hadoop

Understanding & Planning Hadoop Cluster

Contents

1. What is Node?
2. Hard disk of a Hadoop Node
3. What is a DC, Cluster?
4. Hadoop / CDH Versions
5. Hadoop Architecture
6. HDFS Workflow
7. HDFS Write
8. HDFS Read
9. Hadoop General Planning Considerations

What is a Node?

Foreground processes/ Applns

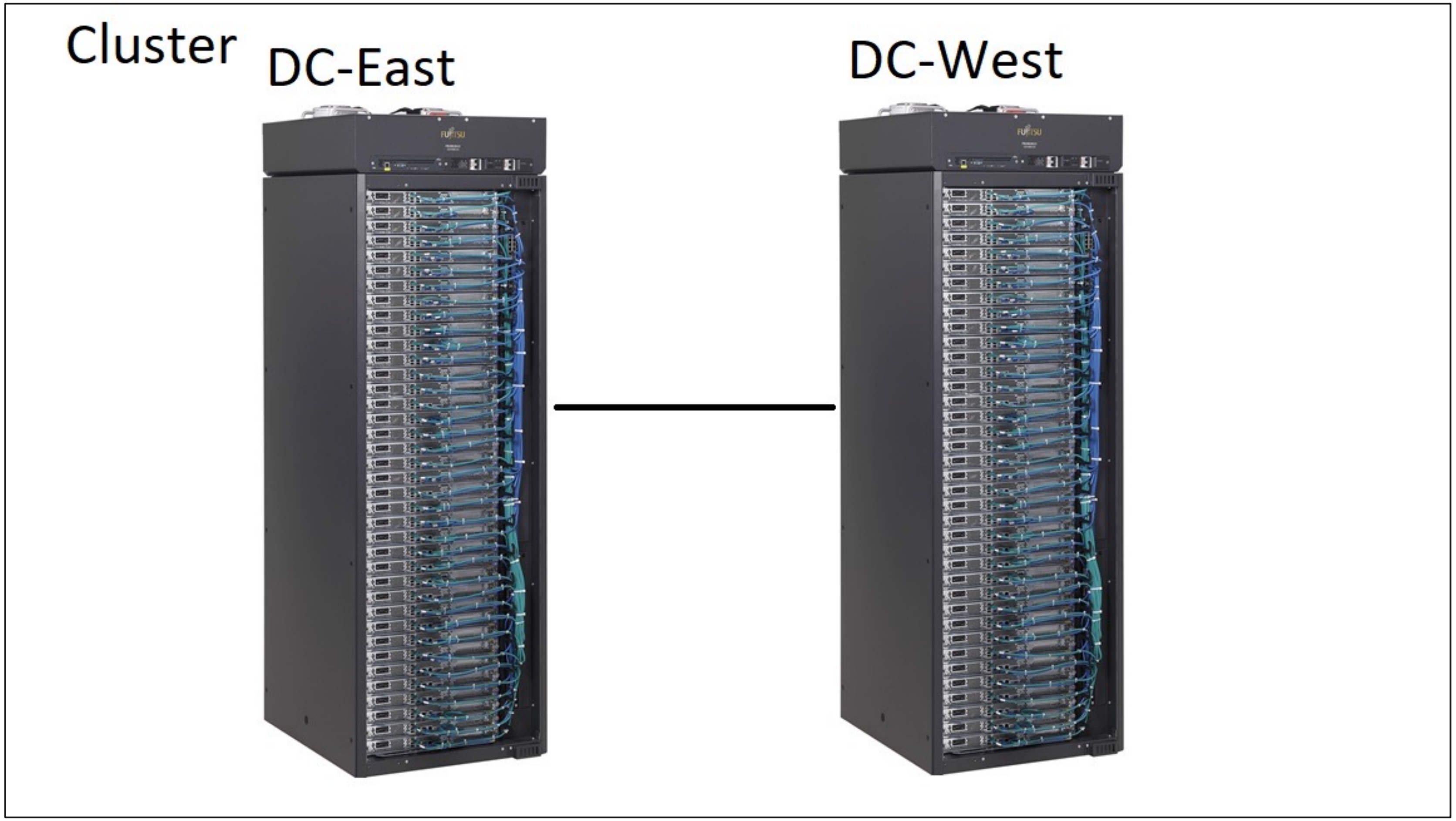
Background processes / Daemons

Operating System

Hardware

Hard Disk of a Hadoop Node

Understanding Hadoop Cluster



Hadoop / CDH Versions

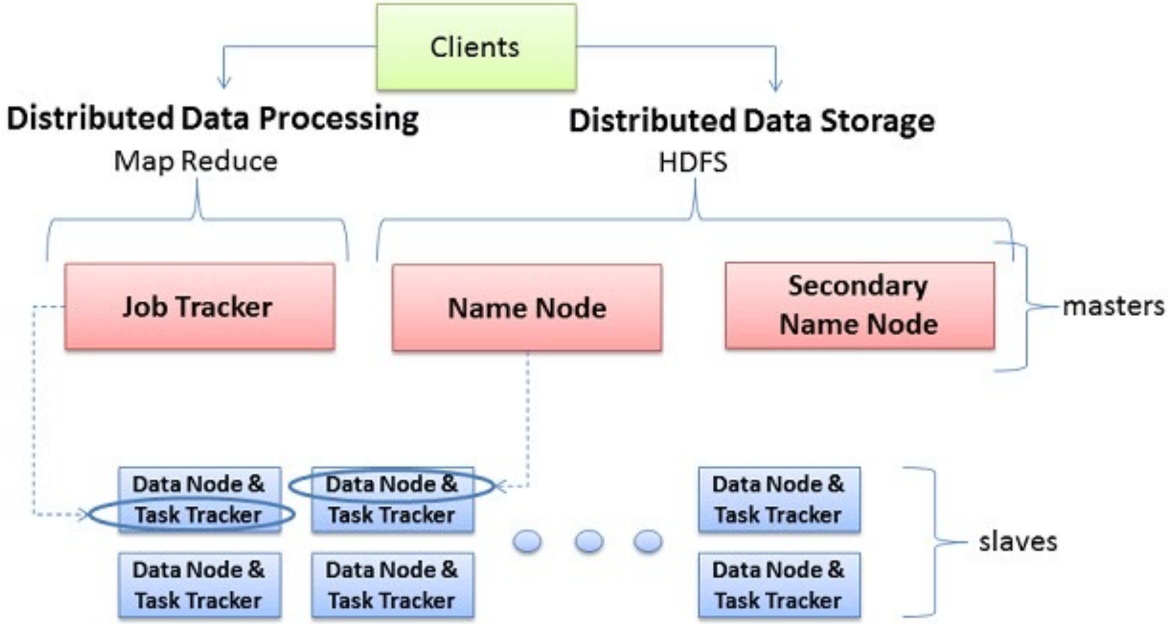
Hadoop	Cloudera(CDH)
0.20	<CDH 4.0
2.0	CDH 4.x
2.3	CDH5.0
2.6	CDH5.4,5.5,5.7
2.7	CDH 5.16
3.0	>CDH 6.0

