Unit-3(part-2)

BIG DATA STORAGE

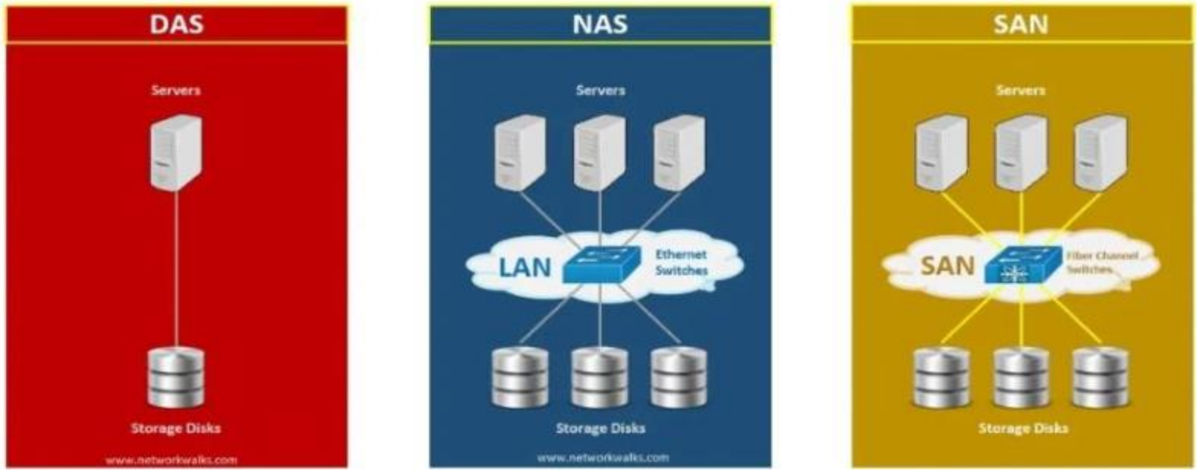
1. Storage system for massive data:

Big data storage is about storing and managing very large amounts of data in a way that's reliable and always available. There are different types of storage systems, but they all have two main parts: the hardware (like servers and storage devices) and the methods used to store the data.

a)Direct Attached Storage (DAS): This is when the storage drives are connected directly to a single server. It's like having an external hard drive connected to your computer. DAS is good for small setups, like personal computers, but it's not great for handling large amounts of data or sharing storage between multiple computers.

b)Network Attached Storage (NAS): NAS is like having a storage device that's connected to a network, so multiple devices can access it. It's good for sharing files across a network, like in an office. NAS devices connect directly to the network, which helps reduce the load on servers.

c)Storage Area Network (SAN): SAN is a network dedicated to storage, with high-speed connections using fiber optics. It's very flexible and allows data to be switched between different devices easily. SANs are great for managing large amounts of data and sharing it between different parts of a system



2. Distributed storage system:

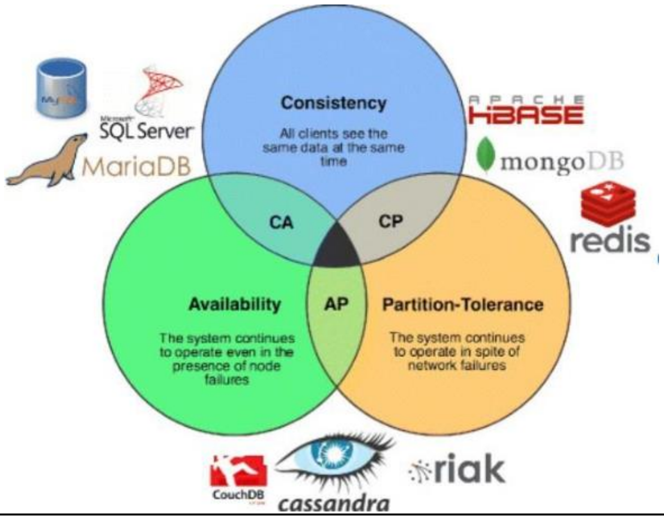
When using a distributed storage system to store a lot of data, we need to think about a few things:

a)Consistency: This means making sure that all copies of the data are the same. In a distributed system, data is split into pieces and stored on different servers. If a server fails, we still want to have the same data available elsewhere.

