

IT 775

Database Technology

Data Stores

Hadoop

Apache Hadoop

- An open source software framework for storage and large scale processing of data-sets on clusters of commodity hardware.
- An Apache top-level project being built and used by a global community of contributors and users (licensed under the Apache License 2.0).
- Created by Doug Cutting and Mike Cafarella in 2005. It was originally developed to support distribution for the Nutch search engine project. Doug, who was working at Yahoo! at the time and is now Chief Architect of Cloudera, named the project after his son's toy elephant.

Hadoop Framework

- Hadoop Common - libraries and utilities needed by other Hadoop modules
- Hadoop Distributed File System (HDFS) - A distributed file-system that stores data on commodity machines, providing very high aggregate bandwidth across the cluster
- Hadoop YARN - A resource-management platform responsible for managing compute resources in clusters and using them for scheduling of users' applications
- Hadoop MapReduce - A programming model for large scale data processing

Hadoop Framework

- All modules assume that hardware failures are common and should be automatically handled in software by the framework
- Apache Hadoop "platform" - Apache Pig, Apache Hive, Apache HBase, and others
- Apache Pig and Apache Hive, among other related projects, expose higher level user interfaces like Pig latin and a SQL variant respectively
- Hadoop framework itself is mostly written in Java with some native code in C and command line utilities written as shell-scripts



Apache Hadoop Ecosystem



Ambari

Provisioning, Managing and Monitoring Hadoop Clusters



Scoop

Data Exchange



Zookeeper

Coordination



Oozie

Workflow



Pig

Scripting



Mahout

Machine Learning

R Connectors

Statistics



Hive

SQL Query



Hbase

Columnar Store



HDFS

Hadoop Distributed File System

YARN Map Reduce v2

Distributed Processing Framework



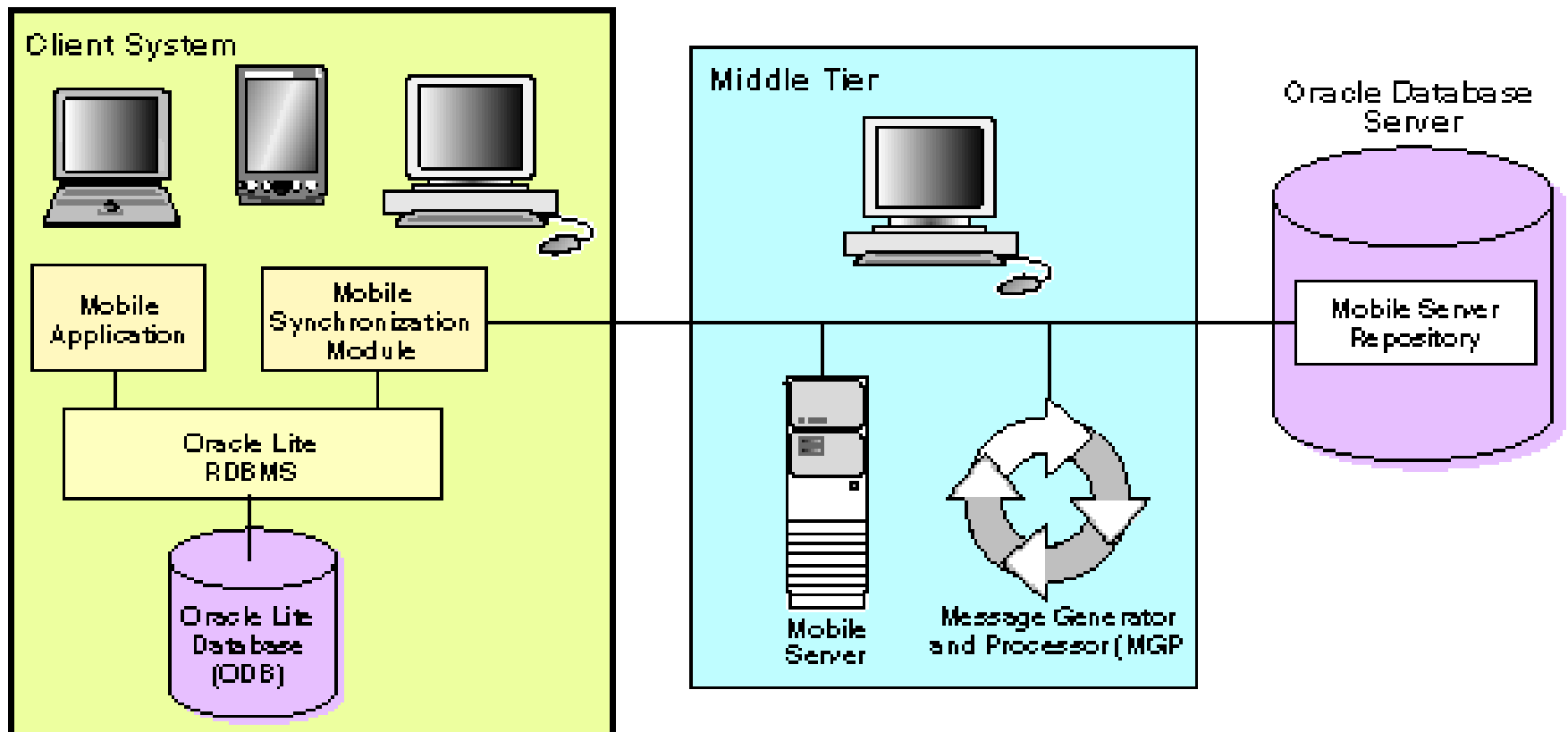
Hadoop Distributed File System (HDFS)

