Yet Another Resource Negotiator

YARN

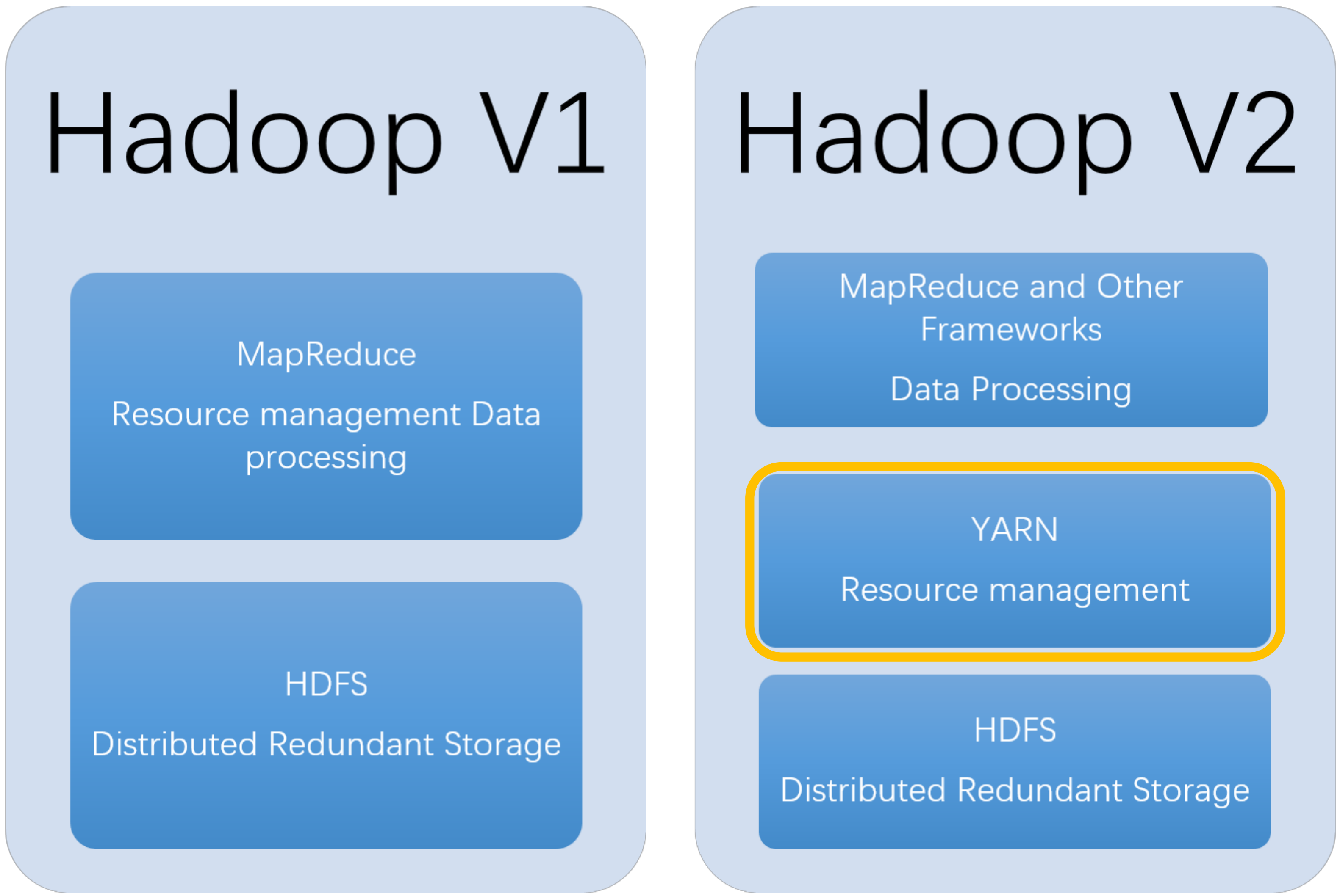
11711335 Zhang Yifan
11712019 Yu Tiancheng

