DOCUMENTO MAESTRO BIG DATA Y NOSQL

Bayron Danilo Ortiz Foronda - 160002925
e-mail: bayron.ortiz@unillanos.edu.co

Juan Manuel Cuero Ortega - 160003039
e-mail: juan.cuero@unillanos.edu.co

CONTENIDO

1. Big Data

1.1. Antecedentes Big Data

No existe un punto de nacimiento del Big Data, pero se relaciona de gran manera con la confluencia de la gran cantidad de tendencias tecnológicas que venían madurando desde la primera década del s.XXI, y que se han consolidado durante los años 2011 y 2013, cuando ha explosionado e irrumpido con gran fuerza en organizaciones y empresas, en particular, y en la sociedad en general, con el uso de las redes sociales, aumento banda ancha, aparición de los smartphones, geolocalización entre otras muchas.[1]

[1]Algunos datos interesantes:

- Durante 2011 se crearon 1.8 zettabytes (ZB)
- Walmart para el año 2013 poseía bases de datos con capacidad para 2.5 petabytes (PB)

- Nace un nueva profesión la analítica de datos, con alta demanda por parte de las organizaciones.

Es por ello que Big Data, se ha convertido en una materia prima de suma importancia en las organizaciones, para la toma de decisiones y competitividad.

1.2. ¿Qué es Big Data?

El término Big Data actualmente no se encuentra estandarizado, ya que diferentes autores plantean definiciones alrededor del término como por ejemplo las siguientes:

- Gartner, "Big Data es un gran volumen, velocidad o variedad de información que demanda formas costeables e innovadoras de procesamiento de información que permitan ideas extendidas, toma de decisiones y automatización del proceso".[4]

- Dan Kusnetzky, señala que "La frase Big Data se refiere a las herramientas, procesos y procedimientos que permitan a una organización crear, manipular y administrar grandes conjuntos de datos e instalaciones de almacenamiento".[4]
- Forrester define Big Data como las "técnicas y tecnologías que hacen que sea económico hacer frente a los datos a una escala extrema".[4]

A pesar de que cada uno de los autores, plantea puntos de vista diferentes, un poco enfocados en su campo de acción, existe un consenso en que Big Data es un gran volumen de información, que para entenderlo y manipularlo, se hacen necesario el uso de técnicas y herramientas.

Por tanto generalizando un poco, Big Data puede definirse simplemente, como un término que incluye diferentes tecnologías asociadas a la administración de grandes volúmenes de datos, provenientes de diferentes fuentes y que se generan con rapidez.[3] Es importante resaltar que Big Data no va dirigido sólo a grandes volúmenes de información, sino que es un concepto mucho más amplio, que abarca variedad de datos, velocidad de acceso y procesamiento [3]; son datos que están en constante cambio y crecimiento (dinámicos).

1.3. Tipos de Datos en Big Data.

En Big Data se hace necesario dividir los tipos de datos en diferentes categorías que incluyen las siguientes:

1.3.1. Estructurados.

La mayoría de las fuentes de datos tradicionales son datos estructurados, datos con formato o esquema fijo que poseen campos fijos. En estas fuentes los datos vienen en un formato bien definido que se especifica en detalle, y que conforma las

bases de datos relacionales.[1] Ejemplos de datos estructurados incluyen fecha de nacimiento, números de cuenta, documentos de identidad.

1.3.2. Semiestructurados.

Los datos semiestructurados tienen un flujo lógico y un formato que puede ser definido, pero no es fácil la comprensión por parte del usuario. Datos que contiene formatos fijos, pero contienen etiquetas y otros marcadores que permiten separar los elementos dato. La lectura de este tipo de datos, requiere uso de reglas complejas que determinan cómo proceder después de la lectura de cada pieza de información.[1] Ejemplos de datos semiestructurados incluyen archivos HTML, XML, logs, CVS.

