Técnicas de Recuperación de Información en Grandes Volúmenes de Datos Heterogéneos con Bases de Datos NOSQL

Damián P. Barry[1], Carlos E. Buckle[1], Renato Mazanti[1], Gustavo Samec[1], Cristian Pacheco[1], Rodrigo Jaramillo[1], Ignacio Real[1], Ignacio Aita[1], Juan Manuel Cortez[1], Fernando G. Tinetti[2]

[1]Depto. de Informática, Fac.de Ingeniería, Universidad Nacional de la Patagonia San Juan Bosco.
Puerto Madryn, Argentina
+54 280-4472885 – Int. 117.
damian_barry@unpata.edu.ar, cbuckle@unpata.edu.ar, renato@cenpat.edu.ar, gsamec@cenpat.edu.ar, cristian@cristian-pacheco.com.ar

[2]III-LIDI, Facultad de Informática - Universidad Nacional de La Plata
[2]Investigador Comisión de Investigaciones Científicas de la Prov. de Bs. As.
La Plata, Argentina

Resumen

El presente proyecto se enfoca en la evaluación de las técnicas existentes para recuperación eficiente de información sobre grandes volúmenes de datos heterogéneos.

Dichas técnicas permitirán establecer las capacidades necesarias con las que debería contar una base de datos de información masiva, tanto desde la perspectiva de almacenamiento y técnicas de indexación, como de distribución de las consultas, escalabilidad y rendimiento en ambientes heterogéneos.

Para ello se diseñarán arquitecturas tanto centralizadas como distribuidas, y se realizarán las correspondientes verificaciones, estableciendo los porcentajes de mejora de rendimiento para cada arquitectura.

Palabras Clave: Recuperación de Información, Bases de Datos NoSQL, Indexación, Particionado horizontal, Shards, Escalabilidad, Rendimiento.

Contexto

Esta presentación corresponde al proyecto de investigación “Técnicas de recuperación de información en grandes volúmenes de datos heterogéneos con bases de datos No-Sql” desarrollado por docentes y alumnos de la Facultad de Ingeniería de la Universidad Nacional de La Patagonia San Juan Bosco (UNPSJB) Sede Puerto Madryn. El proyecto es financiado por la Secretaría de Ciencia y Técnica de dicha Universidad y se vincula con el III-LIDI Facultad de Informática de la Universidad Nacional de La Plata a través del director quien se desempeña como investigador en dicho laboratorio.

Introducción

En la actualidad existe gran cantidad de sitios web que procesan grandes volúmenes de información, la cual es necesario manejar eficientemente. Principalmente por las siguientes razones:

- La popularidad de los sistemas de gestión de contenidos (CMS, Content Management Systems) como portales en general y como plataformas de colaboración en particular.
- La llamada Web 2.0 ha definido un conjunto de aplicaciones que facilitan la interacción con gran volumen de contenido multimedia.
- Crecimiento en la producción de información dentro de las organizaciones ya sea por producción de los sistemas o por la digitalización de información existente.

En suma, se ha pasando de hablar de gigabyte de información a hablar con total normalidad del orden de los petabytes [13] [14] [15].

Esta situación ha generado el desafío de mejorar las herramientas de búsqueda en lo que se denomina “information retrieval” utilizando para ello diversas técnicas. Asociado a este problema, se suma la necesidad de escalabilidad, disponibilidad y desempeño en el manejo de grandes volúmenes de información, situación que requiere de técnicas de sistemas distribuidos. Algunas de las técnicas incluyen: balanceo de carga, replicación y distribución horizontal (sharding) de la información [1] [5].

A modo de ejemplo, podemos mencionar soluciones similares como la adoptada por la Casa Blanca que ha utilizado la combinación de Drupal y Apache Solr en el portal de contenidos documentales [13] [14]. En general, cualquier solución a este problema debe incluir estrategias de búsqueda, ahorro de espacio, escalabilidad, disponibilidad y desempeño.