Outline

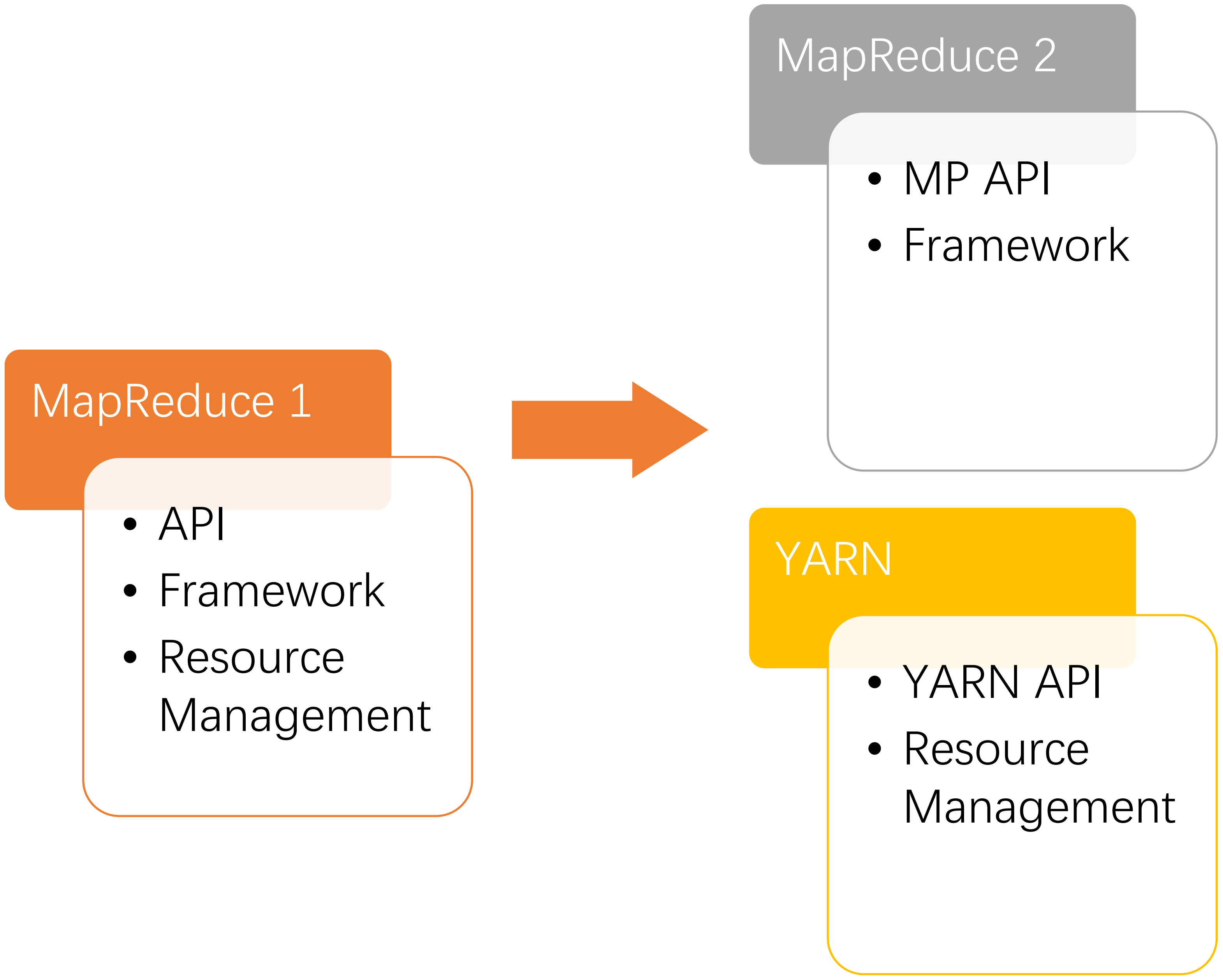
- Introduction
- Motivation for YARN
- YARN Architecture
- Application Workflow
- Real-world Performance
- Performance Optimizations

