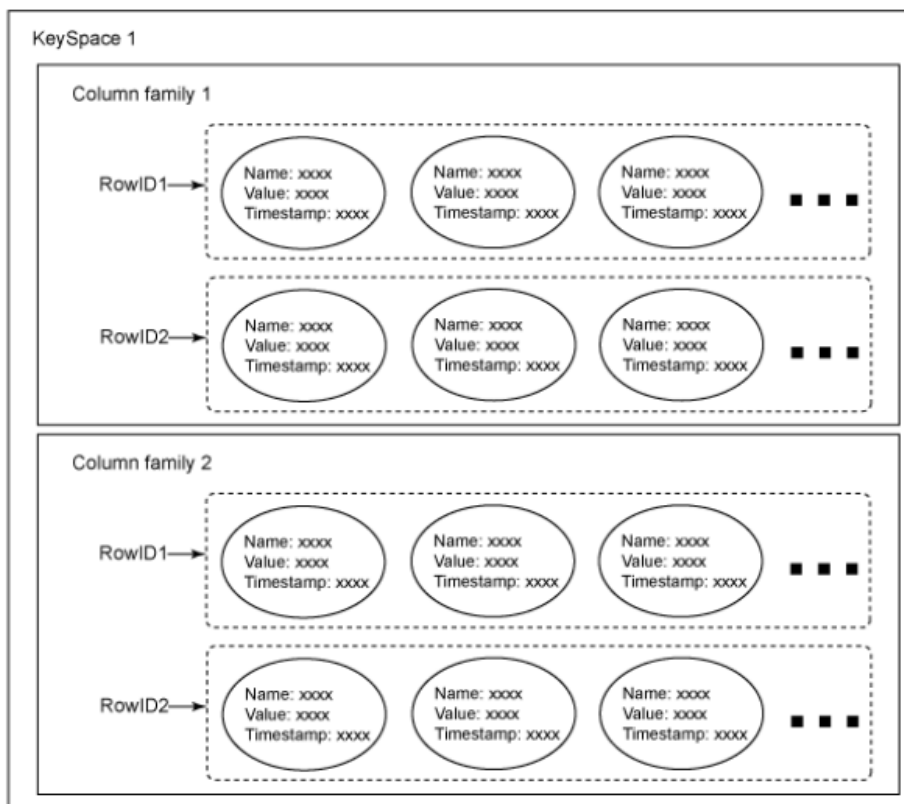


Ilustración 3: Ejemplo de la estructura de una base de datos de columnas (Armillas García, 2015, p. 39).

Este tipo de bases de datos también utiliza JSON como formato de texto para interactuar con el sistema de información, se usa especialmente cuando se requiere estudiar la tendencia en grandes volúmenes de datos registrados, muestra de ello es la relevancia del tiempo en que se registra la información dentro de la base de datos, tal como lo expresa Herranz Gómez, este tipo de base de datos provee mayor rendimiento cuando se utilizan técnicas de Minería de Datos para consultar y analizar grandes cantidades de información para inteligencia de negocio, informes de marketing o de ventas, entre otros (2014), en consecuencia, este tipo de bases de datos funciona de manera óptima cuando es necesario realizar consultas de manera rápida en una porción

concreta de columnas. La aplicación Messenger de Facebook utiliza una base de datos de columnas, también Spotify lo utiliza para conocer las preferencias de sus usuarios.

2.4 Grafos

También se le conoce como bases de datos de gráficos, la cual está basada en la teoría de grafos, Lombardi explica que en este modelo los datos se almacenan en forma de nodos, relaciones y propiedades, con el fin de construir un gráfico de los datos sin un esquema fijo impuesto, por lo tanto, es una buena opción para datos semiestructurados, incluso, cuando los nodos o las relaciones tienen menos o más propiedades no es necesario alterar el diseño (2018).

Cabe anotar que en las bases de datos de grafos se utiliza Cypher como lenguaje programación, este es un lenguaje declarativo, está basado en SQL, pero a su vez utiliza los elementos de la teoría de grafos, tales como nodos, relaciones, propiedades y etiquetas; está diseñado específicamente para la programación de bases de datos de grafos.

