

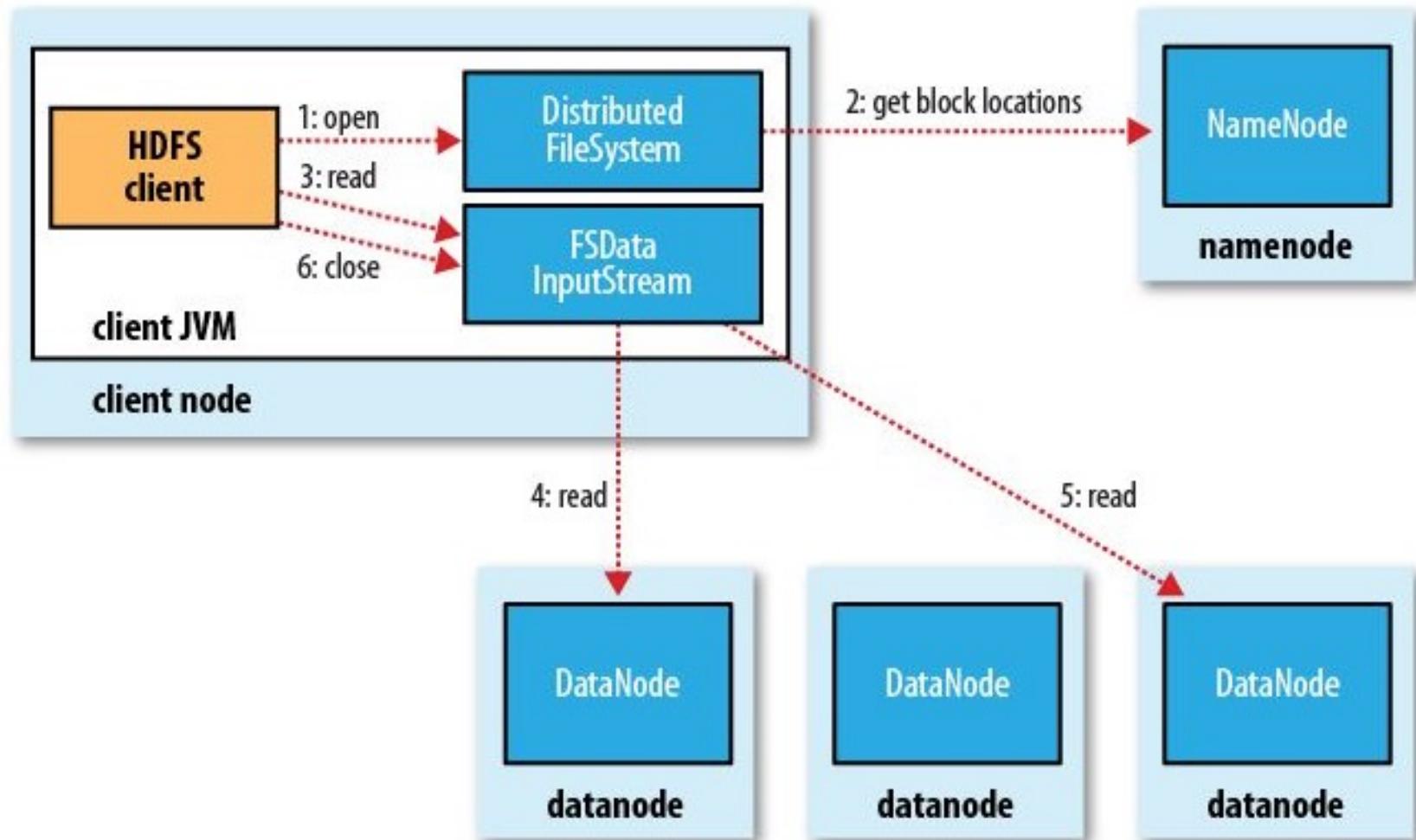
# Spark: HDFS, Yarn, Lab

## Lecture 04-05



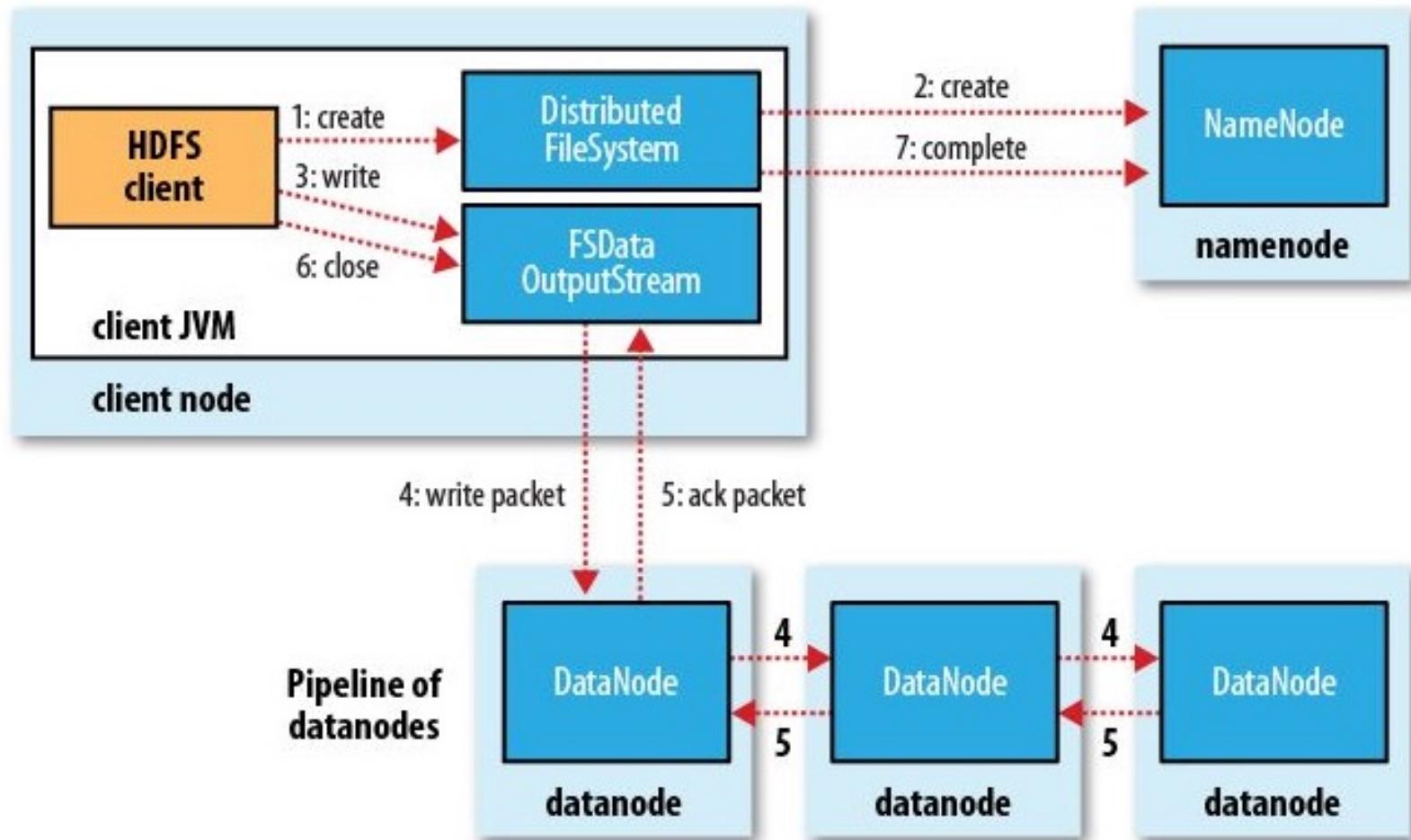
# Hadoop HDFS

## HDFS Read



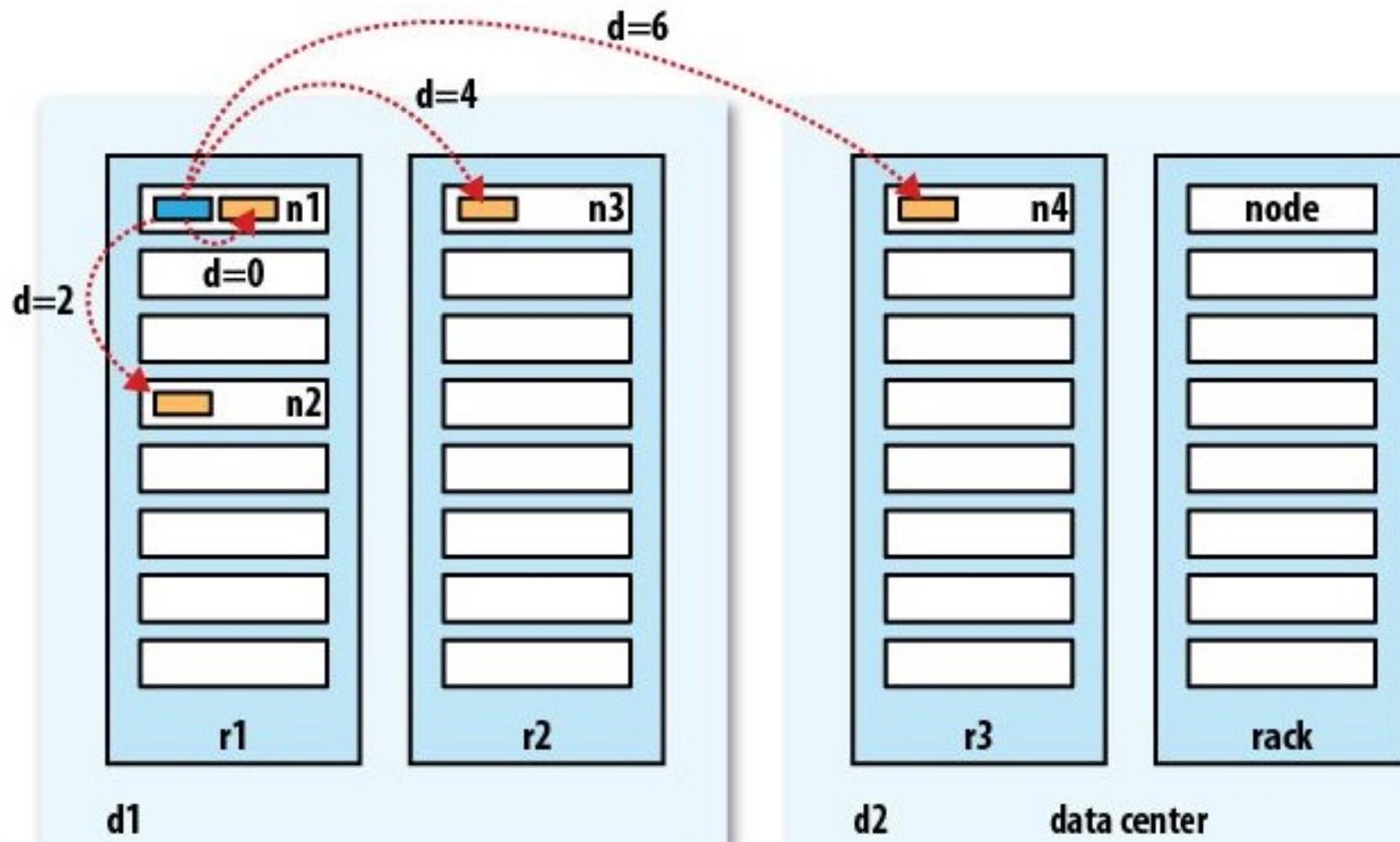
# Hadoop HDFS

## HDFS Write



# Hadoop HDFS

Rack Awareness



# Hadoop HDFS

Network traffic patterns

## Website stack traffic

- Many relatively small requests
- Data flows vertically
- “Sharding” data

[https://en.wikipedia.org/wiki/Shard\\_\(database architecture\)](https://en.wikipedia.org/wiki/Shard_(database_architecture))

## Hadoop stack traffic

- Large continuous data transfers
- Data flows horizontally
- data natively distributed

