Survey on NoSQL Database

Jing Han, Haihong E, Guan Le
PCN&CAD Center
Beijing University of Posts and Telecommunications
Beijing, 100876, *China*
babyblue110128@hotmail.com,
ehaihong@bupt.edu.cn, optimism1226@gmail.com

Jian Du
Super Instruments Corporation
Beijing, 100876, *China*
jamesdu@superinst.com

Abstract

With the development of the Internet and cloud computing, there need databases to be able to store and process big data effectively, demand for high-performance when reading and writing, so the traditional relational database is facing many new challenges. Especially in large scale and high-concurrency applications, such as search engines and SNS, using the relational database to store and query dynamic user data has appeared to be inadequate. In this case , NoSQL database created. This paper describes the background, basic characteristics, data model of NoSQL. In addition, this paper classifies NoSQL databases according to the CAP theorem. Finally, the mainstream NoSQL databases are separately described in detail, and extract some properties to help enterprises to choose NoSQL.

Keywords: NoSQL; key-value; column-oriented;document; database;Big Data;

1. Introduction

With the continuous development of the Internet and cloud computing, various types of applications have emerged, which made database technology more demands, mainly in the following aspects [1][2]:

- *High concurrent of reading and writing with low latency*

Database were demand to meet the needs of high concurrent of reading and writing with low latency, at the same time, in order to greatly enhance customer satisfaction, database were demand to help applications reacting quickly enough.

- *Efficient big data storage and access requirements*

Large applications, such as SNS and search engines, need database to meet the efficient data storage (PB level) and can respond to the needs of millions of traffic.

- *High scalability and high availability*

With the increasing number of concurrent requests and data, the database needs to be able to support easy expansion and upgrades, and ensure rapid uninterrupted service.

- Lower management and operational costs

With the dramatic increase in data, database costs, including hardware costs, software costs and operating costs, have increased. Therefore, need lower costs to store big data.

Although relational databases have occupied a high position in the data storage area, but when facing above requirements, it has some inherent limitations:

- *Slow reading and writing*

A relational database itself has a certain logic complexity, with the data size increases, it is prone to bring about deadlocks and other concurrency issues, this has led to the rapid decline in the efficiency of reading and writing;

- *Limited capacity*

Existing relational database cannot support big data in search engine, SNS or Big System;

- *Expansion difficult*

This work is supported by the National Key project of Scientific and Technical Supporting Programs of China (Grant Nos.2008BAH24B04, 2009BAH39B03); the National Natural Science Foundation of China (Grant No.61072060); the Program for New Century Excellent Talents in University (No.NECET-08-0738); the Co-construction Program with Beijing Municipal Commission of Education; Engineering Research Center of Information Networks, Ministry of Education.

978-1-4577-0208-2/11/$26.00 ©2011 IEEE

Multi-table correlation mechanism which exists in relational database, became the major factor of database scalability

To solve several needs above, a variety of new types of databases have appeared. In general, these new databases are very different with traditional relational databases, so it is referred to as "NoSQL" database. NoSQL also be interpreted as the abbreviation of "NOT ONLY SQL" to show the advantage of NoSQL. After the introduction of the background of NoSQL, we will focus on the advantages and disadvantages NoSQL database.

2. Features, data model and classification

2.1 Features of NoSQL

Main advantages of NoSQL are the following aspects: 1) reading and writing data quickly; 2) supporting mass storage; 3) easy to expand; 4) low cost.

Meanwhile, NoSQL have some inadequacies, such as does not support SQL which is industry standard , lacking of transactions, reports and other additional features, not mature enough for most of the NoSQL database products were created in recent years and so on.

Feature of NoSQL database described above are common ones, in reality, each product comply with the different data models and CAP theorem. Therefore, we will introduce NoSQL database data model, and classify NoSQL according to CAP theorem.

2.2 Data Model

Data model of traditional database are mainly relational, specifically to support associated class operations and ACID transactions, but in the NoSQL database fields, the mainstream data model are the following:

2.2.1. Key-value

Key-value data model means that a value corresponds to a Key, although the structure is simpler, the query speed is higher than relational database, support mass storage and high concurrency, etc., query and modify operations for data through the primary key were supported well.