En la Ilustración 4 se muestra un ejemplo con la estructura de una base de datos de grafos:

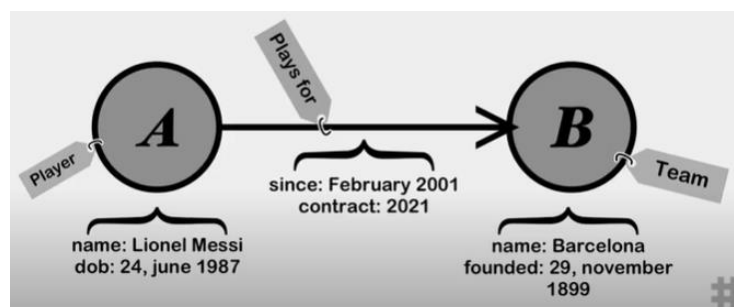


Ilustración 4: Ejemplo de la estructura de una base de datos de grafos (Feregrino, 2018).

Como se mencionó anteriormente, en este tipo de bases de datos es posible almacenar datos tanto en los nodos como en las relaciones, incluso, es posible almacenar datos en las propiedades de cada nodo y relación, de esta manera, se puede tratar a las relaciones con la misma prelación que a los nodos.

Las bases de datos de grafos se destacan dentro del grupo de bases de datos NoSQL por su excepcional velocidad cuando se consultan grandes volúmenes de datos, especialmente cuando se compara con una consulta en una base de datos relacional donde se deben implementar una suma considerable de JOINS, ya que las bases de datos de grafos aprovechan las relaciones existentes entre los datos, optimizando la velocidad de ejecución. Este tipo de base de datos se utiliza actualmente en casi todas las redes sociales, por ejemplo, en el sitio web de Infojobs se emplea este tipo de base de datos para relacionar constantemente las mejores opciones de empleo para un usuario según su formación profesional o su experiencia laboral, también se destaca el uso en la aplicación de eBay Now, el cual coordina el envío de productos en menos de 2 horas a través de una base de datos de grafos.

3. Motores para bases de datos NoSQL.

3.1 Redis

Este motor se especializa en bases de datos de clave-valor, su nombre proviene del acrónimo *Remote Dictionary Server*, Herranz Gómez explica que su funcionalidad básica es almacenar datos en memoria para conferir velocidad en los tiempos de respuesta, pero con un respaldo en

disco para que si un nodo se apaga siempre se pueda recuperar la última información escrita en disco (2014). En Redis se manejan los siguientes tipos de datos: cadenas (*string*), hash, listas (*list*), conjuntos (*set*) y conjuntos ordenados (*sorted set*).

Las bases de datos de clave-valor son muy simples en su manejo, en el siguiente ejemplo se usan los comandos SET y GET para asignar y leer valores en de tipo cadena, y los comandos LPUSH y LRANGE se utilizan para asignar y leer valores de tipo lista:

```
> SET cadena1 hola
OK
> GET cadena1
“hola”

> LPUSH lista1 carlos
(integer) 1
> LPUSH lista1 enrique
(integer) 2
> LPUSH lista1 sanchez
(integer) 3
> LPUSH lista1 pardo
(integer) 4
> LRANGE lista1 0 10
1) “carlos”
2) “enrique”
3) “sanchez”
4) “pardo”
```

3.2 MongoDB

Este motor se especializa en bases de datos de documentos, su nombre deriva de la palabra *humongous* (enorme), Armillas García dice que en MongoDB no es necesario definir un esquema, todo depende de si se necesita trabajar con datos normalizados, desnormalizados o con

una aproximación híbrida, y en gran parte el esquema lo define el tipo de consultas que se ejecuten con mayor frecuencia. (2015). Los tipos de datos que se manejan en MongoDB son: números, cadenas (*string*), arreglos (*arrays*) y subdocumentos.