# Hadoop HDFS

## HDFS Data Integrity

Data can be corrupted during transmission or at rest.

HDFS maintains a CRC-32 for each 512 bytes.

CRC-32 adds an overhead of a little less than 1% of size.

Datanodes verify data coming from clients and other Datanodes during replication.

Also, the “DataNodes” run a background thread called “DataBlockScanner” continuously.

# Hadoop HDFS

HDFS Data Integrity

When the “DataBlockScanner” finds an inconsistency it will use the “replicas” to heal itself.

Hadoop

# HDFS Administration

HDFS Data Integrity

## Cluster Backups

- Typically, we don't have space for backup all the data.

## Solution:

- Backup the “meta-data” of the data from the “NameNodes”.

With popularity of the cloud technology some companies are backing up critical parts using services like Amazon S3 glacier.

Hadoop

# HDFS Administration

HDFS Data Integrity

Periodical File System checks

use command “hdfs fsck -report”

There are options to try to ”fix” things.

I recommend run the ”report” option to detect errors and review results before issuing commands.

# Hadoop

## HDFS Administration

HDFS Adding new nodes.

Adding new nodes currently is done using a Web interface provided by each distribution.

One step that is still manual is to make sure you run the Balancer (`start-balancer.sh`) once you finished to add nodes.