Understanding Hadoop Cluster

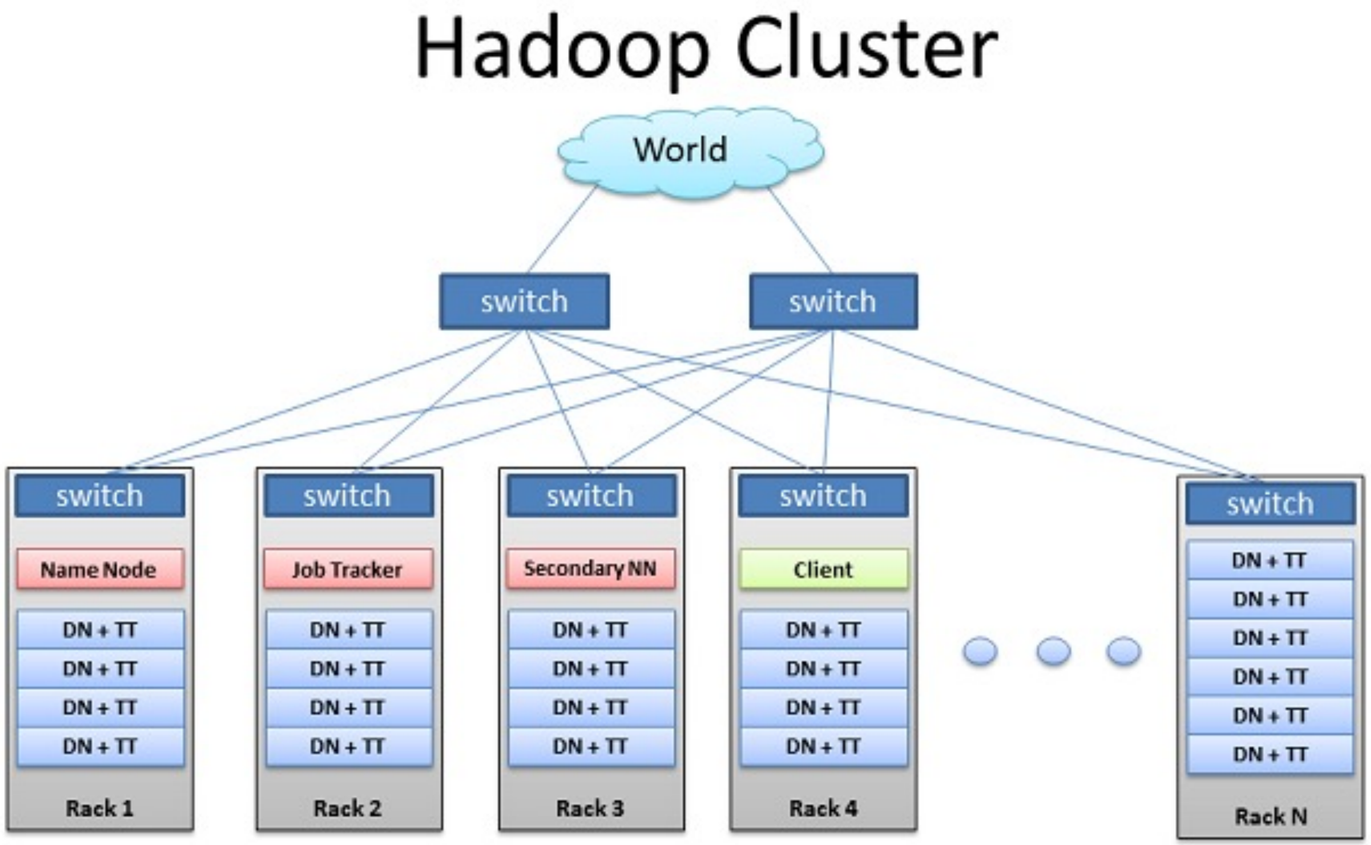
3 Major Categories

- Clients
- Master Nodes
- Slave Nodes

Hadoop Server Roles



Understanding Hadoop Cluster

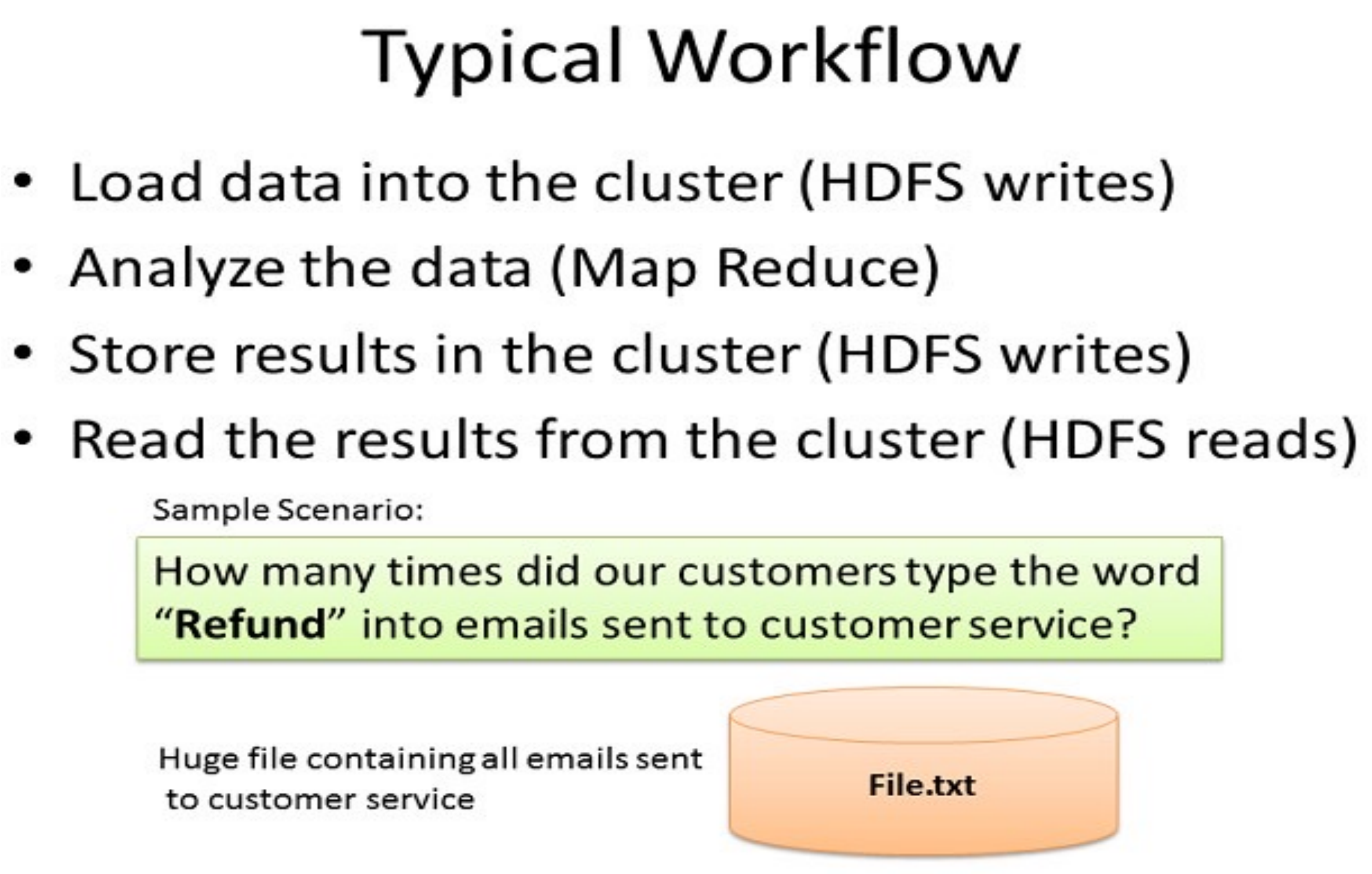


This is the typical architecture of a Hadoop cluster

Understanding Hadoop Cluster

Property File	Property Name/Purpose
hdfs-default.xml	configurations for HDFS **Ex:** dfs.datanode.data.dir, dfs.namenode.name.dir, dfs.replication
mapred-default.xml	configuration for MR **Ex:** mapreduce.jobtracker.http.address
core-default.xml	configuration across the cluster **Ex:** hadoop.tmp.dir, fs.default.name
hdfs-site.xml	Override storage(HDFS) specific properties
mapred-site.xml	Override any processing(MR) specific properties
core-site.xml	Override properties other than above 2

