

Map-Reduce

Scheduling and Data Flow

Mining of Massive Datasets
Leskovec, Rajaraman, and Ullman
Stanford University

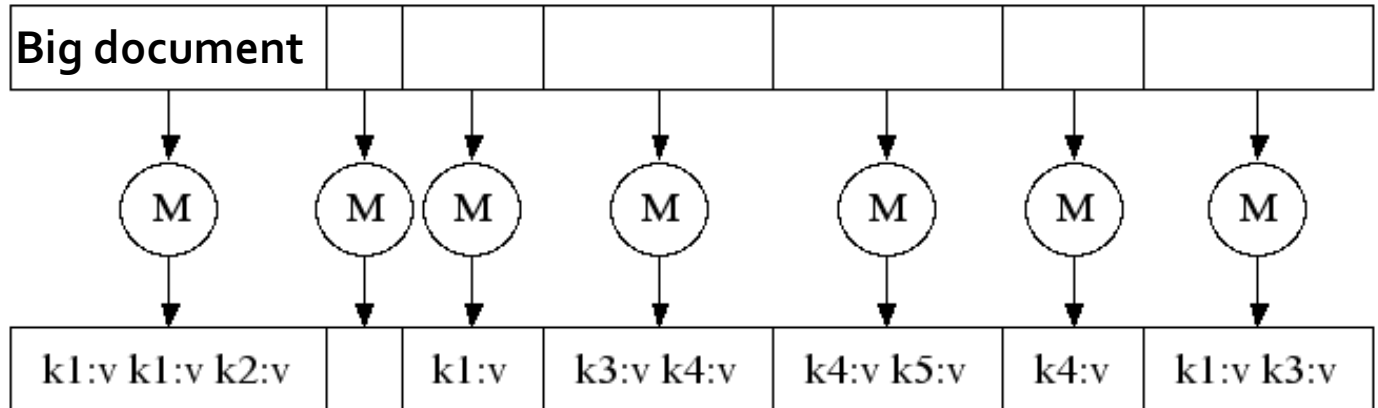


Map-Reduce: A diagram

MAP:

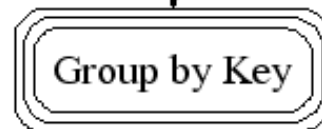
Read input and produces a set of key-value pairs

Input

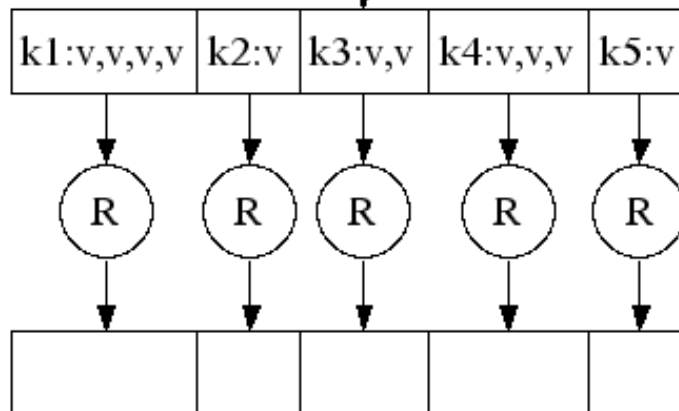


Group by key:

Collect all pairs with same key
(Hash merge, Shuffle, Sort, Partition)