- Distributed, scalable, and portable file-system
- Each node
 - Single namenode
 - Cluster of datanodes
 - Does not require a datanode to be present
- Datanodes serve up data blocks over the network using a HDFS specific block protocol
- File System uses the TCP/IP layer for communication
- Clients use Remote procedure call (RPC) to communicate between each other
- Stores large files (typically gigabytes to terabytes) across multiple machines

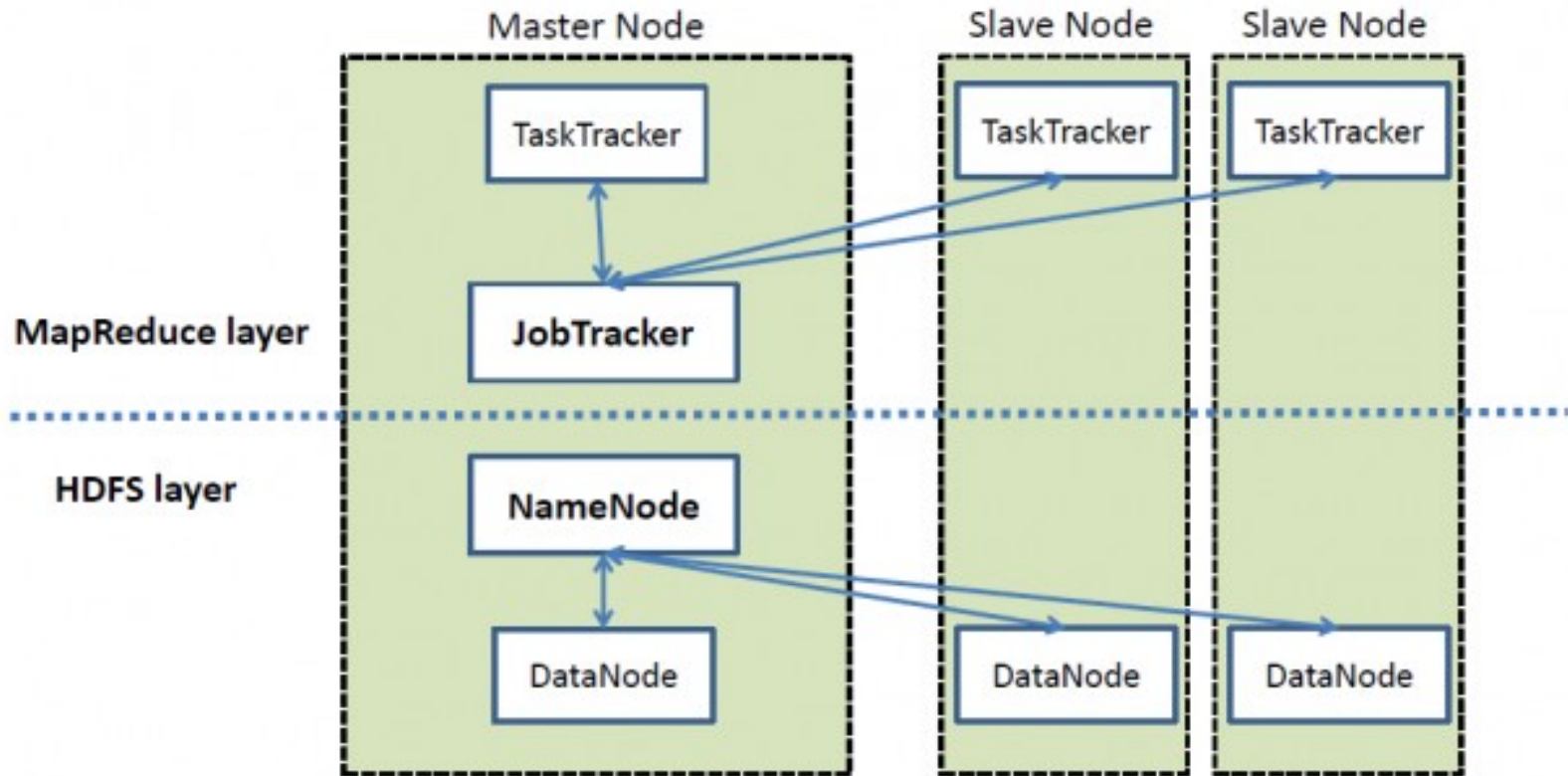
Hadoop Distributed File System (HDFS)