Introduction

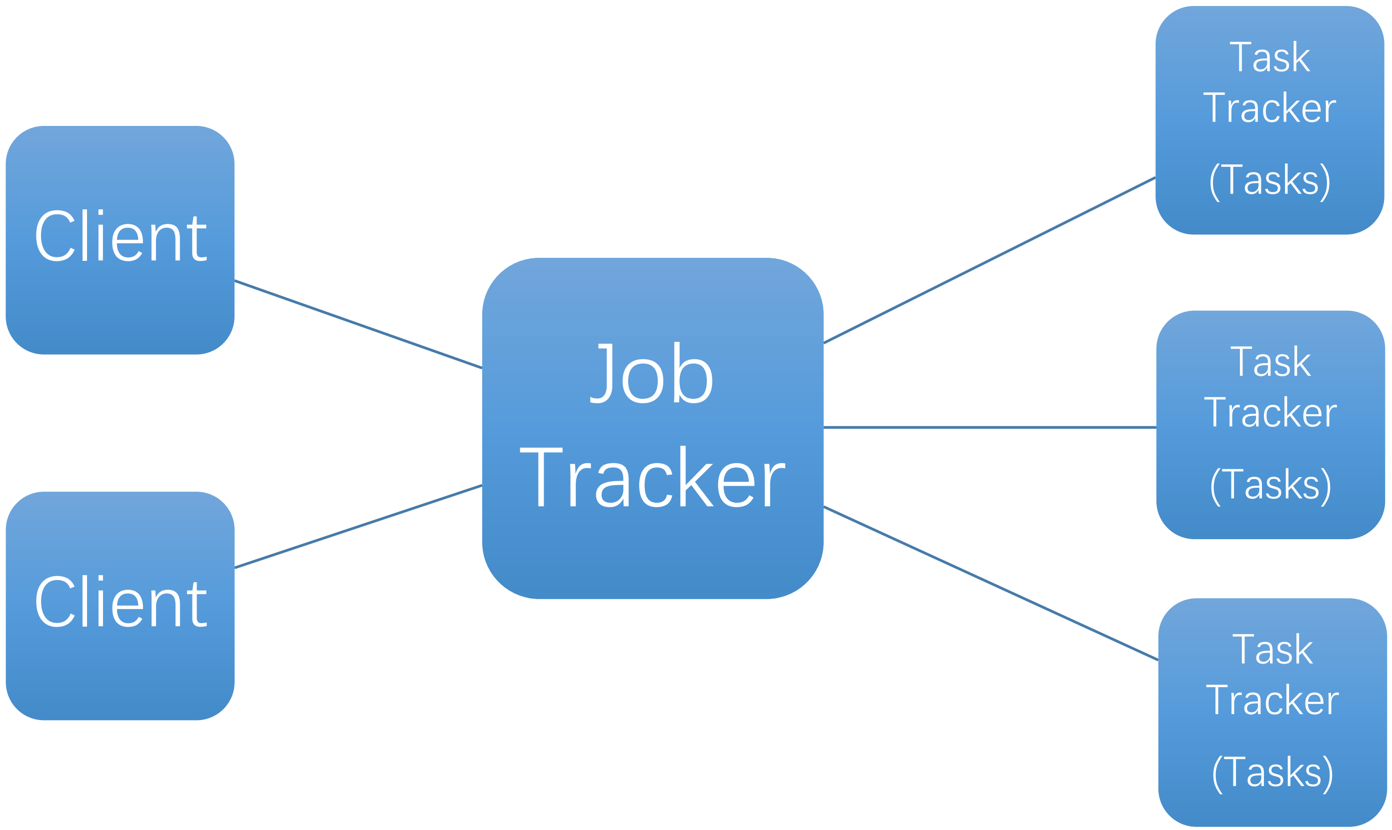
- What is Hadoop?
- What is MapReduce?
- What is YARN?


Hadoop V1
MapReduce
Resource management Data processing
HDFS
Distributed Redundant Storage
Hadoop V2
MapReduce and Other Frameworks
Data Processing
YARN
Resource management
HDFS
Distributed Redundant Storage

What Is YARN?


MapReduce 1
• API
• Framework
• Resource Management
MapReduce 2
• MP API
• Framework
YARN
• YARN API
• Resource Management

