The Operating System for Big Data

Unit II Syllabus

- Data Product, Building Data Products at Scale with Hadoop, Data Science Pipeline and Hadoop Ecosystem

  Ref: (Chapter 1 - Data Analytics with Hadoop By Benjamin Bengfort & Jenny Kim)

- Operating System for Big Data: Concepts, Hadoop Architecture, Working with Distributed file system, Working with Distributed Computation

  Ref: (Chapter 2 - Data Analytics with Hadoop By Benjamin Bengfort & Jenny Kim)

- Framework for Python and Hadoop Streaming, Hadoop Streaming, MapReduce with Python, Advanced MapReduce.

  Ref: (Chapter 3 - Data Analytics with Hadoop By Benjamin Bengfort & Jenny Kim)

- In-Memory Computing with Spark, Spark Basics, Interactive Spark with PySpark, Writing Spark Applications

  Ref: (Chapter 4 - Data Analytics with Hadoop By Benjamin Bengfort & Jenny Kim)

Today's Topics

Operating System for Big Data

- Concepts
- Hadoop Architecture
- Working with Distributed file system
- Working with Distributed Computation

Ref: (Chapter 2 - Data Analytics with Hadoop By Benjamin Bengfort & Jenny Kim)

The Operating System for Big Data

Chapter 2

Data Teams - Members

Data teams are usually structured as small teams of five to seven members who employ a hypothesis-driven workflow using agile methodologies.

Data teams therefore are composed of members who fit into three broad categories:

Data Engineers	Responsible for Practical aspects of wiring and mechanics of data relating to software & Computing Resources
Data Modelers	Focus on Exploration and Explanation of data & creating inferential or Predictive data products
Subject Experts	Provide Domain knowledge to problem solving both in terms of Process & Application

Data Teams - Members…

- Due to the technical nature of distributed computing Data teams that utilize Hadoop tend to place a primary emphasis on the data engineering aspects of data science.
- Big datasets lend themselves to aggregation-based approaches (over instance-based approaches) and a large toolset for distributed machine learning and statistical analyses exists already.
- For this reason, most literature about Hadoop is targeted at software developers, who usually specialize in Java—the software language the Hadoop API is written in.
- Moreover, those training materials tend to focus on the architectural aspects of Hadoop as those aspects demonstrate the fundamental innovations that have made Hadoop so successful at tasks like large-scale machine learning

Basic Concepts…

Hadoop distributes an analytical computation that involves a massive dataset to many machines that each simultaneously operate on their own individual chunk of data

- Using Distributed computing is a technical challenge,
  - requiring distributed algorithms to be developed,
  - machines in the cluster to be managed, and
  - networking and architecture details to be solved

Basic Concepts…

- A distributed system must meet the following requirements:
  - Fault Tolerance
    - If a component fails, it should not result in the failure of the entire system. The system should gracefully degrade into a lower performing state. If a failed com- ponent recovers, it should be able to rejoin the system.
  - Recoverability
    - In the event of failure, no data should be lost.
  - Consistency
    - The failure of one job or task should not affect the final result
  - Scalability
    - Adding load (more data, more computation) leads to a decline in performance, not failure; increasing resources should result in a proportional increase in capacity

Basic Concepts…

- Hadoop addresses these requirements through several abstract concepts as follows:
  - Data is distributed immediately when added to the cluster and stored on multiple nodes.

    Nodes prefer to process data that is stored locally in order to minimize traffic across the network.

  - Data is stored in blocks of a fixed size (usually 128 MB) and each block is duplicated multiple times across the system to provide redundancy and data safety.

