



# **Cloud Computing**

## **Apache Hadoop**

Seyyed Ahmad Javadi

[sajavadi@aut.ac.ir](mailto:sajavadi@aut.ac.ir)

Spring 2024

# Apache Hadoop



- Apache Hadoop is the driving force behind the big data industry.
- It is an open-source, Java-based framework.
- Hadoop ...
  - Stores data (Hadoop Distributed File System – HDFS)
  - Executes jobs (MapReduce) on large clusters of commodity servers.
- Hadoop is highly fault tolerant and it is scalable from a single server to thousands of servers.

# Hadoop Components

---

## ➤ **Common**

- Utilities supporting the other Hadoop modules, components, etc.

## ➤ **MapReduce (YARN)**

- A framework for job scheduling and cluster resource management.
- It is a programming model and an execution engine running on clusters of commodity servers.

## ➤ **HDFS**

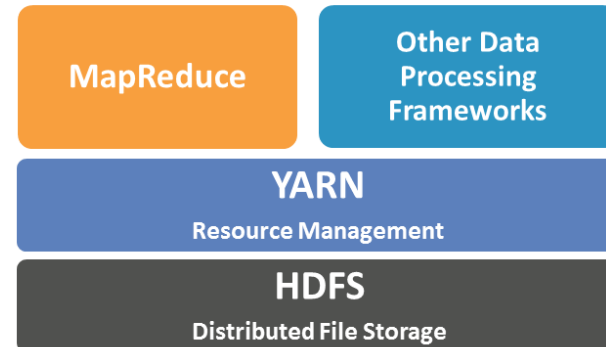
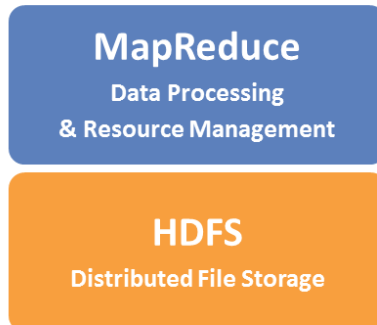
- A distributed file system running on clusters of commodity computers.

# Hadoop MapReduce v.2



## ➤ YARN (Yet Another Resource Negotiator)

- The major new improvement introduced in Hadoop v2.



<https://www.h2kinfosys.com/blog/hadoop-yarn-tutorial-learn-the-fundamentals-of-yarn-architecture/>

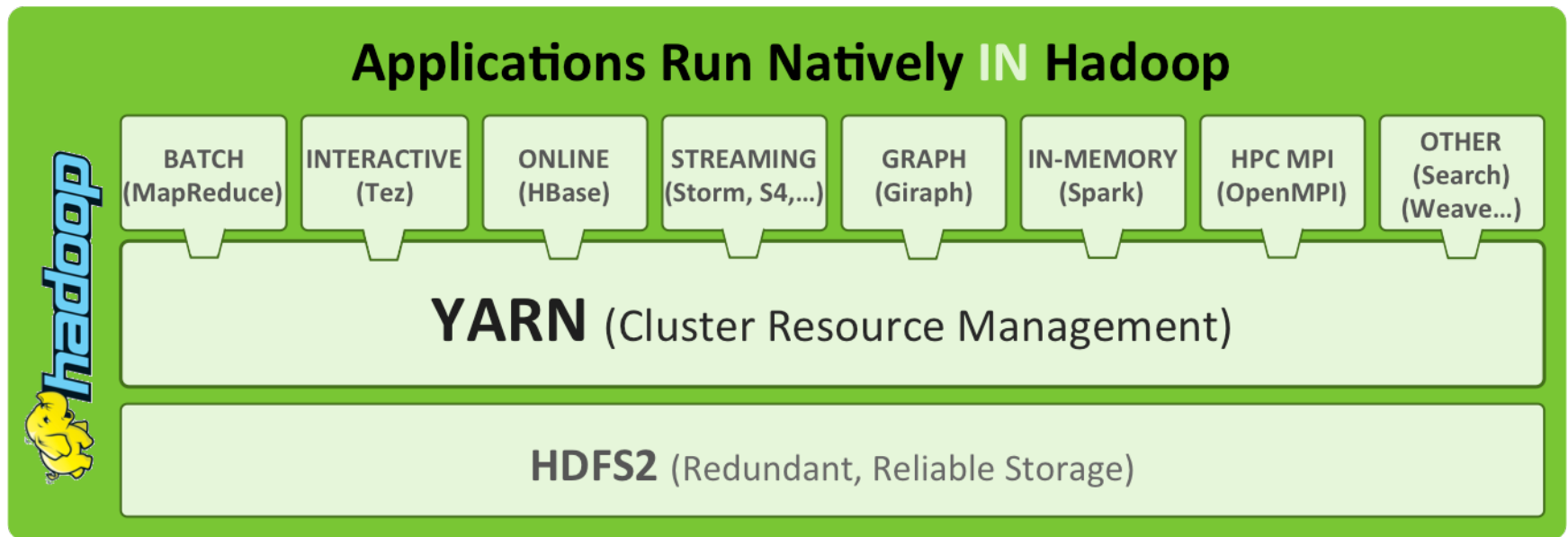
# Hadoop MapReduce v.2 (cont.)