b)Availability: This is about making sure that the system keeps working even if some servers fail. We want to be able to read from and write to the system even if parts of it are down.
c)Partition Tolerance: This is about the system's ability to keep working even if parts of the network fail. If some servers can't talk to each other because of network problems, the system should still be able to function.

According to Brewer's CAP theory, a distributed system can't have all three of these things at the same time. It can only have two out of the three.

- **CA Systems**: These prioritize consistency and availability but can't handle network failures well. They're like traditional databases that work on a single server.
- **CP Systems**: These prioritize consistency and partition tolerance but might not be available all the time. They keep multiple copies of data to make sure it's consistent and can handle network problems.
- **AP Systems**: These prioritize availability and partition tolerance but might not always have the most up-to-date data. They focus on keeping the system running even if some parts of it are not working perfectly.


MySQL
Microsoft SQL Server
MariaDB
Consistency
All clients see the same data at the same time
APACHE HBASE
mongoDB
redis
CA
CP
Availability
The system continues to operate even in the presence of node failures
AP
Partition-Tolerance
The system continues to operate in spite of network failures
CouchDB
cassandra
riak

3. Storage management for big data:

Considerable research on big data promotes the development of storage mechanisms for big

data. Existing storage mechanisms of big data may be classified into three bottom-up levels:

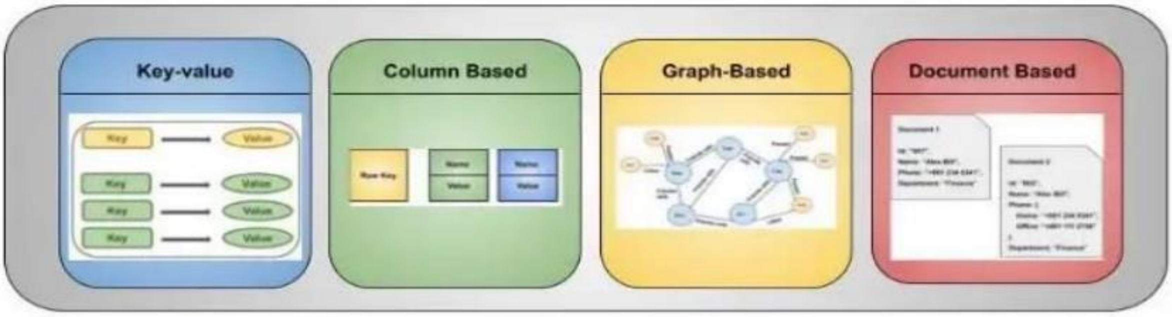
● File systems

● Databases

● Programming models

Database technology:

NoSQL databases are gaining popularity for storing large amounts of data because they're flexible, easy to use, and can handle big data.



a)Key Value Database:

Key-value databases store data as simple pairs of keys and values. These databases are easy to use, scale well, and quickly retrieve data.

Examples:

- **Dynamo**: Used by Amazon, Dynamo stores key-value data and is designed to be highly available and scalable.
- **Voldemort**: Developed for LinkedIn, Voldemort also stores key-value pairs but does not guarantee data consistency.

b)Column Oriented Database:

Column-oriented databases store and process data by columns instead of rows, which can improve performance for certain types of queries.

Examples:

- **BigTable**: Created by Google, BigTable is used for large-scale data processing across many servers.
- **Cassandra**: Developed by Facebook, Cassandra is designed for managing distributed data across multiple servers.

c)Document Database:

Document databases store data in documents, similar to JSON objects, allowing for more complex data structures.