1.3.3. No Estructurados.

Los datos no estructurados son datos sin tipos predefinidos. Se almacenan como documentos u objetos sin estructura uniforme, y se tiene poco o ningún control sobre ellos. Datos de texto, video, audio, fotografía son datos no estructurados. Estos datos son sin duda los más difíciles de dominar en la actualidad, de tal manera que ha provocado el nacimiento de herramientas para su manipulación como es el caso de las bases de datos NoSQL.[1]

1.4. Características del Big Data.

Las principales características que definen el Big Data y que cumple para cualquier industria o sector sobre el cual se esté relacionado, abarcan el módelo de las 3V (Volumen, Velocidad, Variedad). Pero este modelo no es suficiente y algunas fuentes, como es el caso de IBM agrega otra característica adicional al Big Data que es la veracidad y el valor.[1][2]

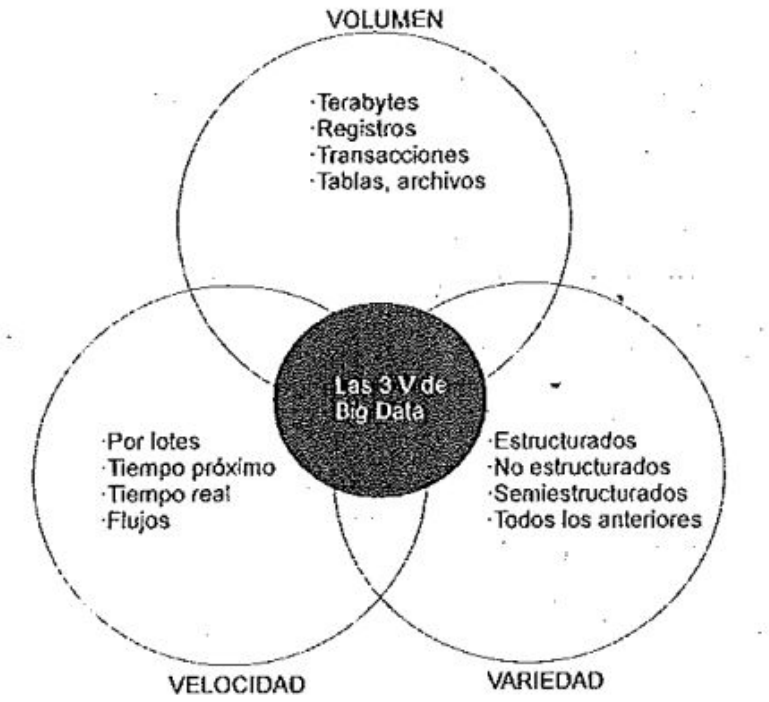

Figura 1. Las 3V del Big Data. Tomado de [1]

A continuación se explica en detalle cada una de las características principales del Big Data, que es necesario conocer.

1.4.1. Volumen.

Las empresas amasan grandes volúmenes de datos, desde terabytes hasta petabytes. Se esta pasando de la era del petabyte a la era del exabyte y para 2015 -2020, se espera entrar en la era del zettabyte. Dado lo anterior es importante para las organizaciones invertir en plataformas adecuadas para analizar todos estos datos con el objetivo de conseguir una mejor comprensión de sus negocios, clientes y el mercado. Ejemplo de empresas u organizaciones que producen gran volumen de datos son Facebook con 10TB diarios y Twitter con 9TB diarios.[1]

1.4.2. Velocidad.

La importancia de la velocidad de los datos o el aumento creciente de los flujos de datos en las organizaciones junto con la frecuencia de las actualizaciones de las grandes bases de datos son características a tener en cuenta. Esto significa que el procesamiento y posterior análisis de la información debe hacerse en tiempo real, para mejorar la toma de decisiones. Ejemplo de ello son los millones de escrutinios de los datos de un banco para detectar algún tipo de fraude, o el análisis de cientos y millones de llamadas de los clientes de una compañía telefónica, con el fin de predecir el comportamiento de estos y evitar que abandonen la compañía.[1] Esta característica tan importante, en algunas ocasiones se une con la siguiente que es la variedad; asociando ya no solo un crecimiento acelerado sino movimiento de los datos.

1.4.3. Variedad.

Abarca todas aquellas fuentes de datos de cualquier tipo. Los datos pueden ser estructurados y no estructurados, y cuando se analizan juntos se requieren nuevas técnicas. Este gran volumen de datos tan variado puede llevar a la confusión que impida ver oportunidades y amenazas dentro de los campos de negocio y acción de las organizaciones, dando como resultado baja competitividad. Por estas razones, las empresas deben capitalizar las oportunidades de los grandes datos, y deben ser capaces de analizar todos los tipos de datos, desde los estructurados y no estructurados.[1]

