SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

19IT601– Data Science and Analytics

III Year / VI Semester

Unit 4 – DATA ANALYTICAL FRAMEWORKS

Topic 4: MapReduce

Map Reduce

Faster Processing

- MapReduce is a programming model for writing applications that can process Big Data in parallel on multiple nodes.

- MapReduce provides analytical capabilities for analyzing huge volumes of complex data.

- MapReduce is a processing technique and a program model for distributed computing based on java.

Map Reduce

- The MapReduce algorithm contains two important tasks, namely Map and Reduce.
- The Map task takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key-value pairs).
- The Reduce task takes the output from the Map as an input and combines those data tuples (key-value pairs) into a smaller set of tuples.
- The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes

Map Reduce

MapReduce program executes in three stages, namely

- map stage,
- shuffle stage, and
- reduce stage.

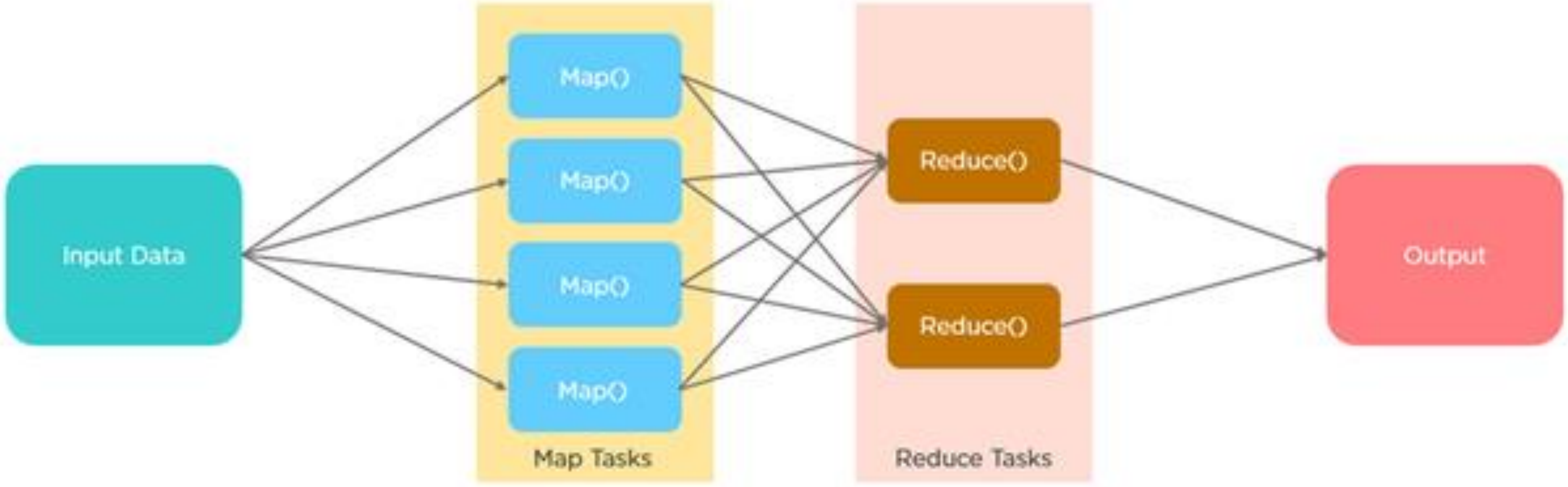
Map stage :

- It process the input data.
- Input data is in the form of file or directory and is stored in the HDFS
- The input file is passed to the mapper function line by line.
- The mapper processes the data and creates several small chunks of data.

Reduce stage:

- This stage is the combination of the Shuffle stage and the Reduce stage.
- It process the data that comes from the maapper.
- After processing, it produces a new set of output, which will be stored in the HDFS.

