Introduction to Hadoop

Background on Big Data Problem

- Big data challenges arises from
  - * volume
  - * variety
  - * velocity
- Traditional solutions become impractical due to
  - * storage limitation
  - * processing capabilities
  - * expensive vertical scaling

Origin of Hadoop :

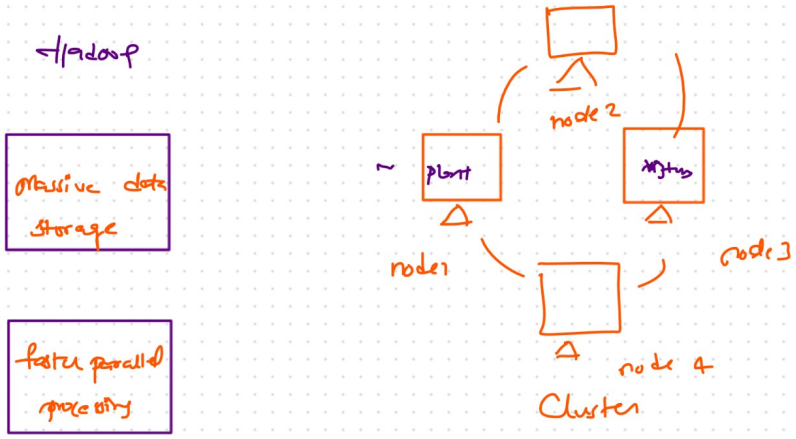
- * in early 2000, Google developed 2 key systems to address web-search data challenges
  - * GFS (Google file storage) — for data storage
  - * GMR (Google mapReduce) — for data processing

Doug Cutting & Mike Cafarella created Hadoop as an open-source implementation of these google concepts

What is Hadoop?

- * An open-source framework designed to handle massive amounts of data
- * uses distributed & scalable architecture
- * runs across multiple commodity hardware machines working together
- * focuses on horizontal scaling rather than vertical scaling

Two main functions of Hadoop



1) Massive Data storage: distribute data across nodes

2) faster Parallel Processing: processing data in parallel across multiple machine

Hadoop Properties & Applications in Big Data:

Key properties of Hadoop:

1) Scalability:
   - Hadoop is horizontally scalable
   - Can add as many machines as needed to cluster
   - enables scaling in & out rather than requiring single computer upgrade

2) Fault Tolerance:
   - maintains replicas (copies) of data across multiple machines
   - System continues to work even if individual machine fails
   - Data redundancy ensures protection against hardware failure

3. Distributed Processing:

* Data is processed where it is stored
* minimizes data movement across the network
* processes data locally, improving efficiency

4. Cost Effectiveness

* works on inexpensive hardware ("commodity machines" for)
* doesn't require specialized high-end servers
* makes it accessible for organizations with different budgets

5. Open Source:

* part of Apache Foundation
* free to use & modify
* increased adaptability & community support

Real-World Application Example: Amazon

Problem:

1. Millions of users generating massive amount of data
2. Data includes searches, purchases, reviews etc
3. Traditional systems cannot handle this volume

Hadoop Solution

* stores data distributed across many low cost machines
* processes data for
  * powered recommendation
  * Trend analysis
  * Delivery route optimization
  * Data-driven operations

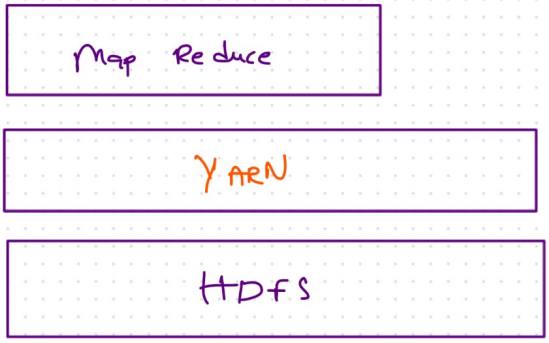
Google — GCP
Microsoft — Azure
Amazon — AWS

→ essential characteristics of big data solutions
Understanding how hadoop implementation with
properties helps evaluate technologies

Core Components of the Hadoop Ecosystem:

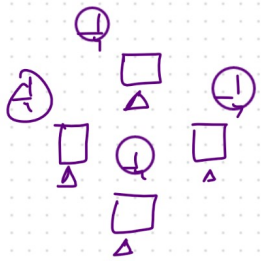
* hadoop ecosystem is a collection of open source tools & projects
* These tools work together to store, process & analyze large amounts of data
* core concepts + added later by various companies (Hadoop)

Core Hadoop Components



1) HDFS (Hadoop Distributed file System)

* Inspired by Google file Storage (GFS)
* purpose: provides distributed storage for large data files
* allows data to be stored across multiple machines in a cluster
* handles the storage aspect of big data problem



2) Map Reduce (MR):

* Inspired by Google map Reduce (GMR)

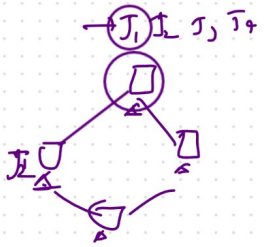
purpose: helps process large-scale data

* works by dividing processing jobs into smaller tasks that can run in parallel
* important to understand for learning distributed processing concept

3) YARN (Yet Another Resource Negotiator)

* Added in Hadoop 2.0

primary purpose: decouples resource management from application execution



* sits b/w HDFS & MapReduce
* manages how resources are allocated when multiple users submit jobs
* handles negotiation & distribution of cluster resources

HDFS

* Storage layer of Hadoop
* Distributed file System for storing large datasets

MapReduce

* processing framework
* enables parallel processing of data
* written primarily in Java

YARN

→ resource management layer

* negotiates computing resources across the cluster