- Reliability by replication of data across multiple hosts; RAID not required
- Default replication value -> 3
 - Data stored on three nodes
 - Two on the same rack
 - One on a different rack
 - Data nodes talk to each other
 - Rebalance data
 - Move copies around
 - Keep data replication high
- Not fully POSIX-compliant; requirements for POSIX file-system differ from Hadoop application goals.
- Tradeoff is increased performance for data throughput and non-POSIX operations like Append.

Typical RDBMS Architecture

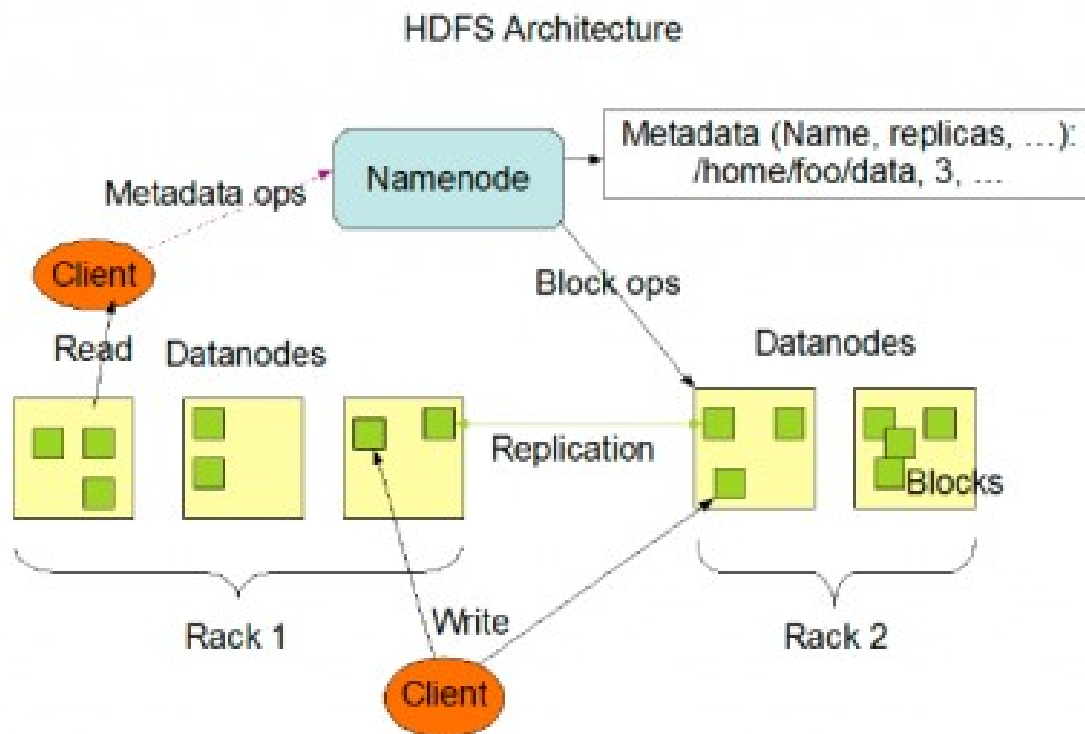


High Level Architecture of Hadoop



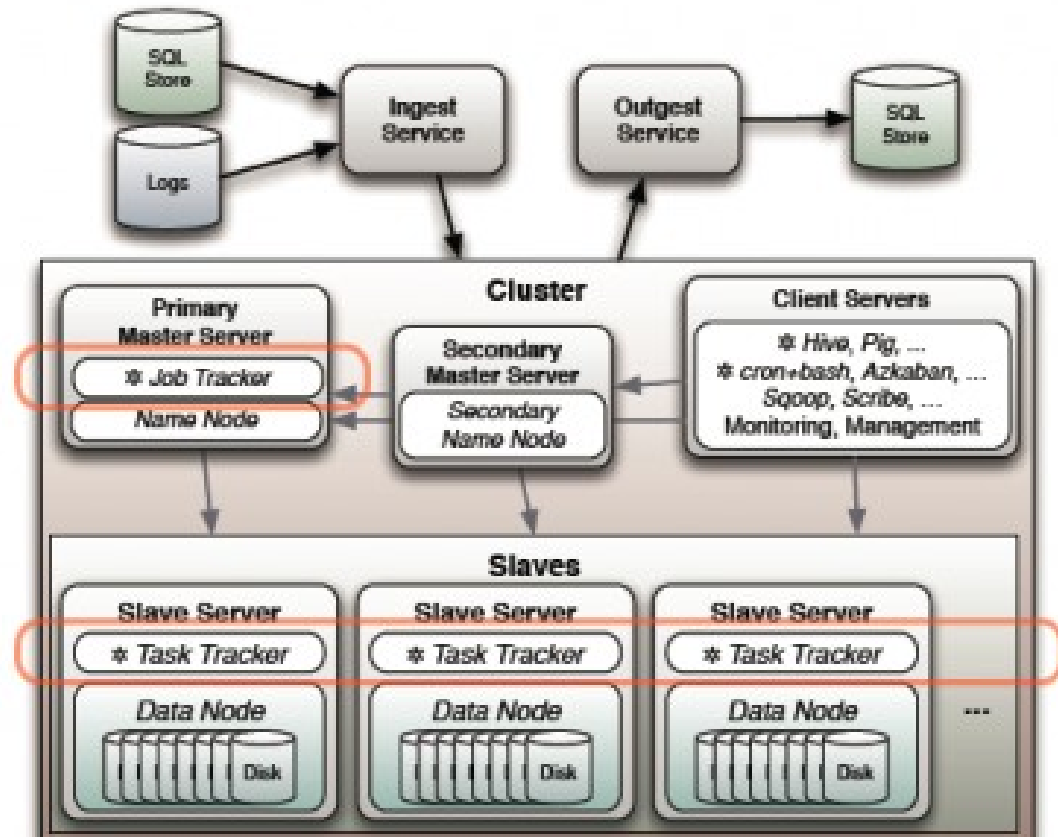
HDFS Terminology

- Namenode
- Datanode
- DFS Client
- Files/Directories
- Replication
- Blocks
- Rack-awareness