1.4.4. Veracidad.

IBM introdujo esta característica, pero la definición de la fuente, es bastante pobre pues relata un poco de que la mayoría de líderes o directivos de organizaciones no creen en la información que utilizan para tomar decisiones; sin embargo el establecimiento de la veracidad o fiabilidad de Big Data supone un gran reto a medida que la variedad y las fuentes de datos crecen.[1]

1.4.5. Valor.

Otra característica señalada por IBM, pues las organizaciones lo que finalmente buscan es obtener información de los grandes volúmenes de datos para ser aprovechados en sus estrategias organizacionales de manera rentable y eficiente.[1]

Actualmente las organizaciones se esfuerzan de alguna manera de gestionar el Big Data, pero no todas esta interesadas por el desconocimiento que existe sobre esta nueva tendencia, que se esta produciendo en la era de la información. Pero es importante resaltar que la información es poder y el Big Data, si se logra gestionar de manera correcta, puede proporcionar gran cantidad de conocimiento; es por ello que se debe hablar un poco de las herramientas que han surgido como es el caso de las bases de datos NoSQL, de las cuales se tratará en el siguiente capitulo.

2. Bases Datos NoSQL

2.1. ¿Que son las bases de datos NoSQL?

Las bases de datos NoSQL son sistemas de almacenamiento de información que no cumplen con el esquema entidad-relación. Las bases de datos NoSQL no imponen una estructura de datos en forma de tablas y relaciones entre ellas sino que proveen un esquema mucho más flexible. [7]

El término NoSQL fue utilizado por primera vez en 1998 por Carlo Strozzi para referirse a una base de datos open-source que omitía el uso de SQL, pero sí seguía el modelo relacional. Dicha base de datos utilizaba como interfaz de usuario lenguaje shell de UNIX.

El término fue re-introducido por Eric Evans, empleado de Rackspace, cuando Johan Oskarsson de Last.fm pregunto en IRC cual sería un buen nombre para la primera reunión de bases de datos distribuidas de código abierto que estaba organizando. [8]

2.2. Características NoSQL

Aunque es difícil determinar propiedades comunes para un conjunto de tecnologías, se proponen seis características específicas para poder encasillar a las bases de datos NoSQL:

- Escalabilidad horizontal: refiriéndose a la facilidad añadir, eliminar o realizar operaciones con elementos (hardware) del sistema, sin afectar el rendimiento.
- Habilidad de distribución: tiene que ver con las escalabilidad horizontal, pero haciendo énfasis en su soporte; para ello se tiene en cuenta la habilidad de replicar y distribuir los datos sobre los servidores.
- Uso eficiente de recursos: aprovecha las nuevas tecnologías, como los discos en estado sólido, el uso eficiente de recursos como la memoria RAM y los sistemas distribuidos en general.
- Libertad de esquema: al no tener un esquema rígido se permite mayor libertad para modelar los datos; además facilita la integración con los lenguajes de programación orientados a objetos, lo que evita el proceso de mapeado.
- Modelo concurrencia débil: no implementa ACID, que reúne las características necesarias para que una serie de instrucciones puedan ser consideradas una transacción, sin embargo sí se tienen en cuenta algunas consideraciones para asegurar estos aspectos, pero no son tan estrictas.
- Consultas simples: las consultas requieren menos operaciones y son más naturales, por la tanto, se gana en simplicidad y eficiencia. [5]

2.3. Tipos de Bases de datos NoSQL

Las bases de datos no relacionales dejan de darle mayor importancia a la consistencia o a la disponibilidad, para solucionar las necesidades en los nuevos sistemas o formas de negocios que están surgiendo en la actualidad; las bases de datos no relacionales se han dividido en cuatro grandes grupos que se describen a continuación. Dependiendo de las taxonomías de las bases de datos no relacionales se pueden dividir en llave-valor, documentos, familias de columnas o por grafos. Esto indica a como se almacena la información en las bases de datos NoSQL. [6]

Clave - valor

Es la más popular, quizá por su sencillez en el aprendizaje y funcionalidad; intensifica su rendimiento en la obtención de información por la forma de almacenarla, de tal forma que cada elemento está referenciado por una clave única. En el elemento se pueden agregar cualquier tipo de valor.

Lo mínimo que se tiene en una NoSQL con características clave-valor son las siguientes:

- Agregar valor y llave: put (key, value).
- Obtener valor: get(key).
- Eliminar llave: remove(key).

