



NoSQL Database

Syllabus

Introduction to NoSQL Database, Types and examples of NoSQL Database- Key value store, document store, graph Performance, Structured versus unstructured data, Distributed Database Model, CAP theorem and BASE Properties, Comparative study of SQL and NoSQL, NoSQL Data Models, Case Study-unstructured data from social media Introduction to Big Data, HADOOP: HDFS, MapReduce.

15.1 Introduction to NoSQL Database

Q. Explain the concept of NoSQL Database and state its advantages over RDBMS.

SPPU - May 18, 4

Marks

Q. Analyze the use of NoSQL databases in current social networking environmental so explain need of NoSQL databases in social networking environment over RDBMS.

SPPU - May 18, 6 Marks

1. History

- The term NoSQL was first used by Carlo Strozzi in the year 1998.
- He mentioned this name for his Open Source Database system in which there was no provision of SQL Query interface.
- In the early 2009, at conference held in USA, NoSQL was comes into picture and actually comes in practice.

2. Overview

- NoSQL is a not a RDBMS (Relational Database Management System).
- NoSQL is specially designed for large amount of data stored in distributed environment.
- The important feature of NoSQL is, it is not bounded by table schema restrictions like RDBMS. It gives options to store some data even if there is no such column is present in table.
- NoSQL generally avoids join operations.

3. Need

- In real time, data requirements are changed a lot. Data is easily available with Facebook, Google+, Twitter and others.
- The data that includes user information, social graphs, geographic location data and other user-generated content.
- To make use of such abundant resources and data, it is necessary to work with a technology which can operate such data.
- SQL databases are not ideally designed to operate such data.
- NoSQL databases specially designed for operating huge amount of data.

4. Advantages

1. Good resource scalability.



2. Lower operational cost.
3. Supports semi-structure data.
4. No static schema.
5. Supports distributed computing.
6. Faster data processing.
7. No complicated relationships.
8. Relatively simple data models.

5. Disadvantages

1. Not a defined standard.
2. Limited query capabilities.

6. Companies working with NoSQL

- | | |
|-------------|--------------------------|
| 1. Google | 2. Facebook |
| 3. LinkedIn | 4. McGraw-Hill Education |

15.2 CAP Theorem (Brewer's Theorem)

Q. State and Explain CAP Theorem.

SPPU - May 18, 6

Marks

Q. Explain the CAP theorem referred during the development of any distributed application.

SPPU - Dec.18, 7 Marks

CAP theorem states three basic requirements of NoSQL databases to design a distributed architecture.

a) Consistency

Database must remain consistent state like before, even after the execution of an operation.

b) Availability

It indicates that NoSQL system is always available without any downtime.

c) Partition Tolerance

This means that the system continues to function even the communication failure happens between servers i.e. if one server fails, other server will take over.

Note : It is very difficult to fulfill all the above requirements.

There are many combinations of NoSQL rules :

1. CA

- It is a single site cluster.
- All nodes are always in contact.
- Partitioning system can block the system.

2. CP

Some data may not be accessible always still it may be consistent or accurate.

3. AP



- System is available under partitioning.
- Some part of the data may be inconsistent.



15.3 BASE Properties for NoSQL

Q. BASE Transactions ensures the properties like Basically Available, Soft State, Eventual Consistency. What is soft state of any system, how it is depend on Eventual consistency property?	SPPU - Dec. 17, Dec.18, 8 Marks
Q. List the different NOSQL Data Models. Explain document store NOSQL data model with example.	SPPU - Dec. 17, 8 Marks
Q. Enlist and explain any three NoSQL Database types.	SPPU - May 18, Dec.18, 6 Marks
Q. State and Explain: BASE properties.	SPPU - May 18, 6 Marks

1. BASE Introduction

- Relational databases have some rules to decide behavior of database transactions.
- ACID model maintains the atomicity, consistency, isolation and durability of database transactions.
- NoSQL turns the ACID model to the BASE model.