Examples:

- **MongoDB**: MongoDB is an open-source document database that stores data as BSON objects and allows for querying on embedded objects and arrays.
- **SimpleDB**: Amazon's SimpleDB organizes data into domains and copies data to different machines and data centers for safety and performance.
- **CouchDB**: CouchDB stores data as JSON objects and provides a unique identifier for each document.

d)Platform for Nimble Universal Table Storage (PNUTS)

PNUTS is a system used by Yahoo! for web applications. It's designed for large-scale data storage across different locations. PNUTS uses a simple relational data model with tables and properties. It supports blob (binary large object) data types for storing large data chunks and structures within records.

Design Factors for Database Systems:

1. **Data Model**: Different databases use various ways to organize data, such as key-value, column, or document models. For example, PNUTS uses a model that organizes data in rows.
2. **Data Storage**: Some systems keep data mainly in fast-access memory (RAM) with backups on slower disk storage. Others store data primarily on disks, using RAM for faster access. Some systems allow using different storage methods.
3. **Concurrency Control**: Database systems use different methods to manage multiple users accessing data at the same time. These methods include locks, which restrict access, and MVCC, which ensures that reading data is consistent even if it's being changed.
4. **Consistency**: The CAP theorem says a database can't be perfectly consistent, available, and able to handle network partitions all at once. Database systems compromise between strong consistency (every read gets the latest write) and eventual consistency (reads may lag behind writes).

5. **CAP Option**: Cloud databases often replicate data across servers to handle failures. This means trading off between keeping data consistent and making sure it's always available, as per the CAP theorem

Database Programming Models

1. **MapReduce**: MapReduce is a programming model used for large-scale computing across many clusters of commercial PCs. It automatically processes data in parallel. Users define two functions: Map, which processes input data and produces intermediate key-value pairs, and Reduce, which merges the intermediate values related to the same key.
2. **Dryad**: Dryad is a distributed execution engine for processing parallel applications with coarse-grained data. It uses a directed acyclic graph structure, where vertices represent programs and edges represent data channels. Dryad executes operations on vertices in computer clusters and transmits data via data channels.
3. **All-Pairs**: All-Pairs is designed for biometrics, bioinformatics, and data mining applications. It focuses on comparing pairs of elements in two datasets using a given function. It involves four phases: system modeling, input data distribution, batch job management, and result collection.

4.Pregel: Pregel is used by Google for processing large graphs, like those found in network analysis and social networking services. It represents computational tasks as directed graphs with vertices (nodes) and edges. Each vertex has a user-defined value, and edges have a value and a target vertex identifier. Pregel conducts iterative calculations until the algorithm completes.

Unit-4

Traditional Data Analysis refers to using statistical methods to analyze large datasets, both first-hand and second-hand data. Many traditional methods can still be applied to big data analysis. Here are some representative traditional data analysis methods:

- **Cluster Analysis**: Grouping objects based on certain features, without the use of training data. It's an unsupervised method.
- **Factor Analysis**: Grouping related variables into factors, which are used to reveal valuable information in the original data.
- **Correlation Analysis**: Determining correlations among observed phenomena to make forecasts and control.
- **Regression Analysis**: Identifying relationships between one variable and several others, revealing dependencies hidden by randomness.
- **A/B Testing**: Comparing tested groups to improve target variables, also known as bucket testing.
- **Statistical Analysis**: Providing descriptions and inferences for large datasets using statistical theory, which models randomness and uncertainty with Probability Theory. It can summarize and describe datasets (descriptive analysis) and draw conclusions from data (inferential analysis).
- **Data Mining**: Extracting hidden, potentially useful information from massive, incomplete, noisy, fuzzy, and random data.