Understanding Hadoop Cluster



Typical Workflow

- Load data into the cluster (HDFS writes)
- Analyze the data (Map Reduce)
- Store results in the cluster (HDFS writes)
- Read the results from the cluster (HDFS reads)

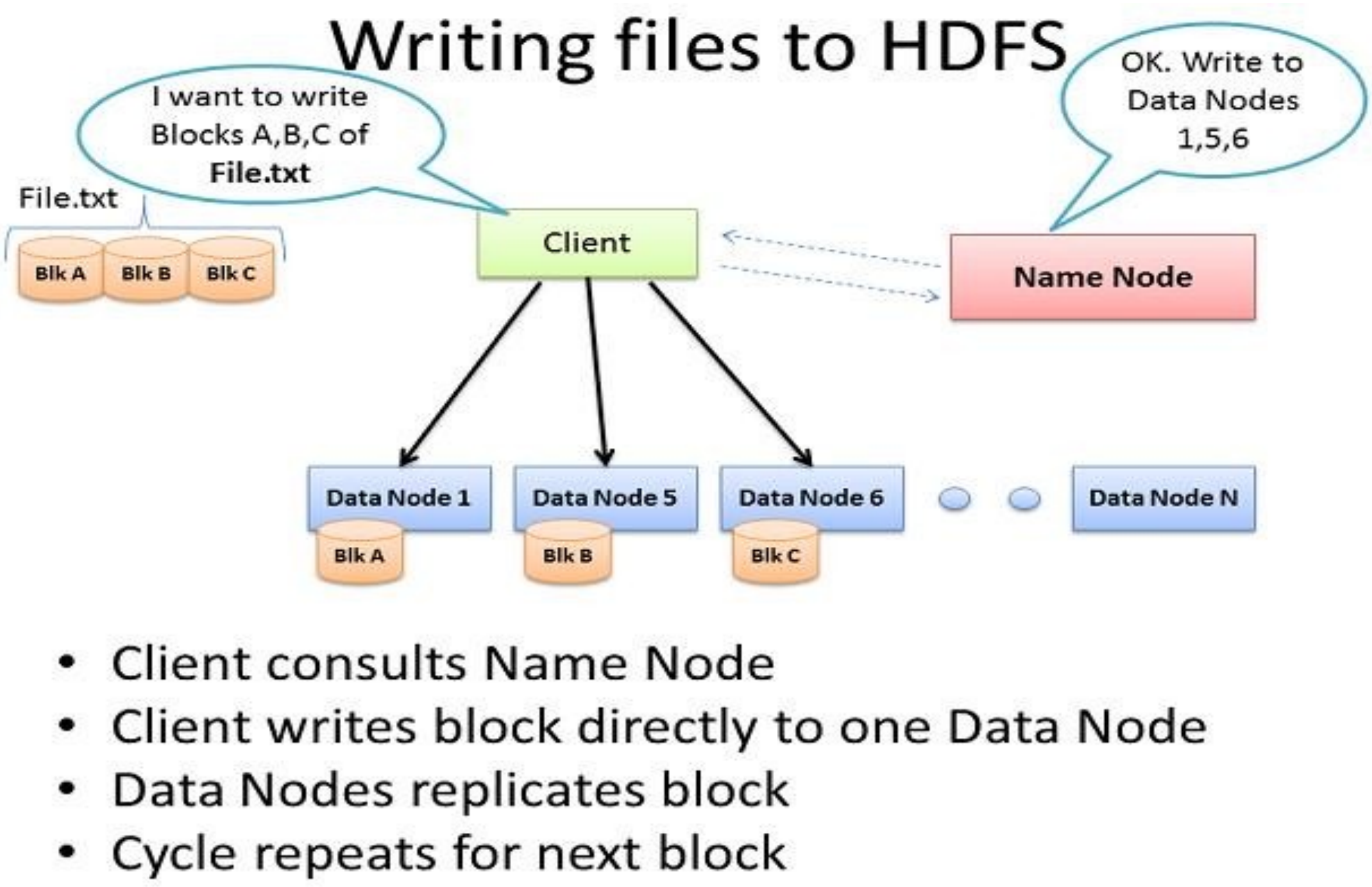
Sample Scenario:

How many times did our customers type the word "**Refund**" into emails sent to customer service?