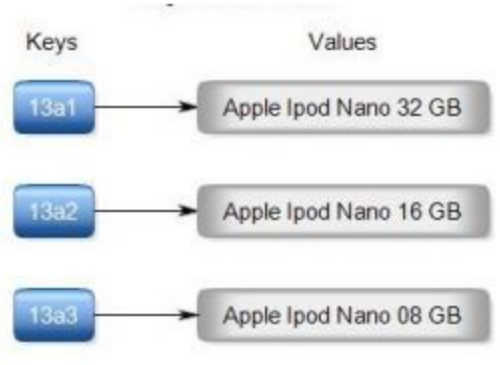

Figura 2. Almacenamiento Clave-valor. Tomado de [8]

Orientado a documentos

Son la extensión a las de clave-valor, solo que tienen un formato definido. Son altamente escalables, tienen conceptos de bases de datos relacionales muy similares como el de bases de datos, tablas que se llaman colecciones y el de registro o fila es el documento en una base de datos de este tipo. Todo documento tiene una clave única, muy similar a como se piensa una base de datos relacional.

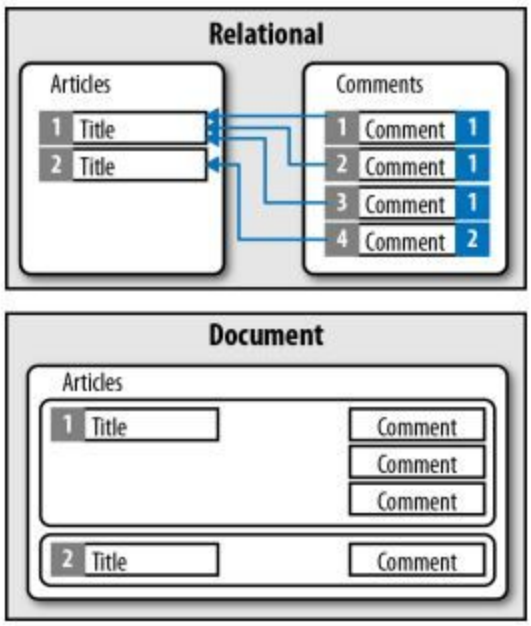

Figura 3. Base de datos documental vs. relacional. Tomado de [8]

Orientado en columnas

Su característica principal es, como lo dice su nombre, guardar la información en columnas en lugar de en filas como el modelo relacional; a cada una de las columnas se le asigna una clave única y evita almacenar valores nulos, ganando velocidad en lectura, pero no es eficiente en realizar escrituras o actualizaciones.

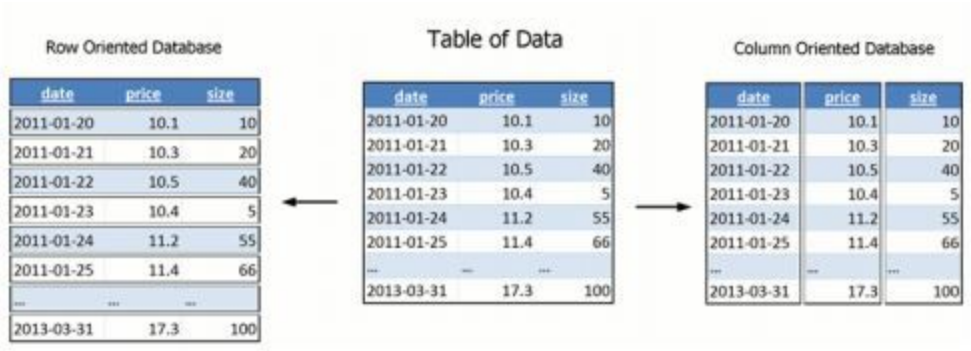

Figura 4. Base de datos orientada a columna vs. orientada a fila. Tomado de [8]

Orientado a grafos

Buscan representar la información como grafos, con sus componentes nodos determinando las relaciones a través de las aristas. Se utilizan para información que es usada a menudo, ofreciendo hash distribuido y estructuras de datos sencillos como arrays asociativos o almacenes como los anteriormente mencionados de clave-valor.