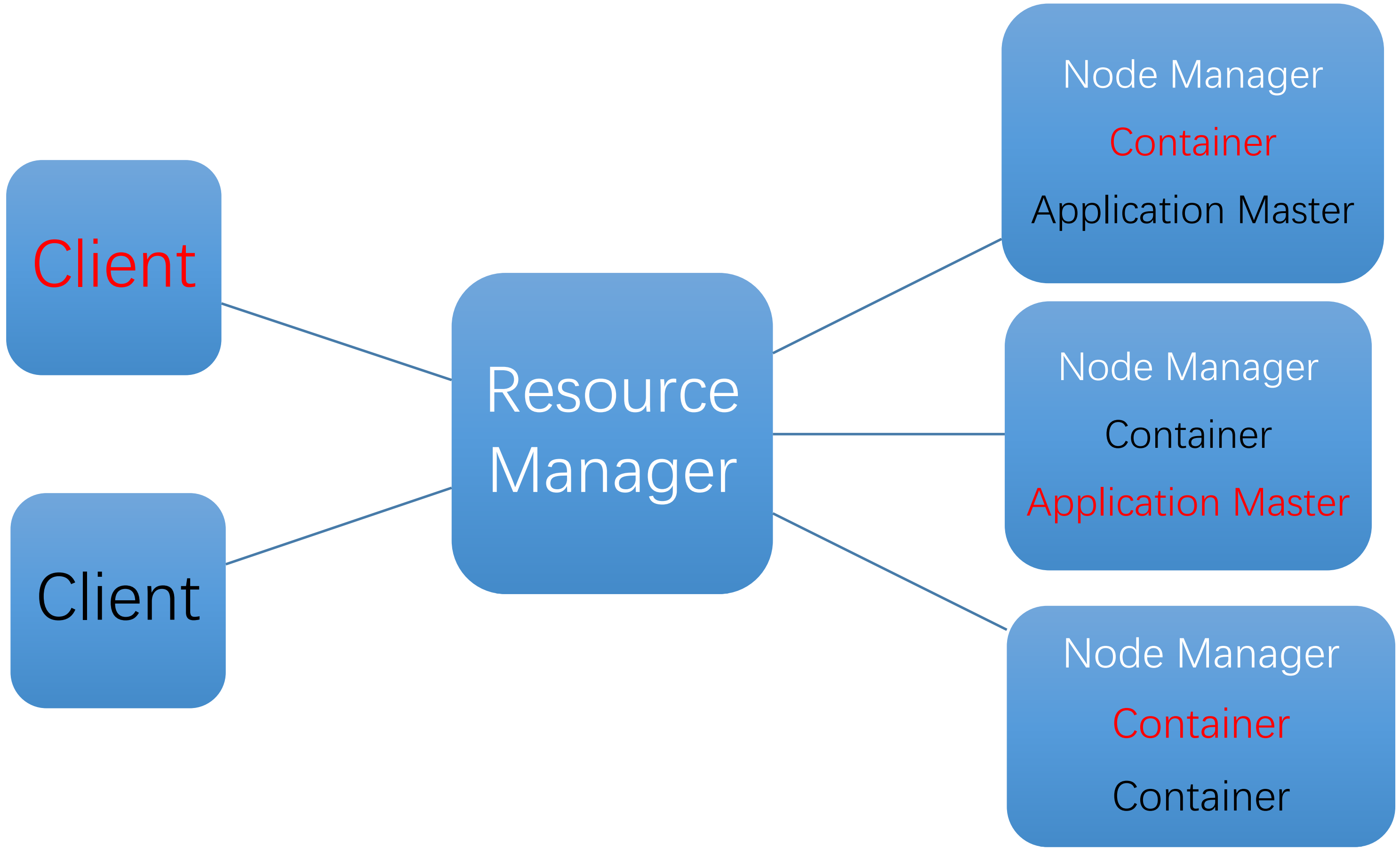
Map Reduce V1 Execution Framework



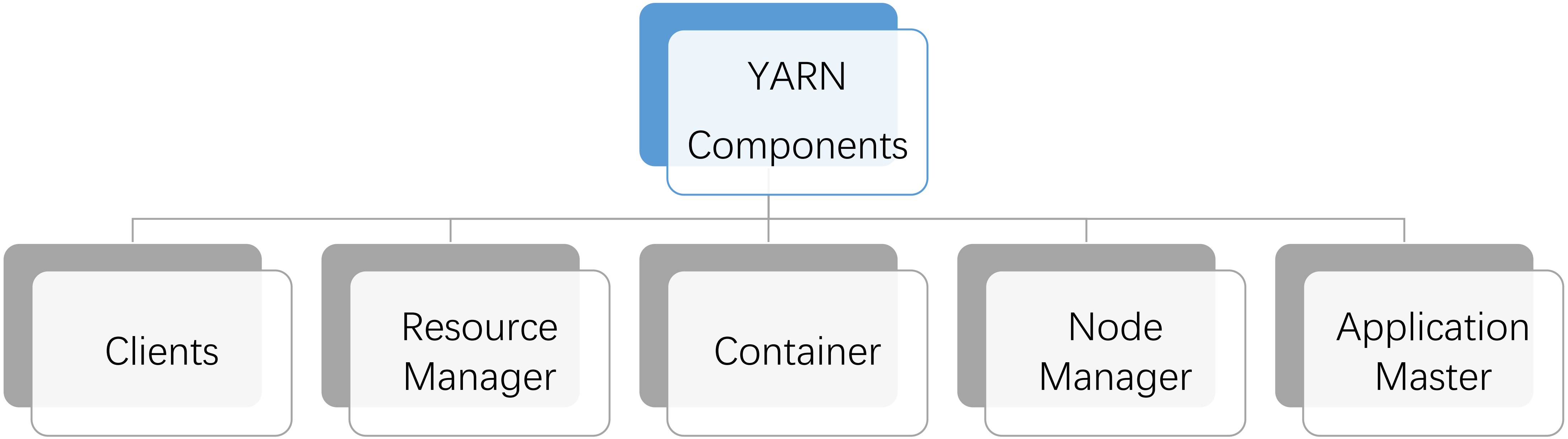
Motivations for V2

- R1: Scalability
- R2: Multi-tenancy
- R3: Serviceability
- R4: Locality awareness
- R5: High Cluster Utilization
- R6: Reliability/Availability
- R7: Secure and auditable operation
- R8: Support for Programming Model Diversity.
- R9: Flexible Resource Model
- R10: Backward compatibility

