(Databases)

a (very) short overview of NoSQL, !rDBMS world

L. Tomassetti

Status

- HEP experiment mostly use rDBMS
  - Oracle + Frontier
  - Mysql
- + Custom software to access (meta)data

Status

- Web portal access (usually Java + Python clients)
- Integrated access into analysis applications (COOL, POOL, CORAL, Custom calls, etc...)

noSQL trend

- Pushed by industry and commercial companies
- Best fitted use case is Web-related
- Large datasets management
- Deployed (and developed) by Big companies like facebook, amazon, google, etc...

noSQL

- Many products available (using different approaches):
  - MongoDB (document store)
  - CouchDB (document store)
  - Cassandra/HBase/etc... [hadoop family] (multi/wide column)
  - Redis (key/value)
  - many others

CouchDB

- Apache CouchDB is a document-oriented database that can be queried and indexed in a MapReduce fashion using JavaScript. CouchDB also offers incremental replication with bi-directional conflict detection and resolution.
- CouchDB provides a RESTful JSON API than can be accessed from any environment that allows HTTP requests. There are myriad third-party client libraries that make this even easier from your programming language of choice. CouchDB's built in Web administration console speaks directly to the database using HTTP requests issued from your browser.

couchdb.apache.org
see also www.couchbase.com

MongoDB

- MongoDB (from "humongous") is a scalable, high-performance, open source, document-oriented database. Written in C++, MongoDB features:
  - Document-oriented storage » JSON-style documents with dynamic schemas offer simplicity and power.
  - Full Index Support » Index on any attribute.
  - Replication & High Availability » Mirror across LANs and WANs for scale.
  - Auto-Sharding » Scale horizontally without compromising functionality.
  - Querying » Rich, document-based queries.
  - Fast In-Place Updates » Atomic modifiers for contention-free performance.
  - Map/Reduce » Flexible aggregation and data processing.

www.mongodb.org

Cassandra

- The Apache Cassandra Project develops a highly scalable second-generation distributed database, bringing together Dynamo's fully distributed design and Bigtable's ColumnFamily-based data model. Cassandra was open sourced by Facebook in 2008, and is now developed by Apache committers and contributors from many companies.
- Cassandra is in use at Digg, Facebook, Twitter, Reddit, Rackspace, Cloudkick, Cisco, SimpleGeo, Ooyala, OpenX, and more companies that have large, active data sets. The largest production cluster has over 100 TB of data in over 150 machines.
- **Fault Tolerant** Data is automatically replicated to multiple nodes for fault-tolerance. Replication across multiple data centers is supported. Failed nodes can be replaced with no downtime.
- **Decentralized** Every node in the cluster is identical. There are no network bottlenecks. There are no single points of failure.
- Synchronous or asynchronous replication for each update. Highly available asynchronus operations are optimized with features like Hinted Handoff and Read Repair.
- **Rich Data Model** Allows efficient use for many applications beyond simple key/value.
- **Elastic** Read and write throughput both increase linearly as new machines are added, with no downtime or interruption to applications.
- **Durable** Cassandra is suitable for applications that can't afford to lose data, even when an entire data center goes down.

cassandra.apache.org
see also hadoop.apache.org

Redis

- Redis is an open source, advanced key-value store. It is often referred to as a data structure server since keys can contain strings, hashes, lists, sets and sorted sets.
- You can run atomic operations on these types, like appending to a string; incrementing the value in a hash; pushing to a list; computing set intersection, union and difference; or getting the member with highest ranking in a sorted set.
- In order to achieve its outstanding performance, Redis works with an in-memory dataset. Depending on your use case, you can persist it either by dumping the dataset to disk every once in a while, or by appending each command to a log.
- Redis also supports trivial-to-setup master-slave replication, with very fast non-blocking first synchronization, auto-reconnection on net split and so forth.
- Other features include a simple check-and-set mechanism, pub/sub and configuration settings to make Redis behave like a cache.

redis.io

R&D activities

- Some tests have been started trying to model the bookkeeping (part of) database used by the production tools with:
  - CouchDB
  - MongoDB

CouchDB test

- Work in progress
- Created a collection with FastSim metadata of 2010_September production (~85 kJobs)
  - Just state-switch records (prepared, submitted, running, finished) ~350 kEntries
  - with and without revisions
  - Unexpectedly slow (>10 minutes for inserts / updates)
  - Large disk occupancy (~GB w.r.t. few MB)

Some lessons already learned

- Db structure strongly depends on querying patterns
- Data access (mostly) through REST interface and JSON
  - Needs to develop proper APIs to provide easy interactions with users/applications

Future R&D work

- Same test in progress with MongoDB
- Collaboration with Bari (D. Diacono), Pisa (A. Fella), Ferrara (LT + students) and hopefully others...
- Studying APIs, best practices in coding, etc... for both products
- Indentifying use cases for test units preparation
- Keep an eye on rDBMS usage evolution and connections with Online + Persistence

Discussion items

- Use cases to start with...
- Conditions DB, Bookkeeping DB, ...
- Online / offline (is it necessary to keep a separation?)
- Data access patterns
- Timeseries (other?) reduction?

XLDB-2011

I'm looking for sponsor(s)!

- 5th Extremely Large Databases Conference, October 18-19, 2011
  SLAC National Accelerator Laboratory
- The XLDB conference focuses on the management and analysis of data at extreme scale. It provides a unique opportunity to meet and learn from leading practitioners from science, industry, and academia working on real-world solutions for handling terabytes and petabytes of data.
- This year's event includes in-depth presentations about tools and practices at Facebook, eBay, Google, Zynga, and the National Center for Biotechnology Information; stories about growing to large scale from Novartis, Netflix and LinkedIn, and discussion panel on cloud computing at scale. Peta-scale data simulation, peta-scale data visualization, as well as scalability of statistical tools such as R will be discussed.