*U6. Large Scale Database

PAGE No. / DATE / /

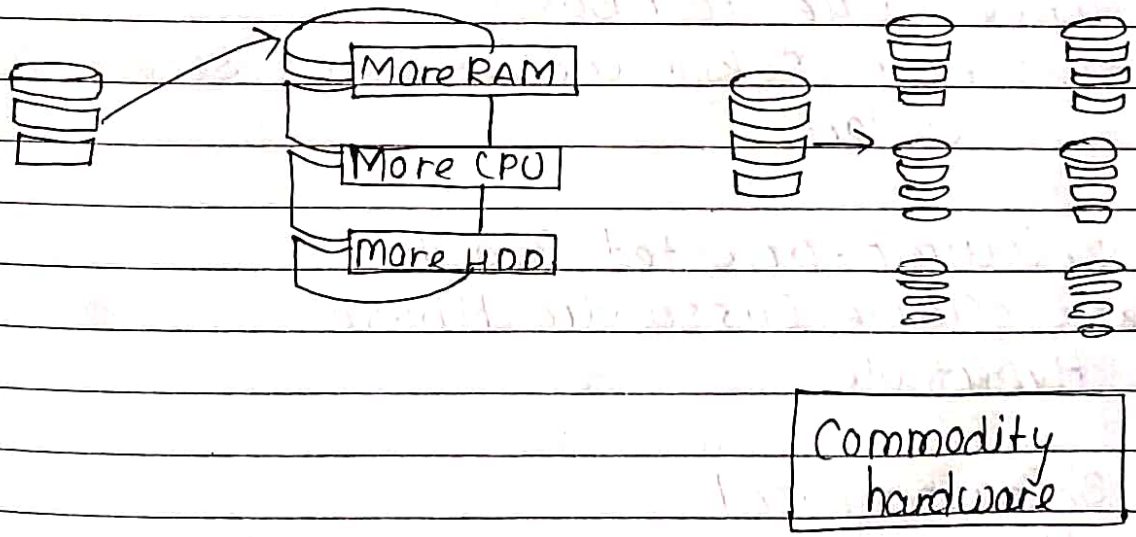
What is NoSql, History of NoSQL, Important characteristic of NoSQL, Type of NoSQL. Comparative study of SQL and NoSQL Introduction to MangoDB. Big Data. HADOOP: HDFS, Map Reduce HBase.

*NoSQL database

NoSQL Database is used to refer a non-SQL or non relational database
It provide a mechanism for storage and retrieval of data other than tabular relation model used in relational databases. NoSQL database doesn't use tables for storing data. It is generally used to store big data and real-time web applications

*Why NoSQL?

Scale up (vertical Scaling) Scale Out (horizontal scaling)



PAGE No.
DATE / /

1998- Carlo Stazzi use the tearm NoSQL, for his lightweight, open source relational Database
2000- Graph database Neo4j is launched
2004- Google Big Table is launched
2005- CouchDB is launched
2007- The research paper on Amazon Dynamo is released
2008- Facebook open source the Cassandra project
2009- The team NoSQL was reintroduced

- Features Of NoSQL
- Non relational
- Distributed Computing
- Schema-free
- NoAPI

* Type Of NoSQL database:-

1) key value Pair Based
Ex:- Riak, Tokyo Cabinet, Redis, Server, Memcached Scalaris

2) Column-oriented Graph
Ex:- Big Table, Cassandra, Hbase Hypertable

3) Graphs based
Ex:- Neo4J, InfoGrid, Infinite

Complex Query Handling	Intensive environment to handle complex queries	Are not good to handle complex queries
Data Set Size	Are not good at handling large data sets	Mostly preferred for large dataset
Support and Adoption	It adopt widely and write support is also variable	Is not adopted widely only local community support is available

* MangoDB

MangoDB is a document-oriented database. This mean that it doesn't use tables and rows to store its data, but instead collection of JSON-like document These documents support embedded fields, so related data can be stored within them
MangoDB is also a schema-less database, so we don't need to specify the number or type of column before inserting the data

PAGE No.
DATE / /

• Comparative Study of SQL and NoSQL

	SQL database	NoSQL database
Another name	Also known as RDBMS or Relational Database	It's non-relational and distributed by nature
Basic	Use SQL to define and manipulate data, based on tables	Document level queries are used and graphics and wide columns can be handled
Data Storage	A Non-~~tria~~ hierarchi -cal database is used	Data is stored in hierarchical order
Types of Data used	Good for these data sets stored in a ~~standard~~ Structured manner	It is good for semi-structured nested and complex data
New Data	New data addition~~s~~ may require schema alteration	Without any alterati -on new data fields can be added
Scalability	Are vertical scalable and with increasing hardware horsepower can be scales	Are horizontal scala -ble and just by