You can add as many nodes and then run the balancer at the end.

# Hadoop

## HDFS Administration

HDFS Removing nodes.

Removing nodes is also done via Web interface.

### **\*\* Important \*\***

In the case of removing more than one "datanode", Its highly recommended to remove one node at time and run the balancer after the removal.

# Hadoop HDFS

## HDFS Globs

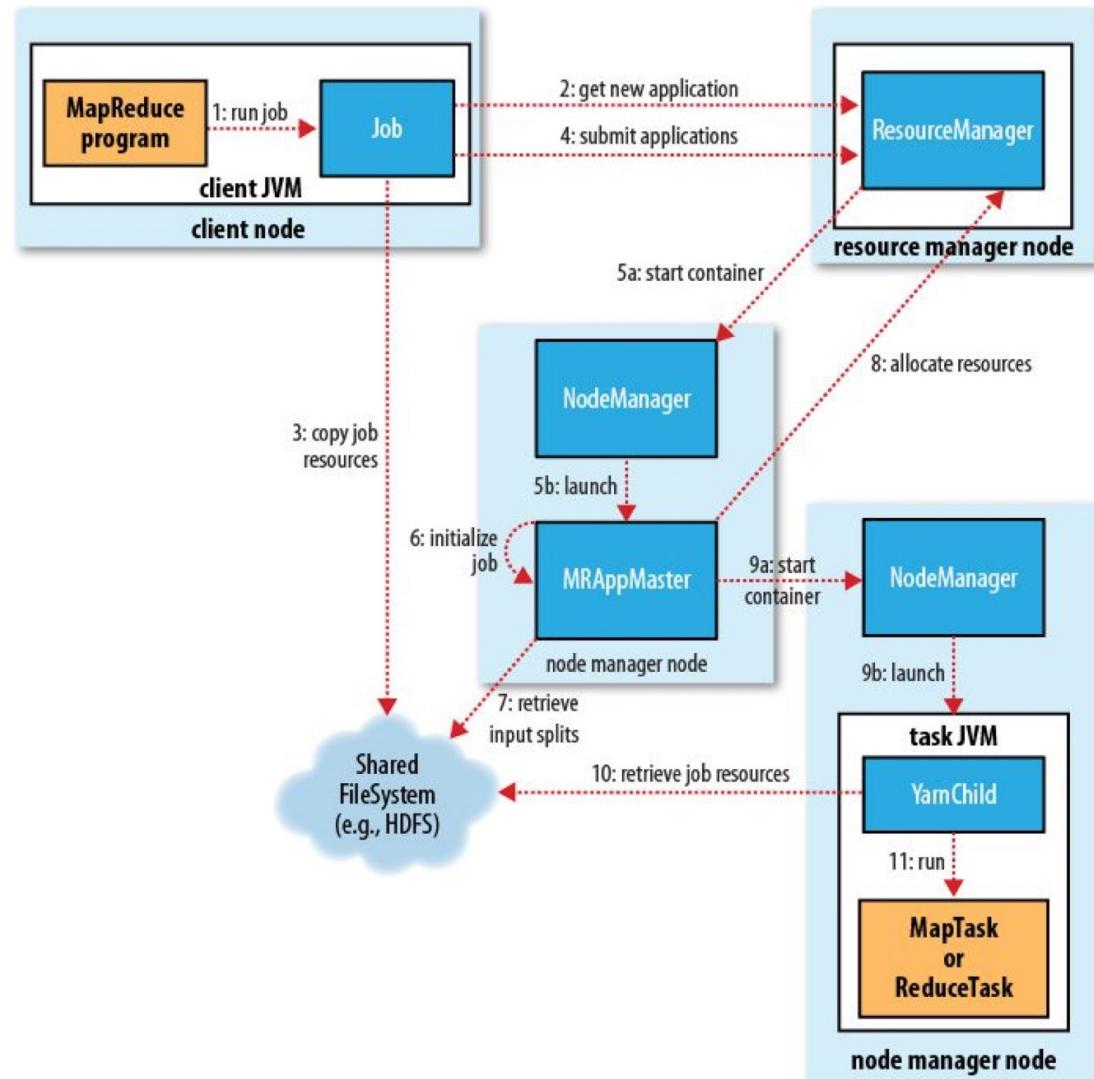
```
/  
└── 2007/  
    └── 12/  
        ├── 30/  
        └── 31/  
    └── 2008/  
        └── 01/  
            ├── 01/  
            └── 02/
```

