Software Libre para Aplicaciones de Big Data

Club de Investigación Tecnológica
San José, Costa Rica 2014.07.16

Theodore Hope • hope@aceptus.com

Big Data: ¿Qué es?

- Conjuntos de datos de:
  - Alto volumen (TBs → PBs)
  - Alta velocidad ("fast data")
  - Alta variedad (estructura cambiante)

  ... que no pueden ser procesados con herramientas convencionales
- Requerimientos de Hardware:
  - Mucho: CPU, RAM, disco
  - Escalamiento horizontal

Big Data: Casos de Uso

- Vista 360º de clientes
- Servicios Públicos
- “Internet of Things”
- Detección de Fraude
- Agregación / Exploración de datos

- RTAP
- OLTP

Herramientas de Software

- ¿Qué es “software libre”?
  - Costo de *licenciamiento* = $ 0
  - Commodity Hardware & Open Systems

- Soluciones no gratuitas ($)
  - Pregúntele a su vendedor

- Herramientas No-RDBMS
- Herramientas RDBMS

Herramientas No-RDBMS

- "NoSQL" / "Non-Relational"
  - MapReduce / Apache Hadoop
  - Apache HBase
  - MongoDB
  - Apache CouchDB
  - Apache Cassandra
  - Apache Solr (Apache Lucene)
  - Elasticsearch (Apache Lucene)
- Escalables / Alta Disponibilidad
- No ACID

MapReduce (Google, 2004)

- Modelo de programación para procesar enormes conjuntos de datos en paralelo, distribuido en un cluster
- Map/Reduce viene de programación funcional
- Implementación abierta: Hadoop

Hadoop (MapReduce)

- Implementación abierta de MapReduce
- Framework incluye:
  - Sistema de archivos distribuido (HDFS)
  - Administración de nodos y tareas
  - MapReduce
- Algunos casos de uso
  - Risk modeling
  - Recomendaciones para usuarios
  - Clusters de 100s PB

Frameworks para Hadoop

- Apache Hive
  - Datos estructurados, queries semi-SQL
- Apache Pig
  - Datos no estructurados
- Cascading
  - Capa de abstracción para workflows Hadoop
- Apache Spark
  - Para apps de alto nivel (Java, Scala, Python)

Hadoop en la nube

- PaaS con modelo “Pay-as-you-go”
- Cluster elástico (según $$)

- AWS Elastic MapReduce
- Azure HDInsight

HBase

- No relacional, orientado a columnas, para Hadoop
- “Wide-column store” (schema-free)
- Modelado en Google BigTable
- v.g., Facebook Messaging Platform

MongoDB

- Orientado a documentos (JSON), schema dinámico
- Ad-hoc queries sobre cualquier campo
- HA con réplicas, escala horizontalmente
- Eventual Consistency

CouchDB

- Orientado a documentos (JSON), schema dinámico
- Funciones de Map en JavaScript
- Operaciones con semántica ACID, pero sólo logra Eventual Consistency

Cassandra

- Híbrido de esquemas Key-Value y Column-Oriented
- Pseudo-SQL queries (no joins)
- Inspirado en Google BigTable y Amazon DynamoDB
- HA (no SPOF), Eventual Consistency
- Mejor throughput de todos los NoSQL

Solr y Elasticsearch

- Basados en Apache Lucene (Java)
- Para búsquedas en texto libre
  - También archivos Word, PDF
- Altamente escalable
- Tiempo real, indexación contínua
- Usuarios, v.g.
  - Elasticsearch: Wikimedia, Foursquare
  - Solr: Netflix, WhiteHouse.gov

Herramientas RDBMS

- MySQL Cluster
- VoltDB

MySQL Cluster

- Full SQL, ACID, HA, Active/Active
- Datos completamente en RAM
- Nodos SQL y Nodos de Datos escalan independientemente y linealmente
- Ideal para OLTP
- Con servidores Intel E5-2500 / 64 GB:
  - 4 data nodes: 2.5 millones de *updates* / seg
  - 16 data nodes: 10 millones de *updates* / seg

VoltDB

- Stonebraker (MIT / UC Berkeley)
  - Ingres, Postgres, C-Store, H-Store, Vertica
- Full SQL, ACID, HA, Active/Active
- Datos completamente en RAM
- Stored Procedures escritos en Java
- Ideal para OLTP, “high velocity” apps
- 3-node cluster ($3k), > 100k *updates*/s
- 12-node cluster, > 1 millón TPS

Observaciones sobre RDBMS

- Big Data y RDBMS: ¿Hay algo nuevo?
- Requerimientos:
  - ACID, HA, todo en RAM, escalabilidad
- Modelo relacional es:
  - Maduro, conocido, y apropiado para muchos problemas de Big Data
- El problema no son los RDBMS sino el volumen y complejidad de los *datos*.