Este motor utiliza JSON para enviar instrucciones a la base de datos, JSON es un acrónimo de *JavaScript Object Notation* (Notación de Objetos de JavaScript), en su momento surgió como alternativa al formato XML, JSON es un formato de texto conocido entre programadores que se usa para el intercambio de datos, soporta los siguientes tipos de datos: numérico, cadenas, booleanos, nulos, arreglos y objetos. En el siguiente ejemplo se muestra su uso para crear un documento con cada tipo de dato que soporta tanto JSON como MongoDB:

```
documento1 =
{
  nombre : "carlos",
  edad : 36,
  gustos : ["informática", "futbol", "videojuegos"],
  correos :
  {
    personal : "carlos@correo.com"
    trabajo : "carlos@trabajo.com"
  }
}
```

3.3 HBase

Es un motor de datos especializado en base de datos de columnas, Herranz Gómez comenta que este motor fue construido sobre el proyecto Hadoop de Apache, el cual le confiere al motor dos características importantes, la primera es HDFS que corresponde a la capa de almacenamiento que le proporciona disponibilidad y fiabilidad, y la segunda es MapReduce

como entorno de computación de alto rendimiento (2014). Los tipos de datos en este motor no requieren ser declarados, ya que todos valores se capturan como cadenas (*string*) entre apóstrofes.

En HBase se utiliza el formato *create* '*<nombre tabla>*', '*<familia de columnas>*' para crear una tabla, y el formato *put* '*<nombre tabla >*', '*numero de fila*', '*<familia de columnas:nombre columna>*', '*<valor>*', en el siguiente ejemplo se crea una tabla con dos familias de columnas, posteriormente, se registran datos a la tabla en donde a cada familia se le agregan dos columnas:

```
> create 'Créditos', 'Datos Iniciales', 'Tasas'
0 row(s) in 1.1300 seconds

> put 'Créditos', '1', 'Datos Iniciales:Valor Desembolso', '$50,000,000.00'
0 row(s) in 0.7600 seconds

> put 'Créditos', '1', 'Datos Iniciales:Plazo en meses', '36'
0 row(s) in 0.5100 seconds

> put 'Créditos', '1', 'Tasas:NAMV', '18.96%'
0 row(s) in 0.6600 seconds

> put 'Créditos', '1', 'Tasas:NMV', '1.58%'
0 row(s) in 0.5600 seconds

> put 'Créditos', '2', 'Datos Iniciales:Valor Desembolso', '$78,000,000.00'
0 row(s) in 0.7800 seconds

> put 'Créditos', '2', 'Datos Iniciales:Plazo en meses', '60'
0 row(s) in 0.4900 seconds

> put 'Créditos', '2', 'Tasas:NAMV', '12.24%'
0 row(s) in 0.6300 seconds

> put 'Créditos', '2', 'Tasas:NMV', '1.02%'
0 row(s) in 0.5900 seconds
```

Con base en el ejemplo anterior, en la Tabla 2 se plasman los datos almacenados:

Tabla 2

Ejemplo de una tabla en una base de datos de columnas.

Clave de la fila	Datos Iniciales		Tasas	
	Valor Desembolso	Plazo	NAMV	NMV
1	\$50,000,000.00	36	18.96%	1.58%
2	\$78,000,000.00	60	12.24%	1.02%

Nota: Autoría Propia.

3.4 Neo4j

Es un motor de datos especializado en base de datos de grafos con código abierto, Lombardi expone que este motor utiliza adyacencia sin índice donde cada nodo mantiene referencias directas a sus nodos adyacentes, esta técnica evita el uso de índices globales, es mucho más económica y reduce los tiempos de consulta porque son independientes del tamaño total del gráfico (2018).

Como se mencionó anteriormente, en las bases de datos de grafos se utiliza Cypher como lenguaje de programación, dicho lenguaje puede manejar varios tipos de datos, los cuales se agrupan en:

Básicos: números enteros (integer), números flotantes (float), cadenas (string), booleanos, listas (list) y clave-valor.