Huge file containing all emails sent to customer service

File.txt

Understanding Hadoop Cluster

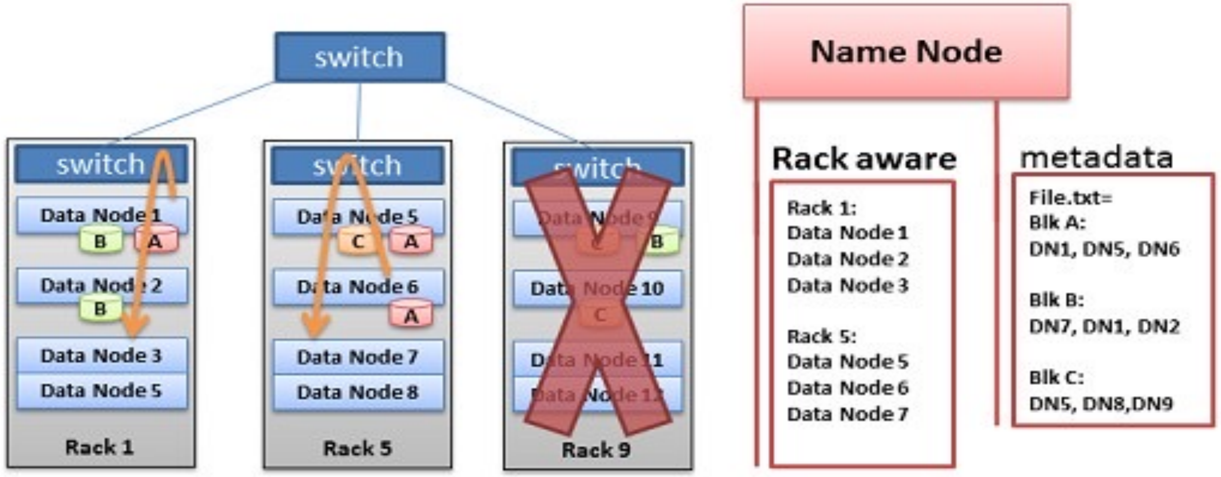


Writing files to HDFS

- Client consults Name Node
- Client writes block directly to one Data Node
- Data Nodes replicates block
- Cycle repeats for next block

Understanding Hadoop Cluster

Hadoop Rack Awareness – Why?



Rack aware

Rack 1:
Data Node 1
Data Node 2
Data Node 3

Rack 5:
Data Node 5
Data Node 6
Data Node 7

metadata

File.txt=
Blk A:
DN1, DN5, DN6

Blk B:
DN7, DN1, DN2

Blk C:
DN5, DN8,DN9

- Never loose all data if entire rack fails
- Keep bulky flows in-rack when possible
- Assumption that in-rack is higher bandwidth, lower latency

