

Incorporando NOSQL en el currículo de Bases de Datos

Gerardo Rossel, Andrea Manna, Fernando Bugni

Departamento de Computación. FCEyN. Universidad de Buenos Aires

{grossel, amanna, fbugni}@dc.uba.ar

Resumen

En este trabajo se presentan aspectos curriculares y docentes relacionados con la incorporación de la enseñanza de las bases de datos denominadas NoSQL dentro del programa de la materia Base de Datos en la carrera de Ciencias de la Computación de la Facultad de Ciencias Exactas y Naturales de la Universidad de Buenos Aires.

Si bien algunos de los temas vinculados a los modelos de bases de datos diferentes al modelo relacional se tratan en la materia desde hace tiempo, los mismos no tenían el peso específico necesario y sólo eran vistos en las clases teóricas.

En este trabajo describimos un cambio cualitativo en la enseñanza de esos temas y su contextualización, tomando en cuenta la ubicuidad de las bases de datos, en el marco de la materia.

Palabras clave: NoSQL, Bases de Datos, Enseñanza de NoSQL

Introducción

En los últimos años ha crecido la utilización de bases de datos que podemos ubicar dentro del paraguas de NoSQL, es decir bases de datos que escapan al modelo relacional. Preferimos utilizar la denominación de bases de datos NoSQL en lugar de NoSQL *datastores*. Este crecimiento que, por ejemplo, ubica a MongoDB, Cassandra, Redis, ranqueadas entre las diez primeras bases de datos por popularidad[1] nos muestra la necesidad de incorporar más fuertemente el tema en el programa de la materia Base de Datos.

Las bases de datos relacionales siguen siendo ampliamente utilizadas pero nuevos desafíos como el crecimiento de internet, junto a la necesidad y disponibilidad de sistemas de alta

disponibilidad, distribuidos y escalables que marcan la emergencia del procesamiento de grandes volúmenes de datos conocido como *Big Data*, impulsan la necesidad de conocer otras formas de almacenamiento de datos [7].

En este artículo describimos cómo realizamos la incorporación de la enseñanza de bases de datos NoSQL, manteniendo un balance con la enseñanza tradicional, basada fundamentalmente en las bases de datos relacionales, en el programa de la materia Base de Datos que se dicta en la carrera de Ciencias de la Computación de la Facultad de Ciencias Exactas y Naturales de la Universidad de Buenos Aires.

El resto de este artículo tiene la siguiente estructura: en primer lugar describimos la distribución de contenido en la materia previo a la incorporación del nuevo tema, luego a partir de explicar la motivación para incorporar la enseñanza de NoSQL detallaremos algunos conceptos y definiciones acerca de bases de datos NoSQL. Con esta base describiremos las modificaciones al programa de la materia y para finalizar expondremos algunas conclusiones.

Contenido Tradicional

La materia, hasta el primer cuatrimestre del año 2015, estaba basada principalmente en la enseñanza de bases de datos relacionales. La aprobación de la materia se materializaba mediante la aprobación de dos parciales, dos trabajos prácticos y el examen final. Cómo correlatividad se exige haber aprobado los prácticos de Ingeniería de Software I (que brinda conocimientos sobre la construcción de software a gran escala y la modelización) y Sistemas Operativos (que ofrece una base en temas de acceso concurrente, manejo de disco, procesos, etc.). Cómo ocurre en muchas materias de la carrera, Bases de Datos tiene

clases teóricas, a cargo de los profesores, y clases prácticas que son conducidas por los jefes de trabajos prácticos. Ambos tipos de clases se coordinan, para que los contenidos teóricos dados sean el sustento de las clases prácticas.

El contenido de la materia se basa en dos enfoques. Por un lado, desde el punto de vista de la construcción de software, es decir la utilización de bases de datos, su diseño y los mecanismos utilizados para consultar las mismas y por otro, el punto de vista de la implementación, es decir del motor de la base datos.

El primer parcial estaba enfocado al primer aspecto y en él se evaluaba:

- Modelización conceptual de datos y modelo relacional
- Diseño de bases relacionales y normalización
- Lenguajes de consulta: SQL, álgebra relacional y cálculo relacional de tuplas.

Para el segundo parcial se abordaban los siguientes temas:

- Optimización de consultas
- Administración de transacciones
- Sistemas de recuperación

Cada tema tiene una guía de ejercicios que permite al alumno practicar y afianzar conocimientos. Para los lenguajes de consulta se realizan clases de laboratorio donde, en una modalidad de aula invertida, los alumnos proceden a realizar diversas consultas de un variado grado de complejidad. Algo similar ocurre con la optimización de consultas donde los alumnos en el laboratorio analizan diversas estrategias.