Glob	Expansion
/*	/2007/2008
/*/*	/2007/12/2008/01
/*/12/*	/2007/12/30/2007/12/31
/200?	/2007/2008
/200[78]	/2007/2008
/200[7-8]	/2007/2008
/200[^01234569]	/2007/2008
/*/*/{31,01}	/2007/12/31/2008/01/01
/*/*/{0,1}	/2007/12/30/2007/12/31
/*/{12/31,01/01}	/2007/12/31/2008/01/01

# Hadoop

## Job Submission & Execution

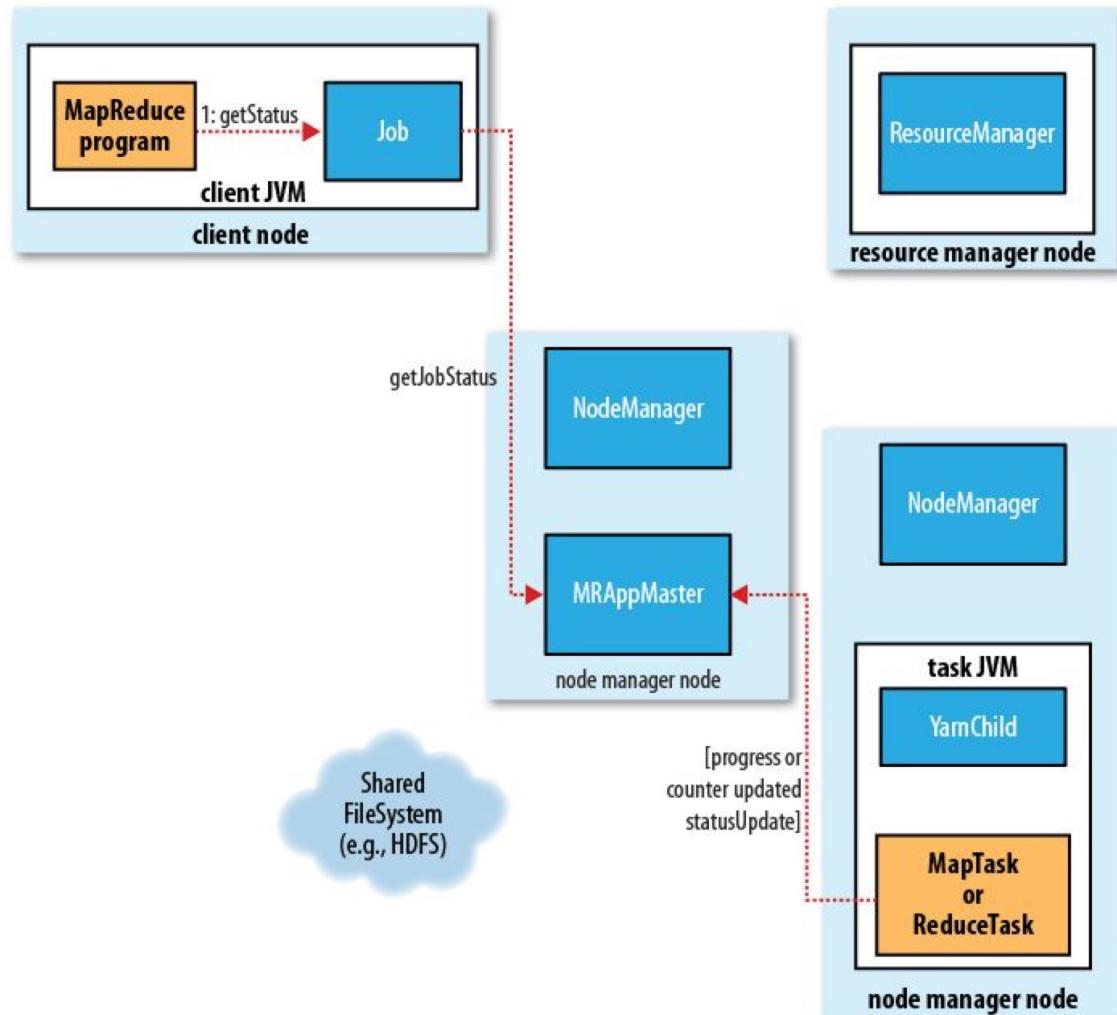
### Map/Reduce V2



# Hadoop

## Job Submission & Execution

### Progress V2



# Hadoop

## Job Submission & Execution

### Types of Job Failures

#### Task Failure

- A task fails, gets restarted and keeps failing until it fails the job.
- Control the number of retries before giving up.  
`mapreduce.map.maxattempts`  
`mapreduce.reduce.maxattempts`
- Control acceptable percentage of failure  
`mapreduce.map.failures.maxpercent`  
`mapreduce.reduce.failures.maxpercent`

