SAGAR INSTITUTE OF RESEARCH & TECHNOLOGY, BHOPAL

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Internship Report

On

"HADOOP TECHNOLOGY"

Submitted in the partial fulfilment for the award of the degree of

Bachelor of Technology

In

Computer Science & Engineering

Submitted to

Prof.Amit Sengarz

Submitted by

Student name: Mujahid Saifuzzma

Enrollment No: 0133CS221108

HOD CSE

Dr. Ritu Shrivastava

ACKNOWLEDGEMENT

I take this opportunity to express my sincere gratitude to all those who have been helpful in the successful completion of my internship. I would like to show my greatest appreciation to the highly esteemed and devoted technical staff, supervisors of the “Sage Group”.

I am highly indebted to them for their tremendous support and help during the completion my internship.

My special thanks goes to “ Mr Gourav Neema, for acceptance of my request for providing internship . I would like to thank to all those people who directly or indirectly helped and guided me to complete my training, including the instructors and technical officers of various section.

I am especially thankful Dr. Rajiv Shrivastava, Director (SIRT, Bhopal) for his kind co- operation and rendering me all possible facilities.

I express my thanks to Dr. Ritu Shrivastava, HOD, Computer Science & Engineering department SIRT Bhopal for kind support.

I am thankful to all staff members of the CSE department and my friends for their timely help co-operation and suggestion during my work. Lastly but not the least, I must express thanks to my family, without their moral support it was impossible for me to complete this work.

Mujahid Saifuzzma

TABLE OF CONTENT

OBJECTIVE OF INTERNSHIP

Hadoop is an open-source framework that allows for the distributed storage and processing of large datasets across clusters of computers. It is designed to handle massive amounts of data in a scalable, fault-tolerant, and cost-effective way, making it a critical technology in big data ecosystems Apart from the core components, Hadoop also has an extensive ecosystem of related tools and technologies:

Objective

1. Store Big Data
2. Process Large Datasets
3. Fault Tolerance
4. Scalability
5. Cost-Effective
6. Handle Different Types of Data

Mission

- **Create a stimulating learning environment**: Foster intellectual growth, creativity, and entrepreneurship
- **Contribute to society's socio-economic development**: Become a center of excellence in education, research, and innovation
- **Prepare students for the challenges of globalization**: Support students' overall personality development and prepare them for a successful career

Values

- Honesty
- Quality
- Client Focus

INTRODUCTION

Hadoop is an open-source framework designed for storing and processing large datasets in a distributed computing environment. It is specifically built to handle big data, which cannot be processed by traditional data management tools due to its volume, variety, and velocity. Hadoop enables organizations to process vast amounts of data across a cluster of computers using simple programming models, ensuring scalability, fault tolerance, and cost-effectiveness.

Hadoop is primarily composed of two main components:

1. **Hadoop Distributed File System (HDFS)**: This is the storage layer that splits large datasets into smaller blocks and distributes them across multiple machines, ensuring data redundancy and high availability.
2. **MapReduce**: This is the computational layer that processes data in parallel across a distributed system by dividing tasks into smaller sub-tasks, making it efficient for large-scale data processing.

In addition to these core components, Hadoop has an ecosystem of related tools and technologies, such as Apache Hive, Apache HBase, Apache Pig, and YARN, that further extend its capabilities for data storage, management, and analysis.

Overall, Hadoop has revolutionized big data analytics by providing a scalable, fault-tolerant, and cost-effective solution for businesses to store and process large volumes of data.

Why HADOOP Internships?

Hadoop is an open-source framework that allows for the distributed storage and processing of large datasets across clusters of computers. It is designed to handle massive amounts of data in a scalable, fault-tolerant, and cost-effective way, making it a critical technology in big data ecosystems.