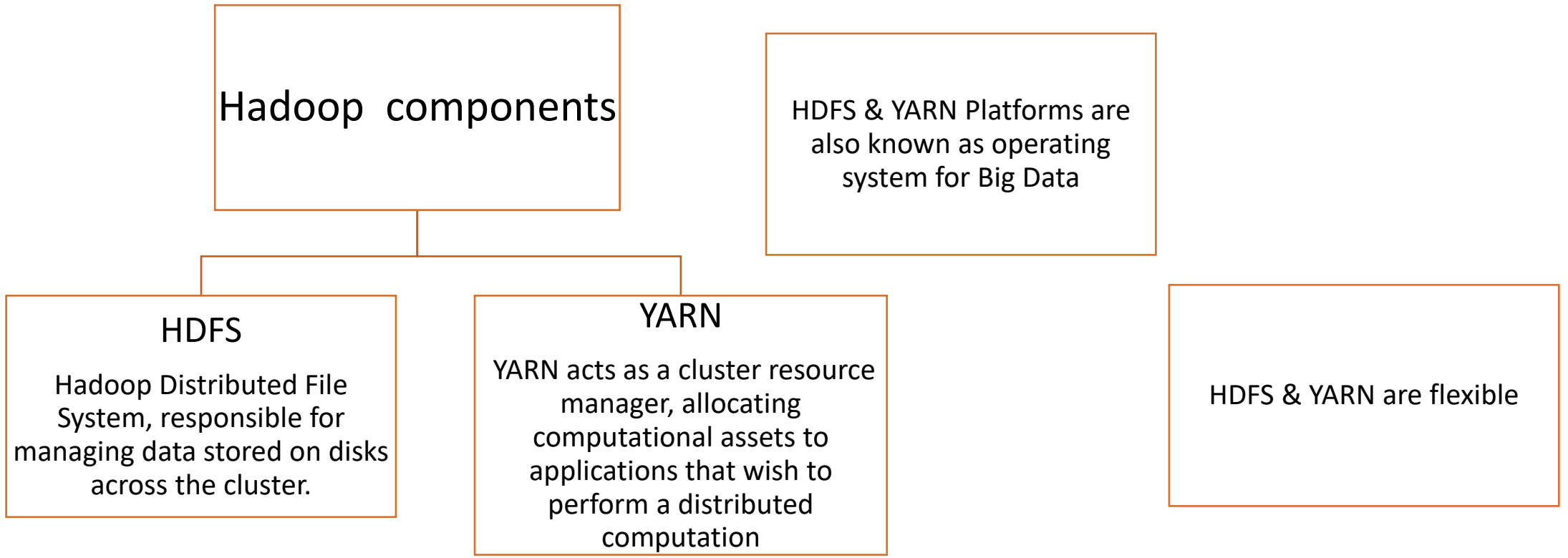
  - A computation is usually referred to as a job; jobs are broken into tasks where each individual node performs the task on a single block of data.

  - Jobs are written at a high level without concern for network programming, time, or low-level infrastructure, allowing developers to focus on the data and computation rather than distributed programming details.

Basic Concepts…

- The amount of network traffic between nodes should be minimized transparently by the system.
- Each task should be independent and nodes should not have to communicate with each other during processing to ensure that there are no inter process dependencies that could lead to deadlock.
- Jobs are fault tolerant usually through task redundancy, such that if a single node or task fails, the final computation is not incorrect or incomplete.
- Master programs allocate work to worker nodes such that many worker nodes can operate in parallel, each on their own portion of the larger dataset.

Hadoop Architecture



Hadoop Architecture…

- HDFS and YARN work in concert to minimize the amount of network traffic in the cluster primarily by ensuring that data is local to the required computation.
- Duplication of both data and tasks ensures fault tolerance, recoverability, and consistency.
- Moreover, the cluster is centrally managed to provide scalability and to abstract low level clustering programming details.
- Together, HDFS and YARN are a platform upon which big data applications are built; perhaps more than just a platform, they provide an operating system for big data.