---

- YARN is a **central component** in the Hadoop v2 ecosystem and provides a common platform for many different types of distributed applications.
- Some examples of the current YARN applications include:
  - The **MapReduce** framework
  - **Spark** processing engine
  - The Storm real-time stream processing framework

# Hadoop YARN

---



<https://devveri.com/hadoop/hadoop-2-0-yarn>



YARN is a resource management system that allows ***multiple distributed processing frameworks*** to effectively share the compute resources of a Hadoop cluster and to utilize the data stored in HDFS.

# Hadoop YARN (cont.)

---

- The **YARN ResourceManager** process is the central resource scheduler that manages and allocates resources to the different applications (also known as jobs) submitted to the cluster .
- **YARN NodeManager** is a **per node process** that manages the resources of a single compute node .



# Hadoop YARN (cont.)

---

- Scheduler component of the ResourceManager allocates resources in response to the applications resource request.
- Factors taken into consideration:
  - The cluster capacity
  - The other scheduling policies that can be specified through the YARN policy plugin framework.

# Hadoop YARN (cont.)

---

- YARN has a concept called **containers**, which is the unit of resource allocation.
  - Each allocated container has the rights to a certain amount of CPU and memory in a particular compute node.
  - Applications can request resources from YARN by specifying the required number of containers and the CPU and memory required by each container.

# Hadoop YARN (cont.)

---

- ApplicationMaster is a per-application process that coordinates the computations for a single application.
- The first step of executing a YARN application is to deploy the ApplicationMaster.