El indexado transforma los datos desde su forma original en una estructura que facilita la búsqueda y recuperación de los mismos en forma rápida y precisa [1] [2]. El proceso de indexado generalmente requiere

un análisis y procesamiento de los documentos a incluir en el índice: lematización, tokenización, análisis fonético, etc. Estos pasos introducen problemas y desafíos importantes al momento de procesar [7] [10] [11] [12].

La búsqueda secuencial de cualquier tipo de información presenta varios problemas, siendo el principal la falta de escalabilidad. Una solución a este inconveniente es el uso de estructuras de datos que permitan ser rápidamente consultadas.

Indagando las distintas alternativas para solucionar la distribución de los índices y de las búsquedas en un ambiente heterogéneo y escalable se definieron un conjunto de propiedades deseables que debiera cumplir una solución [1] [2] [3] [16]: Rendimiento, Tolerancia a Fallas y Ejecución en ambientes heterogéneos. Por otra parte, las soluciones NoSQL para administrar grandes volúmenes de información se basan normalmente en la conformación de un sistema de nodos heterogéneos. Existen diversas técnicas que permiten configurar ambientes heterogéneos y/o mixtos. Algunas de las técnicas utilizadas por estas bases de datos son:

a) Particionamiento horizontal mediante “shards”.

Se utiliza un proceso de indexado transformando los datos desde su forma original en una estructura que facilita la búsqueda y recuperación de los mismos en forma rápida y precisa, por ejemplo un índice invertido [5], un índice de citas, una matriz o un árbol. Estas técnicas facilitan la implementación y escalabilidad en ambientes heterogéneos mediante el uso del concepto de base de datos de partición horizonal o sharding [1] [2] [17] [18].

b) Shared Nothing

Según Michael Stonebraker: “Consiste en una arquitectura distribuida en el que cada nodo es independiente y autosuficiente, y tiene un único punto de contención en todo el sistema”. El concepto de Shared Nothing parte de la independencia de los nodos, mediante la distribución de la información y de las acciones sobre dichos nodos.

Se podría decir que un Shrad es un nodo Shared Nothing donde se administra un conjunto de documentos indexados según algún criterio y en donde se los puede someter a mecanismos de búsqueda de información y de jerarquización y ordenamiento de la información recuperada dependiendo de necesidades particulares de información. Por ejemplo se podrían realizar distribuciones: geográficas, temáticas, ontológicas, segmentación según preferencias, etc. o inclusive combinaciones de ellas.

En todos los casos se puede combinar con técnicas de bases de datos tradicionales, como la replicación y la paralelización mediante esquemas de Shared Disk (cluster tradicional, como por ejemplo la implementación de un blade con una Storage Area Network) [1] [2] [19].

c) Replicación con balanceo de carga

La arquitectura debe garantizar un conjunto de nodos con la información replicada y consistente en todos los nodos. En este caso el motor de búsqueda cuenta con un pooling de nodos de datos en los cuales buscar la información. La arquitectura misma no paraleliza las búsquedas, simplemente distribuye la carga entre los nodos. Como los nodos son independientes y auto-suficientes son capaces de responder consistentemente a la consulta realizada ya que la responsabilidad de recuperación está en el nodo de datos y la responsabilidad de distribución de carga en el balanceador. La desventaja de este método es que ni resuelve el problema espacial de la información ni paraleliza la búsqueda[3].

d) Scatter and Gather

El método realiza un broadcast de la búsqueda de información requerida en sus nodos conocidos, realizando una dispersión de la misma. Cada nodo (independiente de los demás) tiene la capacidad de elaborar una respuesta con la información que contiene dicho nodo. Todas las respuestas se concentran en el nodo que realizó la dispersión y éste es responsable de consolidar las mismas en una única (y consistente) respuesta a a la petición. Una ventaja adicional del método es que a su vez los nodos de datos pueden ser dispersores en nuevos nodos (conocidos por él). Conformando de esta forma una red de nodos independientes que contienen información.