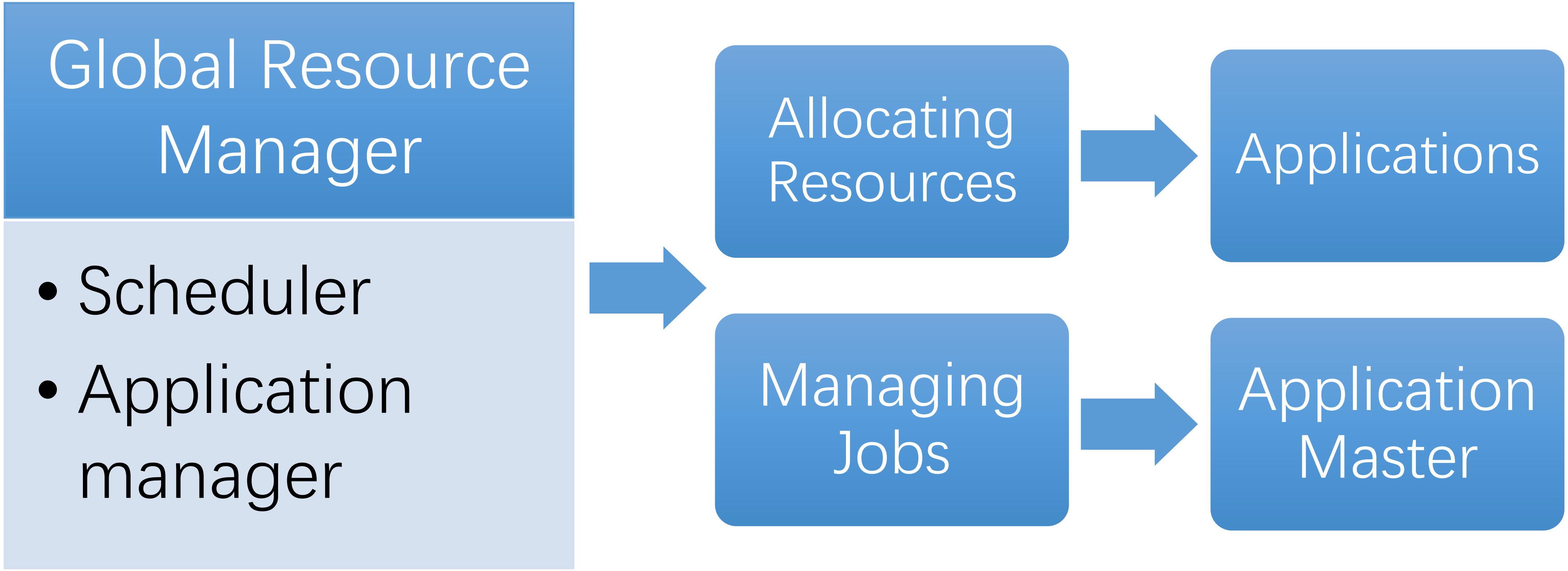
The New Architecture



The New Architecture

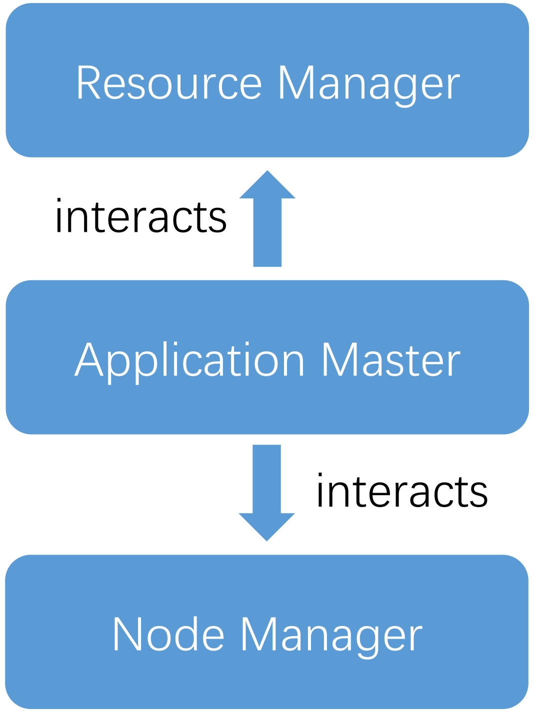


YARN Daemons



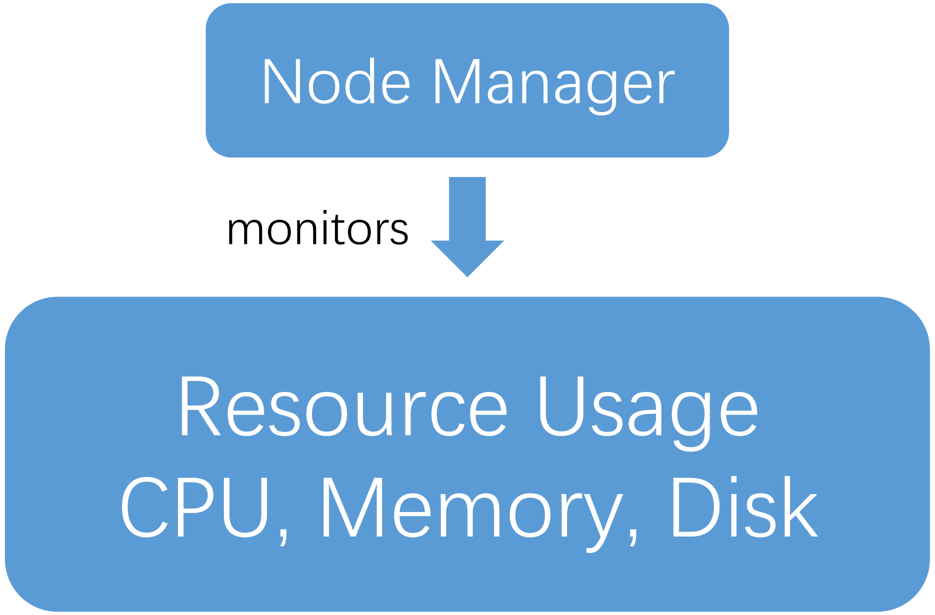
Application Master

- Manages application life and task scheduling



Node Manager

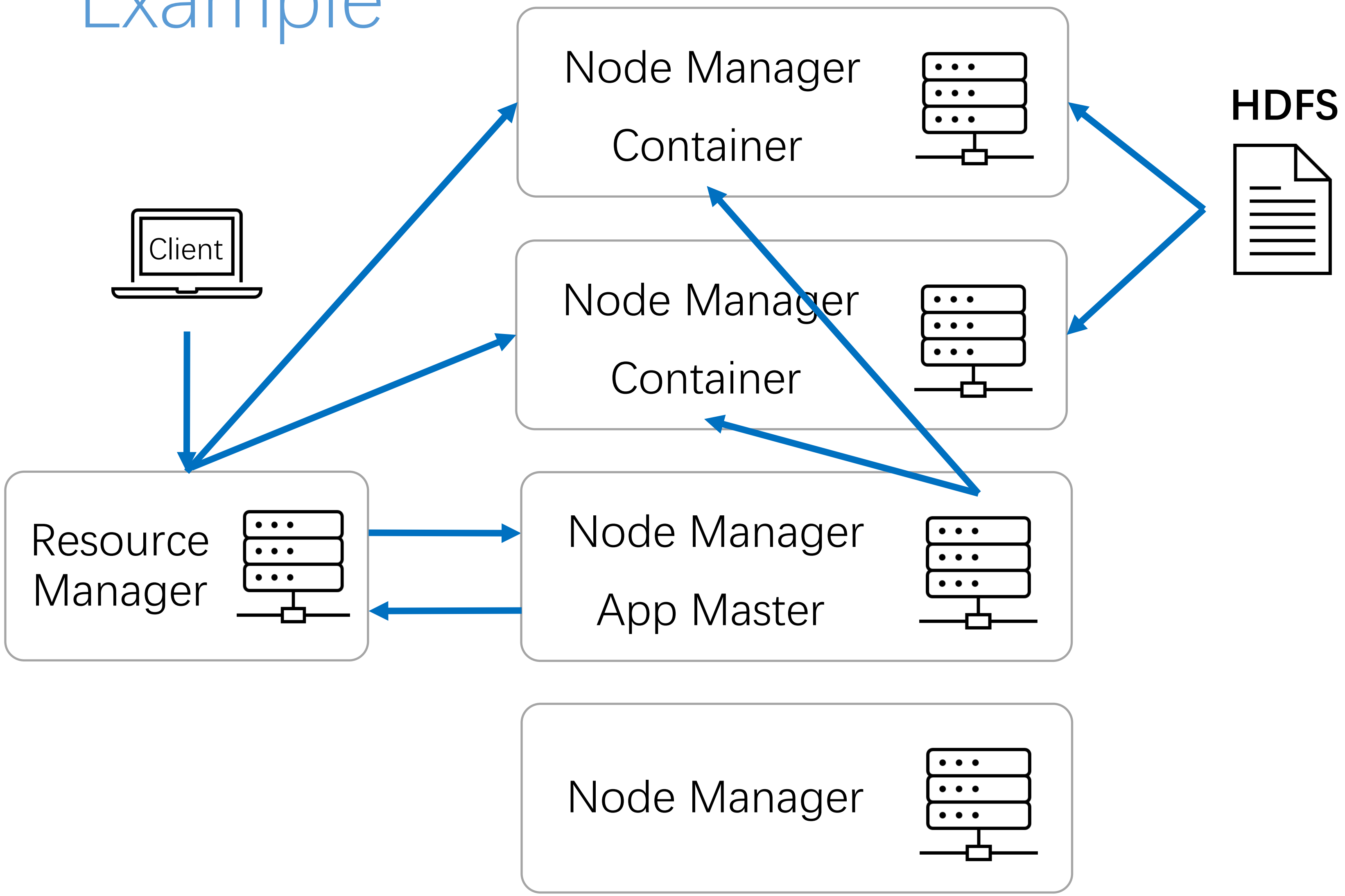
- Per node agent
- Manages single node resource allocations



Container/Slot

- Basic unit of resource allocation
- Example: Container x = 2GB, 1CPU
- Fine grained resource

Example


Node Manager
Container
HDFS
Client
Node Manager
Container
Resource Manager
Node Manager
App Master
Node Manager

YARN's Real-World Performance

- Hadoop held both the Daytona and Indy GraySort benchmark records in 2013.
- The current record is held by Tencent Sort.