Junto a estos temas, en las clases teóricas, se incorporan otros como ser: mapeo relacional-objetos, XML, *datawarehousing* y minería de datos.

Bases de Datos NoSQL

El enorme desarrollo de “Big Data” en los últimos años ha producido un impacto importante en la tecnología de almacenamiento de datos. Esto es así debido a los particulares requerimientos de “Big Data”,

entendido como el análisis, procesamiento y almacenamiento de grandes cantidades de datos. Entre los desafíos que plantea se encuentra la necesidad de escalar en forma horizontal, el trabajo con fuentes de datos diversas, la falta de esquema o estructura de los datos con los que trabaja. Estas demandas, junto a la necesidad de alcance global y disponibilidad permanente, tuvieron como respuesta el surgimiento de una familia de bases de datos conocidas como NoSQL.

Definir qué significa NoSQL es complejo, no hay una definición precisa sino más bien 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. Tal es así, que Fowler y Sadalage afirman que: “NoSQL es un neologismo accidental. No hay una definición prescriptiva, todo lo que se puede hacer es una observación de las características comunes” [5].

El origen del nombre puede rastrearse a la base de datos de código abierto de Carlo Strozzi en 1998, que es en sí una base de datos relacional pero que expofeso no utiliza SQL y no tiene relación con las nuevas bases de datos NoSQL. El uso actual del término para referirse a un conjunto de tecnologías de almacenamiento de datos que se separaban del modelo relacional fue acuñado posteriormente por Jon Oskarsson para una reunión que organizó en San Francisco, USA, en el año 2009 [5]. Si bien hay muchísimas implementaciones, podemos ver como características comunes a las siguientes [7]:

- No relacionales
- Distribuidas, con soporte de replicación sencillo
- Código abierto (aunque hay algunas que no los son)
- Escalables horizontalmente
- Libre de esquema

Veremos entonces que esta característica de paraguas de tecnología que se utiliza para diferentes modelos de almacenamiento, plantea un importante desafío para la enseñanza.

Clasificación de bases de datos NoSQL

Una forma de establecer un marco adecuado para la comprensión de los conceptos subyacentes y de las tecnologías involucradas consiste en clasificar las bases de datos NoSQL.

La clasificación nos permite analizar diferencias y similitudes entre diferentes implementaciones. Hay varias dimensiones que pueden utilizarse para clasificar las bases de datos NoSQL. Podría, por ejemplo, utilizarse el modelo de consistencia usado tomando el teorema CAP como referencia [5]. En la materia preferimos utilizar la dimensión asociada a la forma de almacenar la información. En ese sentido, las dividimos en cuatro modelos de almacenamiento:

- Clave/Valor (*Key-Value*)
- Basadas en Documento.
- Familia de columnas y *wide column*
- Bases basadas en Grafos.

Esta clasificación permite estudiar más uniformemente como diseñar una base de datos NoSQL y ver la aplicabilidad de cada categoría según el contexto de utilización. A continuación, describimos brevemente cada una de las categorías:

Bases de Datos Clave-Valor. Este es el modelo más simple de almacenamiento de datos, pero también sumamente poderoso para el acceso concurrente. En una base de datos NoSQL basada en clave/valor cada dato consiste de un par compuesto por una única clave y un valor. Para almacenar información la aplicación debe generar una clave que es asociada al valor. Los valores son, entonces, recuperados mediante la clave. El único medio de acceder al valor es mediante la clave.

Entre las bases de datos clave-valor más usadas se encuentran Redis y Riak KV. Redis cuenta con un almacenamiento clave-valor en memoria (*in-memory*) con persistencia opcional.

Bases de Datos Basadas en Documento. Las bases NoSQL basadas en documento tienen a éste como el principal concepto. Documento se refiere a una estructura jerárquica de datos cuya forma más usual de representarla es mediante JSON (*Java Script Object Notation*) aunque también utilizan otros formatos como

XML y BSON. Estas bases soportan en general operaciones atómicas a nivel documento. Es decir, un documento puede guardarse en una única operación indivisible. Los documentos se agrupan en colecciones y pueden indexarse por diversas propiedades. La característica de ser de esquema libre hace que una colección de documentos no imponga que todos los documentos sigan el mismo esquema. El modelo de almacenamiento basado en documentos es muy flexible y ampliamente usado. MongoDB que es una de las bases de datos representantes de esta familia y está entre las cuatro bases de datos más populares [1].