# Hadoop

## Job Submission & Execution

### Types of Job Failures

#### App Master Failure

- After failure is restarted or recovered.
- Control the number of retries before giving up.  
`yarn.resourcemanager.am.max-retries`
- Control if retrieve state from previous runs  
`yarn.app.mapreduce.am.job.recovery.enable`

Hadoop

# Job Submission & Execution

Types of Job Failures

## Node Manager Failure

- Node manager fails to send heartbeat
- Blacklisted

# Hadoop

## Job Submission & Execution

### Types of Job Failures

#### Resource Manager Failure

- The most serious
- Property to set a class to handle state  
`yarn.resourcemanager.store.class`
  - Normally kept in memory, but there is also a “Zookeeper” based that retain state.

Hadoop

# Job Submission & Execution

Jobs

## Job Schedulers

- FIFO – First in First out

Jobs executed in sequence of scheduling.

Pros:

Simple

Cons:

One large job blocks everybody.

# Hadoop

## Job Submission & Execution

Jobs

### Job Schedulers

- FIFO – First in First out
  - Recent versions allow to set Job priority.
- This only rearranges jobs waiting on the queue.
- If a large job is running other jobs are still blocked.

# Hadoop

## Job Submission & Execution

Jobs

### Job Schedulers

- Fair Scheduler

Jobs are allocated in pools

- Support preemption of jobs.

New job comes and forces a running job to give up resources.

- Avoids blocking jobs.

# Hadoop

## Job Submission & Execution

Jobs

### Job Schedulers

- Capacity Scheduler

Jobs are allocated in “queue”

Each queue is processed using FIFO w/ Priorities

- Can Create hierarchy.

Ex: prod/critical prod/normal and office.

- Avoids blocking jobs.

Hadoop

## Reduce - Copy

M/R

- First stage of reduce phase

Maps complete at different times. So the copy phase starts the before the “Map” phase is complete.

`mapred.reduce.parallel.copies`: Defines how many threads each reducer daemon can use to pull data from the mapper.

# Hadoop navigation

## Yarn Web UI



### All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Pending
7	0	0	7	0	0 B	8 GB	0 B	0	8	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted
1	0	0	0	0	0

User Metrics for dr.who

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Containers Pending	Containers Reserved	Memory Used	Memory Pending	Memory Reserved	VCores Used	VCores Pending
0	0	0	0	0	0	0	0 B	0 B	0 B	0	0

Show 20 entries Search:

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Progress
<a href="#">application_1518037618377_0008</a>	cloudera	Partitioned Wordcount Job.	MAPREDUCE	root.cloudera	Sat Feb 10 21:53:49 -0800 2018	Sat Feb 10 21:54:21 -0800 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	
<a href="#">application_1518037618377_0007</a>	cloudera	Partitioned Wordcount Job.	MAPREDUCE	root.cloudera	Sat Feb 10 21:52:49 -0800 2018	Sat Feb 10 21:53:22 -0800 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	
<a href="#">application_1518037618377_0006</a>	cloudera	Partitioned Wordcount Job.	MAPREDUCE	root.cloudera	Sat Feb 10 21:19:15 -0800 2018	Sat Feb 10 21:19:41 -0800 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	
<a href="#">application_1518037618377_0004</a>	cloudera	Plain Wordcount Job	MAPREDUCE	root.cloudera	Sat Feb 10 21:07:39 -0800 2018	Sat Feb 10 21:08:07 -0800 2018	FINISHED	SUCCEEDED	N/A	N/A	N/A	

Clicking on one application link takes to the application run page.

# Hadoop navigation

## Yarn Web UI

Logged in as: dr.who

**hadoop**

**Application Overview**

User: cloudera
Name: Partitioned Wordcount Job.
Application Type: MAPREDUCE
Application Tags:
State: FINISHED
FinalStatus: SUCCEEDED
Started: Sat Feb 10 21:53:49 -0800 2018
Elapsed: 31sec
Tracking URL: <a href="#">History</a>
Diagnostics:

**Application Metrics**

Total Resource Preempted: <memory:0, vCores:0>
Total Number of Non-AM Containers Preempted: 0
Total Number of AM Containers Preempted: 0
Resource Preempted from Current Attempt: <memory:0, vCores:0>
Number of Non-AM Containers Preempted from Current Attempt: 0
Aggregate Resource Allocation: 132503 MB-seconds, 88 vcore-seconds

**ApplicationMaster**

Attempt Number	Start Time	Node	Logs
1	Sat Feb 10 21:53:49 -0800 2018	quickstart.cloudera:8042	<a href="#">logs</a>

"History" link shows more details.

