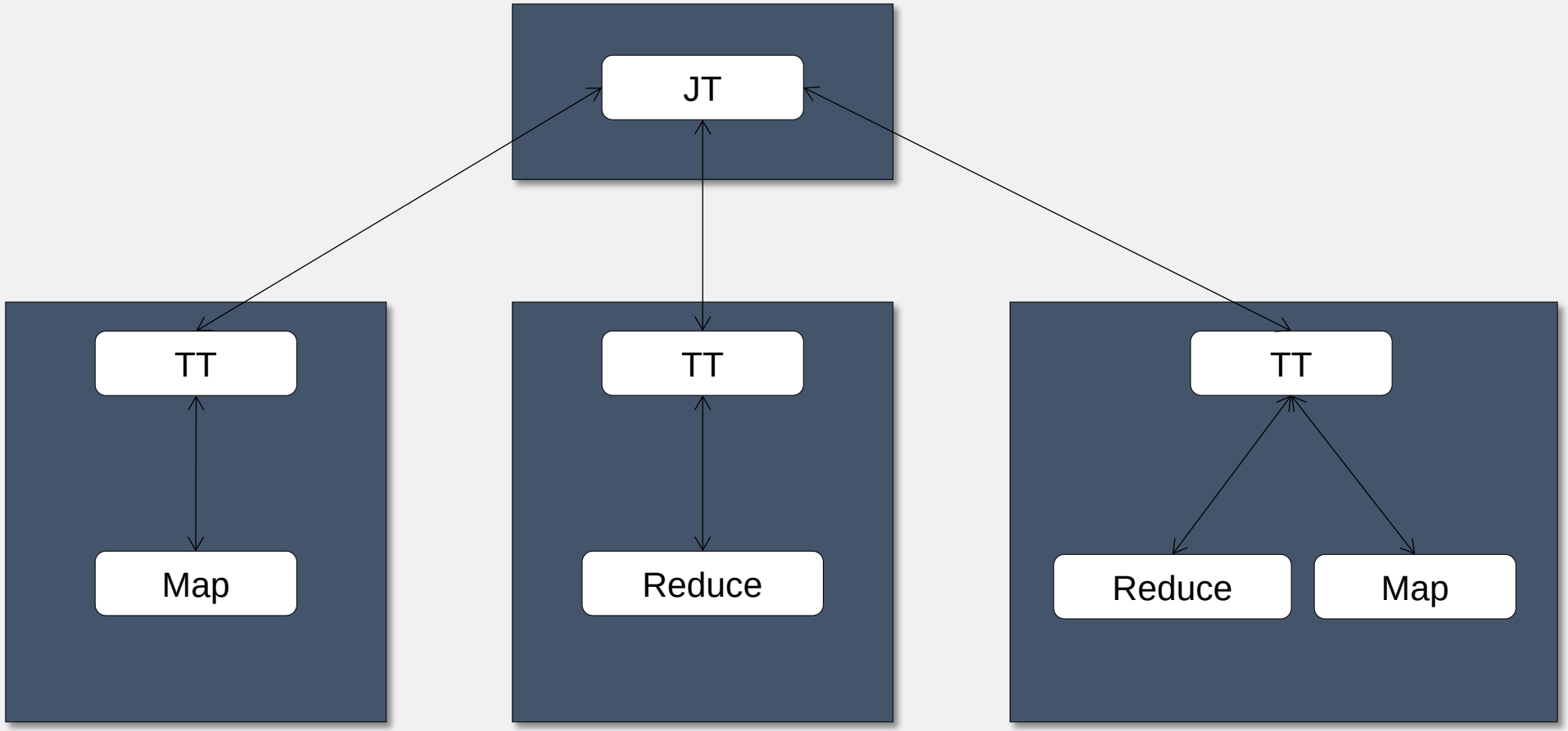
CLOUD COMPUTING APPLICATIONS

YARN Introduction

Roy Campbell & Reza Farivar

Hadoop 1.x



Issues with Hadoop

- Hadoop JobTracker was a barrier for scaling
  - Primary reason Hadoop 1.x is recommended for clusters no larger than 4000 nodes
  - Thousands of applications each running tens of thousands of tasks
  - JobTracker not able to schedule resources as fast as they became available
  - Distinct map and reduce slots led to artificial bottlenecks and low cluster utilization

Issues with Hadoop

- MapReduce was being abused by other application frameworks
  - Frameworks trying to work around sort and shuffle
  - Iterative algorithms were suboptimal
- YARN strives to be application framework agnostic
- Different application types can share the same cluster
- Runs MapReduce “out of the box” as part of Apache Hadoop

What is YARN?

- Yet Another Resource Negotiator
- Provides resource management services
  - Scheduling
  - Monitoring
  - Control
- Replaces the resource management services of the JobTracker
- Bundled with Hadoop 0.23 and Hadoop 2.x

YARN High-Level Architecture

- ResourceManager
  - Single, centralized daemon for scheduling containers
  - Monitors nodes and applications
- NodeManager
  - Daemon running on each worker node in the cluster
  - Launches, monitors, and controls containers
- ApplicationMaster
  - Provides scheduling, monitor, control for an application instance
  - RM launches an AM for each application submitted to the cluster
  - AM requests containers via RM; launches containers via NM
- Containers
  - Unit of allocation and control for YARN
  - ApplicationMaster and application-specific tasks run within containers

YARN High-Level Architecture