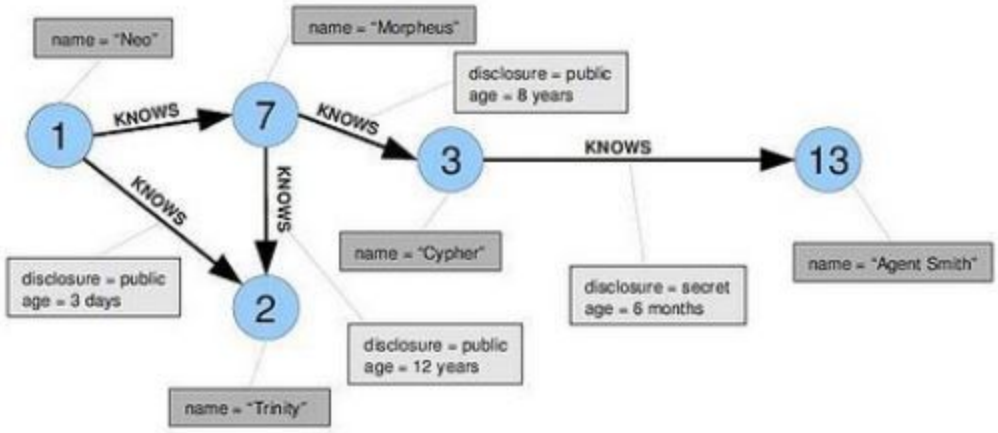

Figura 5. Base de datos orientada a grafos. Tomado de [8]

2.4. Teorema de CAP

Figura 5. Base de datos orientada a grafos. Tomado de [6]

Durante el simposio de “Principios de computación distribuida” de ACM en el año 2000, Eric Brewer, un profesor de la universidad Berkeley de California y cofundador de Inktomi, a través de una presentación titulada “Hacia sistemas distribuidos robustos”, estableció la conjetura que los servicios web no pueden asegurar en forma conjunta las siguientes propiedades: Consistencia (C), Disponibilidad (A) y Tolerancia a particiones (P), esto es lo que se conoce como el teorema de CAP. Posteriormente en el año 2002, Seth Gilbert y Nancy Lynch de MIT publicaron una demostración formal de la conjetura de Brewer, convirtiéndola en un teorema.
El teorema de CAP establece que en un sistema distribuido con datos compartidos, se debe optar por favorecer dos de las tres características: Consistencia, Disponibilidad y Tolerancia a particiones. [8]

- Consistencia: Los usuarios pueden acceder simultáneamente a un mismo registro.

- Disponibilidad: Cada solicitud debe tener una respuesta.
- Tolerancia de reparto: El sistema sigue funcionando a pesar de la pérdida de información.

2.5. Ejemplos base de datos NoSql

Base de Datos	Características	Aplicabilidad
MongoDB	● Escrito en C++Algunas características SQL (Query, index) ● Protocolo binario ● Replicación maestro-esclavo ● Sharding ● Permite ejecutar Javascript	● Cuando necesitas CouchDB con muchos cambios ● Para no utilizar MySQL
CouchDB	● Escrito en ErlangProtocolo HTTP/REST ● Replicación bidireccional con detección de conflictos ● MVCC ● Incluye librería JQuery	● Sistemas CRM ● Sistemas con replicación
Redis	● Escrito en C++ ● Protocolo estilo Telnet ● Bases de datos en memoria con backup en disco	● Comunicación en tiempo real ● Analíticas
Cassandra	● Escrito en JavaLo mejor de BigTable y Dinamo ● Protocolo binario ● Búsqueda por columnas o rango de claves ● Escrituras más rápidas que lecturas.	● Más escritura que lectura (logging). ● Análisis tiempo real
Neo4J	● Escrito en JavaBase de datos de grafos ● Protocolo HTTP/REST o Java ● Web de administración	● Datos ricos interconectados ● Redes sociales ● Topologías de red

Tabla 1. Bases de datos NoSql conocidas.. Tomado de [7]

3. Neo4j

Neo4j es una base de datos que almacena los datos en un grafo, que se define como una estructura genérica, capaz de representar elegantemente cualquier tipo de dato de manera altamente accesible. Neo4j se basa en el modelo de grafo de propiedades que se explicara a continuación.