De estructura: nodo (node), relación (relationship) y ruta (path).

En el siguiente ejemplo se muestra el código que se emplearía para crear los nodos y la relación de la Ilustración 4 (Feregrino, 2018), vale la pena resaltar que la información de los nodos queda almacenada en m y b , mientras que la información de la relación queda almacenada en p , de manera intuitiva se deduce el propósito de los comandos *CREATE*, *WITH*, *SET* y *RETURN*:

```
CREATE (m: Player {name: "Leonel Messi", dob: date("24-06-1987")}),
      (b: Team {name: "Barcelona", founded: date("29-11-1899")})
WITH m, b
CREATE (m)-[p: PlayFor]->(b)
SET p.since=date("2001-02-01")
RETURN m, p, b
```

4. Aplicación de bases de datos NoSQL en Colombia.

En Colombia la gran mayoría de ciudades principales y municipios han implementado un sistema de transporte masivo basado en Autobuses de Transito Rápido (BRT por sus siglas en inglés), por tal motivo, un grupo de investigadores diseñó un prototipo de sensor inteligente ITS para ayudar a la planificación del transporte para BRT, este sensor detecta señales Bluetooth de teléfonos móviles e implementa un almacenamiento en la nube a través de una base de datos NoSQL, durante las pruebas realizadas se demostró que el tiempo de respuesta tiende a aumentar a 700 ms y posteriormente se estabiliza, lo que se atribuye al autoescalado de la nube en respuesta a la alta demanda de recursos (Herrera-Quintero, Vega-Alfonso, Banse, y Carrillo Zambrano, 2018). En este caso, NoSQL facilita el almacenamiento masivo de señales Bluetooth

que se recolectan a través de los sensores, además, el almacenamiento en la nube requiere de la escalabilidad que provee este tipo de tecnología.

Por otra parte, existe también un diseño de software denominado SCADA, orientado para las granjas avícolas en Colombia, puntualmente el software pretende predecir el posible peso futuro de los pollos y con ello tener bases para tomar decisiones inteligentes al respecto, el software cuenta con un sistema de recolección de datos sensóricos instalados en los galpones de las granjas, diariamente se podrían recolectar una gran cantidad de datos a través de estos sensores, por lo que se implementó una base de datos NoSQL (Valderrama Mendoza, Rodríguez Urrego, Cobo, y María Martínez, 2019).

Finalmente, en el sistema de pronóstico agroclimático digital denominado *Pronosticos AClimateColombia*, el cual está disponible en <https://pronosticos.aclimatecolombia.org>, se emplea el motor MongoDB en la capa del sistema de bases de datos, este sistema de pronóstico como tal formaliza el procesamiento de la información climática y de cultivos, por lo que este tipo de base de datos le permite almacenar diariamente grandes cantidades de datos meteorológicos, tanto de información histórica como de pronósticos generados por el sistema, además, permite fácilmente la adición de más localidades o de más fuentes de datos meteorológicos, asimismo, se pueden agregar otros modelos de cultivos para generar nuevos pronósticos agroclimáticos (Sotelo et al., 2020).

Conclusiones

NoSQL es una temática que ha ganado relevancia en los últimos años, especialmente con la masificación de la información que se produce minuto a minuto no solamente en las redes sociales, información que en su gran mayoría se conforma de datos semiestructurados y que requiere consultarse con rapidez.

Muestra de ello son los resultados obtenidos por Pérez Román en su investigación (2020), donde comparó el rendimiento entre los motores PostgreSQL y MongoDB a la hora de realizar la consulta de 10.000 operaciones, los resultados indicaron que en PostgreSQL se tardó 2.774 ms, mientras que en MongoDB se tardó solo 7 ms, una diferencia de tiempo excepcional.