Online applications
- Excels at distributed database and multi-data center operations
- Eliminates the need for a specific caching layer to store data
- Offer a flexible schema design which can easily be altered without downtime or service disruption

Disadvantages of NoSQL

- No standardization rules
- Limited query capabilities
- RDBMS database and tool are comparatively mature
- It does not offer any ~~traditional~~ transaction ~~are~~ per database capabilities, like consistency when multiple transaction are performed simultaneously
- When the volume of data increase it is difficult to maintain unique value as key become difficult
- Doesn't work as well with relational data.
- The learning curve is stiff for new developers
- Open source option so ~~that~~ not so popular for enterprise

PAGE No. DATE / /

4) Document-Oriented
Ex:- MangoDB, CouchDB, OrienteDB, RavenDB

-There are many different type of NoSQL, database, with different specification, some of these are-
Column :- Data is stored in a columnar form some examples of this type of database are: Accumulo, Cassandra, Druid, Vertica, etc
key value :- These database are organized as key value pairs, where each key appears exactly once. The key are usually arranged in sorted fashion
Graph :- The database are arranged in the form of a graph with the element connected using the relations bet them
Documents :- The database is stored in the form of documents that are accessed using unique key. A single key references a ~~databa~~ document

* Advantages of NoSQL
- Can be used as Primary or Analytic Data Source
- Big Data Capability
- No single Point of Failure
- Easy Replication
- It provides fast performance and horizontal scaleability
- Can handle structured, semi-structured and unstructured data with equal effect
- NoSQL database don't need a dedicated high-performance server
- Simple to implement than using RDBMS
- It can serve as the primary data source for

PAGE No.
DATE / /

Example of unstructured data → the output in the form of Google output

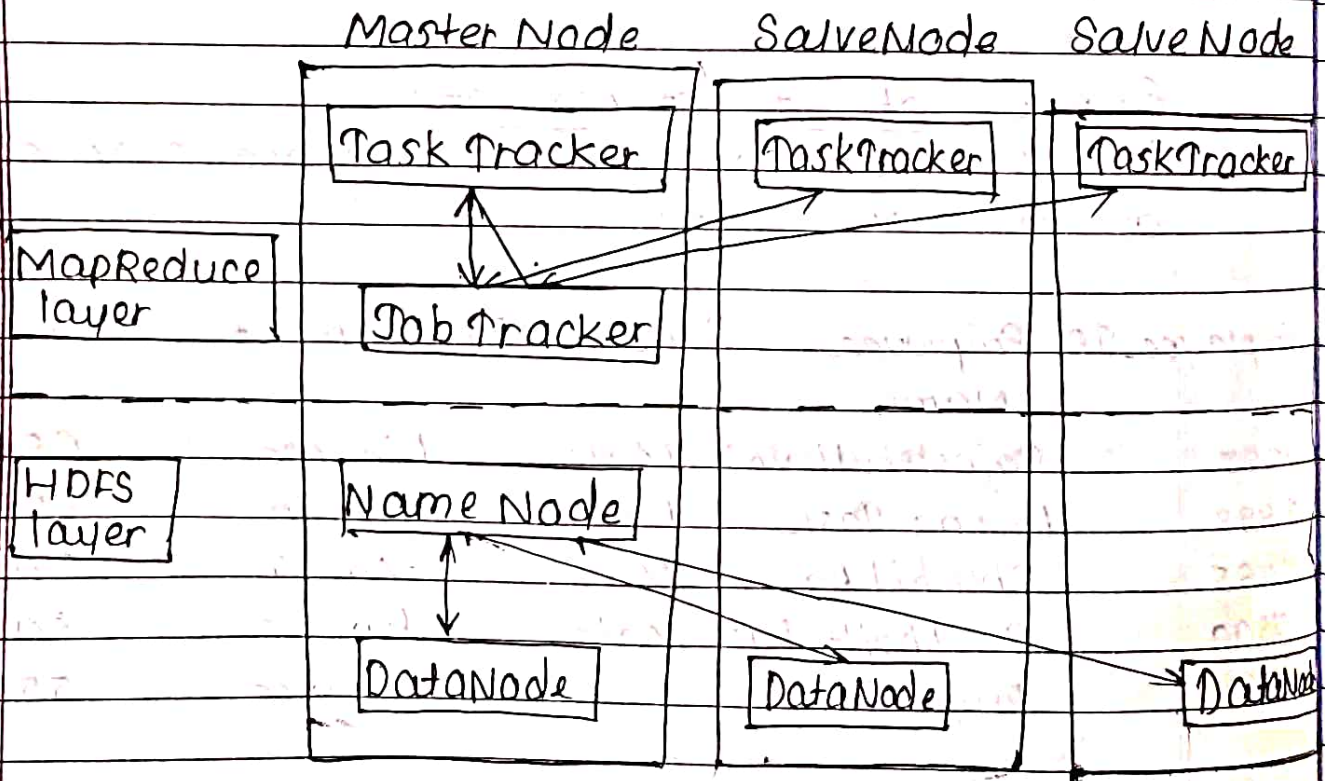
Example of Semi-Structured data → Personal data stored in XML file