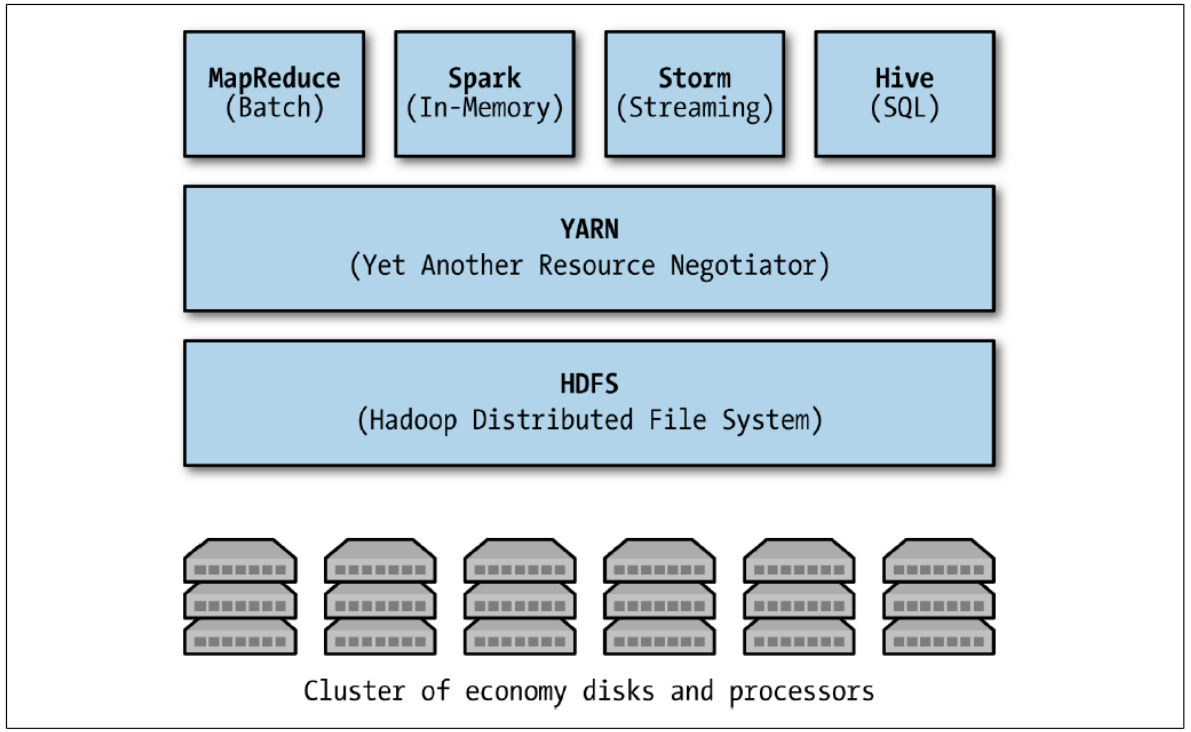


Figure 2-1. Hadoop is made up of HDFS and YARN

Hadoop Architecture…

- Like any good operating system, HDFS and YARN are flexible.
- Other data storage systems aside from HDFS can be integrated into the Hadoop framework such as Amazon S3 or Cassandra.
- Alternatively, data storage systems can be built directly on top of HDFS to provide more features than a simple file system.
- For example, HBase is a columnar data store built on top of HDFS and is one the most advanced analytical applications that leverage distributed storage.
- In earlier versions of Hadoop, applications that wanted to leverage distributed computing on a Hadoop cluster had to translate user-level implementations into MapReduce jobs.
- However, YARN now allows richer abstractions of the cluster utility, making new data processing applications for machine learning, graph analysis, SQL-like querying of data, or even streaming data services faster and more easily implemented.
- As a result, a rich eco- system of tools and technologies has been built up around Hadoop, specifically on top of YARN and HDFS

A Hadoop Cluster…

Hadoop is the name of the software that runs on a cluster—namely, the distributed file system, HDFS, and the cluster resource manager, YARN, which are collectively composed of six types of background services running on a group of machines.