2. Guidelines

- Basic availability
- Soft state
- Eventual consistency

3. Data storage

- NoSQL databases use the concept of a key / value store.
- There are no schema restrictions for NoSQL database.
- It simply stores values for each key and distributes them across the database, it offers efficient retrieval.

4. Redundancy and Scalability

- To add redundancy to a database, we can add duplicate nodes and configure replication.
- Scalability is simply a matter of adding additional nodes. There can be hash function designed to allocate data to server.

15.4 Structured and Unstructured Data

15.4.1 Structured Data

(a) Introduction

- Information stored in database is represented in strict format. Hence it is called as structured data.
- Data recorded in form of rows and columns is also a type of structured data.
- Only 2% of data available in world is exists in form of Structured data.

(b) Example

Consider EMPLOYEE table below

Table 15.4.1: Employee Table

FName	LName	SSN	DOB	Sex	DNO
John	Wayne	11101101	1/1/1971	M	1
Bruce	Willis	1111001	1/1/1981	M	5
Jane	Honda	9123876	1/1/1991	F	10
Drew	Barry	9989951	1/1/1986	F	7

Table 15.4.2 : Project Table

PNo	PName	Location	DNO
1	Proj_X	Hinton	5
5	Proj_Y	Philistine	10
10	Proj_Z	New Castle	7

Table 15.4.3 : Works on Table

SSN	PNo	Hours
11101101	5	33.5
9123876	7	100.0
9989951	1	Null

Above is known as structured data, because each records in any of above table follows the same format as other record in the table.

- (c) The DBMS checks to ensure that all the data follows the structure and constraints specified in the schema.

15.4.2. Unstructured Data

(a) Introduction

- If there is very limited restriction to the type of data it is called as unstructured data.
- Images, Blogs, Videos and audio files on internet comes under this category as there is no fixed format of data.

- Data which is not in form of rows and columns is unstructured data.
- More than 90% of data available in world is unstructured data.

(b) Example

- For an example text document that contains information in it.
- Web pages in HTML contain unstructured data.

Example : Consider following HTMLfile

```
<HTML>
  <HEAD>
    <TITLE>XavierCollege Of Engineering </TITLE>
  </HEAD>
  <BODY>
    <H1>Computer Engineering</H1>
    <H2>Mahim Causeway, Mumbai</H2>
    <FONT size="2" face="Arial">this college is one of the best college located in
    central Mumbai<BR>
    We Offers you Degree course in various branches<BR>
  </FONT>
</BODY>
</HTML>
```

15.5 Comparative Study of SQL and NoSQL (SQL vs NoSQL)

Q. Explain how NOSQL databases are different than relational databases?

SPPU - Dec.17, 8 Marks

Q. Explain the difference SQL Vs NoSQL.

SPPU - May 18, Dec.18, 4

Marks

SQL databases are Relational Databases (RDBMS); whereas NoSQL database are non-relational database.

1. Data Storage

- SQL databases stores data in a table whereas NoSQL databases stores data as document based, key-value pairs, graph databases or wide-column stores.
- SQL data is stored in form of tables with some rows.
- NoSQL data is stored as collection of key-value pair or documents or graph based data with no standard schema definitions.

2. Database Schema

SQL databases have predefined schema which cannot be change very frequently, whereas NoSQL databases have dynamic schema which can be change any time for unstructured data.

3. Complex Queries

- SQL databases provides standard platform for running complex query.
- NoSQL does not provide any standard environment for running complex queries.
- NoSQL are not as powerful as SQL query language.

