studocu

DA Unit-5 notes for One Shot video by brevilearning YT compressed

B.tech (Dr. A.P.J. Abdul Kalam Technical University)

Scan to open on Studocu

Studocu is not sponsored or endorsed by any college or university
Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

Data Analytics

UNIT - 5 [One Shot]

Most important topics:

1. MapReduce, Hadoop, Pig, Hive, HBase, HDFS
2. Sharding, DV and S3.
3. No-SQL databases
4. R-programming (PYQ).
(to check no. is prime or not).

@bravuilearning

MapReduce:

- It is a programming model for processing large datasets with a parallel, distributed algorithm on a cluster.

Key features:

1. Scalability: Handels large volumes of data by distributing processing accross multiple nodes.
2. Fault tolerance: Automatically reruns tasks on different nodes if failure occur, ensuring data is replicated and safe.
3. Ease of use: Simplifies distributed computing by letting developers focus on writing maps and reduce functions.
4. Load balancing: Dynamically balances workload across the cluster for efficient resource use.
5. Flexibility: Process various data formats and sources, including structured and unstructured data.

Workflow:

Step 1. Input Split: Data is split into distributed chunks and across nodes.

Step 2. Map Phase: Each node processes its

This document is available on studocu
Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

chunk, generating intermediate key-value pairs.

Step 3 Shuffle and Sort: Intermediate pairs are shuffled and sorted by key.

Step 4 Reduce Phase: Groups of key-value pairs are processed to produce the final output.

Step 5 Output: Final results are written to distributed file system.

* Applications in big-data analytics:

1. Data processing and transformation
2. Distributed search
3. log analysis
4. Data mining and ML
5. Real-time analytics

@brewilearning

Hadoop:

- It is a open-source framework that facilitates the processing of large datasets across clusters of computers using simple programming models.
- It is designed to scale from single servers to thousands of machines, offering local computation and storage.

* Workflow:

Step 1: Data ingestion: Data is ingested into HDFS from various sources like databases, sensors, and logs.

Step 2: Data storage: HDFS stores the data across the cluster with replication for fault tolerance.

Step 3: Data processing: MapReduce are used to process the data. Map function transforms the input data into intermediate key-value pairs, and Reduce function aggregates the results.

Step 4: Data Output: Processed data is writte

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

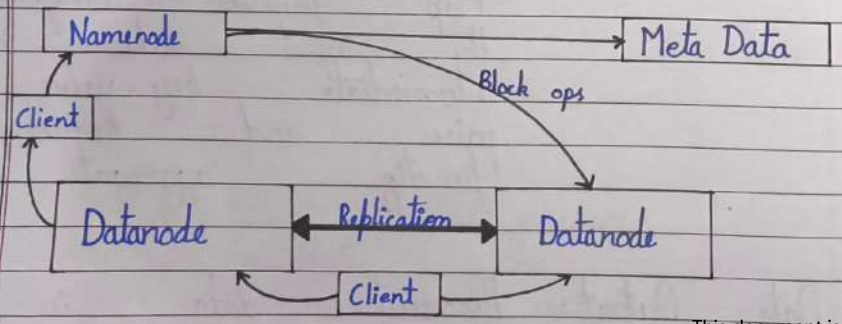
back to HDFS for further analysis.

Core components:

1. HDFS (Hadoop distributed file system):

- It is the primary storage system used by Hadoop applications.
- It is designed to store very large files across multiple machines in a large cluster.
- The architecture of HDFS is highly fault-tolerant and designed to be deployed on low cost hardware.

Architecture



@brevilearning

1. Namenode:

- Acts as master server.
- Manages filesystem namespace and metadata.
- Keeps track of the locations of data blocks.

2. Datanode:
- Acts as slave nodes that stores actual data blocks.
- Performs block creation, deletion, and replication based on the NameNode's instructions.

3. Metadata:

- Information about the data such as the directory.

4. Client:

- Client interact with NameNode and DataNode for metadata, file operations and read/write data respectively.

5. Replication:

- Blocks are replicated across multiple DataNodes to ensure fault tolerance.

This document is available on studocu
Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

6. Block operations: DataNodes handle read/write requests for blocks.

* Data Flow:

1. File write:

- The client requests the NameNode to write a file.
- It writes data to the first DataNode, which then replicates it to other DataNodes.

2. File read:

- Client requests NameNode for file location.
- NameNode responds with the list of DataNodes storing the data blocks.
- The client reads data directly from the DataNodes.

SUBSCRIBE @brwilearning YT

Pig and Hive:

* Pig:

- Apache pig is a high level platform for processing large datasets.
- It uses language called Piglatin, which is designed to handle parallel processing of data.

Key features:

1. Ease of programming:

- Piglatin is simpler to write in Java as compared to MapReduce.

2. Sequencial data flow:

- Piglatin scripts describe a sequence of data transformation.

3. User - friendly:

- Users can create there own functions to process data using Pig.

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

4. Flexibility:

- Pig can handle both structured and unstructured data.

5. Optimization:

- Pig framework optimizes the execution of Piglatin scripts to improve performance.