Big Data Analytic Methods include:

- **Bloom Filter**: Storing Hash values of data in a bit array to conduct loss compression storage.
- **Hashing**: Transforming data into shorter fixed-length numerical values or index values.
- **Indexing**: Effective method to reduce disk reading and writing expenses and improve data manipulation speeds.
- **Trie**: Used for rapid retrieval and word frequency statistics by reducing comparison on character strings.

- **Parallel Computing**: Utilizing multiple computing resources to complete a computation task by decomposing a problem into independent processes.

Architecture for Big Data Analysis:

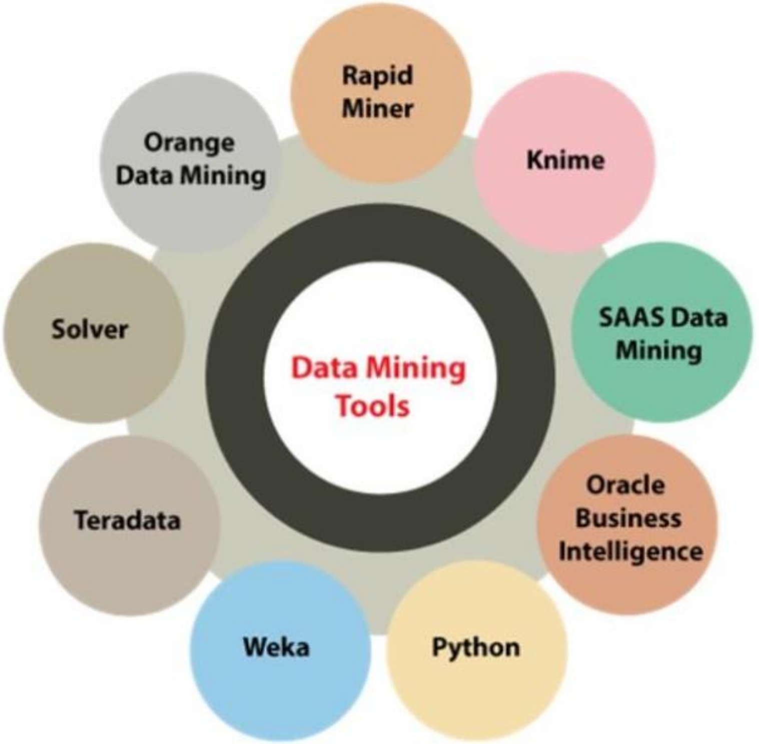
Big data analysis requires different architectures based on data sources, structures, and application needs:

1. **Real-Time vs. Offline Analysis**: Big data analysis can be real-time or offline. Real-time analysis is crucial for industries like Ecommerce and finance. Existing architectures include:
    - Parallel processing clusters using traditional relational databases.
    - Memory-based computing platforms.
2. **Analysis at Different Levels**:
    - **Memory-Level Analysis**: When data volume is within cluster memory limits. Suitable for real-time analysis. MongoDB is a representative architecture.
    - **Business Intelligence (BI) Level Analysis**: For data scales beyond memory but manageable by BI tools supporting TB-level data.
    - **Massive Level Analysis**: When data scale surpasses BI and traditional relational databases' capacities. Typically uses HDFS or Hadoop for storage and MapReduce for analysis, mostly for offline analysis.
3. **Analysis with Different Complexity**:
    - Algorithms vary in time and space complexity based on data and application demands.
    - Distributed algorithms and parallel processing models are used for applications amenable to parallel processing.

Tools for Big Data Mining and Analysis

When it comes to mining and analyzing big data, there are several tools available. Here are the top five widely used ones:

1. **R (30.7%)**: An open-source programming language and software for data analysis and visualization. It's popular because it's free and powerful.
2. **Excel (29.8%)**: Part of Microsoft Office, Excel is great for data processing and statistical analysis. It has plug-ins like Analysis ToolPak for advanced analysis.
3. **RapidMiner (26.7%)**: This open-source software is used for data mining and machine learning. It helps with data processing, modeling, and more.
4. **KNIME (21.8%)**: KNIME is a user-friendly, open-source platform for data integration, processing, and analysis. It offers many functions through plugins.
5. **Weka/Pentaho (14.8%)**: Weka is a free, open-source software for machine learning and data mining. Pentaho is a commercial software based on Java, offering business intelligence functionalities. It integrates Weka's algorithms for analysis.