3.1. Grafo de Propiedades Neo4j

Un grafo de propiedades se puede definir en términos teóricos, como un multigrafo dirigido, etiquetado en el vértice y etiquetado en la arista (relación), con aristas propias, donde estas tienen su propia identidad. En el grafo de propiedades, usamos el término nodo para denotar un vértice y una relación para denotar una arista. Ejemplo:

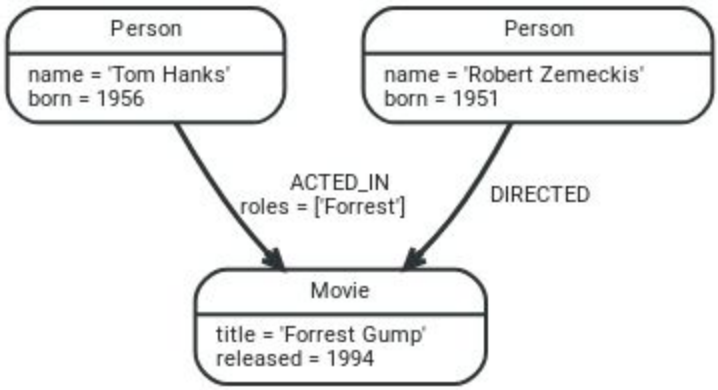

Figura 6. Ejemplo grafo de propiedades. Tomado de [9]]

Dado que el grafo de propiedades utiliza el concepto de Nodo y Arista (relación) se hace necesario definirlos y mostrar su uso en Neo4j.

Nodo

Se utilizan a menudo para representar entidades, siendo el grafo más simple posible aquel que esta conformado por un solo nodo. Un nodo puede tener diferentes propiedades como puede verse en la figura.

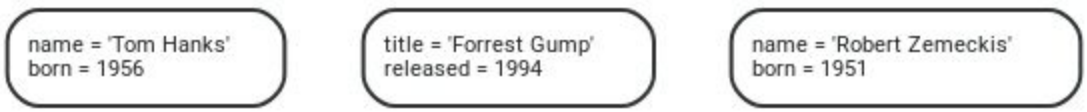

Figura 7. Ejemplo de Nodos con diferentes propiedades. Tomado de [9]

Arista (Relación)

Las relaciones entre nodos son la característica clave de las bases de datos basadas en grafos, ya que permiten encontrar datos relacionados. Una relación conecta dos nodos y se garantiza que tiene un nodo de origen y destino válido.

Las relaciones organizan los nodos en estructuras arbitrarias, lo que permite que un grafo se asemeje a una lista, un árbol, un mapa o una entidad compuesta, cualquiera de los cuales puede combinarse en estructuras aún más complejas y ricamente interconectadas. Puede referirse a la Figura 6, para tener una idea de como sería las relaciones entre nodos.

Propiedades

Describen los valores donde el nombre o la llave del nodo es un string. Neo4j sólo soporta 4 tipos de valores:

1. Valores Numéricos.
2. Valores Cadenas (String).
3. Valores Booleanos.

Type	Description	Value range
`boolean`	binary logic value	true/false
`integer`	64-bit integer	-9223372036854775808 to 9223372036854775807, inclusive
`float`	64-bit IEEE 754 floating-point number	-
`String`	sequence of Unicode characters	infinite

Figura 8. Rango valores tipos de datos Neo4j. Tomado [9]

Etiquetas

Una etiqueta es una construcción de un grafo nombrado que se usa para agrupar nodos en conjuntos; todos los nodos etiquetados con la misma etiqueta pertenecen al mismo conjunto. Muchas consultas de bases de datos pueden funcionar con estos conjuntos en lugar de con todo el grafo, lo que hace que las consultas sean más fáciles de escribir y más eficientes de ejecutar. Un nodo puede etiquetarse con cualquier cantidad de etiquetas, incluida ninguna, lo que hace que las etiquetas sean una adición opcional al grafo. Ejemplo de ellos es la Figura 6, donde puede apreciarse como cada uno de los nodos tiene etiquetas diferentes, tanto Person (Persona) y Movie (Película).

Las propiedades anteriores básicas, definen cómo se tratan los grafos en Neo4j, a continuación veremos algunas características relevantes.

3.2. Características de Neo4j

Neo4j, algunos autores la clasifican incluso fuera de las motores NoSQL y posee 3 características muy importantes:

- Rendimiento: Las bases de datos orientadas a grafos como Neo4j tienen mejor rendimiento que las relacionales (SQL) y las no relacionales (NoSQL). La clave es que, aunque las consultas de datos aumenten exponencialmente, el rendimiento de Neo4j no desciende, frente a lo que sí sucede con las BD relacionales como MySQL.

  Las BDOG responden a las consultas actualizando el nodo y la relaciones de esa búsqueda y no todo el grafo completo. Eso optimiza mucho el proceso.