Application Master Node

# Hadoop navigation

## Yarn Web UI



### MapReduce Job job\_1518037618377\_0008

Logged in as: dr.who

Application

- Job

- Overview
- Counters
- Configuration
- Map tasks
- Reduce tasks

Tools

Job Overview

Job Name: Partitioned Wordcount Job.  
User Name: cloudera  
Queue: root.cloudera  
State: SUCCEEDED  
Uberized: false  
Submitted: Sat Feb 10 21:53:49 PST 2018  
Started: Sat Feb 10 21:53:56 PST 2018  
Finished: Sat Feb 10 21:54:21 PST 2018  
Elapsed: 24sec  
Diagnostics:  
Average Map Time 13sec  
Average Shuffle Time 5sec  
Average Merge Time 0sec  
Average Reduce Time 0sec

**ApplicationMaster**

Attempt Number	Start Time	Node	Logs
1	Sat Feb 10 21:53:51 PST 2018	quickstart.cloudera:8042	logs

Task Type	Total	Complete
Map	3	3
Reduce	1	1

Attempt Type	Failed	Killed	Successful
Maps	0	0	3
Reduces	0	0	1

The diagram shows a blue curved arrow originating from the 'Maps' link under the 'Attempt Type' section of the ApplicationMaster table and pointing towards the 'Maps' link under the same section of the 'Attempt Type' table below.

"Maps" or Reduce links  
Show the tasks.

# Hadoop navigation

## Yarn Web UI

Logged in as: dr.who

**Map Tasks for job\_1518037618377\_0008**

Task							Successful Attempt		
Name	State	Start Time	Finish Time	Elapsed Time	Start Time	Finish Time	Elapsed Time		
<a href="#">task_1518037618377_0008_m_000000</a>	SUCCEEDED	Sat Feb 10 21:53:59 -0800 2018	Sat Feb 10 21:54:12 -0800 2018	12sec	Sat Feb 10 21:53:59 -0800 2018	Sat Feb 10 21:54:12 -0800 2018	12sec		
<a href="#">task_1518037618377_0008_m_000001</a>	SUCCEEDED	Sat Feb 10 21:54:00 -0800 2018	Sat Feb 10 21:54:14 -0800 2018	14sec	Sat Feb 10 21:54:00 -0800 2018	Sat Feb 10 21:54:14 -0800 2018	14sec		
<a href="#">task_1518037618377_0008_m_000002</a>	SUCCEEDED	Sat Feb 10 21:54:01 -0800 2018	Sat Feb 10 21:54:15 -0800 2018	13sec	Sat Feb 10 21:54:01 -0800 2018	Sat Feb 10 21:54:15 -0800 2018	13sec		

Showing 1 to 3 of 3 entries      First Previous 1 Next Last

Lists the tasks for a particular map.  
Clicking on the “task” link...

# Hadoop navigation

## Yarn Web UI

The screenshot shows the Yarn Web UI interface. On the left, there's a sidebar with a yellow elephant logo and the word "hadoop". The main area is titled "Attempts for task\_1518037618377\_0008\_m\_000000". The page displays a table of task attempts. The first attempt listed is "attempt\_1518037618377\_0008\_m\_000000\_0", which is in the "SUCCEEDED" state, running as a "map" task on node "/default-rack/quickstart.cloudera:8042". The "Logs" link for this attempt is highlighted with a blue arrow. The table has columns for Attempt, State, Status, Node, Logs, Start Time, Finish Time, Elapsed Time, and Note.

Attempt	State	Status	Node	Logs	Start Time	Finish Time	Elapsed Time	Note
attempt_1518037618377_0008_m_000000_0	SUCCEEDED	map	/default-rack/quickstart.cloudera:8042	<a href="#">Logs</a>	Sat Feb 10 21:53:59 -0800 2018	Sat Feb 10 21:54:12 -0800 2018	12sec	

Showing 1 to 1 of 1 entries

We can see the logs for a particular attempt on the 'logs' Link.