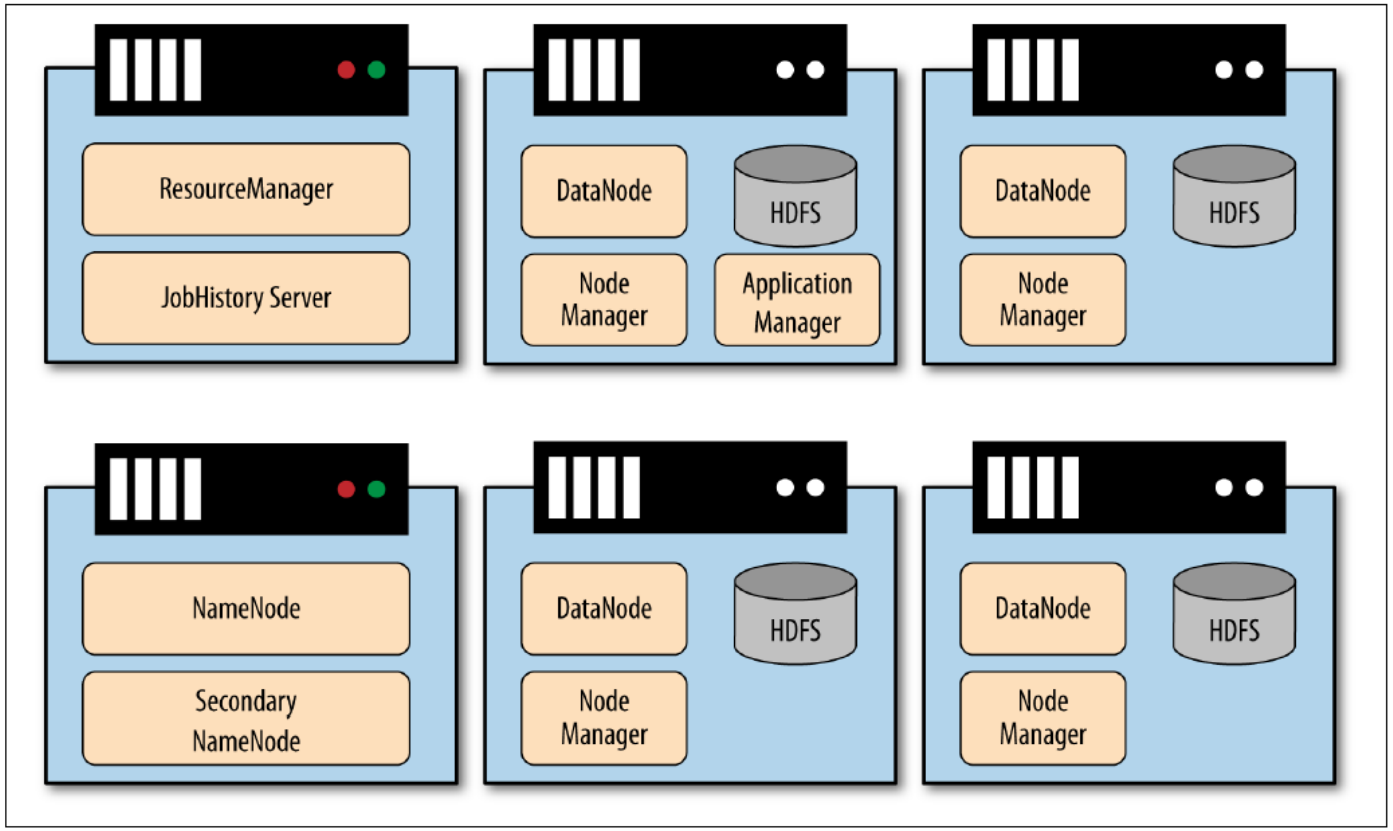


Figure 2-2. A small Hadoop cluster with two master nodes and four workers nodes that implements all six primary Hadoop services

A Hadoop Cluster…

- HDFS and YARN expose an application programming interface (API) that abstracts developers from low-level cluster administration details.
- A set of machines that is running HDFS and YARN is known as a cluster, and the individual machines are called nodes.