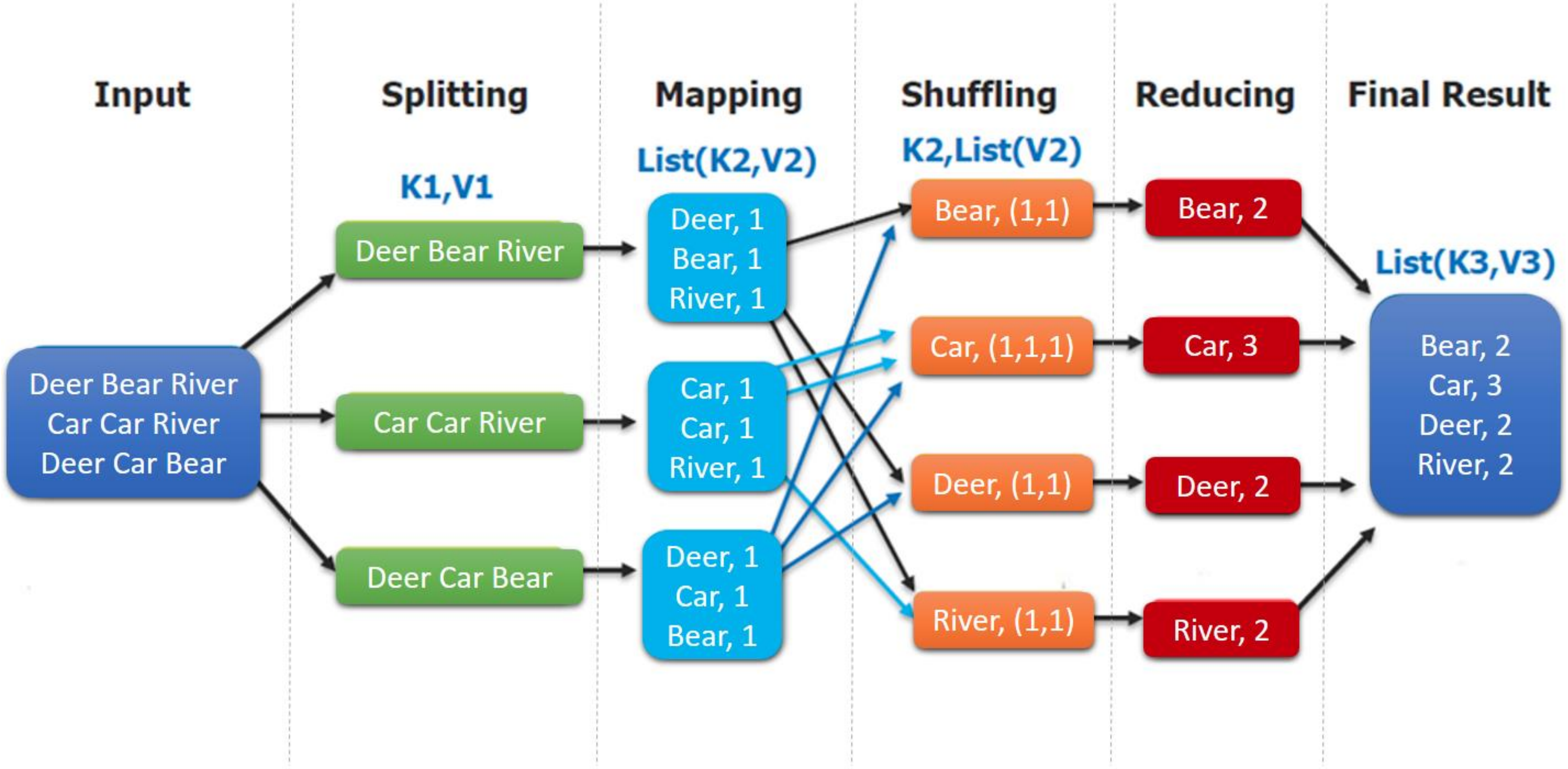
Map Reduce


Input Data
Map()
Map()
Map()
Map()
Map Tasks
Reduce()
Reduce()
Reduce Tasks
Output

Map Reduce

Phases

- Input Phase
- Map
- Intermediate Keys
- Combiner
- Shuffle and Sort
- Reducer
- Output



MapReduce Word Count Process

Map Reduce

Phases

Input Phase – Here we have a Record Reader that translates each record in an input file and sends the parsed data to the mapper in the form of key-value pairs.

Map – Map is a user-defined function, which takes a series of key-value pairs and processes each one of them to generate zero or more key-value pairs.

Intermediate Keys – They key-value pairs generated by the mapper are known as intermediate keys.

Combiner – A combiner is a type of local Reducer that groups similar data from the map phase into identifiable sets. It takes the intermediate keys from the mapper as input and applies a user-defined code to aggregate the values in a small scope of one mapper. It is not a part of the main MapReduce algorithm; it is optional.

Map Reduce

Phases

Shuffle and Sort – The Reducer task starts with the Shuffle and Sort step. It downloads the grouped key value pairs onto the local machine, where the Reducer is running. The individual key-value pairs are sorted by key into a larger data list. The data list groups the equivalent keys together so that their values can be iterated easily in the Reducer task.