# Hadoop YARN (cont.)

---

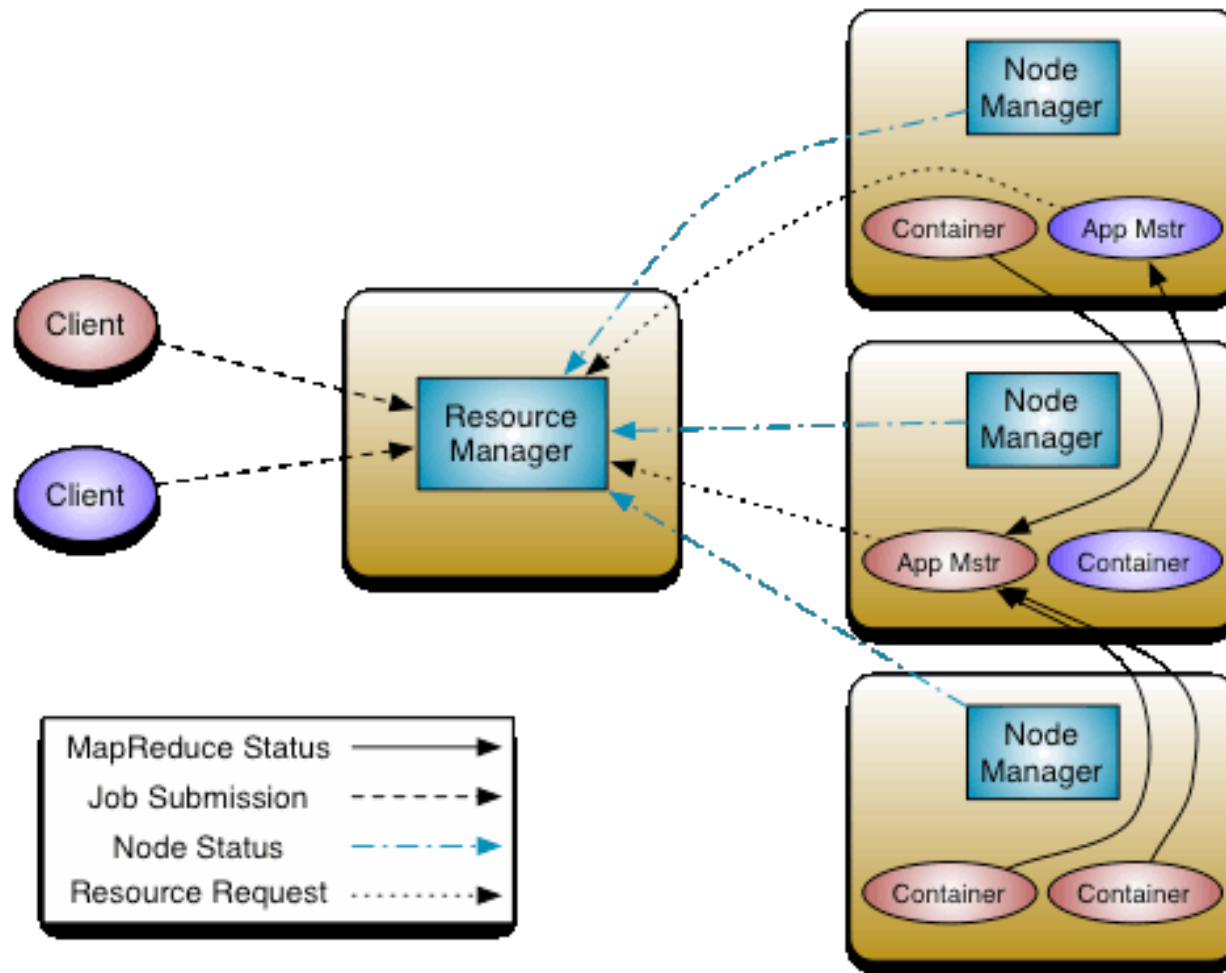
- After an application is submitted by a YARN client, the ResourceManager allocates a container and deploys the ApplicationMaster for that application.
- Once deployed, the ApplicationMaster is responsible for requesting and negotiating the necessary resource containers from the ResourceManager.

# Hadoop YARN (cont.)

---

- Once the resources are allocated by the ResourceManager, ApplicationMaster coordinates with the NodeManagers to launch and monitor the application containers in the allocated resources.

# Hadoop MapReduce (cont.)



<https://hadoop.apache.org/docs/stable/hadoop-yarn/hadoop-yarn-site/YARN.html>

# Hadoop MapReduce (cont.)

---

- A MapReduce **job** is a unit of work that the client wants to be performed.
  - It consists of the **input data**, the [MapReduce program](#), and **configuration information**.
- Hadoop runs the job by dividing it into **tasks**, of which there are two types: **map tasks** and **reduce tasks**.

# Hadoop MapReduce (cont.)

---

- Hadoop divides the input to a MapReduce job into fixed-size pieces called **input splits**, or just **splits**.
- Hadoop creates **one map task for each split**, which runs the user defined map function for each record in the split.



# Hadoop MapReduce (cont.)

---

- For most jobs, a good split size tends to be the size of an HDFS block, 128 MB by default.
- Hadoop does its best **to run the map task on a node where the input data resides in HDFS.**
- **Why does Hadoop do this?**

# Hadoop MapReduce (cont.)

---

- Hadoop does its best **to run the map task on a node where the input data resides in HDFS.**
- This is called the **data locality optimization**
  - It should now be clear why the optimal split size is the same as the block size.
- **Map tasks write their output to the local disk, not to HDFS**
  - Because the Map output is intermediate output, not the final output.