Key Components of Hadoop:

1. **Hadoop Distributed File System (HDFS)**:
    a. HDFS is the storage layer of Hadoop, which splits large data files into smaller blocks (typically 128MB or 256MB) and stores them across multiple machines in the cluster.
    b. It ensures fault tolerance by replicating each block multiple times on different machines.
2. **MapReduce**:
    a. MapReduce is the computational layer of Hadoop that processes data in parallel across multiple nodes.
    b. The Map phase breaks down the task into smaller sub-tasks, and the Reduce phase aggregates the results.
    c. This makes Hadoop highly effective for processing large datasets.
3. **YARN (Yet Another Resource Negotiator)**:
    a. YARN is the resource management layer of Hadoop, which manages and schedules the resources across the cluster.
    b. It allows multiple applications (like MapReduce, Apache Spark, etc.) to run on a Hadoop cluster and efficiently utilize the cluster's resources.

4. **Hadoop Common**:
    a. This includes libraries and utilities that support the other Hadoop modules. It provides essential services such as file management, communication, and configuration management.

Hadoop Ecosystem:

Apart from the core components, Hadoop also has an extensive ecosystem of related tools and technologies:

- **Hive**: A data warehouse system that provides a SQL-like interface for querying and managing large datasets.
- **Pig**: A scripting language that simplifies the process of working with MapReduce by using higher-level abstractions.
- **HBase**: A NoSQL database built on top of HDFS for real-time data access and storage.
- **ZooKeeper**: A centralized service for maintaining configuration information, naming, and providing synchronization across distributed applications.
- **Spark**: While not part of the Hadoop core, Apache Spark is often used in conjunction with Hadoop for in-memory processing of big data.

In summary, Hadoop is a powerful tool for processing large-scale data in a distributed environment, making it essential for many big data applications.

The main objectives of Hadoop in simple points are:

1. **Store Big Data**: Hadoop allows the storage of massive amounts of data in a distributed way across many machines.
2. **Process Large Datasets**: It enables the processing of vast datasets in parallel, making tasks like data analysis faster and more efficient.
3. **Fault Tolerance**: Hadoop ensures that data is replicated across multiple nodes, so even if one machine fails, data is not lost, and processing can continue.

4. **Scalability**: Hadoop can scale easily by adding more machines to the system, allowing it to handle increasing data loads over time.
5. **Cost-Effective**: It uses commodity hardware, making it a cheaper alternative to traditional data storage and processing systems.
6. **Handle Different Types of Data**: Hadoop can manage structured, semi-structured, and unstructured data, making it versatile for various data types.
7. **Distributed Computing**: It divides tasks into smaller sub-tasks and processes them across many machines, speeding up computations.

These objectives make Hadoop a key technology for managing and analyzing large-scale data in modern businesses.

You said:

only headings

ChatGPT said:

ChatGPT

1. **Store Big Data**
2. **Process Large Datasets**
3. **Fault Tolerance**
4. **Scalability**
5. **Cost-Effective**
6. **Handle Different Types of Data**
7. **Distributed Computing**

Hadoop is an open-source framework designed for storing and processing large datasets in a distributed computing environment. It is specifically built to handle big data, which cannot be processed by traditional data management tools due to its volume, variety, and velocity. Hadoop enables organizations to process vast amounts of data across a cluster of computers using simple programming models, ensuring scalability, fault tolerance, and cost-effectiveness.

Hadoop is primarily composed of two main components:

1. **Hadoop Distributed File System (HDFS)**: This is the storage layer that splits large datasets into smaller blocks and distributes them across multiple machines, ensuring data redundancy and high availability.
2. **MapReduce**: This is the computational layer that processes data in parallel across a distributed system by dividing tasks into smaller sub-tasks, making it efficient for large-scale data processing.

In addition to these core components, Hadoop has an ecosystem of related tools and technologies, such as Apache Hive, Apache HBase, Apache Pig, and YARN, that further extend its capabilities for data storage, management, and analysis.

Overall, Hadoop has revolutionized big data analytics by providing a scalable, fault-tolerant, and cost-effective solution for businesses to store and process large volumes of data.

1. Hands-on Experience with Big Data Tools

- Hadoop is a widely used framework in the industry for processing and managing large datasets. An internship provides practical experience with the tools and technologies like **HDFS, MapReduce, YARN, Hive, Pig, and HBase**, which are essential for big data processing.