Reducer – The Reducer takes the grouped key-value paired data as input and runs a Reducer function on each one of them. Here, the data can be aggregated, filtered, and combined in a number of ways, and it requires a wide range of processing. Once the execution is over, it gives zero or more key-value pairs to the final step.

Output Phase – In the output phase, we have an output formatter that translates the final key-value pairs from the Reducer function and writes them onto a file using a record writer.

Map Reduce

- During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.

- The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.

- Most of the computing takes place on nodes with data on local disks that reduces the network traffic.

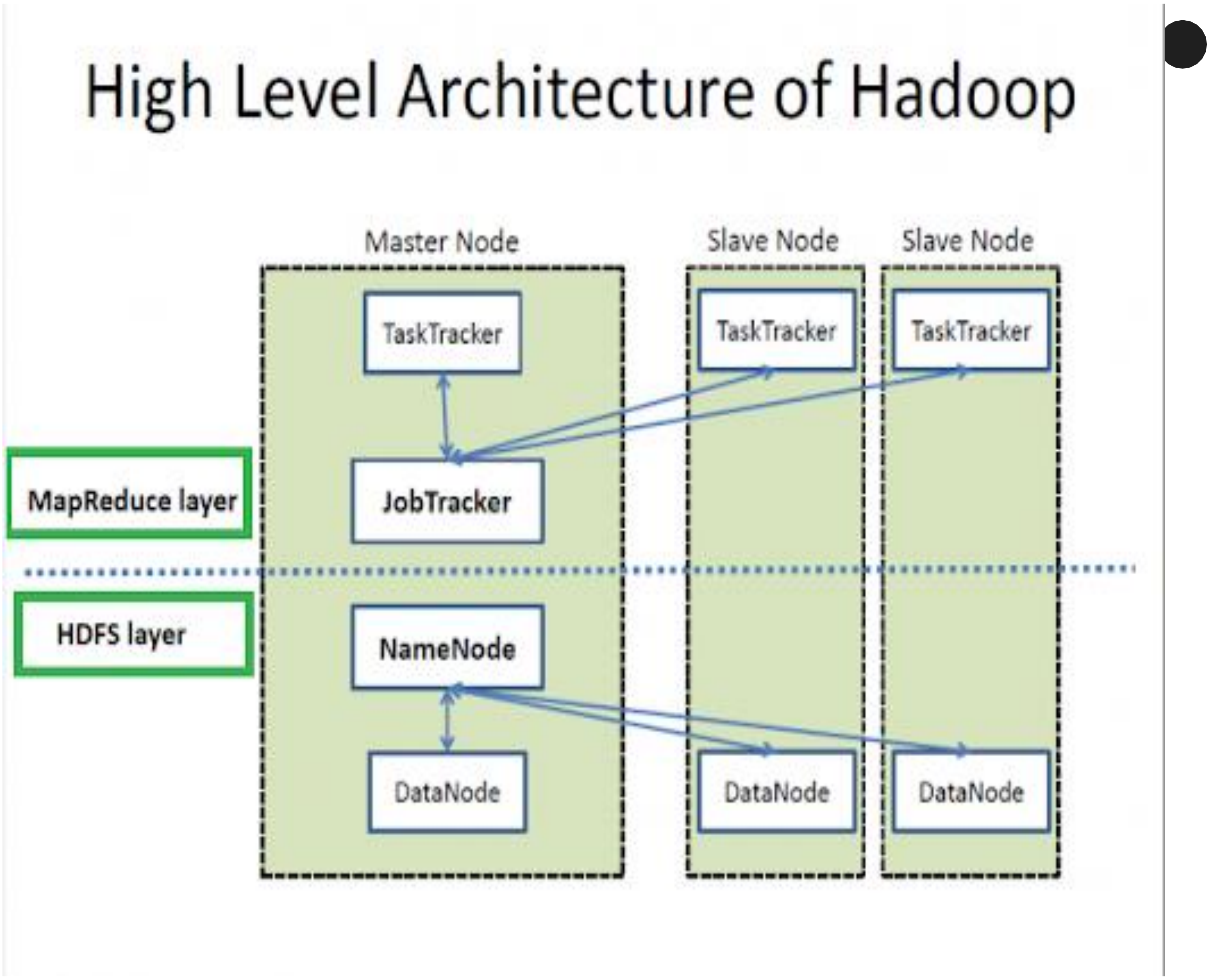
- After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.

