Introducción al business intelligence y al big data

Bases de datos NoSQL: Introducción

Hola, bienvenidos a una nueva presentación dentro del curso

de Introducción al business intelligence donde vamos a introducir conceptos

relativos a bases de datos NoSQL. Esta presentación ha sido hecha

conjuntamente por Jordi Conesa Caralt y yo misma; mi nombre es Elena Rodríguez

y ambos somos profesores de los Estudios de Informática, Multimedia

y Telecomunicación de la UOC. Hemos estructurado esta presentación

en tres partes: en primer lugar, veremos qué es NoSQL para pasar a describir

las principales características asociadas a estas bases de datos,

así como algunos de sus inconvenientes. Bajo la denominación de NoSQL

se engloba un conjunto de bases de datos para gestionar datos en entornos

de aplicación donde las bases de datos
relacionales no son la mejor solución.

Dichos entornos de aplicación verifican
al menos una y frecuentemente ambas

de las características que están puestas
en esta transparencia. En primer lugar,

se trata de entornos de aplicación
que requieren esquemas de datos más flexibles

con relación a lo que ofrecen
las bases de datos relacionales.

Como ya sabéis, las bases de datos relacionales
ofrecen una estructura de datos única

y uniforme para estructurar los datos
que son las tablas, tablas que se organizan

en un conjunto fijo de columnas
y que almacenan conjuntos de filas.

En segundo lugar, estos entornos de aplicación
son sistemas altamente distribuidos

que necesitan gestionar
grandes volúmenes de datos

y que necesitan siempre estar disponibles,
es decir, necesitan proporcionar

de forma continuada servicio a los usuarios.

El término NoSQL aparece
por primera vez en el año 1998

para denominar una base de datos relacional
construida por Carlo Strozzi

y que tenía como principal característica
el hecho de que no utilizaba

el lenguaje SQL para acceder

y manipular los datos. Esta implementación
no tiene nada que ver

con lo que hoy se conoce como base de
datos NoSQL. El uso del término

base de datos NoSQL,
para lo que nos interesa en esta presentación,

se empezó a popularizar alrededor del año 2009
para identificar desarrollos

de bases de datos que no seguían
los fundamentos de lo que serían

las bases de datos relacionales.

Cuando tomamos NoSQL como un acrónimo,
generalmente se acostumbra

a asimilar al hecho de no solo SQL.
Esta denominación, en esencia,

es poco afortunada, porque las bases de datos
NoSQL van mucho más allá del hecho

de que no provean un lenguaje estándar
tipo SQL para acceder y manipular los datos,

sino que cuestionan muchos
de los fundamentos sobre los que

se sustenta el desarrollo clásico
de bases de datos relacionales.

Algunas de las primeras implementaciones
más reconocibles de bases de datos NoSQL

son BigTable, de Google y Marklogic.

A modo de ejemplo, BigTable se utiliza
para almacenar los datos de Google Maps

y de Google Earth, entre otros.
Y también otras bases de datos NoSQL

especialmente emblemáticas son DynamoDB,
de Amazon, que cuestionó

los esquemas de consistencia
de las bases de datos relacionales,

o Cassandra, que actualmente es un proyecto
Apache y que fue desarrollado

por Facebook para mejorar la funcionalidad
de búsqueda en su bandeja de entrada.

Si pensamos en las características
de estas empresas pioneras en el desarrollo

de bases de datos NoSQL,
podemos llegar a la conclusión

de que se enfrentaban a una serie
de circunstancias que hasta ese momento

no se conocían o no se habían

enfrentado a la tecnología de base de datos. Por un lado, tenemos

que se trata de desarrollos que se orientan a un número elevado y no determinado

a priori de usuarios, usuarios que, además, pueden subir datos al sistema y, por otro lado,

se trata de un número importante e indeterminado de usuarios repartidos

alrededor del mundo y que están cursando miles, cientos de miles

de peticiones de servicio diariamente, peticiones que tienen que estar servidas,

o sea, el sistema siempre tiene que estar online. En esencia, lo que hay detrás

de estos desarrollos tiene que ver con que internet y la web 2.0 ha cambiado

las reglas y ha posibilitado la aparición de nuevos modelos de negocio

basados en los datos, donde los datos cobran cada vez más importancia.

La situación, con el paso del tiempo, ha ido a más: disponemos de sensores

y dispositivos inteligentes, como sería el caso de smartphones o tablets, y también existen

las plataformas de comercio electrónico.
La disponibilidad, insistimos,

la necesidad de estar siempre online,
es importantísima, es un requisito

fundamental de este tipo de sistemas.

Tenemos que pensar que en internet,
a diferencia del mundo real,

nuestra competencia está únicamente
a un clic de distancia.