2. Industry-Relevant Skills

- As big data becomes increasingly important across industries, gaining expertise in Hadoop gives you a competitive edge in the job market. It is a crucial skill for roles in **data engineering, analytics, and machine learning**.

3. Understanding of Distributed Systems

- Hadoop is built on distributed computing principles. By working with Hadoop, interns gain insights into how large-scale, fault-tolerant, and distributed systems work, a key concept in modern data management.

4. Exposure to Real-world Data Challenges

- Interns get the opportunity to work with real datasets, facing challenges such as scalability, data storage, processing speed, and fault tolerance. This exposure helps interns learn how to solve complex problems that arise in big data environments.

5. Career Opportunities

- Many companies are actively looking for professionals with big data expertise. Completing a Hadoop internship can increase your chances of securing a full-time job or a permanent role in data-related fields.

6. Collaboration and Networking

- Internships often offer the chance to collaborate with experienced professionals, providing valuable networking opportunities and mentorship. Building relationships in the industry can open doors for future career growth.

7. Learning New Tools and Technologies

- Hadoop is part of a broader **big data ecosystem** that includes tools like **Apache Spark, Kafka, and Flume**. An internship allows you to learn these complementary technologies, increasing your versatility as a data professional.

8. Exposure to Industry Practices

- Working in an internship allows you to learn about best practices, industry standards, and methodologies used in big data processing, which helps you develop professional expertise.

9. Boosts Resume

- An internship with Hadoop experience adds significant value to your resume, demonstrating to potential employers that you have practical skills in handling large-scale data processing, which is in high demand.

10. Problem-Solving and Innovation

- Working with Hadoop often involves troubleshooting, performance optimization, and solving data-related challenges. Interns can develop critical thinking and problem-solving skills that are highly valued in any tech-oriented career.

In summary, a Hadoop internship provides **valuable hands-on experience**, improves your understanding of **big data technologies**, and increases your employability in the growing field of **data engineering** and **data science**.

Key Skills and Knowledge:

To excel in a CLOUD computing internship, you should have a solid understanding of:

- Basic Understanding of Computer Networks
- Operating Systems Knowledge
- Virtualization Concepts
- Basic Programming Skills
- Understanding Databases
- Cloud Service Model

INTERNSHIP EXPERIENCE

Internship Experience:

1. **Training on Hadoop Tools**: Learn to work with Hadoop components like HDFS, MapReduce, and Hive.
2. **Real-World Projects**: Work on data processing tasks, such as creating MapReduce jobs and managing large datasets.
3. **Collaboration**: Partner with senior engineers and data scientists to solve big data challenges.
4. **Problem-Solving**: Debug issues, optimize performance, and ensure fault tolerance in Hadoop jobs.
5. **Feedback and Mentorship**: Receive guidance and feedback to improve skills and prepare for future career opportunities.

Major Accomplishments

- **Successfully Implementing a Hadoop Job**: Completing a MapReduce job or data processing task that helps optimize workflows or solve real-world data problems.
- **Improving Data Processing Efficiency**: Identifying performance bottlenecks in Hadoop jobs and implementing optimizations to improve speed or resource usage.
- **Contributing to a Real-world Project**: Playing an integral role in a big data project, such as building a data pipeline or processing large datasets for analysis.
- **Gaining Proficiency with Hadoop Ecosystem Tools**: Mastering tools like Hive, Pig, or HBase, and applying them effectively in project work.

- **Solving Complex Data Challenges**: Successfully troubleshooting and resolving data-related issues, such as data skew or replication problems in HDFS.

Day **1: Introduction to Hadoop**

- **Understanding Big Data**: Characteristics of big data (volume, velocity, variety).
- **Introduction to Hadoop**: What is Hadoop, why it's important for big data.
- **Core Components of Hadoop**:
  - HDFS (Hadoop Distributed File System)
  - MapReduce
  - YARN (Yet Another Resource Negotiator)

Day **2: Cloud Computing Architecture**

- **Understanding MapReduce**: How MapReduce works (Map phase and Reduce phase).
- **MapReduce Job Structure**: Writing a simple MapReduce job in Java.
- **Input and Output Formats**: Working with different input/output formats in MapReduce
- **Goal**: Learn how cloud systems are structured and how they interact.
-

DAY 3: Hadoop Ecosystem and Advanced Tools

- **Introduction to Hive**: A data warehouse system for SQL-like querying of data stored in HDFS.
- **Introduction to Pig**: A high-level platform for creating MapReduce programs using a scripting language.
- **Introduction to HBase**: A NoSQL database built on top of HDFS for real-time data access.