La utilización de JSON para guardar documentos alinea esta categoría de bases de datos con el desarrollo de aplicaciones WEB y las nuevas tendencias al respecto como el uso de AJAX y las aplicaciones adaptables (*Responsive Web Applications*).

Con la generación de un identificador por documento, que algunas bases lo hacen en forma automática, las bases de datos basadas en documentos pueden ser vistas como un modelo clave-valor con más sofisticación.

Bases de datos de Familia de Columnas.

La principal inspiración para esta categoría es *BigTable* de Google [10] aunque la base de datos *Cassandra* es actualmente la más popular [1]. Esta categoría también usa tablas como los sistemas de base de datos relacionales, aunque en este caso los datos se guardan en columnas. Las filas tienen familias de columnas y se cuenta con un identificador de fila. Este modelo permite un crecimiento dinámico de las columnas. Puede verse a una base de datos de columnas con un mapeo multidimensional, donde cada valor es accedido por el identificador de columna, la columna y una marca de tiempo. Una característica distintiva es que se pueden almacenar datos utilizando no sólo los valores de una columna sino también el nombre de la misma. La utilización de los nombres de columna para almacenar datos puede verse con el siguiente ejemplo. Supongamos una base de datos que almacene datos de series de televisión y sobre la cual se desea consultar los datos de los actores de una serie dada. Una

forma de guardar esa información es utilizar el nombre de la serie como identificador de fila y el nombre del actor como columna variable. El diagrama de Chebotko[2] siguiente muestra el diseño para responder la consulta dada.

ActoresXSerie	
Titulo	K
Nombre	C ↑
Edad	
Personaje	

Entre las ventajas de este tipo de almacenamiento se encuentra la facilidad que tiene para escalar. De esta manera, es posible almacenar gran cantidad de información en las columnas y minimizar el tiempo de acceso.

Bases de Datos de Grafos. Esta clase de bases de datos es notoriamente diferente a las otras. En este caso la información se almacena como una red interconectada de nodos. La más conocida y utilizada base de datos de esta categoría es *Neo4j* [1][3]. Almacenar la información en un grafo implica almacenar datos tanto en los ejes como en los nodos. Esto permite que determinado tipo de información sea representado en forma más natural que en las otras categorías de bases NoSQL, como así también en bases relacionales. Además de representar más naturalmente ciertos dominios se puede utilizar sobre esta base muchos de los algoritmos de grafos. Este tipo de base de datos tiene un fuerte fundamento teórico en la teoría de grafos [9]. El hincapié principal está dado por las relaciones entre los datos más que por la estructura de los mismos. Hay, además, diferencias con las otras clases en lo que hace a consistencia y distribución. En general este tipo de base de datos no soporta bien la distribución entre diversos servidores. En un simple servidor se garantiza la consistencia ACID, por lo menos en *Neo4j* que es la más utilizada. Desde el punto de vista de la

ejecución en un *cluster Neo4j* soporta un modelo maestro-esclavo.[5]

Modificaciones Realizadas al programa de la materia

El primer inconveniente para incorporar a la currícula los fundamentos de bases de datos NoSQL, está dado por la falta de antecedentes ya que es un tema que se empezó a discutir recientemente [6][4][11]. Tomando las ideas presentadas en las conferencias TLDA[6][11] y manteniendo el espíritu de la materia de brindar conocimientos de base que el alumno pueda después desarrollar en diversas implementaciones tecnológicas, hemos organizado el enfoque del tema, la práctica de enseñanza y de evaluación.

Para mantener la estructura de tres ítems por parcial y a su vez poder incorporar los temas asociados a NoSQL hemos unificado en la práctica dos temas que se abordaban como ítems separados: administración de transacciones y sistemas de recuperación en un solo tema: concurrencia y recuperación. De esta manera, para la segunda parte de la materia, en la parte práctica, tenemos los siguientes temas:

- Optimización de consultas,
- Concurrencia y recuperación
- Bases de datos NoSQL

La organización del cronograma queda con dos semanas para bases de datos NoSQL. Con esa limitación temporal debíamos decidir qué temas incluir y sobre qué conceptos hacer foco. Por un lado, existen aspectos generales comunes, relacionados con los modelos consistencia y distribución que tienen características transversales a las categorías que hemos definido. Por otra parte, tenemos para cada categoría definida diferentes formas de encarar el diseño de la base de datos. Respecto a la consulta de grandes volúmenes de información, dado que las bases NoSQL operan con lo que denominan *vistas materializadas* en general mediante computaciones utilizando *map-reduce*[8], dedicamos atención al diseño de este tipo de consultas.