# Hadoop MapReduce (cont.)

---

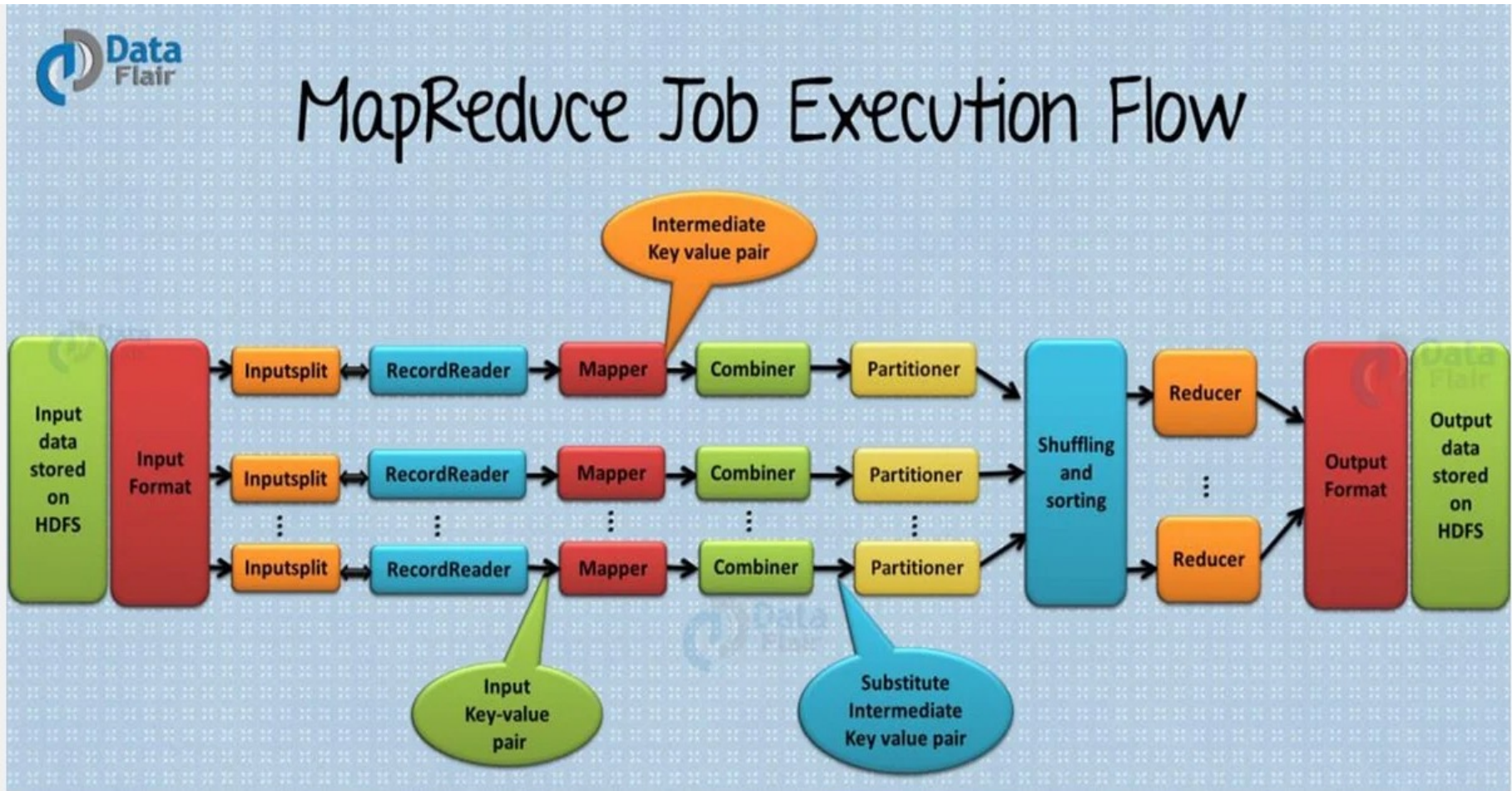
- Reduce tasks don't have the advantage of data locality.
- When there are multiple reducers, the map tasks **partition** their output, each creating one partition for each reduce task.