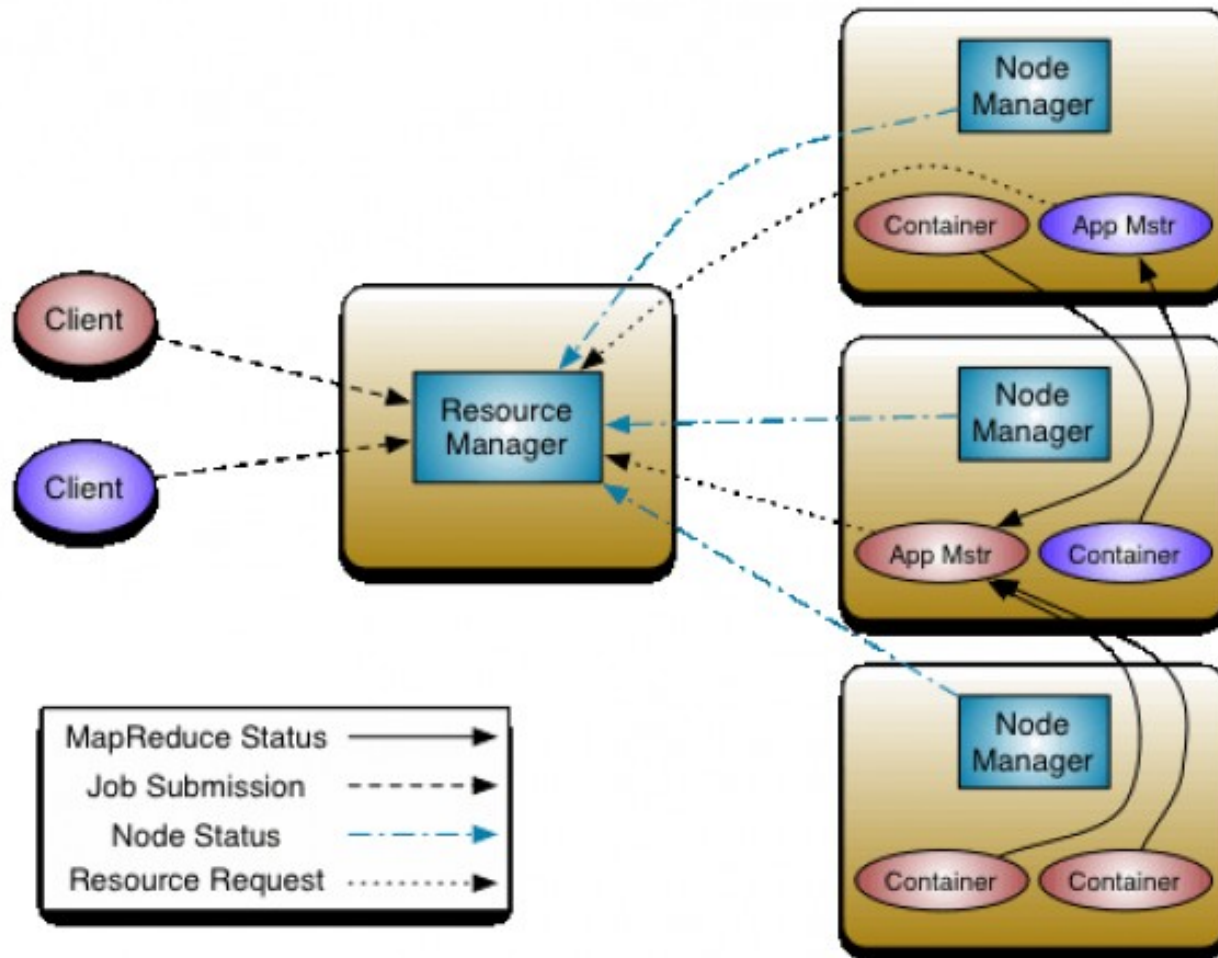


Jobs and Tasks

- Services
- *Job Tracker*
- *Task Trackers*



YARN



Is Hadoop Enough?

Limitations:

- Takes time for processing
- Lack of Incremental processing
- Weak with Graph use cases
- Not very easy to create a processing pipeline (addressed with HIVE, Pig, etc.)
- Too close to programmers
- Faster implementations are possible

Is Hadoop Enough?

Alternatives:

- Apache Drill (<http://drill.apache.org/>)
- Spark (<http://spark.apache.org/>)
- Graph Processors (<http://giraph.apache.org/>)