* Hadoop

Hadoop is an Apache open source framework written in java that favours distributed processing of large dataset across cluster of computer using simple programming language

* Hadoop Architecture

Hadoop has two major layer namely -
- Processing / Computation layer (Map Reduce), and
- Storage layer (Hadoop Distributed File System)



PAGE No.
DATE / /

Examples Of Big Data

Following are some of the example of Big Data
- The New York Stock Exchange generates about one terabytes of new trade data per data
- Social Media

The ~~site~~ Static shows that 500+ terabytes of new data get ingested into the database of social media site Facebook, every day. This data is mainly generated in term of photo and video uploads, message exchanges, putting comments, etc

Type Of Big Data

Big Data could be found in three forms:
1. Structured
2. UnStructured
3. Semi-Structured

Example of Structured Data

An 'Employee' table in a database is an example of Structured data

Employee ID	Employee Name	Gender	Department	Salary in lacs
2365	Rajeshkulkarni	Male	Finance	650000
3398	Prabha Joshi	Female	Admin	650000
7465	Shushil Roy	Male	Admin	500000
7500	Shubhajit Das	Male	Finance	500000
7699	Priya Sane	Female	Finance	550000

PAGE No.
DATE / /

Conversion / mapping of application objects to database object not needed
Uses internal memory for storing the working set, enabling faster access of data

* Big Data

The term "big data" refers to data that is so large, fast or complex that it's difficult or impossible to process using traditional method. The act of accessing and storing large amount of information for analytics has been around a long time But the concept of big data gained momentum in the early 2000s when industry analyst Doug Laney articulated the now-mainstream defination of big data as the threes Vs:: Volumn, Velocity, Veriety

• Why Is Big Data Important

When you combine big data with high-powered analytics, you can accomplish business-related task such as:
Determining root causes of failure, issues and defects in near-real time
Generating coupons at the point of sale based on the customer buying habit
Recalculating entire risk portfolios in minutes
Detecting fraudulent behavior before it affects your organization.

PAGE No.
DATE / /

Relationship of traditional RDBMS and MongoDB

RDBMS	MongoDB
Database	Database
Table	Collection
Tuple / Row	Document
Column	field
Table Join	Embadded Document
Primary Key	Primarykey (Default key _id, provided by mongodb itself)
Database Server and Client	
Mysqld / Oracle	mongodb
mysql / sqlplus	mongo

Advantages of MongoDB over RDBMS

- **Schema less:-** MongoDB is document database in which one collection holds different documents. Number of field, content and size of the document can be differ from one different to another

- **Deep query-ability:-** MongoDB support dynamic queries on document using a document-based query language that's nearly as powerful as SQL Tuning

- **Ease of Scale-Out:** MongoDB is easy to scale

PAGE No.
DATE / /

Terminology

- **Payload**:- Application implement the Map and the Reduce function, and form the core of the job
- **Mapper**:- Mapper map the input key/value pair to a set of intermediate key/value pair
- **NamedNode**:- Node that message the Hadoop Distributed File System (HDFS)
- **DataNode**:- Node where data is presented in advance before any processing take place
- **Master Node**:- Node where JobTracker run and which accept job request from client
- **SlaveNode**:- Node where Map and Reduce Program runs
- **JobTracker**:- Schedule job and tracks the assign job to Task tracker
- **Task Tracker**:- Traks the task and report status to JobTracker
- **Job**:- A program is an execution of a mapper and Reducer across a dataset
- **Task**:- An execution of a Mapper or a Reducer on a slice of data
- **Task Attempt**:- A particular instance of an attempt to execute a task on a SlaveNode

PAGE No.
DATE / /

MapReduce :-

MapReduce is processing technique and a program model for distributed computing based on java. The MapReduce algorithm contain two imp. important tasks, namely Map and Reduce. Map takes a set of data and convert it into another set of data, where individual element are broken down into tuple. Secondly reduce task which take the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce task is always performed after the map job

- HDFS follows the master-slave architecture and it has the following element

Namenode :-

it does the following tasks -

Manage the file system namespace

Regulate client's access to files

It also execute file system operation such as renaming, closing, and open file and directories

Datanode :-

These node manages the data storage of their system.

Datanode perform read-write operation on the file systems as per client request

They also perform operation such as block creation, deletion, and replication according to the instruction of namenode.

PAGE No.
DATE / /

*features of 'Hadoop'

Hadoop is Open source
Hadoop Cluster is Highly Scalable
Hadoop provides Fault Tolerance
Hadoop provides High Availability
Hadoop is very Cost-Effective
Hadoop is faster in Data Processing
Hadoop is based on Data Locality concept
Hadoop provides Feasibility