★ Hive:

- It is a data warehouse infrastructure built on top of Hadoop.
- It facilitates querying and managing large datasets stored in HDFS, using SQL-like language called HiveQL.

Key features:

1. HiveQL: Similar to SQL, makes it easier to users familiar with traditional database to write queries

2. Schema on Read: Hive applies the schema to data at the time of reading rather than writing

3. Storage independence: Efficient to support and handle various storage formats.

4. Scalability: Can handle large datasets and scale out with Hadoop's cluster.

5. Partitioning: Improves query performance by organizing tables into partitions and buckets.

@brewilearning

HBase:

- It is an open-source, distributed, NoSQL database.
- It is duigned for storing and managing large-scale, sparse data.

This document is available on studocu

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

Key features :

1. Distributed architecture :

Runs on a cluster of machines.

2. Columnar storage: Stores data in a columnar format for efficient access.

3. Schema flexibility : Supports dynamic addition and removal of columns without affecting existing data.

4. Strong consistency :

Ensures data consistency with a single row.

5. Low - latency access :

Suitable for real time access to large datasets.

Like and Share !!!

@brevilearning

Sharding :

- It is a database partitioning technique where a large database is divided into smaller, faster, and more manageable pieces called shards.

Need of sharding :

- Example : A college database with 100000 students record.
- Problem : Searching through a large, unshardid database is costly and inefficient.
- Solution : Dividing the database by years reduces the no. of records per shard, enhancing manageability and reducing costs.

Features of sharding :

1. Smaller database : Sharding reduces the size of individual databases.

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

2. Faster performance: Queries and transactions are faster with fewer records to process.

3. Manageablity: Smaller databases are easier to handle and maintain.

4. Complexity: Implementing sharding can be quite complex.

5. Cost - efficiency: Reduces transaction costs significantly.

@bxeuilearning

Data visualization:

- Graphical representation of information and data using visual elements like charts, graphs, and maps to understand trends and patterns.

Techniques:

- Box Plots: Show data spread and compare distributions b/w gro

- Histograms: Display the shape and spread of continuous sample data with bars grouping no. into ranges.

- Heat maps: Use color to represent data values, similar to how bar graphs use hight and width.

- Charts:

  → Pie charts: Circular graphs with slices proportional to numerical values.

  → Line charts: Plot the relationship between two variables.

  → Bar charts: Compare quantities of different categories or groups.

@bxeuilearning

S3: (Amazon's Simple Storage Service)

- Amazon S3 is a scalable object storage service by AWS

This document is available on studocu

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

for secure, durable, and highly available cloud storage.

Key features:

- Scalability: Automatically handles growing amount of data.
- Durability: Highly durable.
- Availability: 99.9% availability with SLA (service level agreement)
- Security: Encryption, access management, and bucket policies.
- Performance: Fast data retrieval and performance optimization.

Uses of S3:

- Backup and restore
- Data archiving
- Content distribution
- Big data analytics
- Application hosting.

@brewilearning

NoSQL databases:

- These are non-relational databases designed to handle large volumes of unstructured, semi-structured, and structured data as well.

Key characteristics:

i. Schema-less: Flexible schema allows variety of data structures.

ii. Scalability: Easily scales horizontally by adding more servers.

iii. Performance: Optimized for high read and write throughput.

iv. Data model flexibility: Supports various data models including key-value, doc., graph etc.

Some popular NoSQL databases:

- MongoDB: Document oriented (e.g. JSON doc.)
- Cassandra: Column-family store.

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

- Redis: key-value, stores in memory.
- Neo4j: Graph database, more complex.
- Amazon DynamoDB: Fully managed key-value and document database, offered by AWS.

★ Advantages:
- High availability
- Flexible data models
- Distributed architecture

★ Disadvantages:
- Consistancy trade-offs
- Complex
- Limited transactions

Subscribe → @brewilearning YT

R-programming:

- It is a environment specific programming language for data analysis, statistical computing, and graphical representation.

Key features:

- Statistical analysis: Extensive library of statistical tools and techniques.
- Data manipulation: Powerful data manipulation capabilities with pre-defined packages.
- Open source: Free and open-source programming language.
- Community: Server a large community.
- Integration: Easy to integrate with other programming languages and big data platforms.
- Flexible: Can be extended via packages available on R-based repositories.

This document is available on studocu
Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)

* Core Components :

- R Language : A programming language focused on statistical analysis and data visualization.
- R environment : Includes R interpreters, standard libraries, and development tools.
- R Studio : A popular integrated development environment (IDE) for R.

R- program to check if a number is prime or not ?

```
check <- function (n)
{
    if (n<=1)
    {
        return (FALSE)
    }
    if (n==2)
    {
        return (TRUE)
    }
    if (n%%2==0)
    {
        return (FALSE)
    }
    for (i in 3:sqrt(n))
    {
        if (n%%i==0)
        {
            return (FALSE)
        }
    }
    return (~~True~~ TRUE)
}
```

Test function :

```
number <- 29
if (check (number))
{
    print (paste (number, " is prime "))
}
else
{
    print (paste (number, " is not prime "))
}
```

Thanks for watching !!

Downloaded by Anuj Singh (anujsingh73.anuj@gmail.com)