Understanding Hadoop Cluster

Name node replica placement Strategy

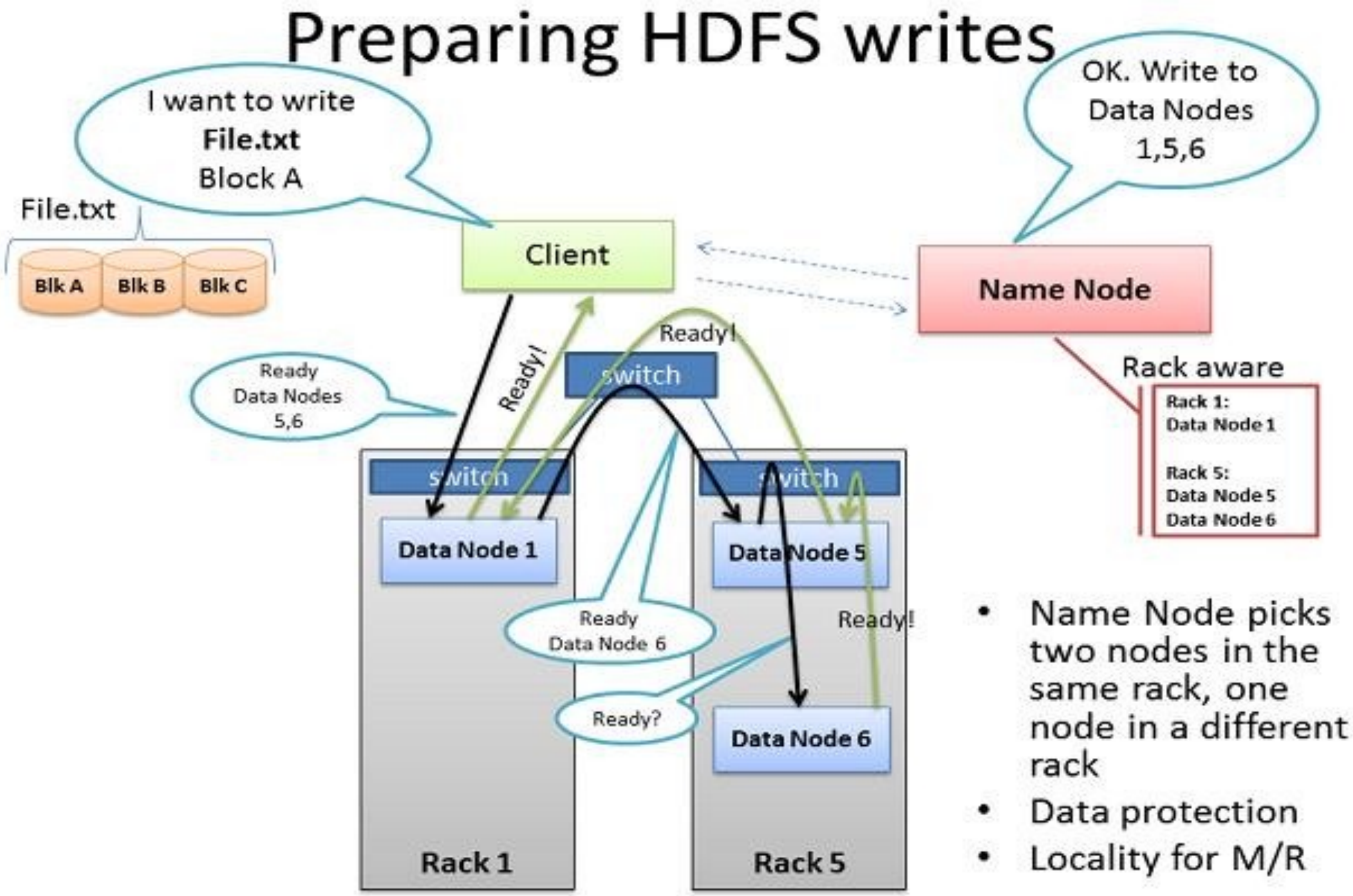
1st Replica → 1 Node (Usually client Node) say DN1 of Rack 1.
2nd Replica →Another Node Say, DN5 of another Rack 2.
3rd Replica →Same Rack R2 another Node say DN6.

By Rule of thumb

1/3rd of Replicas in 1 Rack.
2/3rd of Remaining Replicas in Rack 2.
Remaining replicas are distributed across other racks equally.

Ex: Replication factor =9
1/3rd = 3 replicas in 1 Rack.
2/3rd =2/3*6 =4 in another Rack.
Remaining 2 replicas in other racks.