Daytona GraySort	Indy GraySort
2013, 1.42 TB/min **Hadoop** 102.5 TB in 4,328 seconds 2100 nodes x (2 2.3Ghz hexcore Xeon E5-2630, 64 GB memory, 12x3TB disks) Thomas Graves Yahoo! Inc.	2013, 1.42 TB/min **Hadoop** 102.5 TB in 4,328 seconds 2100 nodes x (2 2.3Ghz hexcore Xeon E5-2630, 64 GB memory, 12x3TB disks) Thomas Graves Yahoo! Inc.
2016, 44.8 TB/min **Tencent Sort** 100 TB in 134 Seconds 512 nodes x (2 OpenPOWER 10-core POWER8 2.926 GHz, 512 GB memory, 4x Huawei ES3600P V3 1.2TB NVMe SSD, 100Gb Mellanox ConnectX4-EN) Jie Jiang, Lixiong Zheng, Junfeng Pu, Xiong Cheng, Chongqing Zhao Tencent Corporation Mark R. Nutter, Jeremy D. Schaub	2016, 60.7 TB/min **Tencent Sort** 100 TB in 98.8 Seconds 512 nodes x (2 OpenPOWER 10-core POWER8 2.926 GHz, 512 GB memory, 4x Huawei ES3600P V3 1.2TB NVMe SSD, 100Gb Mellanox ConnectX4-EN) Jie Jiang, Lixiong Zheng, Junfeng Pu, Xiong Cheng, Chongqing Zhao Tencent Corporation Mark R. Nutter, Jeremy D. Schaub

Source: **sortbenchmark.org**

MapReduce Benchmarks

Benchmark	Avg runtime(s)		Throughput (GB/s)	
	1.2.1	2.1.0	1.2.1	2.1.0
Random Writer	222	228	7.03	6.94
Sort	475	398	3.28	3.92
Shuffle	951	648	-	-
AM Scalability	1020	353/303	-	-
Terasort	175.7	215.7	5.69	4.64
Scan	59	65	-	-
Read DFSIO	50.8	58.6	-	-
Write DFSIO	50.82	57.74	-	-

- This benchmark compares the YARN performance in Hadoop 2.1.0 to the standard Hadoop-1 release 1.2.1
- The Sort and Shuffle performs better on YARN
- The Scan and Read / Write DFSIO performs worse
- Why?