Table 15.5.1

Sr. No.	SQL	NoSQL
1.	Full form is Structured Query Language.	Full form is Not Only SQL or Non-relational database.
2.	SQL is a declarative query language.	This is Not a declarative query language.
3.	SQL databases works on ACID properties, Atomicity Consistency Isolation Durability	NoSQL database follows the Brewers CAP theorem, Consistency Availability Partition Tolerance
4.	Structured and organized data	Unstructured and unreplicable data
5.	Relational Database is table based.	Key-Value pair storage, Column Store, Document Store, Graph databases.
6.	Data and its relationships are stored in separate tables.	No pre-defined schema.
7.	Tight consistency.	Eventual consistency rather than ACID property.
8.	Examples : MySQL Oracle MS SQL PostgreSQL SQLite DB2	Examples : MongoDB Big Table Neo4j Couch DB Cassandra HBase

15.5.1 Business Drivers of NoSQL

1. The Growth of Big Data

- Big Data is one of the main driving factor of NoSQL for business.
- The huge array of data collection act as driving force for data growth.

2. Continuous Availability of Data

- The competition age demands less downtime for better company reputation.
- Hardware failures are possible but NoSQL database environments are built with a distributed architecture so there are no single points of failure.
- If one or more database servers goes down, the other nodes in the system are able to continue with operations without any loss of data.
- So, NoSQL database environments are able to provide continuous availability.

3. Location Independence

- It is ability to read and write to a database regardless of where that I/O operation is done.
- The master/slave architectures and database sharding can sometimes meet the need for location independent read operations.



4. Modern Transactional Capabilities

The transactions concept is changing and ACID transactions are no longer a requirement in database systems.

5. Flexible Data Models

- NoSQL has more flexible data model as compared to others.
- A NoSQL data model is schema-less data model not like RDBMS.

6. Better Architecture

- The NoSQL has more business oriented architecture for a particular application.
- So, Organizations adopt a NoSQL platform that allows them to keep their very high volume data.

7. Analytics and Business Intelligence

- A key driver of implementing a NoSQL database environment is the ability to mining data to derive insights that offers a competitive advantage.
- Extracting meaningful business information from very high volumes of data is a very difficult task for relational database systems.
- Modern NoSQL database systems deliver integrated data analytics and better understanding of complex data sets which facilitate flexible decision-making.

15.6 NoSQL Data Models ---

Different Architectural Patterns in NoSQL

- Key-Value databases Eg. Riak, Redis, Memcached, BerkeleyDB, upscaledb, Amazon DynamoDB.
- Document databases Eg. MongoDB, CouchDB, Terrastore, OrientDB, RavenDB
- Column family stores Eg. Cassandra, HBase, HyperTable.
- Graph Databases Eg. Neo4j, InfiniteGraph, FlockDB.

15.6.1 Key-Value Store Databases

- This is very simple NoSQL database.
- It is specially designed for storing data as a schema free data.
- Such data is stored in a form of data along with indexed key.

Examples :

- Cassandra
- Azure Table Storage (ATS)
- DyanmoDB



Fig. 15.6.1

Use Cases

- This type is generally used when you need quick performance for basic Create-Read-Update-Delete operations and data is not connected.

Example :

- Storing and retrieving session information for a Web pages.
- Storing user profiles and preferences
- Storing shopping cart data for ecommerce

Limitations

- It may not work well for complex queries attempting to connect multiple relations of data.
- If data contains lot of many-to-many relationships, a Key-Value store is likely to show poor performance.

15.6.2 Column Store Database

- Instead of storing data in relational tuples (table rows), it is stored in cells grouped in columns.
- It offers very high performance and a highly scalable architecture.

Examples :

1. HBase
2. Big Table
3. Hyper Table

Use Cases

- Some common examples of Column-Family database include event logging and blogs like document databases, but the data would be stored in a different fashion.
- In logging, every application can write its own set of columns and have each row key formatted in such a way to promote easy lookup based on application and timestamp.
- Counters can be a unique use case. It is possible to design application that needs an easy way to count or increment as events occurs.

15.6.3 Document Database (Document Store)

- Document databases works on concept of key-value stores where “documents” contains alot of complex data.