- A cluster can have a single node, or many thou- sands of nodes, but all clusters scale horizontally, meaning as you add more nodes, the cluster increases in both capacity and performance in a linear fashion

A Hadoop Cluster…

- YARN and HDFS are implemented by several daemon processes—that is, software that runs in the background and does not require user input.
- Hadoop processes are services, meaning they run all the time on a cluster node and accept input and deliver output through the network, similar to how an HTTP server works.

- Each of these processes runs inside of its own Java Virtual Machine (JVM) so each daemon has its own system resource allocation and is managed independently by the operating system.

A Hadoop Cluster…

- Each node in the cluster is identified by the type of process or processes that it runs:
- Master nodes
  - These nodes run coordinating services for Hadoop workers and are usually the entry points for user access to the cluster. Without masters, coordination would fall apart, and distributed storage or computation would not be possible.
- Worker nodes
  - These nodes are the majority of the computers in the cluster. Worker nodes run services that accept tasks from master nodes—either to store or retrieve data or to run a particular application. A distributed computation is run by parallelizing the analysis across worker nodes.

A Hadoop Cluster…

- Both HDFS and YARN have multiple master services responsible for coordinating worker services that run on each worker node.
- Worker nodes implement both the HDFS and YARN worker services.

- For HDFS, the master and worker services are as follows:
  - NameNode (Master)
  - Secondary NameNode (Master)
  - DataNode (Worker)

A Hadoop Cluster…

- NameNode (Master)
  - Stores the directory tree of the file system, file metadata, and the locations of each file in the cluster. Clients wanting to access HDFS must first locate the appropri- ate storage nodes by requesting information from the NameNode.
- Secondary NameNode (Master)
  - Performs housekeeping tasks and checkpointing on behalf of the NameNode. Despite its name, it is not a backup NameNode.
- DataNode (Worker)
  - Stores and manages HDFS blocks on the local disk. Reports health and status of individual data stores back to the NameNode.

A Hadoop Cluster…

- At a high level, when data is accessed from HDFS, a client application must first make a request to the NameNode to locate the data on disk.
- The NameNode will reply with a list of DataNodes that store the data, and the client must then directly request each block of data from the DataNode.
- Note that the NameNode does not store data, nor does it pass data from DataNode to client, instead acting like a traffic cop, pointing clients to the correct DataNodes.

A Hadoop Cluster…

- Similarly, YARN has multiple master services and a worker service as follows:
  - ResourceManager (Master)
  - ApplicationMaster (Master)
  - NodeManager (Worker)

A Hadoop Cluster…

- ResourceManager (Master)
  - Allocates and monitors available cluster resources (e.g., physical assets like mem- ory and processor cores) to applications as well as handling scheduling of jobs on the cluster.
- ApplicationMaster (Master)
  - Coordinates a particular application being run on the cluster as scheduled by the ResourceManager.
- NodeManager (Worker)
  - Runs and manages processing tasks on an individual node as well as reports the health and status of tasks as they’re running

A Hadoop Cluster…

- Similar to how HDFS works, clients that wish to execute a job must first request resources from the Resource Manager, which assigns an application-specific Application Master for the duration of the job.

- The ApplicationMaster tracks the execution of the job, while the ResourceManager tracks the status of the nodes, and each individual NodeManager creates containers and executes tasks within them.

- Note that there may be other processes running on the Hadoop cluster as well—for example, JobHistory servers or ZooKeeper coordinators, but these services are the primary software running in a Hadoop cluster.

Single Node Cluster

- Finally, one other type of cluster is important to note: a single node cluster.
- In “pseudo-distributed mode” a single machine runs all Hadoop daemons as though it were part of a cluster, but network traffic occurs through the local loopback network interface.
- In this mode, the benefits of a distributed architecture aren’t realized, but it is the perfect setup to develop on without having to worry about administering several machines.

- Hadoop developers typically work in a pseudo-distributed environment, usually inside of a virtual machine to which they connect via SSH.

- Cloudera, Hortonworks, and other popular distributions of Hadoop provide pre-built virtual machine images that you can download and get started with right away

Thank You….

Revise the topics from Syllabus References…

Fill Your Attendance Form….!