

NoSQL DATABASE



DATA MANAGEMENT SYSTEMS

Today we have to manage lots of data. It's getting harder and harder to choose a Data Management System when evaluating relational and non-relational systems in non-single-server environments. Today there are 4 major data models on the market:

- Relational (supports ACID)
- Key-value (supports get, put, and delete operations based on a primary key)
- Column-oriented (store data by column as opposed to traditional row-oriented databases)
- Document-oriented (stores structured "documents" such as JSON or XML)

GOALS

 In most Data Management projects we have to achieve the following items:

- Availability
- Consistency
- Scalability
- Low cost

MAJOR TRADE-OFFS

- You can only choose 2 of the following:
 - Availability (all clients should always read and write)
 - Partitioning (the system should work across physical network partitions)
 - Consistency (all clients should have the same view of the data)
- One of the major goals of a NoSQL system is to provide robust horizontal scalability(partitioning). To achieve that we have to give up either consistency or availability.

WHAT TO CHOOSE

- ✦ Choosing consistency and availability: In this case we have trouble with partitions and typically deal with it with replication(RDBMSs like MySQL, Oracle, MS SQL, Greenplum).
- ✦ Choosing consistency and partitioning: In this case we have trouble with availability while keeping data consistent across partitioned nodes such as MongoDB(document-oriented), Hbase(column-oriented) and Redis(key-value).
- ✦ Choosing availability and partitioning: In this case we have trouble with “eventual consistency” across replicas such as Cassandra(column-oriented), CouchDB(document-oriented).

TODAYS CHALLENGES

- The biggest challenge today is data mining. There is no NoSQL database that provides data mining tools similar to what's available for regular RDBMS
- In many existing solutions people use RDBMS for data mining which has been aggregated in chunks from a NoSQL source.
- There is a common hybrid: Storing data in NoSQL (provides fast inserts). Doing data mining in RDBMS (provides variety of tools)
- NoSQL is good, but today it can't fully replace RDBMS for every project, but it's getting there.