

Entendiendo el movimiento NoSQL (Not Only SQL)

Antonio Silva Sprock
antonio.m.silva@ucv.ve

Universidad Central de Venezuela, Facultad de Ciencias, Escuela de Computación,
Av. Los Ilustres, Los Chaguaramos, Caracas, 1043, Venezuela.

En 1970, en los laboratorios IBM en San José (California), el Sr. Edgar Codd postuló la base de datos relacional (BDR), la cual permite modelar problemas reales y administrar datos dinámicamente, cumpliendo con el modelo relacional. Las BDR permiten establecer interconexiones (relaciones) entre los datos (que están guardados en tablas), y trabajar con ellos conjuntamente (mediante el lenguaje de consulta SQL).

Las transacciones de una BDR, deben cumplir con cuatro propiedades básicas, definidas como: atomicidad, consistencia, aislamiento y durabilidad (ACID), el cumplimiento de esas propiedades es una garantía de la correcta operación del sistema manejador de BDR, sin embargo, lograrlo es costoso, más aún cuando el sistema intenta escalar a la par del crecimiento de los volúmenes de datos.

En determinadas aplicaciones modernas, que requieren el uso de datos de forma intensiva, que incluya el indexado de un gran número de documentos, millones de registros, la presentación de páginas de gran demanda o indexación y presentación de sitios de *streaming*, las BDR han mostrado poca eficiencia. Los Sistemas Manejadores de BDR se han perfeccionado para pequeñas cantidades de datos, pero alta frecuencia de lecturas y escrituras o para un gran conjunto de transacciones que tiene pocos accesos de escritura.

El movimiento NoSQL (Not only SQL) se presenta como la forma de destronar la tiranía de las BDR, costosas y lentas, en favor de una alternativa mucho más eficiente y barata para manipular datos. Surgió de desarrolladores Web y Java, muchos de los cuales, ajustados en presupuesto, completaban aplicaciones sin utilizar manejadores de BDR propietarios y costosos.

Para esto construyeron sus propias soluciones para almacenar datos, y luego las publicaron como código abierto. Grandes industrias de internet, como Google, Facebook, Twitter y Amazon han apostado a NoSQL, y defienden sus trabajos afirmando que: no están utilizando una base de datos, sino “sistemas de almacenamiento distribuido para gestionar datos estructurados”, se pueden ejecutar en clusters de servidores de computadoras baratas, superan los cuellos de botella de

rendimiento al manipular los grandes volúmenes de datos, y evitan contar con mucho más de lo que puede cubrir sus necesidades.

Por otro lado, NoSQL puede servir gran cantidad de carga de lecturas y escrituras. Implementaciones de NoSQL usadas en el mundo real incluyen los 6 TB de la base de datos del "ENSEMBLE" de la Comisión Europea usada en los modelos de comparación y calidad del aire, los nuevos 500 TB diarios de Facebook, entre sus 300 millones de nuevas fotos y los 2 billones de *Likes* diarios, los 90 PB de eBay, los 51 TB de Amazon y los 2.5 PB de Walmart, entre otras.

Sin embargo, aún muchas organizaciones no están muy convencidas, ya que no tienen el mismo soporte que los proveedores tradicionales de bases de datos relacionales, y a las organizaciones les resulta alarmante trabajar sin "culpables" cuando las cosas resultan mal y sobre, al tratarse de avances iniciales muy puntuales y ad-hod de empresas actualmente exitosas.

Tipos de Sistemas de Bases de Datos NoSQL

Existen múltiples tipos de Sistemas de BD NoSQL, y múltiples Sistemas disponibles (Figura 1).



Figura 1. Tipos de Sistemas de Bases de Datos NoSQL

Base de Datos Documental: están diseñadas alrededor de una noción abstracta de "Documento". Entre las codificaciones usadas en la actualidad se encuentran XML, YAML, JSON y BSON, formatos binarios como PDF y documentos Microsoft Office. Los documentos, internamente son similares, de algún modo, a registros, tuplas o filas en una BDR pero menos rígidos. No requiere ajustarse a un esquema estándar ni tener todas las mismas secciones, atributos, claves o cosas por el estilo.

Base de Datos orientada a Grafos: representan la información como nodos de un grafo y sus relaciones con las aristas del mismo. Se puede recorrer la BD utilizando la teoría de grafos ya que esta puede describir atributos de los nodos (entidades) y las aristas (relaciones).

Base de Datos Orientada a Objetos: incorpora todas las ventajas de la Orientación a Objetos, pudiéndose conservar, gestionar y compartir, directamente sin tener que hacer la traducción a tablas o registros, manteniendo su integridad y sus relaciones.

Base de Datos Clave/Valor: posee contenedores o *cabinets*, donde se pueden tener tantas parejas de clave-valor como queramos, hay sistemas que permiten tener claves duplicadas y hay que no, o que se puede indicar que no queremos que se dupliquen. En cada contenedor podemos tener datos de la misma naturaleza o totalmente diferentes.

Base de Datos Columnar: cada tabla contiene filas y columnas, de tal manera que los elementos de una familia de columnas son almacenados en un solo conjunto. Eso es distinto a las bases de datos relacionales orientadas a filas, donde todas las columnas de una fila dada son almacenadas en conjunto.

Base de Datos Multivalor: permiten el uso de atributos que tienen una lista de valores, siendo más flexible su actualización y modificaciones en el tiempo, como por ejemplo la adición de atributos, que no requiere reconstrucción de la BD. Son un tipo especial de BD multidimensionales.

El movimiento NoSQL no representa la panacea, representa una la especialización para poder adaptar soluciones de software a las necesidades y escenarios de cada organización y no en general, sino a problemas concretos. NoSQL no es una amenaza para las BDR, es un complemento a ellas.