De modo que, si en un proyecto de TI es necesario almacenar datos y consultarlos a velocidades ínfimas, entonces es pertinente considerar el desarrollo de una base de datos NoSQL para dicho proyecto como alternativa a las bases de datos relacionales, no obstante, si en Colombia se pretende dar un salto de calidad en cuanto a la gestión de proyectos TI, entonces es indispensable la conformación de profesionales capaces de desarrollar dichos proyectos, de cara a las nuevas exigencias de TI y a las nuevas tendencias de desarrollo para bases de datos no relacionales.

Referencias bibliográficas

- Armillas García, P. (2015). Estudio comparativo de BBDD NoSQL y relacionales aplicadas al ámbito de la información geográfica (Bachelor's thesis, Universitat Politècnica de Catalunya).
- Cáceres, N. R., Tolaba, C., Pérez, R. D., Quispe, J. J. M., Rodríguez, C. S., & Sandoval, I. L. (2018). Benchmarking de Bases de Datos NoSQL para el almacenamiento de Modelos Semánticos. In XX Workshop de Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste).
- Camargo-Vega, J. J., Camargo-Ortega, J. F., & Joyanes-Aguilar, L. (2015). Conociendo big data. Facultad de Ingeniería, 24(38), 63-77.
- Capacho Portilla, J. R., & Nieto Bernal, W. (2017). Diseño de base de datos. Universidad del Norte.
- Feregrino, A. [That C# guy] (2018, agosto 14). Bases de datos de grafos - Neo4j [Archivo de video]. https://youtu.be/k3h_y9w_714
- Herranz Gómez, R. (2014). Bases de datos NoSQL: arquitectura y ejemplos de aplicación (Master's thesis).
- Herrera-Quintero, L., Vega-Alfonso, J., Banse, K., & Carrillo Zambrano, E. (2018). Smart ITS Sensor for the Transportation Planning Based on IoT Approaches Using Serverless and Microservices Architecture. IEEE Intelligent Transportation Systems Magazine, Intelligent Transportation Systems Magazine, IEEE, IEEE Intell. Transport. Syst. Mag, 10(2), 17–27. <https://doi-org.bibliotecavirtual.unad.edu.co/10.1109/MITS.2018.2806620>

- Lombardi, I. (2018). Análisis y estudio de la tecnología de bases de datos orientadas a grafos, focalizado en el sistema Neo4j.
- Pérez Román, A. (2020). Comparación de rendimiento entre bases de datos Relacionales, NoSQL y Blockchain Comparación de rendimiento entre PostgreSQL, MongoDB y Kaleido.
- Rossel, G., Manna, A., & Bugni, F. (2016). Incorporando NoSQL en el currículo de Bases de Datos. In XI Congreso de Tecnología en Educación y Educación en Tecnología (TE&ET 2016).
- Sotelo, S., Guevara, E., Llanos-Herrera, L., Agudelo, D., Esquivel, A., Rodriguez, J., Ordoñez, L., Mesa, J., Muñoz Borja, L. A., Howland, F., Amariles, S., Rojas, A., Valencia, J. J., Segura, C. C., Grajales, F., Hernández, F., Cote, F., Saavedra, E., Ruiz, F., ... Ramirez-Villegas, J. (2020). Pronosticos AClimateColombia: A system for the provision of information for climate risk reduction in Colombia. *Computers and Electronics in Agriculture*, 174. <https://doi-org.bibliotecavirtual.unad.edu.co/10.1016/j.compag.2020.105486>
- Strozzi, C. (1998). NoSQL: A relational database management system. *Lainattu*, 5, 2014.
- Tabla Hash (s.f.). En Wikipedia. https://es.wikipedia.org/wiki/Tabla_hash
- Valderrama Mendoza, M., Rodríguez Urrego, L., Cobo, L., & María Martínez, G. (2019). Sistema de análisis para el incremento de la producción de granjas avícolas en Colombia. Caso de estudio: proyecto proavícola. (Spanish). *Revista Avances: Investigación En Ingeniería*, 16(1), 7.
- Venkatraman, S., Fahd, K., Kaspi, S., & Venkatraman, R. (2016). SQL versus NoSQL movement with big data analytics. *Int. J. Inform. Technol. Comput. Sci*, 8, 59-66.