2.2.2 Column-oriented

Column-oriented database using Table as the data model, but does not support table association. Column-oriented database has the following characteristics: 1) data is stored by column, that is data stored separately for each column; 2) each column of data is the index of database; 3) only access the columns involving the queries result to reduce the I/O of system; 4) concurrent process queries, that is, each column treat by one process; 5) there have the same type of data, similar characteristics and good compression ratio. Overall, the advantage of this data model is more suitable application on aggregation and data warehouse.

2.2.3 Document

Document database and Key-value is very similar in structure, but the Value of document database is semantic, and is stored in JSON or XML format. In addition, the document databases can generally a Secondary Index to value to facilitate the upper application, but Key-value database cannot support this.

2.3 CAP theorem and NoSQL database classification

In 2000, Professor Eric Brewer put forward the famous CAP theorem. That is, Consistency, Availability, tolerance of network Partition.CAP theorem's core idea is a distributed system cannot meet the three district needs simultaneously, but can only meet two.

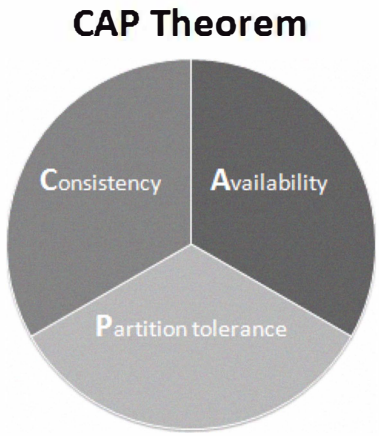

Figure 1. CAP Theorem

According to CAP theorem and different concerns of NoSQL database, a preliminary classification of NOSQL databases is as follows [3]:

- *Concerned about consistency and availability (CA)*

Part of the database is not concerned about the partition tolerance, and mainly use of Replication approach to ensure data consistency and availability.

Systems concern the CA are: the traditional relational database, Vertica (Column-oriented), Aster Data (Relational) , Greenplum (Relational) and so on.

- *Concerned about consistency and partition tolerance(CP)*

Such a database system stores data in the distributed nodes, but also ensure the consistency of these data, but support not good enough for the availability. The main CP system: BigTable (Column-oriented), Hypertable (Column-oriented), HBase (Column-oriented), MongoDB (Document), Terrastore (Document), Redis (Key-value), Scalaris (Key-value) , MemcacheDB (Key-value), Berkeley DB (Key-value).

- *Concerned about availability and partition tolerance (AP)*

Such systems ensure availability and partition tolerance primarily by achieving consistency, AP's system: Voldemort (Key-value), Tokyo Cabinet (Key-value), KAI (Key-value), CouchDB (Document-oriented), SimpleDB (Document-oriented), Riak (Document-oriented).

3 Mainstream NoSQL database

3.1 Key-value databases

3.1.1 Redis

Redis[4] is a very new project, the following is its characteristics: 1) Redis is the a Key-value memory database: when Redis run, data were entire load into memory, so all the operations were run in memory, then periodically save the data asynchronously to the hard disk. The characteristics of pure memory operation makes it very good performance, it can handle more than 100,000 read or write operation per second; 2) Redis support List and Set and various related operations; 3) The maximum of value limit to 1GB; 4) the main drawback is that capacity of the database is limited by physical memory, so Redis cannot be used as big data storage, and scalability is poor.

Therefore, Redis is suitable for providing high-performance computing to small amount of data.

3.1.2 Tokyo Cabinet- Tokyo Tyrant

TC and TT were developed by Mikio Hirabayashi in Japan mainly for Japan's largest SNS site mixi.jp, it has been a very mature project yet. TC is a high-performance storage engine, while the TT provide multi-threaded high-concurrency servers, it can handle 4-5 million times read and write operations per second.

While ensure the high performance of read and write concurrent, TC uses a reliable data persistence mechanism. TC supports data structure not only Key-value, but also hashtable which is similar with relational database, in addition, supports simple operations like conditional query, paging and sorting.

TC's main drawback is that when the amount of data grows to the billion levels, concurrent write performance will drop dramatically.

3.1.3 Flare

Flare was developed by the second largest SNS website green.jp in Japan, Flare is stronger than TC because of the scalability has been extends. Flare added a node server before data servers in order to manage data servers at back, so user can dynamically add or remove data servers, and it also supports failover. However, Flare supports only the memcached protocol, when using Flare, you cannot use the table data structure of TC, but only can use TC's Key-value data structure.

