Goals of HDFS

1. Manage Large Datasets
2. High Availability
3. Scalability
4. Cost-effectiveness
5. Fault Tolerance
6. Streaming Data Access
7. Portability
8. Easy Integration

HDFS, or Hadoop Distributed File System, aims to achieve several key goals:

1. Manage Large Datasets

HDFS is designed to store and manage massive datasets efficiently, exceeding terabytes or even petabytes in size. It achieves this by dividing data into fixed-size blocks and distributing them across multiple nodes in a cluster, enabling parallel processing and efficient storage utilization.

2. High Availability

HDFS prioritizes data availability by replicating data blocks across different nodes. This ensures that data remains accessible even if individual nodes fail, minimizing downtime and guaranteeing data integrity.

3. Scalability

HDFS scales horizontally to accommodate growing data volumes. By adding additional nodes to the cluster, HDFS can seamlessly expand its storage capacity and processing power to handle even the largest datasets.

4. Cost-effectiveness

HDFS leverages commodity hardware, eliminating the need for expensive specialized storage solutions. This makes it a cost-effective solution for storing and managing large datasets, especially compared to traditional enterprise storage systems.

5. Fault Tolerance

HDFS is designed to be fault-tolerant, meaning it can continue to operate even when hardware or software failures occur. This is achieved through data replication, automatic failover mechanisms, and distributed architecture that avoids single points of failure.

6. Streaming Data Access

HDFS optimizes for large data sets and streaming data access patterns. It facilitates reading and writing large amounts of data sequentially, making it ideal for big data applications that require continuous data ingestion and processing.

7. Portability

HDFS is designed to be platform-independent, allowing it to run on different operating systems and hardware configurations. This makes it a versatile solution for diverse computing environments.

8. Easy Integration

HDFS integrates seamlessly with other components of the Hadoop ecosystem, such as MapReduce and YARN, enabling smooth data flow and efficient processing of large data sets.

Write down the goals of HDFS ?