Hadoop, Yarn and Beyond

B. RAMAMURTHY

Overview

- We learned about Hadoop1.x or the core.
- Just like Java evolved, Java core, Java 1.X, Java 2.. So on, software and systems evolve, naturally..
- Lets examine how Hadoop evolved.
- Just like Java core is still the important principles, we will learn to work with Hadoop.
- Also think about this, in our regular OS, we use the file system through the interface an operating system provides. In a similar fashion, we will use Hadoop 2.x for our needs.
- Have you wondered about this? RDBMS vs Hadoop vs MongoDB

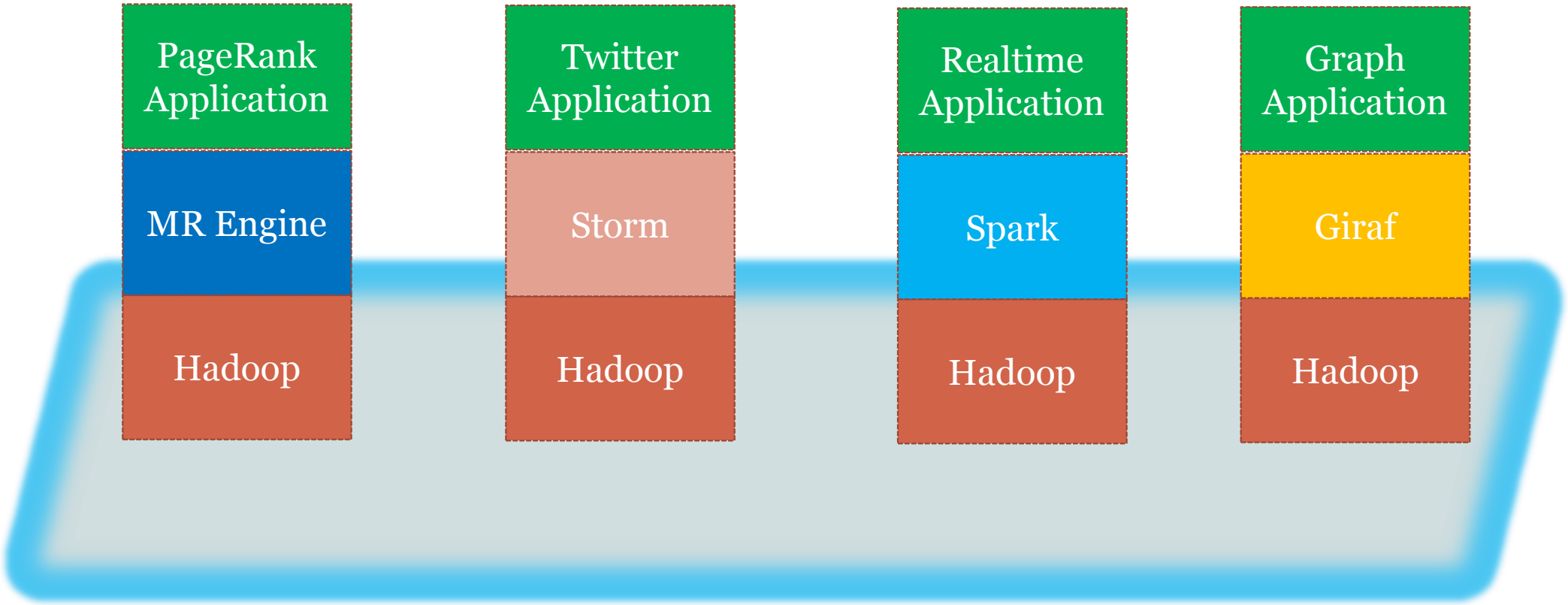
Execution Framework: Hadoop 2.x

B. RAMAMURTHY

Introduction

- Think about this: Hadoop started off as the infrastructure for MapReduce (MR).
- Then many other data models were built on Hadoop: twitter storm, Giraf (graph processing), Spark Streaming etc.
- Initially individual Hadoop infrastructure for built for each of these.
- Hadoop 2.x evolves from a distributed filesystem into an operating system that manages not only files/data storage but other resources.
- Because of this it can support more than the Mapreduce like engines.
- It can support Spark, and other programming models.
- You have to stand up a HDFS for each of the application models!