- Every document contains a unique key, used to retrieve the document.
- Key is used for storing, retrieving and managing document-oriented information also known as semi-structured data.

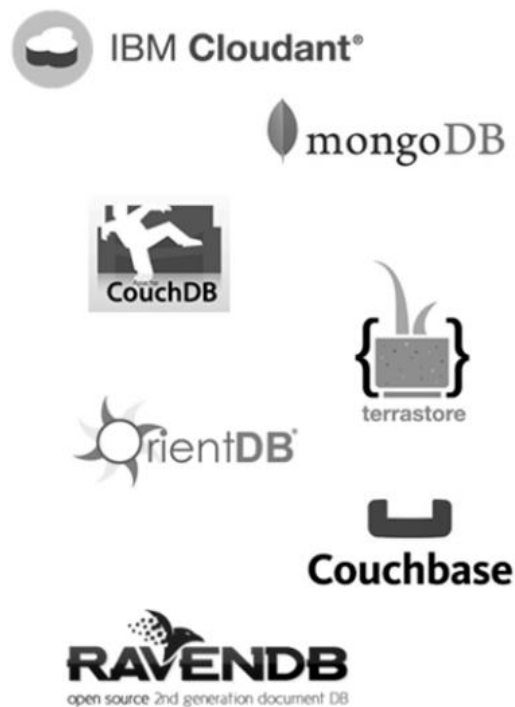


Fig. 15.6.3

Examples :

1. MongoDB
2. Couch DB

Use Cases

- The example of such system would be event logging system for an application or online blogging.
- In online blogging user acts like a document; each post a document; and each comment, like, or action would be a document.
- All documents would contain information about the type of data, username, post content, or timestamp of document creation.

Limitations

- It's challenging for document store to handle a transaction that on multiple documents.
- Document databases may not be good if data is required in aggregation.

15.6.4 Graph Database (Distributed Document Store)

Data is stored as a graph and their relationships are stored as a link between them whereas entity acts like a node.

Examples :

1. Neo4j
2. Polyglot

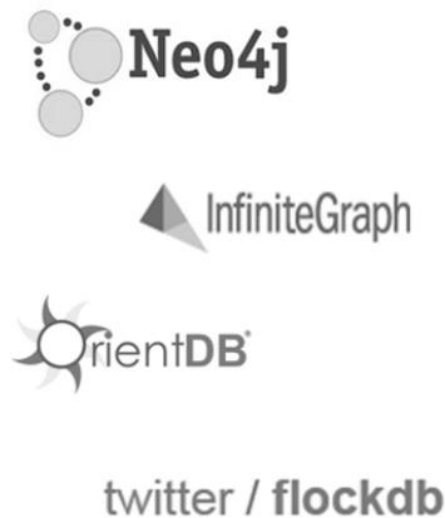


Fig. 15.6.4

Use Cases

- The very important and popular application would be social networking sites can benefit by quickly locating friends, friends of friends, likes, and so on.
- The Google Maps can help you to use graphs to easily model their data for finding close locations or building shortest routes for directions.
- Many recommendation systems makes effective use of this model.

Limitations

- Graph Databases may not be offering better choice over other NoSQL variations.
- If application needs to scale horizontally this may introduces poor performance.
- Not very efficient when it needs to update all nodes with a given parameter.

15.6.5 Performance NoSQL Data Models

Database model	Performance	Scalability	Flexibility
Key value store database	High	High	High
Column store database	High	High	Moderate
Document store database	High	Variable (High)	High
Graph database	Variable	Variable	High



15.6.6 Benefits of NoSQL

1. Big Data Analytics

- Big data is one of main feature promotes growth and popularity of NoSQL.
- NoSQL has good provision to handle such big data.

2. Better Data Availability

- NoSQL database works with distributed environments.
- NoSQL database environments should provide good availability across multiple data servers.
- NoSQL databases supply high performance.

3. Location Independence

NoSQL data base can read and write database regardless of location of database operation.