25110670036

ABD Assignment - 1

1) Explain the factors leading to Big data. List and explain major sources of Big data.

⇒ **Advancement in storage Technology**

* The cost of data storage has fallen sharply and hence organization can afford to keep more data.
* Cloud computing provides scalable infrastructure for handling large datasets.
* Distributed processing frameworks make analyzing data easier.

Rapid Growth of Data Sources

* Social media sites generate data in millions through likes, posts and comments.
* Smartphones, IoT devices, sensors and other devices generate continuous stream of data
* Businesses, governments and healthcare system are moving from paper-based records to digital systems

Internet & Connectivity Growth

* Widespread internet access and smartphones allows billions of people to stay connected
* Data traffic is created by Real-time communication and cloud-based applications.

2) List and explain the characteristics of Big Data

⇒ The characteristics of Big Data are:

(i) **Volume:** this refers to the huge size of data being generated every second.

* The data could be a large piece of

data coming at once or collection of small data over a period of time.

(ii) Velocity: this refers to the speed at which data is created, collected and processed.

* The data in today's world requires real-time or near-real time processing. Hence organizations need tool that can process and react instantly.

(iii) Variety: this refers to the different formats and types of data being generated.

Types of Data under variety:

* Structured Data
* Unstructured Data
* Semi structured Data

(iv) Veracity: this refers to the accuracy, quality and reliability of data.

* Not all data is useful, some may be incomplete, inconsistent or misleading

Types of Veracity:

* High Veracity Data
* Low Veracity Data

3) List and explain the major challenges of Big data system.

⇒

* Scaling up and down Big data according to current demand

→ The volume of data handled or processed in a big data system can vary based on several factors such as current demand,

→ Being able to predict the required capacity required to proces such data is difficult, which may lead to wastage of resources or money.

* Collecting and Integrating Massive and Diverse datasets

→ Data comes from multiple sources and needs to be combined and made consistent for analysis.

→ The data can be of different formats and if the speed of generation is high it can get complex.

* Picking the right NoSQL tools

→ Since traditional relational databases cannot handle big data well, NoSQL tools like MongoDB, Cassandra etc. are used.

→ Each NoSQL is designed for a specific use case.

* Maintaining Data Integrity, Security and Privacy

→ Ensuring data remains accurate, protected from attacks and compliment with privacy laws.

→ Data comes from unverified sources and could pose threats like hacking, ransomware and data leaks.

* Overcoming Big Data talent and resource constraints

→ Big data system require expert in Data engineering, data science, machine learning but people with the required skills are less

→ The tools, also, evolve at a rapid rate, resulting in scarcity of people skilled on the newer version of the tool

4) Discuss the problems faced by traditional database systems.

⇒

* Scalability Issues

→ RDBMS scale vertically, not horizontally. For Big data, vertical scaling expensive and limited.

* Handling Unstructured and Semi-structured data

→ Traditional DBs work best with structured data. Big data includes video's, images, social media posts etc. which is difficult to store in fixed schemas.

* High velocity of Data

→ Data is generated in real-time. Traditional databases cannot handle continuous high speed writes efficiently.

* Complex Data Integration

→ RDBMS struggles to integrate data from multiple heterogeneous sources.

* Limited Parallel Processing

→ Query processing in RDBMS is not optimized for Distributed Parallel computation.

→ Hence frameworks like Hadoop are required to distribute work across many servers.

5) Discuss the required properties for Big Data systems

⇒ When designing Big data systems, certain properties are essential:

(i) Scalability: The system must be able to handle increasing amounts of data without performance degradation.

(ii) Fault tolerance & Reliability:
The system should continue operating even if some hardware or software components fail.

(iii) High Throughput & Low Latency
The system must be able to support high-speed data ingestion, processing & querying.

(iv) Data variety support: should handle structured, semi-structured and unstructured data. Also must integrate multiple sources like databases, IoT sensors

(v) Consistency & Data Integrity
correctness of data and trustworthiness of results should be maintained.

(vi) Security and Privacy

Should implement strong mechanisms for Authentication, authorization and encryption.

6) Explain the different layers of Lambda architecture.

=> The 3 layers (Basic Blocks) of Lambda architecture are:

i) Batch Layer: stores the master copy of the dataset and precomputes batch views on that master dataset (based on Recomputation)

The batch layer should be able to:

→ store an immutable, constantly growing master dataset

→ compute arbitrary functions on that dataset.

(ii) Serving Layer: When batch views are emitted from batch layer, serving layer, which is a specialized distributed database loads the batch view and makes it possible to perform random reads on it.

(iii) Speed Layer: Based on incremental model, this layer handles real-time data streams

→ The goal is to ensure new data is represented in query function as quickly as needed for application requirements.

7) Differentiate b/w recomputation and incremental algorithm.

⇒ Recomputation

The system recomputes results from scratch whenever new data arrives, based on the full dataset.

→ This ensures accuracy and consistency and also makes the implementation easier

→ One disadvantage of following this approach is the high computational cost

→ Also the response time is drastically slow

Incremental algorithm

The system updates the previous results by incorporating only the new incoming data.

→ This approach is faster and can support real time data processing

→ Has low latency and uses resources efficiently

→ On the other hand, if there were errors in the previously computed results, this increases chances in error of the new data

→ It can be complex to design & maintain

8) List the requirements and responsibilites of batch layer.

⇒ The batch layer should:

→ store an immutable, append-only master dataset

→ should be scalable
→ support recomputation
→ High throughput over low latency
→ Efficient storage and processing

* Responsibilities of Batch Layer:
→ Managing master dataset
→ Running batch processing jobs
→ supplying batch views to the serving layer.

9) Explain the requirements of serving Layer in Lambda architecture.

⇒ The serving layer is where the precomputed results from the Batch Layer are stored and exposed for querying.

The Requirements are:

→ Low latency query performance:
* Must allow fast querying on large datasets

→ Scalability:
* should handle increasing query loads and dataset sizes
* support horizontal scaling

→ Fault tolerance and High availability:
* Must ensure data is available even if node fails
* Replication and automatic failover

→ Flexibility of querying:
* should support flexible query types
* Ideally provides API or SQL-like query support.

→ Support for Precomputed views
* Designed to store and serve batch views efficiently
* Must allow quick retrieval without complex computation.

10) With example show how low latency and high throughput can be achieved in serving layer of lambda architecture.

=) Assume that 1 query need to perform on an average 20 seeks.
Let us assume 1 cluster has 100 disks. Each disk performs 500 seek/second.
Total seeks allowed in a single cluster is 50,000
Total queries handled by one cluster per second is 2500.

Solution:
Place "PageView" information for a single URL on the same partition and store it sequentially.
"PageView" information require single seek and scans depending upon time range.
"PageView" information for single URL is present in single server, hence no variance issues.

11) List the requirements and responsibilities of speed layer.

=> Requirements

→ Low Latency Processing:
→ Incremental computation
→ Fault tolerance
→ Integration with Batch & Serving layers.

Responsibilities

→ Generating real times views
→ Process real time data streams
→ Compensate for batch latency
→ Handle out of order data

12) Differentiate between Batch and speed layers.

=>

* Speed layer only looks at recent data, whereas batch layer looks at all data at once.
* To achieve low latency speed layer doesn't look at all the new data at once.
* Batch layer uses recomputational approach whereas Speed layer uses incremental approach.
* Batch layer has strong fault tolerance due to recomputation whereas speed layer is more error prone as it updates the precomputed data