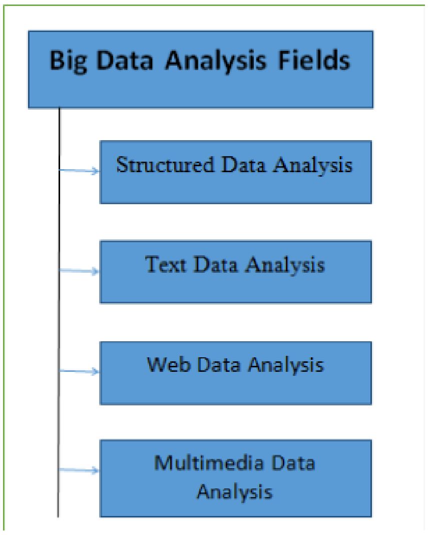


Unit-5

Big Data Applications:

1. **Commercial Applications**: Websites and networks allow businesses to interact with customers online, collecting data like clicks and user behavior for analysis.
2. **Network Applications**: The web hosts diverse data types, spurring the development of tools to analyze this mix of structured and unstructured data.
3. **Scientific Research**: Research fields are using big data to extract insights from large datasets, such as the iPlant dataset, which includes various types of data like specifications, experiments, and observations.

Big Data Analysis Fields



1. **Structured Data Analysis**: This involves analyzing structured data like databases using technologies such as data mining and statistical analysis.

2. **Text Data Analysis**: Text mining extracts useful information from unstructured text like emails and social media using natural language processing.

3. **Web Data Analysis**: Analyzing web content involves retrieving, extracting, and evaluating information from web documents using techniques like content mining and structure mining.

4. **Multimedia Data Analysis**: Analyzing multimedia data like images and videos involves tasks such as audio summarization and video summarization.

5. **Network Data Analysis**: Analyzing data from online social networking services involves tasks like link prediction and content-based analysis.

6. **Mobile Traffic Analysis**: Analyzing mobile data involves dealing with challenges like noise and redundancy, with applications in areas like RFID for inventory management and logistics.

Key Applications of Big Data:

1. **Enterprise Applications**: Big data is used in marketing, supply chain optimization, and finance to predict consumer behavior and improve business strategies.
2. **IoT-Based Big Data**: The Internet of Things (IoT) generates a vast amount of data used for tracking processes and improving efficiency. For example, UPS tracks truck positions to prevent failures.
3. **Online Social Network-Oriented Big Data**: Social media data is used to analyze user preferences and behaviors, helping improve content and user engagement.
4. **Healthcare and Medical Big Data**: Big data is used in healthcare for storing, processing, and analyzing medical data to improve patient care and predict health outcomes.
5. **Collective Intelligence**: Mobile devices with sensors are used for crowd sensing, where users participate in sensing tasks. This data is used for various applications like traffic management and environmental monitoring.
6. **Smart Grid**: The Smart Grid integrates traditional energy networks with technology for optimized energy generation, supply, and consumption, helping in grid planning and renewable energy integration.

Challenges of Big Data in Smart Grid Simplified

1. **Grid Planning**: Analyzing data helps identify regions with high electrical load or frequent power outages. This data helps in upgrading and maintaining the grid.
2. **Interaction Between Power Generation and Consumption**: Traditional power grids are one-directional and cannot adjust generation based on demand, leading to energy waste. Smart meters enable better balance between generation and consumption.
3. **Accessing Intermittent Renewable Energy**: Integrating wind and solar energy into the grid is challenging due to their dependence on weather conditions. Managing these energy sources efficiently is a key challenge.