- Typically both the input and the output are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

Map Reduce

Execution Environment

- The MapReduce execution environment employs a master/slave execution model,
- One **master node called the Job Tracker** manages a pool of slave computing resources
- **Slaves called Task Trackers** that are called upon to do the actual work.



Map Reduce

The MapReduce framework

•Single master Job Tracker and

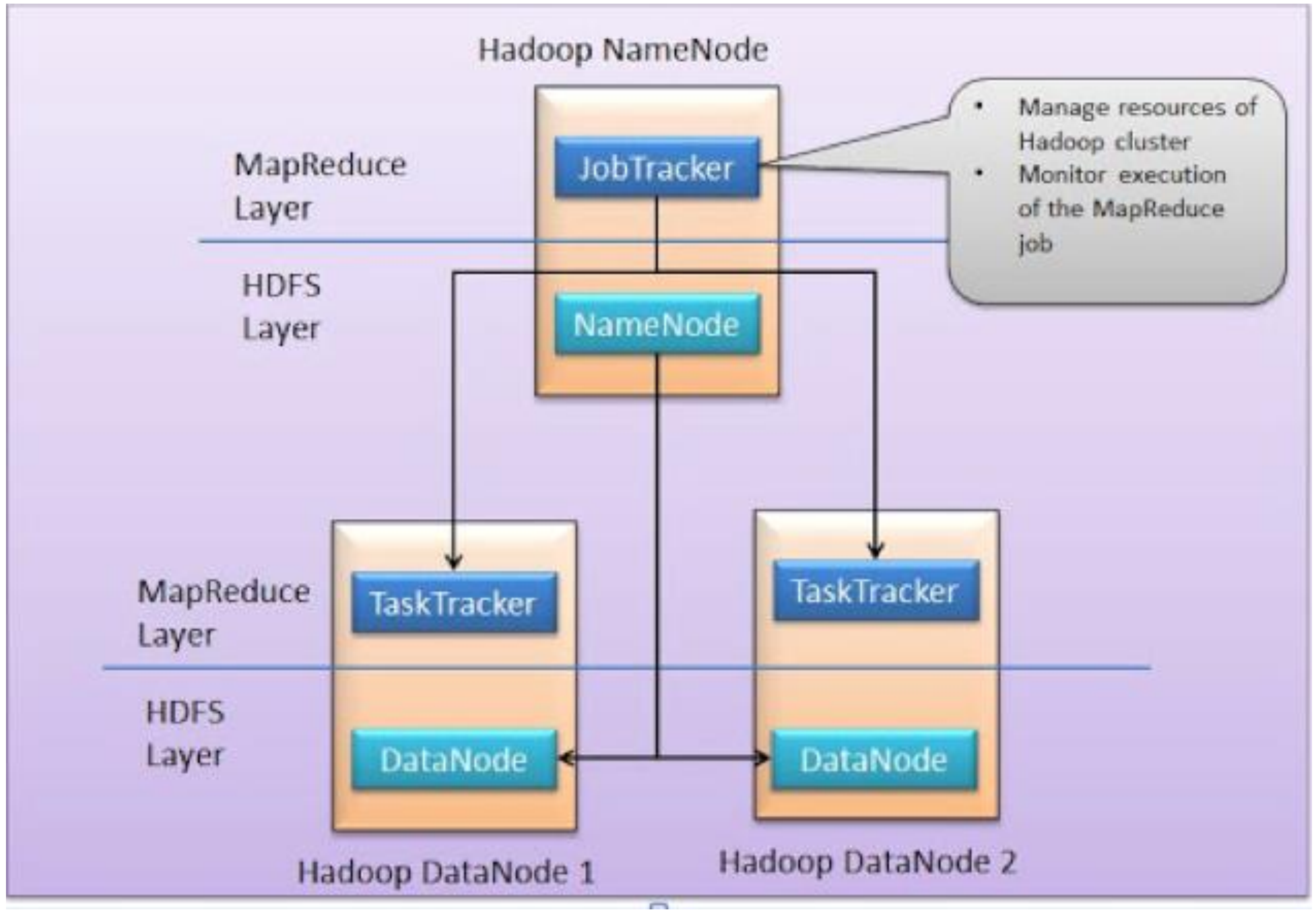
•One slave Task Tracker per cluster-node.