Organización del temario

Tomando en cuenta lo antedicho hemos organizado el temario con los siguientes ítems:

- Motivación y orígenes de NoSQL
- Consistencia y distribución.
- *Map-Reduce*
- Bases de datos clave-valor
- Bases de datos basadas en documento
- Bases de datos de grafos

Respecto a las diferentes categorías se ofrecen los aspectos comunes sin hacer foco en una implementación concreta, aunque se referencian en los temas, los diferentes motores existentes.

La guía de ejercicios sobre los temas vistos apunta a cómo resolver cuestiones de diseño en contextos específicos y de esta forma que el alumno aprenda a evaluar la utilidad de las diferentes implementaciones de acuerdo al problema y contexto. También se trata de que el alumno pueda diseñar la forma de almacenar la información de tal manera que optimice el uso de la herramienta.

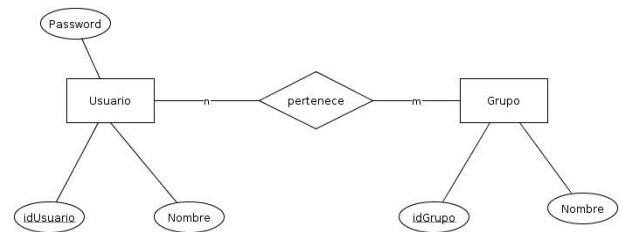
Un ejercicio clásico de la guía es por ejemplo el siguiente:

Una empresa de video juegos realiza un juego en línea y necesita guardar el estado de las partidas de los jugadores. Dicho estado debe almacenar: posición, nivel de vida, objetos encontrados y enemigos abatidos. El jugador deberá poder jugar desde cualquier estado guardado eligiendo la fecha y hora en el que lo guardó. Se pide realizar un modelo de almacenamiento para una que soporte lo descrito.

En este caso se establece previamente el tipo de base de datos a usar y el diseño presentado por el alumno debe ser adecuado a dicha categoría. En otros ejercicios se plantean problemas que deben solucionarse usando diversos tipos de bases de datos para que el alumno pueda evaluar ventajas y desventajas.

En bases de datos basadas en documento o en familia de columnas utilizamos el Diagrama de Entidad Relación como herramienta de modelización conceptual que permite tener una visión detallada de los datos que deben manejarse, de esta forma en varios ejercicios se pide trasladar el DER a un esquema apropiado. Un ejemplo clásico sería que dado

el diagrama de entidad relación de la siguiente figura se definan tres formas diferentes de implementarlo en una base de datos basada en documentos.



Este tipo de ejercicio permite el debate sobre las diferentes formas de organizar los documentos y las ventajas/desventajas relativas de cada una de ellas.

Para la categoría de bases de datos de familia de columnas usamos los diagramas de *Chebotko*[2] que a pesar de estar orientados a una implementación concreta, en este caso *Cassandra*, son muy útiles para visualizar el diseño y además son parte de una metodología que parte de un modelo conceptual.

El trabajo en grupos es una estrategia de enseñanza que ha ganado gran espacio pedagógico y didáctico desde hace ya varias décadas. Permiten desarrollar las habilidades de interacción social entre las personas y su capacidad de hacer elecciones y tomar decisiones [13]. Las experiencias de aprendizaje colaborativo y trabajo en grupo son cualitativamente diferentes de las experiencias de trabajo individual: promueven el aprendizaje activo centrado en el alumno, se crean condiciones que favorecen el aprendizaje profundo, brinda al estudiante múltiples oportunidades para confrontar sus ideas con las de los otros miembros del grupo. Se analizan diferentes razonamientos y el alumno se coloca en puntos de observación desde donde puede responder o investigar diferentes perspectivas, así como evaluar, aceptar e intercambiar argumentos y refutar otros, justificar sus posiciones y propuestas, incluso en las reuniones con el docente tutor. Estas son operaciones cognitivas que requieren la interacción con otros y que difícil y excepcionalmente pudieran desarrollarse en actividades realizadas sólo individualmente.

Para ayudar aún más en la elaboración de un trabajo práctico grupal, se realiza una clase de laboratorio sobre un motor específico, que es utilizado en la resolución del trabajo. En esta primera experiencia utilizamos *MongoDB* por su popularidad y por ser el representante más conocido de las bases orientadas a documentos. La clase de laboratorio tenía como objetivo familiarizar al alumno con el entorno de cliente de *MongoDB* para la realización del trabajo práctico. Los ejercicios de la clase se centraban principalmente en desnormalizar y aplicar consultas sobre diversos modelos.