3.2 Column-oriented database

Although column-oriented database has not subverted the traditional stored by row, but in architecture with data compression, shared-nothing, massively parallel processing, column-oriented database can maintain high-performance of data analysis and business intelligence processing. Column-oriented databases are HBase, Yale University HadoopDB, Facebook's Cassandra, Hypertable, Google's Bigtable[9], and Yahoo's PNUTS and so on.

3.2.1 Cassandra

Cassandra [5] is an opensource database of facebook.

Its characteristics are: 1) the schema is very flexible and does not require to design database schema at first , and add or delete field is very convenient; 2) support range queries, that is it can range queries for Key ; 3) high scalability: a single point of failure does not affect the whole cluster, and it support linear expansion.

Cassandra system is a distributed database system which was composed of lots of database nodes, a write operation will be replicated to other nodes, and read request will be routed to a certain node. For a Cassandra cluster, only to add node can achieve the goal of scalability. In addition, Cassandra also supports rich data structure and powerful query language.

3.2.2 Hypertable

Search engine Zvents develop opensource distributed data storage system Hypertable [6] by drawing Bigtable. The system is designed for the proportion of 1000 nodes, and it can be deployed on HDFS and KFS. Although Hypertable is still in its infancy, but the performance is not bad: when write 28M column data to Hypertable, the write rate of each node is up to 7MB / s, read speed is up to 1M cells / s. however, there is not much higher loads and large storage application use Hypertable.

3.3 Document database

Document database is not concerned about high-performance read and write concurrent, but rather to ensure that big data storage and good query performance. Typical document database are MongoDB, CouchDB.

3.3.1 MongoDB

MongoDB[7] is a database between relational databases and non-relational database, its features are: 1) it is non-relational database, which features the richest and most like the relational database; 2) support complex data types: MongoDB support bjson data structures to store complex data types; 3) powerful query language: it allows most of functions like query in single-table of relational databases, and also support index. 4) High-speed access to mass data: when the data exceeds 50GB, MongoDB access speed is 10 times than MySQL. Because of these characteristics of MongoDB, many projects with increasing data are considering using MongoDB instead of relational database.

3.3.2 CouchDB

Apache CouchDB[8] is a flexible, fault-tolerant database, which supports data formats such as JSON and AtomPub, it provides REST-style API. To ensure data consistency, CouchDB comply with ACID properties. In addition, CouchDB provides a P2P-based distributed database solution that supports bi-directional replication. However, it also has some limitations, such as only providing an interface based on HTTP REST, concurrent read and write performance is not ideal and so on.

4 Conclusions

Based on the above description of the mainstream NoSQL databases, when companies decide whether to use NoSQL, it needs to consider the business model, ACID transactions demand, cost and other requirements.

In addition , companies need to consider the following options when deciding which properties NoSQL:

- Data Model
- CAP Support
- Multi Data Center Support
- Capacity
- Performance
- Query API
- Reliability
- Data Persistence
- Rebalancing
- Business Support

This paper describes the background of NoSQL first, after understand the requirements to database of existing applications and the limitations of traditional database, then the advantages and disadvantages of NoSQL database were analyzed .According to the characteristics of NoSQL, summarized data model of NoSQL database .After a brief introduction of the CAP theorem, classification of NoSQL database was proposed. Finally, according to each type of data models, introduce the current mainstream NoSQL database, and objective analysis of their strengths and weaknesses respectively, which will help user to choose NoSQL database in practice. However, NoSQL database still have various limitations, and using NoSQL in cloud computing is also not clear enough; we will be to strengthen research in these two areas.

References

[1] Kai Fan, “Suvey on Nosql”, *Programmer*, 2010(6): pp.76-78

[2] Jing Han, Meina Song, and Junde Song. “A Novel Solution of Distributed Memory NoSQL Database for Cloud Computing”, In ICIS 2011, 10th IEEE/ACIS International Conference on Computer and Information Science, 2011.

[3] Nathan Hurst, “Visual Guide to NoSQL Systems.”, http://blog.nahurst.com/visual-guide-to-NoSQL-systems/

[4] Redis http://redis.io/

[5] Avinash Lakshman, Prashant Malik, “Cassandra-A Structured Storage System on a P2P Network”, http://cassandra.apache.org/

[6] Hypertable, http://hypertable.org/

[7] Mongodb,http://www.mongodb.org/display/DOCS/Home

[8] Couchdb, http://couchdb.apache.org/.

[9] Fay Chang et al., "Bigtable: A Distributed Storage System for Structured Data", OSDI 06'