Traditional Hadoop Shop



Yarn OS and Hadoop File System

6

B. RAMAMURTHY

Yarn OS

- Yarn (Yet another Resource Negotiator) is an operating system with HDFS as the file system
- Now you can run any big data application on a single system
- You don’t have to be deploying a HDFS each for each of your application.
- And here are the differences,

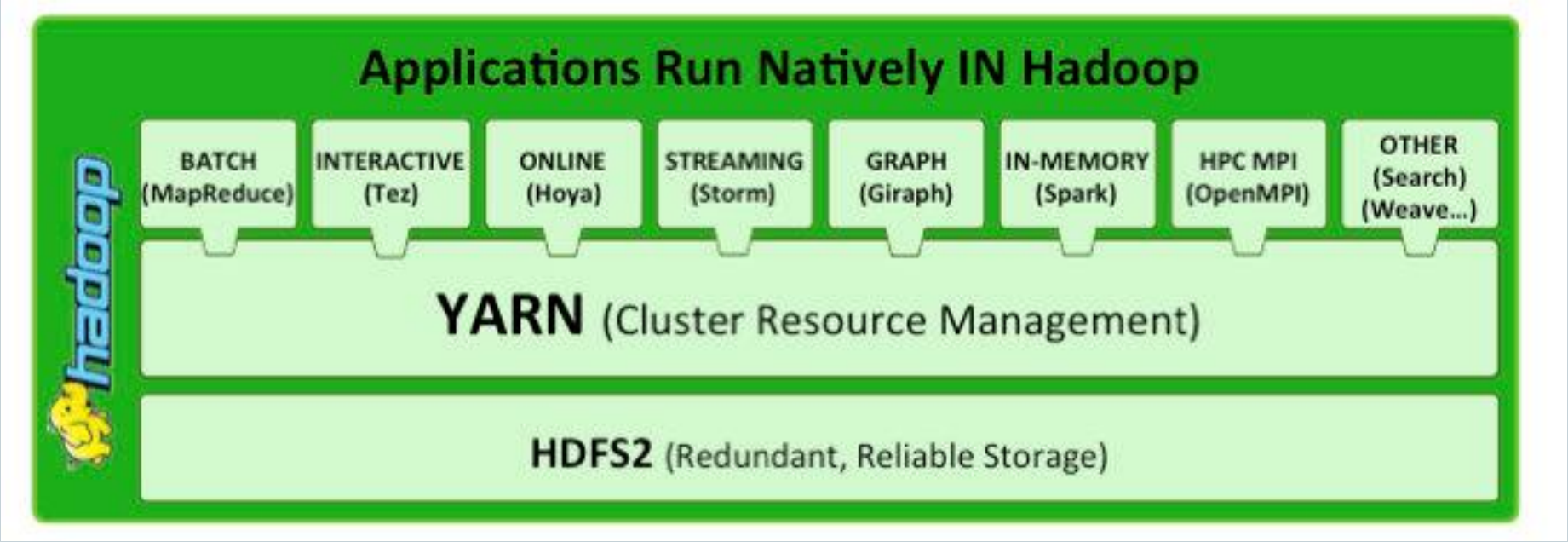
Improvements in Yarn-based Hadoop

1. HDFS Federation: Separation of namespaces and storage, this allows for many applications to exist on the same system, performance gains.
2. NameNode high availability: HA Hadoop
3. Introduction of YARN: decoupling of MR and Hadoop; can now do batch processing, interactive processing, stream processing, graph processing (we will look at examples next)
4. Resource manager high availability (HA): Main Resource manager of yarn is replicated by a standby providing high availability.

A look at the application models: where si Hadoop used?

- Batch processing: Indexing data, crawling the web, and data crunching; Examples: MapReduce, Tez, Spark, Hive; Take a look at Tez https://tez.apache.org/
- Interactive processing: To answer unknown questions: data exploration; EDA; Examples: Impala, Tez
- Stream Processing: social media trending; sentiment analysis; Examples: Spark stream, Apache storm
- Graph processing: Distributed graph processing of linked world; Examples: Apache Giraf, Sparkx