# Hadoop MapReduce (cont.)

---

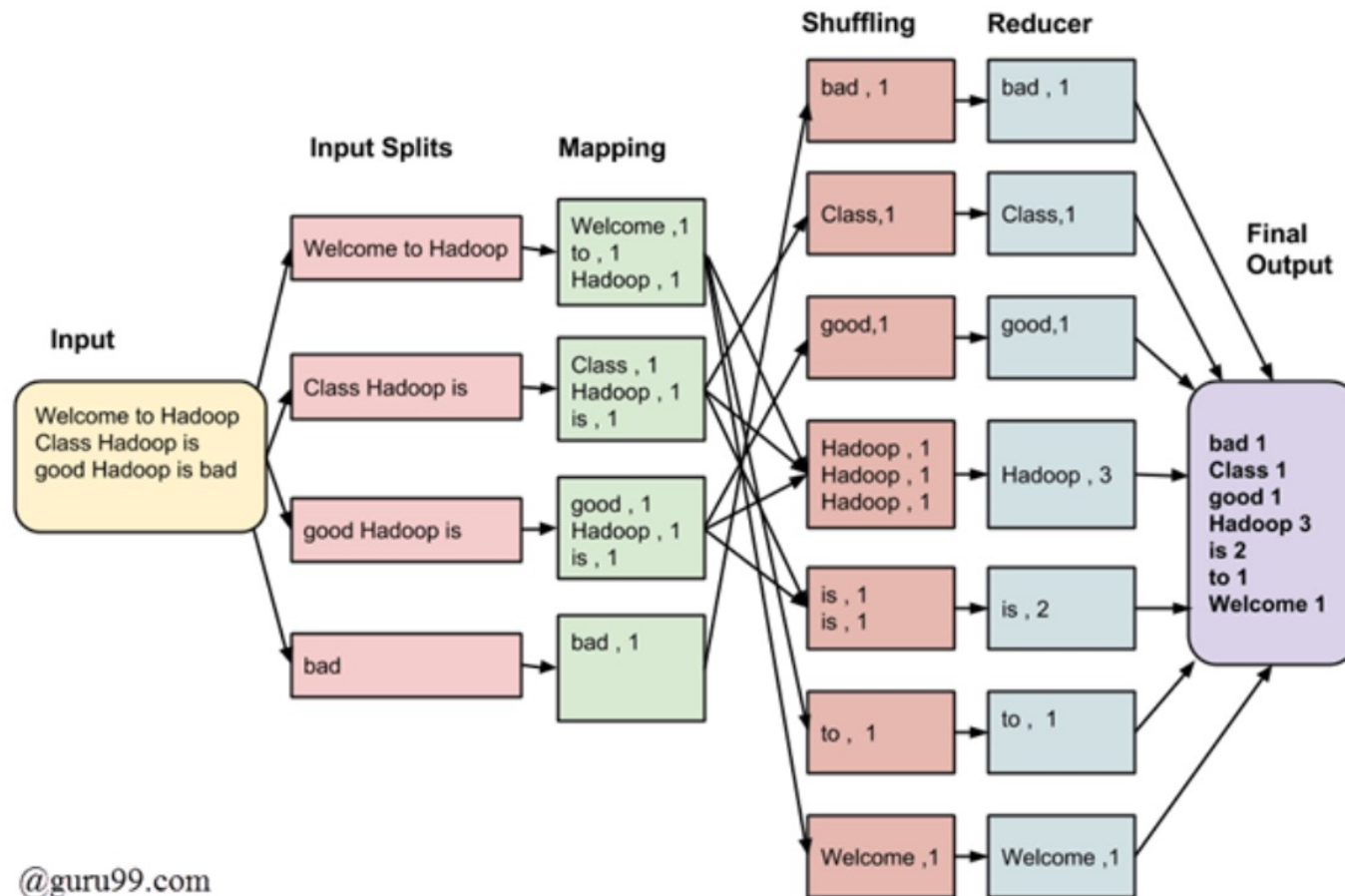
- There can be many keys (and their associated values) in each partition, but the records for **any given key are all in a single partition.**
- The partitioning can be controlled by a user-defined partitioning function, but normally the default partitioner works very well.
  - Default partitioner buckets keys using a **hash function.**

# MapReduce Job Execution Flow



<https://data-flair.training/blogs/how-hadoop-mapreduce-works/>

# MapReduce Example



@guru99.com

<https://www.guru99.com/introduction-to-mapreduce.html>

# MapReduce on Utility Clouds

---

- Amazon offers an analytic cloud service called [Amazon Elastic MapReduce \(EMR\)](#) which provides a managed Hadoop MapReduce on Amazon EC2 instances.
- Microsoft also offers a big data service, which is called [Windows Azure HDInsight](#) Service, which deploys and provisions ApacheHadoop clusters in the Azure cloud.

# MapReduce on Utility Clouds (cont.)

---

- **AppEngine-MapReduce** is an open-source library for doing MapReduce-style computations on the Google App Engine platform with pricing that is competitive with Amazon EMR.