- Agilidad: Neo4J tiene muchas ventajas, pero una es su agilidad en la gestión de datos. Si se quisiera llevar al límite sus capacidades, se tendría que superar un volumen total de 34.000 millones de nodos (datos), 34.000 millones de relaciones entre esos datos, 68.000 millones de propiedades y 32.000 tipos de relaciones.

- Flexibilidad y escalabilidad: Cuando los desarrolladores de una empresa trabajan con grandes datos, buscan flexibilidad y escalabilidad. Las bases de datos orientadas a grafos aportan mucho en este sentido porque cuando aumentan las necesidades, las posibilidades de añadir más nodos y relaciones a un grafo ya existente son enormes.

3.3. Campos de Aplicación Neo4j

Algunos campos de aplicación realmente relevantes para las bases de datos orientadas a grafos serán mencionados a continuación, pero cabe aclarar las posibilidades van mucho más allá a otros tipos de áreas.

1. Detección de Fraude: Neo4j ya trabaja con varias corporaciones en la detección del fraude en sectores como la banca, los seguros o el comercio electrónico. Esta base de datos puede descubrir patrones que con otro tipo de BD sería difícil de detectar.

   Las redes de fraude tienen mecanismos para delinquir que no son detectables con el análisis lineal de los datos. Pero con un análisis escalable de las múltiples relaciones entre los datos, esto es mucho más fácil.

   Un fraude habitual es la apertura de líneas de crédito con identidades falsas con la idea no pagar: en la actualidad, entre el 10% y el 20% de la deuda sin respaldo en los bancos líderes tanto en EEUU como en Europa se debe a este fraude.

2. Recomendaciones en tiempo real y redes sociales: Neo4j permite conectar de forma eficaz a las personas con otros productos y servicios, en función de la información personal, sus perfiles en redes sociales y su actividad online reciente. En este sentido, las bases de datos orientadas a grafos son interesantes porque son capaces de conectar personas e intereses.

   Con esa información, una empresa puede ajustar sus productos y servicios a su público objetivo y personalizar las recomendación en función de los perfiles. Eso es lo que permite que se aumente la precisión comercial y el compromiso del cliente.

3. Gestión de Centros de Datos: Las bases de datos gráficas son el antídoto perfecto ante el crecimiento desbordante de los datos. La gran cantidad de información, dispositivos y usuarios hacen que las tecnologías tradicionales no puedan gestionar tantos datos. La flexibilidad, rendimiento y escalabilidad de Neo4j permite gestionar, monitorizar y optimizar todo tipo de redes físicas y virtuales pese a la gran cantidad de datos.

4. Referencias

[1] Joyanes, L., (2013), *Big Data Análisis De Grandes Volúmenes De Datos En Organizaciones,* México DF, México, Alfaomega Grupo Editor.

[2] Fernández, E. P. (2017). *Big Data: Eje Estratégico En La Industria Audiovisual*. Recuperado de http://ebookcentral.proquest.com

[3] Hernández, E., Duque, N., Moreno, J.,(15 de marzo de 2017). Big Data: una exploración de investigaciones, tecnologías y casos de aplicación, *TecnoLógica*. Recuperado de http://www.scielo.org.co/pdf/teclo/v20n39/v20n39a02.pdf

[4] Camargo, J., C. Ortega, J., Joyanes, L. (1 de diciembre de 2014). Conociendo Big Data. *Facultad de Ingeniería*. Recuperado de http://www.scielo.org.co/pdf/rfing/v24n38/v24n38a06.pdf

[5] Romero, A. C., Sanabria, J. S. G., & Cuervo, M. C. (2012). *Utilidad y funcionamiento de las bases de datos NoSQL*. Facultad de Ingeniería, 21(33), 21-32.

[6] Pérez, M. A. C. (2017). NoSQL,¿ es necesario ahora?. Tecnología Investigación y Academia, 5(2), 174-179.

[7] Martín, A., Chávez, S. B., Rodríguez, N. R., Valenzuela, A., & Murazzo, M. A. (2013, June). Bases de datos NoSQL en cloud computing. In XV Workshop de Investigadores en Ciencias de la Computación.

[8] Antiñanco, M. J. (2014). Bases de Datos NoSQL: Escalabilidad y alta disponibilidad a través de patrones de diseño (Doctoral dissertation, Facultad de Informática).

[9] Neo4j, Inc. (2017). The Neo4j Developer Manual, (v3). Recuperado de http://neo4j.com/docs/developer-manual/current/