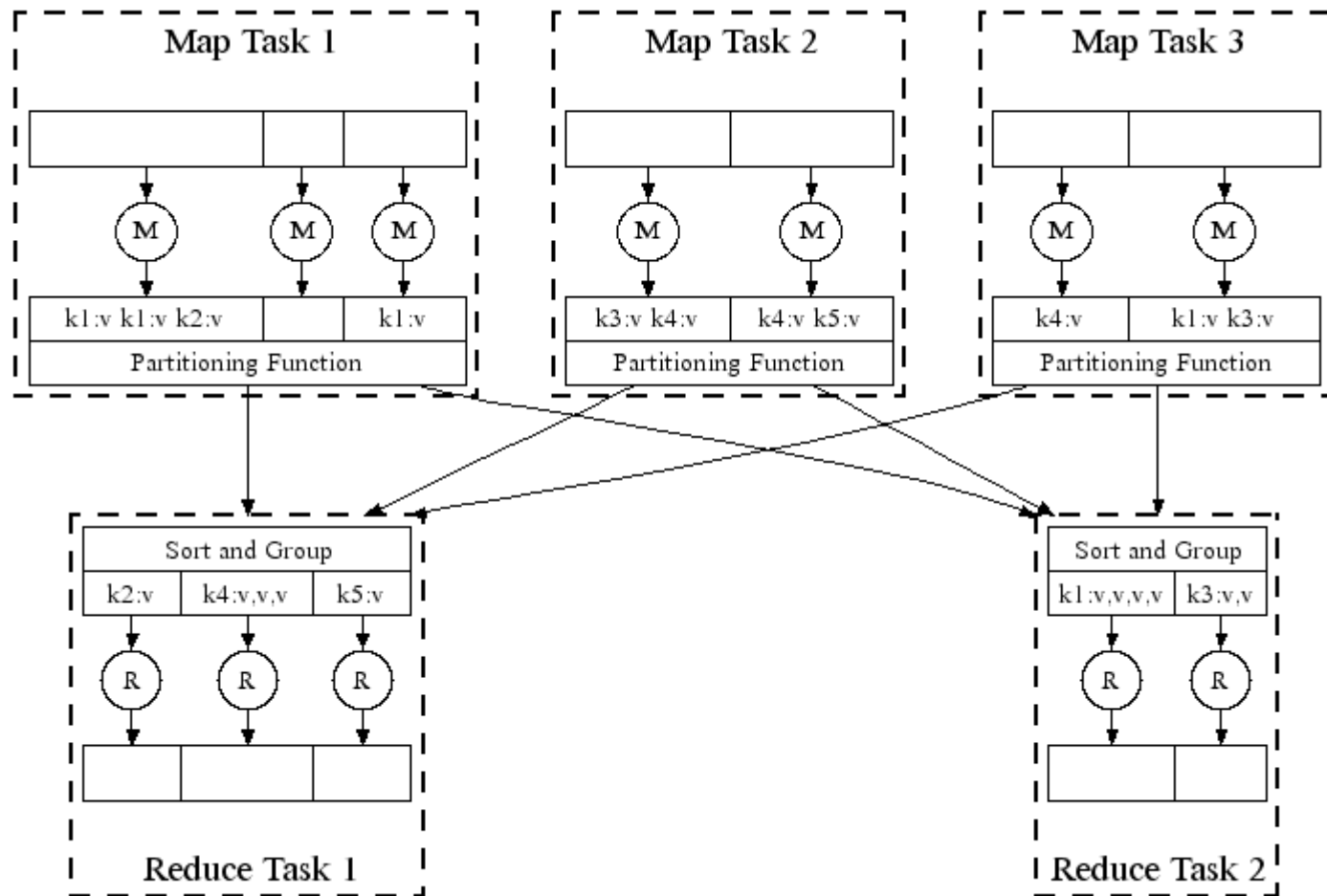
Grouped



Reduce:

Collect all values belonging to the key and output

Map-Reduce: In Parallel



All phases are distributed with many tasks doing the work

Map-Reduce: Environment

Map-Reduce environment takes care of:

- Partitioning the input data
- Scheduling the program's execution across a set of machines
- Performing the **group by key** step
- Handling node failures
- Managing required inter-machine communication

Data Flow

- Input and final output are stored on the distributed file system (DFS):
 - Scheduler tries to schedule map tasks “close” to physical storage location of input data
- Intermediate results are stored on local FS of Map and Reduce workers
- Output is often input to another MapReduce task

Coordination: Master

- **Master node takes care of coordination:**
 - **Task status:** (idle, in-progress, completed)
 - **Idle tasks** get scheduled as workers become available
 - When a map task completes, it sends the master the location and sizes of its R intermediate files, one for each reducer
 - Master pushes this info to reducers
- Master pings workers periodically to detect failures

Dealing with Failures

- **Map worker failure**

- Map tasks completed or in-progress at worker are reset to idle
- Idle tasks eventually rescheduled on other worker(s)

- **Reduce worker failure**

- Only in-progress tasks are reset to idle
- Idle Reduce tasks restarted on other worker(s)

- **Master failure**

- MapReduce task is aborted and client is notified

How many Map and Reduce jobs?

- M map tasks, R reduce tasks
- **Rule of thumb:**
 - Make M much larger than the number of nodes in the cluster
 - One DFS chunk per map is common
 - Improves dynamic load balancing and speeds up recovery from worker failures
- **Usually R is smaller than M**
 - Because output is spread across R files