El trabajo práctico se enfocó en las principales ideas de las base de datos NoSql. Los temas abarcados en éste son desnormalización, *map-reduce* y *sharding*. Se dividió en tres partes. La primera parte consistía en dado un modelo conceptual representado en un DER, aplicar desnormalización de modo tal se puedan realizar diferentes consultas en forma eficiente. En este ejercicio era importante entender las formas de desnormalizar y cómo se debe llevar un esquema de DER hacia un modelo de documentos desnormalizado que pueda servir para las consultas que se pedían realizar. La segunda parte consistía, a partir de varios *datasets*, realizar consultas utilizando el esquema de consultas de *map-reduce*. Esta parte tiene el objetivo de que el alumno se familiarice con grandes conjuntos de datos y por otro lado pueda realizar consultas sobre dicho esquema. La tercera parte del trabajo práctico consistía en realizar un experimento utilizando *sharding*. Dado un esquema de base de datos basado en documentos, los alumnos debían elegir un atributo para aplicar *sharding* y luego debían insertar una gran cantidad de registros (aproximadamente del orden de los 500.000). Sobre eso se pedía comprobar la manera en que iban cambiando los distintos tamaños de las particiones. El énfasis del ejercicio se ponía en probar distintos atributos para la división de las particiones. A partir de ese experimento los alumnos debían exponer cuáles características debería cumplir un atributo para trabajar con *sharding*. Para finalizar, se le pedía elegir una base de datos de otra categoría (clave-valor, columnas o

grafos) y describir cuáles son los cambios que deberían realizar si se utilizara esa base de datos elegida. La idea no es dar muchos detalles técnicos sino investigar a grandes rasgos cómo se modificarían los ejercicios del trabajo práctico cambiando el tipo de base de datos utilizado y ver si la característica de dicha categoría se adapta más o menos al problema con respecto a la utilizada inicialmente.

Evaluación

Para evaluar este tema en los parciales se pensó en cuáles son las actividades predominantes cuando se decide utilizar bases de datos NoSql. En las prácticas de enseñanza, la actitud evaluadora invierte el interés de conocer por el interés por aprobar en tanto se estudia para aprobar y no para aprender [12]. Por eso, es importante que los docentes recuperemos el lugar de la evaluación como el lugar que genera información respecto de la calidad de su propuesta de enseñanza. En este sentido, se elige evaluar con dos ejercicios. El primero corresponde a realizar un diseño de una base de datos NoSql para un problema dado. Un ejercicio planteado en esta instancia es, por ejemplo:

El servicio meteorológico nacional debe guardar, para análisis posteriores, la información de las mediciones de temperatura y presión atmosférica de diversas estaciones meteorológicas.

Cada estación meteorológica se identifica con un número y dos letras que representan la ubicación geográfica. La información obtenida se guarda cada 10 minutos. Es necesario poder graficar la evolución de la temperatura y/o de la presión en una estación dada en un día dado. También puede ser necesario comparar esa evolución en dos fechas diferentes.

Se pide que realice un modelo para una base de datos de familia de columnas tal que sea adecuado para lo enunciado.

Entendemos que el modelado de problemas similares a los encontrados en la práctica real constituye un proceso de toma de conciencia de los aprendizajes adquiridos y eso es lo que se quiso lograr en esta instancia de evaluación.

El segundo ejercicio corresponde a realizar una consulta utilizando el esquema *map-reduce*. Un ejemplo de ejercicio tomado es: *Una red social posee una base de datos basada en Documentos con la siguiente colección.*

```
Persona: {  
  id: id de persona  
  
  nombre: nombre de la persona,  
  
  amigos: [{  
    id: id de persona,  
  
    afinidad: valor entre 0 y 1  
  }]  
}
```

Dada una persona A se define como amigo recomendado a otra persona C si la persona A posee como mejor amigo a B y esa persona B posee como mejor amigo a otro C. Utilizando un sólo map-reduce devolver todos los pares id de persona que sean amigos recomendados.

Pensamos que una consulta de este esquema es novedosa y rompe lo habitual en las consultas de base de datos permitiendo apreciar la comprensión del alumno en las consultas sobre grandes volúmenes de información. Pensar la evaluación desde la perspectiva de los principios que fomentan la comprensión, la investigación cognitiva, la reflexión y el pensamiento crítico significa las posibles maneras de comprender de los estudiantes, tanto por parte de los docentes como de los propios alumnos [12].