Shows information about task execution attempts.  
In this case the first attempt succeed (end with '\_0')

# Hadoop navigation

## Yarn Web UI



▼ Application
<a href="#">About</a>
<a href="#">Jobs</a>
▶ Tools

Log Type: stderr  
Log Upload Time: Sat Feb 10 21:54:29 -0800 2018  
Log Length: 0

Log Type: stdout  
Log Upload Time: Sat Feb 10 21:54:29 -0800 2018  
Log Length: 0

Log Type: syslog  
Log Upload Time: Sat Feb 10 21:54:29 -0800 2018  
Log Length: 3687

```
2018-02-10 21:54:02,237 WARN [main] org.apache.hadoop.metrics2.impl.MetricsConfig: Cannot locate configuration: tried hadoop-metrics2-maptask.properties,hadoop-metrics2.properties
2018-02-10 21:54:02,639 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: Scheduled snapshot period at 10 second(s).
2018-02-10 21:54:02,631 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: MapTask metrics system started
2018-02-10 21:54:02,674 INFO [main] org.apache.hadoop.mapred.YarnChild: Executing with tokens:
2018-02-10 21:54:02,674 INFO [main] org.apache.hadoop.mapred.YarnChild: Kind: mapreduce.job, Service: job_1518037618377_0008, Ident: (org.apache.hadoop.mapreduce.security.token.JobTokenIdentifier@1e75d892)
2018-02-10 21:54:03,675 INFO [main] org.apache.hadoop.mapred.YarnChild: Sleeping for 0ms before retrying again. Got null now.
2018-02-10 21:54:05,523 INFO [main] org.apache.hadoop.mapred.YarnChild: mapreduce.cluster.local.dir for child: /var/lib/hadoop-yarn/cache/yarn/nm-local-dir/usercache/cloudera/appcache/application_1518037618377_0008
2018-02-10 21:54:07,888 INFO [main] org.apache.hadoop.conf.Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
2018-02-10 21:54:09,896 INFO [main] org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter: File Output Committer Algorithm version is 1
2018-02-10 21:54:09,897 INFO [main] org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2018-02-10 21:54:09,981 INFO [main] org.apache.hadoop.mapred.Task: Using ResourceCalculatorProcessTree : []
2018-02-10 21:54:11,373 INFO [main] org.apache.hadoop.mapred.MapTask: Processing split: hdfs://quickstart.cloudera:8020/data/wordcount/input/HadoopPoem1.txt@0+136
2018-02-10 21:54:12,142 INFO [main] org.apache.hadoop.mapred.MapTask: (EQUATOR) 0 kv=26214396(104857584)
2018-02-10 21:54:12,142 INFO [main] org.apache.hadoop.mapred.MapTask: mapreduce.task.io.sort.mb: 100
2018-02-10 21:54:12,142 INFO [main] org.apache.hadoop.mapred.MapTask: soft limit at 83886688
2018-02-10 21:54:12,142 INFO [main] org.apache.hadoop.mapred.MapTask: bufstart = 0; bufvoid = 104857600
2018-02-10 21:54:12,142 INFO [main] org.apache.hadoop.mapred.MapTask: kvstart = 26214396; length = 6553669
2018-02-10 21:54:12,160 INFO [main] org.apache.hadoop.mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
2018-02-10 21:54:12,287 INFO [main] org.apache.hadoop.mapred.MapTask: Starting flush or map output
2018-02-10 21:54:12,287 INFO [main] org.apache.hadoop.mapred.MapTask: Spilling map output
2018-02-10 21:54:12,287 INFO [main] org.apache.hadoop.mapred.MapTask: bufstart = 0; bufend = 216; bufvoid = 104857600
2018-02-10 21:54:12,287 INFO [main] org.apache.hadoop.mapred.MapTask: kvstart = 26214396(104857584); kvend = 26214388(104857232); length = 89/6553600
2018-02-10 21:54:12,299 INFO [main] org.apache.hadoop.mapred.MapTask: Finished spill 0
2018-02-10 21:54:12,323 INFO [main] org.apache.hadoop.mapred.Task: Task@attempt_1518037618377_0008_m_000000_0 is done. And is in the process of committing
2018-02-10 21:54:12,645 INFO [main] org.apache.hadoop.mapred.Task: Task 'attempt_1518037618377_0008_m_000000_0' is done.
2018-02-10 21:54:12,646 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: Stopping MapTask metrics system...
2018-02-10 21:54:12,646 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: MapTask metrics system stopped.
2018-02-10 21:54:12,647 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: MapTask metrics system shutdown complete.
```

Notice that in a real cluster these logs are being produced in the node that executed that attempt.

Logged in as:

Q & A