Las ventajas en este caso son el particionamiento de la información y la paralelización de las búsquedas. La desventaja es una sobrecarga en la distribución de la información, especialmente si se desea realizar con alguna lógica de segmentación en particular: Geográfica, tipo de contenido, atributos ontológicos, etc. En este último caso se requiere conocimiento e información sobre los contenidos (datos) a ser almacenados, siendo en algunos casos relativamente compleja su resolución, especialmente ante la aplicación de reglas ontológicas sobre los contenidos [2] [3] [7].

Para este caso es interesante poder aplicar la técnica de Map / Reduce que es una buena técnica para procesar gran volumen de datos en paralelo. El modelo provee un mecanismo de particionamiento de información que permite distribuir “inteligentemente” de acuerdo a reglas pre-definidas los datos en distintos nodos auto-contenidos.

Map /reduce es una técnica que implica paralelizar los datos que es distinto a paralelizar las tareas. Obviamente la paralelización de los datos permite paralelizar el procesamiento en las búsquedas, pero la clave de la técnica se basa en la inteligencia de separación de los datos. A su vez una ventaja adicional radica en el ahorro de espacio en el resultado de las

claves compartidas al reducirlas dentro de un documento[8] [9] [17] [18].

Líneas de Investigación y Desarrollo

Las principales líneas de investigación se podrían resumir en la lista siguiente:

- Investigar técnicas de particionamiento, replicación y distribución de información tanto en las denominadas Bases de Datos relacionales (RDBMS) como en las denominadas NoSql.
- Investigar las Bases de Datos no estructuradas (NoSql) actuales y cóomo implementan la administración de sus recursos, especialmente en lo que respecta al particionamiento, almacenamiento, distribución y recuperación de información.
- Determinar la factibilidad y aplicabilidad de los métodos teóricos en los entornos prácticos estudiados, especialmente en lo referido a la distribución de información e information retrieval.
- Realizar una comparación entre las distintas Bases de datos NoSql específicamente en lo referido al particionamiento, distribución, escalabilidad, disponibilidad y performance.
- Proponer mejoras o nuevas técnicas y/o re-formulaciones a las técnicas existentes para el manejo de recursos, en lo que se refiere a las técnicas de distribución y recuperación.
- Implementar y validar las técnicas y métodos propuestos sobre plataformas de desarrollo concretas.

Resultados y Objetivos

Los resultados y objetivos de este proyecto de investigación se pueden enumerar como sigue:

- Seleccionar material bibliográfico y generar una base de conocimiento sobre las técnicas y métodos empleados en los esquemas de particionamiento, replicación, distribución e indexado de los Sistemas de Bases de Datos NoSql.
- Investigar y seleccionar una o varias de las Bases de Datos NoSql con código abierto. (posibles: Hadoop/Hbase, Cassandra, CouchDB, Lucene/Solr, etc.)
- Investigar y seleccionar uno o varios métodos y lenguajes de consultas sobre Bases de Datos NoSql. (posibles: MapReduce, HQL, SparQL, GQL, etc.)
- Investigar y seleccionar uno o varios métodos de particionamiento y replicación sobre Bases de Datos NoSql. (posibles: HDFS Replication, Master Master, Master Slave, etc.)
- Definir métricas que permitan obtener conclusiones relevantes respectos a las técnicas y métodos implementados.
- Definir y desarrollar uno o varios métodos de pruebas de stress sistematizadas para someter a comparación las distintas Bases de Datos NoSql Seleccionadas.
- Armar un banco de pruebas que permita comprobar las distintas implementaciones y métodos utilizados en las Bases de Datos NoSql seleccionadas.
- Diseñar distintas arquitecturas de particionamiento y escalamiento en función de los resultados obtenidos y compararlas con los mismos métodos definidos.