La evaluación desde el punto de vista práctico de estos temas, consiste entonces de una instancia individual a través de los parciales y un aspecto grupal con el desarrollo de un trabajo práctico guiado por un tutor asignado a quien deberán responder. La guía del tutor es de primordial importancia y debería ser su función, entre otras la de coordinar y dirigir el grupo, estimular la participación de todos los miembros, evitar la competencia en el interior del grupo, generar debates y dudas con respecto a alguna decisión o un argumento presentado, redirigir y focalizar el trabajo de los alumnos si pierden el rumbo. En lo que

respecta a la evaluación, se quiere hacer foco no solamente en evaluar el proceso y el desarrollo del trabajo por parte del grupo, sino también el producto del trabajo final.

Conclusiones

Hemos presentado la modificación realizada a la materia Bases de Datos en función de incorporar en el temario los fundamentos de las bases de datos NoSQL. Esta modificación nos obligó a redistribuir los temas, implementar una nueva guía de trabajos prácticos y un laboratorio específico del tema. Naturalmente al ser un tema nuevo y con pocos antecedentes en las currículas tradicionales de bases de datos, hay muchos detalles que se deben mejorar, pero el retorno por parte de los estudiantes ha sido muy bueno ya que expresan la satisfacción por tener un temario actualizado y que brinde formación de base.

A pesar de que la diversidad de modelos y mecanismos de consulta dificulta lograr un conocimiento profundo sin hacer foco en una implementación particular [11] hemos conseguido con el enfoque dado, brindar los fundamentos básicos haciendo énfasis en conseguir mediante la ejercitación la habilidad en los alumnos de evaluar el contexto para poder seleccionar el modelo de almacenamiento adecuado.

Cómo trabajo futuro planificamos, entre otras cosas, depurar la bibliografía y resolver la puesta a punto de un apunte general que acompañe esta forma de encarar la enseñanza. Esto último es una necesidad ya que la mayor parte de la bibliografía actual está orientada a un producto particular o no se adapta al objetivo que nos planteamos en la materia.

Bibliografía

- [1] solid IT, «<http://db-engines.com/en/ranking>,» 23 12 2015. [En línea].
- [2] A. K. S. L. Artem Chebotko, «A Big Data Modeling Methodology for Apache Cassandra,» de *IEEE International*

- Congress on Big Data (BigData'15)*, New York, USA, 2015.
- [3] R. V. Bruggen, *Learning Neo4j*, Packt Publishing, 2014.
- [4] Y. Silva, L. Tsosie, S. Dietrich y J. Reed, «Integrating Big Data into the Computing Curricula,» de *SIGCSE '14 Proceedings of the 45th ACM technical symposium on Computer science education*, New York, 2014.
- [5] P. J. S. Martin Fowler, *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*, Addison-Wesley Professional, 2012.
- [6] C. Stanier, «Introducing NoSQL into the Database Curriculum,» de *TLAD 2012 10th International Workshop on the Teaching, Learning and Assessment of Databases*, 2012.
- [7] J. Han, E. Haihong, G. Le y J. Du, «Survey on NoSQL database,» de *6th International Conference on Pervasive Computing and Applications (ICPCA)*, 2011.
- [8] S. G. Jeffrey Dean, «MapReduce: simplified data processing on large clusters,» *Commun. ACM*, vol. 51, n° 1, pp. 107--113, 2008.
- [9] J. W. E. E. Ian Robinson, *Graph Databases*, O'Reilly MEdia, 2013.
- [10] J. D. S. G. W. C. H. D. A. W. M. B. T. C. A. F. R. E. G. Fay Chang, «Bigtable: A Distributed Storage System for Structured Data,» *ACM Trans. Comput. Syst.*, vol. 26, n° 2, pp. 4:1--4:26, 2008.
- [11] E. G. James H. Paterson, «Teaching NoSQL with RavenDB and Neo4j,» de *TLAD 2014 12th International Workshop on the Teaching, Learning and Assessment of Databases*, 2014.
- [12] A. Camillioni, «La evaluación de los aprendizajes en el debate didáctico contemporáneo,» de PAIDOS 1° Edición, 1998.
- [13] R. Anijovich, A. Camillioni, A. Cappelletti, G. Hoffmann, J. Katykowicz, R. López Mottier «La evaluación significativa,» de PAIDOS, 2010.