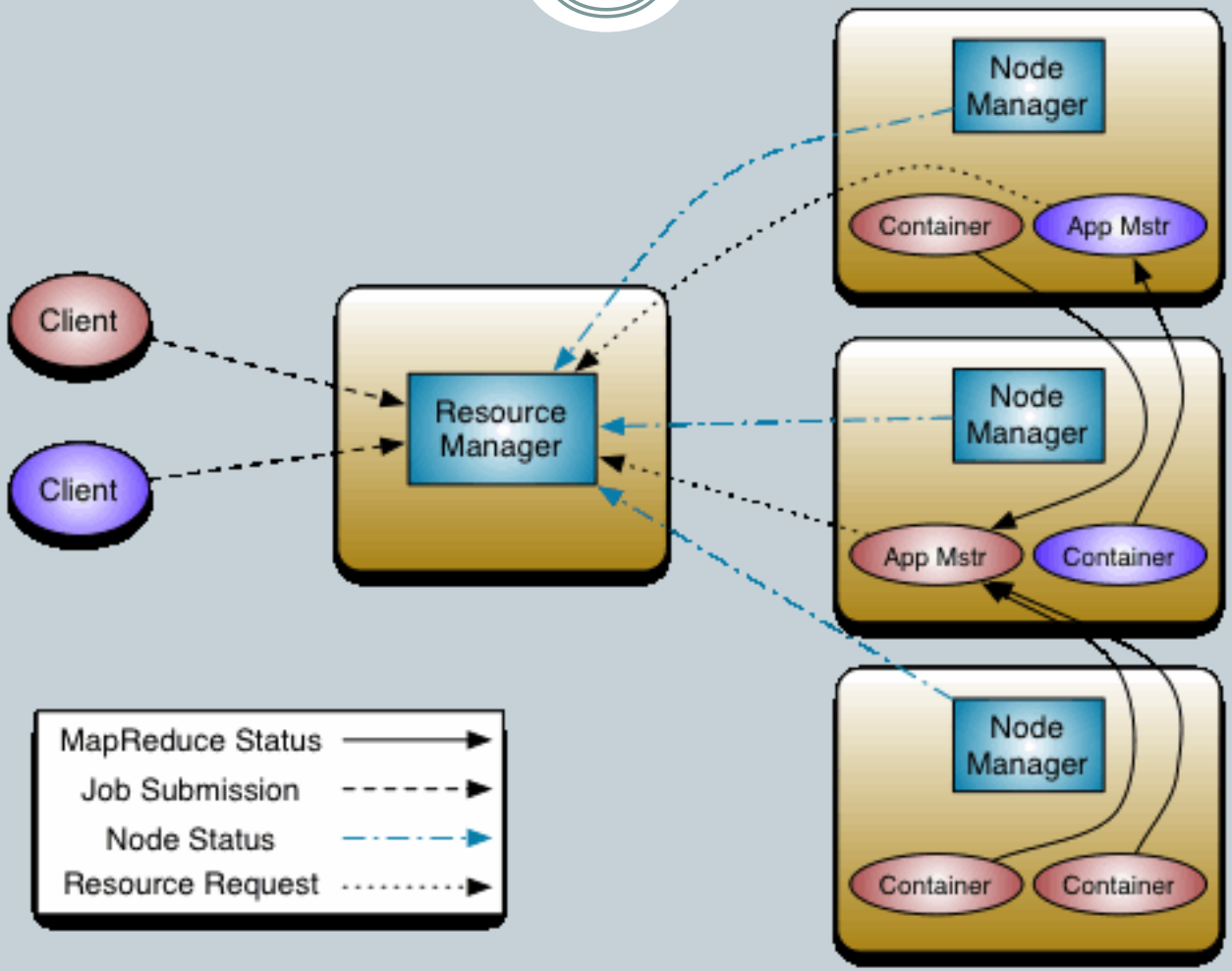
Yarn: Different applications are supported by YARN



YARN

- Enables both batch applications such MR and other streaming application run natively on Hadoop.
- Resource manager, node manager and application master.
- Application manager negotiates on behalf of the application for resources.
- Container-based execution units.

YARN Architecture



Hadoop vs MongoDB vs RDBMS

- "If you have requirements for processing low-latency real-time data, or are looking for a more encompassing solution (such as replacing your RDBMS or starting an entirely new transactional system), MongoDB may be a good choice. If you are looking for a solution for batch, long-running analytics while still being able to query data as needed then Hadoop is a better option. Depending on the volume and velocity of your data, Hadoop is known to handle larger scale solutions, so that should certainly be taken into account for scalability and expandability. Either way, both can be excellent options for a scalable solution that process large amounts of complex data sets more efficiently than a traditional RBDMS."
- Reference: [4]

LAMP vs HDFS vs MEAN

- LAMP: Linux (OS), Apache (web server), Mysql (RDBMS), PHP (pipeline): for transaction data, typically structured data
- HDFS: Big data
- MongoDB: also for common data, unstructured, more a replacement for LAMP not for HDFS.
- For example, MEAN and HDFS may co-exist in a given location. RDBMS too!

Exploring these technologies without much upfront cost?

- Yes you can. Using any of the cloud infrastructures?
- You can use many from Amazon AWS to Google Cloud.
- Lets look at AWS.
- UB is a institutional educator member.
  - Every students gets $100 credit on AWS for using most of the services it hosts
  - See here: http://www.buffalo.edu/ubit/service-guides/teaching-technology/aws.html
- Lets try to access AWS and check it out
- Our goal today is to be able to
  - login into aws: explore some common services and some emerging technology services
  - provision a Linux server + Windows Server and interact with them

Consider your application characteristics

- EDA? Lab 1
- Data pipeline? Lab 2?
- Data Product? Lab 2?
- Data Analysis: ML

References

1. http://hadoop.apache.org/
2. https://www.digitalocean.com/community/tutorials/an-introduction-to-hadoop
3. R. Raposa, An Introduction to Hadoop 2.0. https://www.oreilly.com/ideas/an-introduction-to-hadoop-2-0-understanding-the-new-data-operating-system, O'Reilly, 2017.
4. Hadoop vs MongoDB, http://www.aptude.com/blog/entry/hadoop-vs-mongodb-which-platform-is-better-for-handling-big-data, last viewed 2018.