Formación de Recursos Humanos

En lo referido a Formación de Recursos Humanos este proyecto propone las siguientes metas:

- Consolidar mediante el proyecto, un grupo de investigación de la Universidad Nacional de la Patagonia San Juan Bosco sede Puerto Madryn, sobre la disciplina Bases de Datos NoSql. Este grupo se integra actualmente de 4 profesores, 1 JTP y 1 Auxiliar, además participan del mismo 4 alumnos del ciclo superior que realizarán sus trabajos de tesina de grado enmarcados en el proyecto. Los miembros, los cuales se encuentran abocados a la investigación a fin de crear nuevos métodos, desarrollos y trabajos de publicación científica para revistas, congresos de orden nacional e internacional.
- Fomentar, incentivar y difundir las tareas de investigación.
- Mejorar la formación de recursos humanos altamente calificados, con capacidades de investigación y desarrollo. Lograr la categorización de los docentes participantes y la jerarquización del departamento de informática y de la universidad en todos sus niveles.
- Contribuir a la creación en un futuro Centro o Instituto en investigación informática.
- Interactuar con otros grupos de investigación de las sedes de la universidad y de otras universidades, en tareas conjuntas de investigación y desarrollo, como también en la formación de recursos humanos.

- Incrementar el número de proyectos acreditados y de trabajos publicados por la universidad y la sede.

Referencias

[1] M.T. Özsu & P. Valduriez. “Principles of Distributed Database Systems, 2nd edition”. Prentice-Hall, 1999. Sitio web: http://softbase.uwaterloo.ca/~tozsu/ddbook/notes.html

[2] David Taniar & Clement H. C. Leung & Wenny Rahayu & Sushant Goel. “High Performance Parallel Database Processing and Grid Databases”. John Wiley & Sons, 2008.

[3] P. Valduriez. “Data Management and Parallel Proessing”. Chapman and Hall, 1992.

[4] M.Cohn, “Succeeding with Agile: Software Development Using Scrum”, Pearson Education, 2010.

[5] Ahmed K. Elmagarmid & Marek Rusinkiewicz & Amit Sheth. “Management of Heterogeneous and Autonomous Database Systems”. Morgan Kaufmann Publishers, 1999.

[6] Kristina Chodorow & Michael Dirolf. “MongoDB: The Definitive Guide”. O'Reilly, 2010.

[7] Satnam Alag. “Collective Intelligence in Action”. Manning Publication, 2009.

[8] Jason Venner. “Pro Hadoop”. Apress, 2009.

[9] Tom White. “Hadoop: The Definitive Guide”, O'Reilly, 2011.

[10] Michael McCandless & Erik Hatcher. “Lucene in Action, Second Edition: Covers Apache Lucene 3.0”. Manning Publication, 2010.

[11] David Smiley & Eric Pugh. “Solr 1.4 Enterprise Search Server”. Packt Publishing, 2009.

[12] Erik Hatcher, Otis Gospodnetić. “Lucene in Action”, 2nd. ed, Manning Publications Co. 2004.

[13] WhiteHouse.gov Goes Drupal,

http://personaldemocracy.com/node/15131

[14] Thoughts on the Whitehouse.gov switch to Drupal, http://radar.oreilly.com/2009/10/whitehouse-switch-drupal-opensource.html

[15] Cal Henderson: “Building Scalable Web Sites”, O'Reilly Media, 2006

[16] Ricky Ho: Scalable System Design Patterns, Pragmatic Programming Techniques. http://horicky.blogspot.com/2010/10/scalable-system-design-patterns.html

[17] Azza Abouzeid, Kamil BajdaPawlikowski, Daniel Abadi1, Avi Silberschatz, Alexander Rasin: HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads.

[18] Jeffrey Dean and Sanjay Ghemawat: MapReduce: Simplied Data Processing on Large Clusters. Google Inc.

[19] Michael Stonebraker: The Case for Shared Nothing. University of California, Berkeley, Ca.