- **Overview of Other Hadoop Ecosystem Tools**: Flume, Sqoop, Oozie, and ZooKeeper.
- **Running Queries in Hive**: Writing simple SQL-like queries in Hive for data analysis.
- **Using Pig**: Writing simple Pig Latin scripts to process large datasets.

BRIEF DESCRIPTION OF PROJECTS

Overview: Processing and Analyzing Web Server Logs with Hadoop

The goal of this project is to process large web server log files using Hadoop's MapReduce framework to extract useful information such as the most visited pages, the number of unique visitors, and the peak traffic times

My Role and Contributions

- **Setting Up the Hadoop Environment** You were responsible for setting up the **Hadoop cluster** (single-node or multi-node), configuring **HDFS**, and ensuring the environment was ready for data processing.

- **Data Ingestion**:You handled the task of **loading web server log files** into **HDFS** for processing, ensuring that data was correctly stored and accessible for
- **MapReduce Programming**:
- You wrote and optimized the **MapReduce programs** to process the web log files. This included writing the **Mapper** to extract necessary fields from the logs (e.g., URL, IP address, timestamp) and the **Reducer** to aggregate data (e.g., counting visits, identifying unique visitors).
- **Data Analysis**:
- You analyzed the output from MapReduce jobs, identifying patterns like the **most visited pages**, **unique visitors**, and **traffic trends** during different times of the day.
- **Performance Optimization**:
- You worked on improving the **performance** of the MapReduce jobs, troubleshooting issues like slow job execution, and ensuring that the processing was done efficiently.

Challenges and how they were overcome

- **Challenge: Handling Large Log Files**
- **Issue**: The web server log files were too large to process efficiently on a single machine, which could result in slower processing and memory issues.
- **Solution**: By using **HDFS**, the log files were distributed across multiple nodes in the Hadoop cluster, allowing parallel processing. Additionally, the log files were split into smaller blocks, making it easier to manage and process the data without overwhelming a single node.
- **Challenge: Data Quality and Inconsistent Log Formats**

- **Issue**: The web server logs had inconsistent formats, with some logs missing fields or having malformed entries, making it difficult to process them without errors.
- **Solution**: Implemented **data cleaning** steps in the **Map phase** of the MapReduce program to filter out invalid or incomplete log entries. Regular expressions were used to ensure the correct extraction of relevant fields like IP addresses, timestamps, and URLs.
- **Challenge: Slow Performance of MapReduce Jobs**
- **Issue**: The MapReduce jobs were running slower than expected, especially with large log datasets, leading to delays in analysis.
- **Solution**: Optimized the **MapReduce logic** by minimizing data shuffling between the Mapper and Reducer phases. Improved performance by **combining smaller tasks** and leveraging **combiner functions** to reduce the amount of intermediate data sent between nodes. Also, optimized the cluster's resource management by adjusting the configuration settings of **YARN**.
- **Challenge: Handling Data Skew in MapReduce**
- **Issue**: Some log entries, such as certain URLs, had disproportionately high numbers of occurrences, causing data skew and inefficient resource utilization in the **Reducer** phase.
- **Solution**: Implemented **custom partitioners** to distribute the data more evenly across reducers and prevent some reducers from being overwhelmed. This helped balance the load and improved overall job performance.

Learning and Skills Gained

.