Respecto a las características de las bases
de datos NoSQL, aquí se resumen

las principales: en primer lugar, no hay
un modelo de datos único para estructurar

los datos,

a diferencia de las bases de datos relacionales.
Bajo el paraguas NoSQL

se ofrecen dos grandes familias
de modelos de datos,

las conocidas como modelos de agregación
que, a su vez, incluyen

las clave-valor, las orientadas a columnas
y las orientadas a documentos, y las bases

de datos NoSQL orientadas a grafos.
Adicionalmente proporcionan un esquema

de datos flexible.

Esto lo que significa
es que la definición del esquema de los datos,

si es que existe, se puede definir a la vez
que se insertan los datos.

Esto es una diferencia fundamental
con respecto a las bases de datos relacionales,

en las que, en primer lugar,

declaramos el esquema de la base
de datos, en esencia, las tablas que tendrá

nuestra base de datos, y luego
se insertan los datos en estas tablas.

Se trata de modelos que permiten tratar
tanto con datos no estructurados

como semiestructurados e incluso
datos estructurados y, además, esta flexibilidad

incluye el hecho
de que podemos definir con estructuras

diferentes instancias de datos
que pertenecen a una misma entidad del mundo real.

Otra de las características
de las bases de datos NoSQL

es que no proporcionan un lenguaje estándar
para acceder y manipular los datos,

como sería el caso de lenguaje SQL

en las bases de datos relacionales.

De hecho, en ocasiones la única manera
de acceder a la base de datos

es a través de API.

En general, se trata de bases de datos
distribuidas, y otra de las

características es que acostumbran
a no garantizar las propiedades ACID

a la hora de gestionar las transacciones.
De hecho, frecuentemente proporcionan un modelo

transaccional alternativo que se conoce
bajo el nombre de modelo BASE.

Finalmente, frecuentemente se trata
de desarrollos de bases de datos que son

de código abierto.

Si no entendéis en profundidad
estas características, no os preocupéis,

porque en una presentación separada entraremos
en más detalle a explicarlas. Para finalizar,

queremos comentar algunos de los inconvenientes
de las bases de datos NoSQL.

En primer lugar, la falta de estándares como,
por ejemplo, el hecho de que no exista

un lenguaje de consulta unificado
como es el caso de las bases de datos

relacionales, o el hecho de que subyazcan diferentes modelos de datos.

Esta falta de estándares complica la portabilidad de nuestras aplicaciones

y no facilita precisamente que podamos cambiar de base de datos, de fabricante

en definitiva. El hecho de disponer de un esquema de datos flexible,

si bien permite

tratar con

diversidad de tipologías de datos y mezclar,

estar trabajando simultáneamente con datos heterogéneos,

también es importante destacar que esto puede complicar el esquema de la base

de datos.

Gestionar dicho esquema puede ser complejo y en muchos casos ese esquema

está disperso y escondido en los programas de aplicación que están

accediendo a la base de datos.

En otras palabras, el gestor de la base

de datos en ocasiones no es consciente

del esquema de la base de datos.

Esto compromete la independencia lógica de los datos que, como sabéis, es uno

de los pilares fundamentales sobre el que se sustenta el desarrollo de bases

de datos relacionales.

Todo esto puede causar dificultades en la administración

de la base de datos,

de hecho, las tareas clásicamente atribuidas al administrador de la base

de datos se complican.

Asimismo, existen funcionalidades, que se acepta de forma consensuada

que un sistema gestor de base de datos tiene que proporcionar,

que no están implementadas en las bases de datos NoSQL.

A modo de ejemplo, hemos dicho que no existe un lenguaje estándar de acceso,

a veces ni siquiera existe tal lenguaje, por lo que la única manera de acceder es a través de API.

Otra de las funcionalidades que frecuentemente no está implementada en este tipo de base

de datos tiene que ver con la seguridad y el control de acceso de los usuarios

a los datos. Esta responsabilidad recae en los programas de aplicación y no en el gestor

de la base de datos. Como tecnología nueva, o relativamente nueva,

existe una falta de madurez y de soporte por parte de los fabricantes de este tipo

de soluciones,

al menos si lo comparamos con los proveedores de soluciones de bases

de datos relacionales.

Asimismo y también como consecuencia de lo anterior, hay una ausencia

o una cierta falta de especialistas en esta tecnología; es más,

en la instalación, el mantenimiento y el desarrollo de una base de datos

NoSQL puede ser complejo y, por lo tanto, hay que estar atentos a que la curva

de aprendizaje puede ser lenta.

Finalmente, es complicado saber quién es el mejor, de hecho, todos afirman

que son el mejor; esto en épocas
recientes se está resolviendo parcialmente debido

a la aparición de benchmarks que permiten
comparar y validar las virtudes

y los defectos de las diferentes implementaciones.
Bien, hasta aquí esta presentación

de introducción a las bases de datos NoSQL.

Aquí os dejamos un conjunto de referencias
por si os interesa profundizar

en los temas que hemos tratado.
Simplemente esperamos que hayáis

disfrutado de este vídeo
y que tengáis un buen día.