Benefits of Preemption

- Consider a cluster running CapacitiyScheduler, with two queues:
  - Queue A: 80% of total capacity
  - Queue B: 20% of total capacity
- Submit one MapReduce jobs to each queue
- What will happen if:
  - Fixed capacity for each queue
  - Queues may take 100% of capacity, no preemption
  - Queues may take 100% of capacity, preemption

Benefits of Preemption

- Each of the graphs shows the result of the scheduling policies
- Work-preserving preemption allows the scheduler to overcommit resources without worrying about starvation
- Fixed capacity wastes resources
- No preemption cause capacity rebalancing take a long time

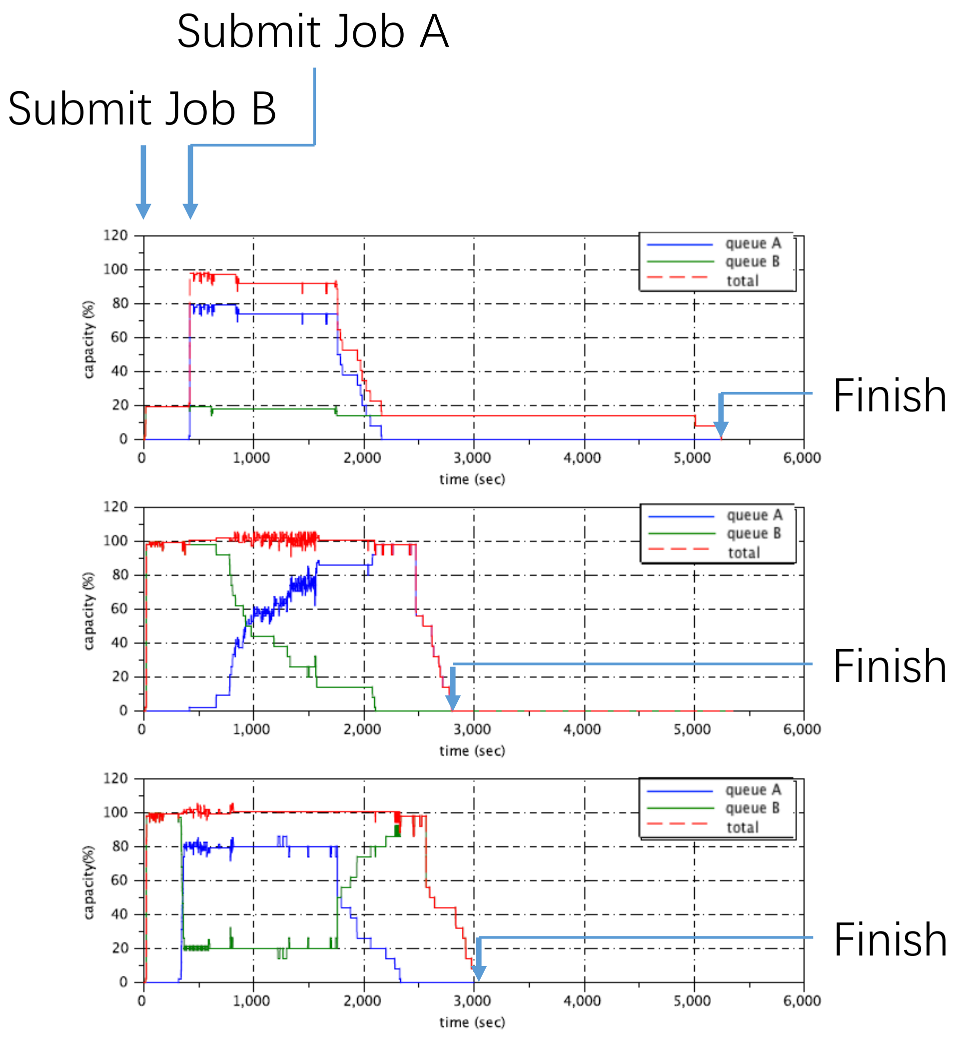


Figure 5: Effect of work-preserving preemption on the CapacityScheduler efficiency.

Questions?