Technical Skills

- **Hadoop Ecosystem Knowledge**:
  Gained in-depth knowledge of **Hadoop components** such as **HDFS**, **MapReduce**, and **YARN**. Learned how data is stored in a distributed environment and how to process it using parallel computing.
- **MapReduce Programming**:
  Developed proficiency in writing **MapReduce programs** for data processing. Gained hands-on experience with the **Map** and **Reduce** functions and learned how to optimize them for better performance.
- **Data Ingestion and Management**:
  Learned how to load, store, and manage large datasets in **HDFS**. Gained experience using Hadoop commands to interact with HDFS and manipulate files.
- **Performance Optimization**:
  Gained skills in optimizing **MapReduce jobs** by minimizing data shuffling, handling data skew, and using techniques like **combiner functions** and custom **partitioners** to improve job efficiency.
- **Data Analysis and Querying**:

Learned how to analyze data stored in HDFS and process it with tools like **Hive** and **Pig**. Acquired the ability to write SQL-like queries in Hive and create scripts in Pig to simplify complex data processing tasks.

- **Troubleshooting and Debugging**:
  Gained experience in debugging MapReduce jobs, analyzing logs, and using **Hadoop monitoring tools** (like **ResourceManager UI** and **JobTracker UI**) to track job status and resolve issues.
- **Understanding of Distributed Systems**:
- Learned the fundamentals of **distributed computing** and how **Hadoop's distributed architecture** works to store and process data across multiple nodes, ensuring fault tolerance and scalability.
- **Collaboration and Teamwork**:
  Worked closely with team members, contributing to brainstorming sessions, solving problems collaboratively, and enhancing communication skills for project success.
- **Problem-Solving and Critical Thinking**:
  Overcame challenges like handling large datasets, optimizing job performance, and ensuring data quality, which helped improve problem-solving and analytical thinking skills.
- **Real-world Data Processing Experience**:
  Gained practical experience in working with real-world big data problems, enhancing skills that are directly applicable to roles in **data engineering**, **data analysis**, and **big data solutions.**

REFLECTION AND EVALUATION

Personal Growth

•Confidence Boost: The hands-on experience and successful completion of complex projects significantly boosted my confidence in my abilities

as a developer. I now feel more prepared to take on new challenges and responsibilities in the tech industry.

•Adaptability: Working on diverse projects and overcoming various challenges helped me develop a flexible mindset. I learned to adapt quickly to new technologies and methodologies, which is crucial in the ever-evolving tech landscape.

Technical Skills

•Enhanced Proficiency: My proficiency in the Hadoop improved substantially. I gained practical knowledge of hadoop

•Problem-Solving Abilities: Tackling real-world coding challenges sharpened my problem-solving skills. I developed a methodical approach to debugging and finding efficient solutions, which will be invaluable in my future career.

Soft Skills

•Teamwork and Collaboration: Collaborating with peers and mentors taught me the importance of effective communication and teamwork. I learned to give and receive constructive feedback, which is essential for personal and professional growth.

•Time Management: Managing multiple tasks and projects within deadlines improved my organizational skills. I learned to prioritize tasks effectively and stay focused on achieving goals.

Overall Evaluation

Strengths:

- **Excellent for beginners** to learn the basics of Hadoop and web hosting.
- Low cost and fast setup, with potential for growth and further learning.

Areas for Improvement:

- **Advanced Performance Optimization**:
- While basic optimizations were done, there's room for improvement in **fine-tuning MapReduce jobs** further, such as understanding and applying **advanced tuning parameters** for memory management, processing efficiency, and garbage collection.
- **Mastering Hadoop Ecosystem Tools**:
- Gaining deeper expertise in other Hadoop ecosystem tools like **Apache Spark**, **Apache Flume**, or **Apache HBase** could improve the ability to handle different types of data processing tasks, real-time data ingestion, and storage solutions.

CONCLUSION

In conclusion, the Hadoop project provided valuable hands-on experience in managing and processing large-scale data, enhancing skills in MapReduce programming, Hadoop ecosystem tools, and distributed systems. Despite facing challenges such as performance optimization, data

quality issues, and debugging, the project was a great opportunity to learn advanced data processing techniques, troubleshoot complex problems, and collaborate with team members. The experience also highlighted areas for growth, such as mastering performance tuning, expanding knowledge of additional Hadoop tools, and improving scalability and security practices. Overall, the project was a significant step in developing the expertise needed for big data solutions and data engineering.