The master Job Tracker

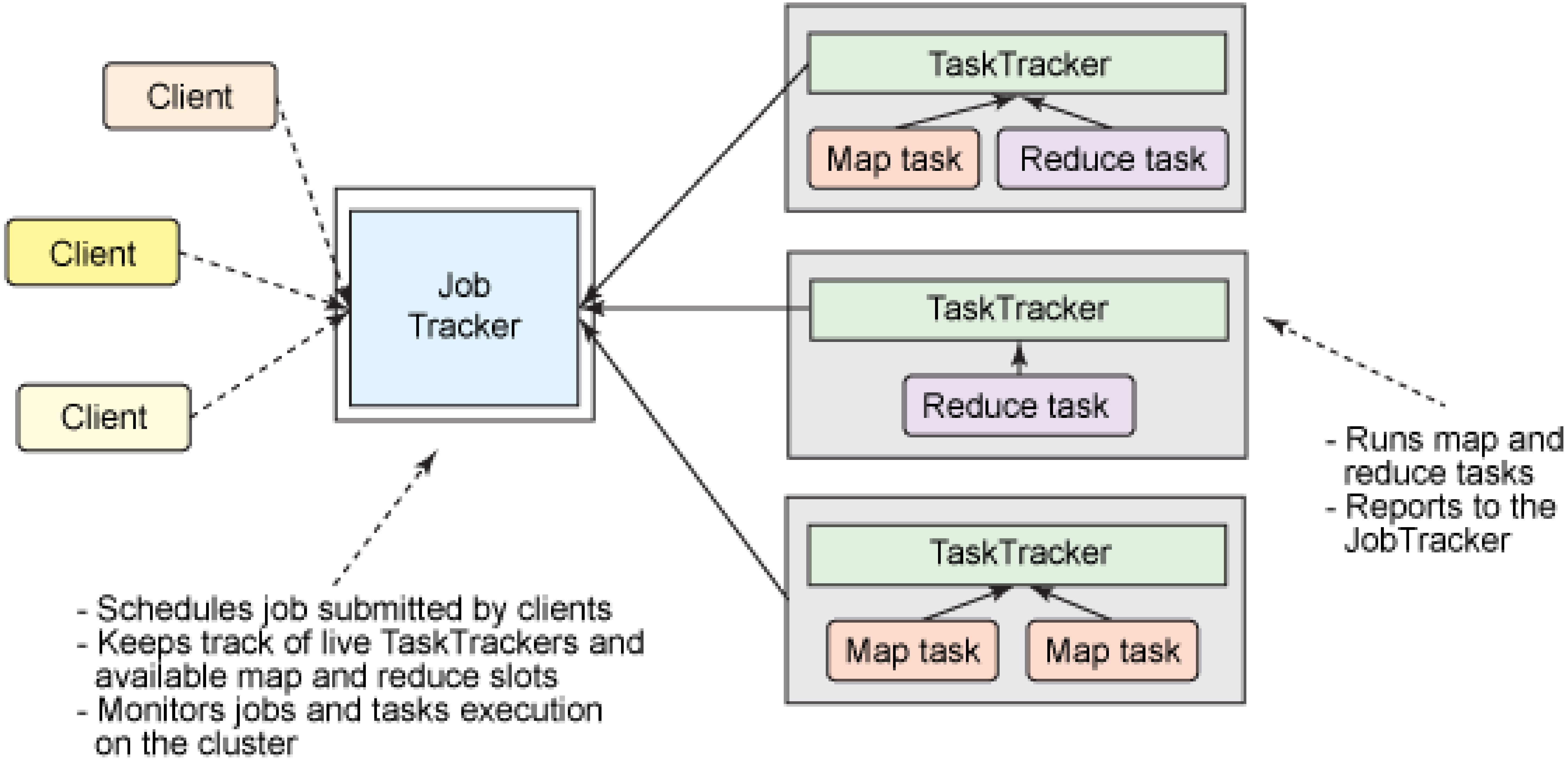
- Responsible for resource management,
- Tracking resource consumption/availability
- Scheduling the jobs component tasks on the slaves,
- Monitoring them and re-executing the failed tasks. The Job Tracker is a single point of failure for the Hadoop MapReduce service which means if Job Tracker goes down, all running jobs are halted

Slaves TaskTracker

- It execute the tasks as directed by the master
- provide task-status information to the master periodically.



Map Reduce



- Schedules job submitted by clients
- Keeps track of live TaskTrackers and available map and reduce slots
- Monitors jobs and tasks execution on the cluster

- Runs map and reduce tasks
- Reports to the JobTracker

THANK YOU