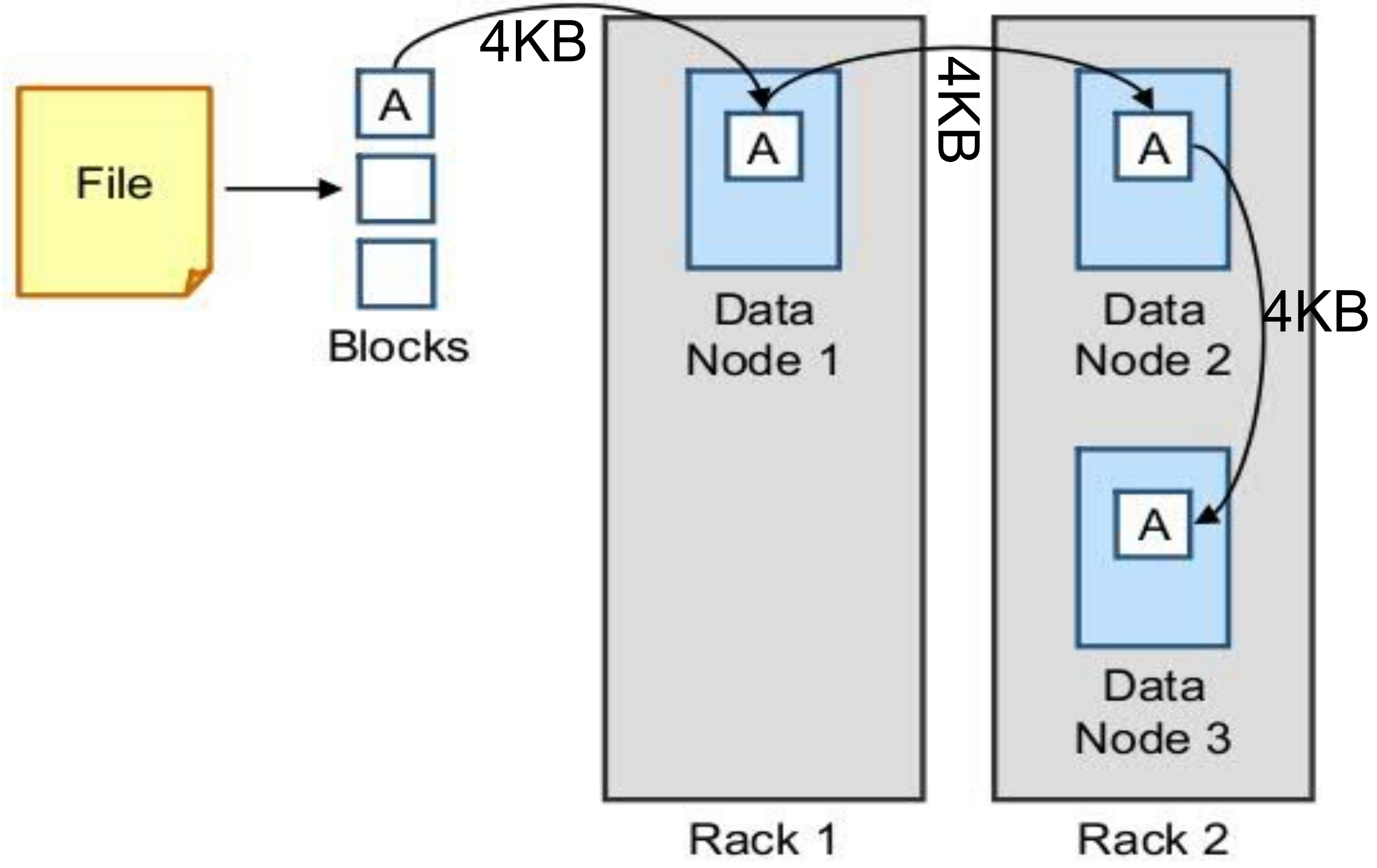
Understanding Hadoop Cluster



Preparing HDFS writes

- Name Node picks two nodes in the same rack, one node in a different rack
- Data protection
- Locality for M/R

Understanding Hadoop Cluster

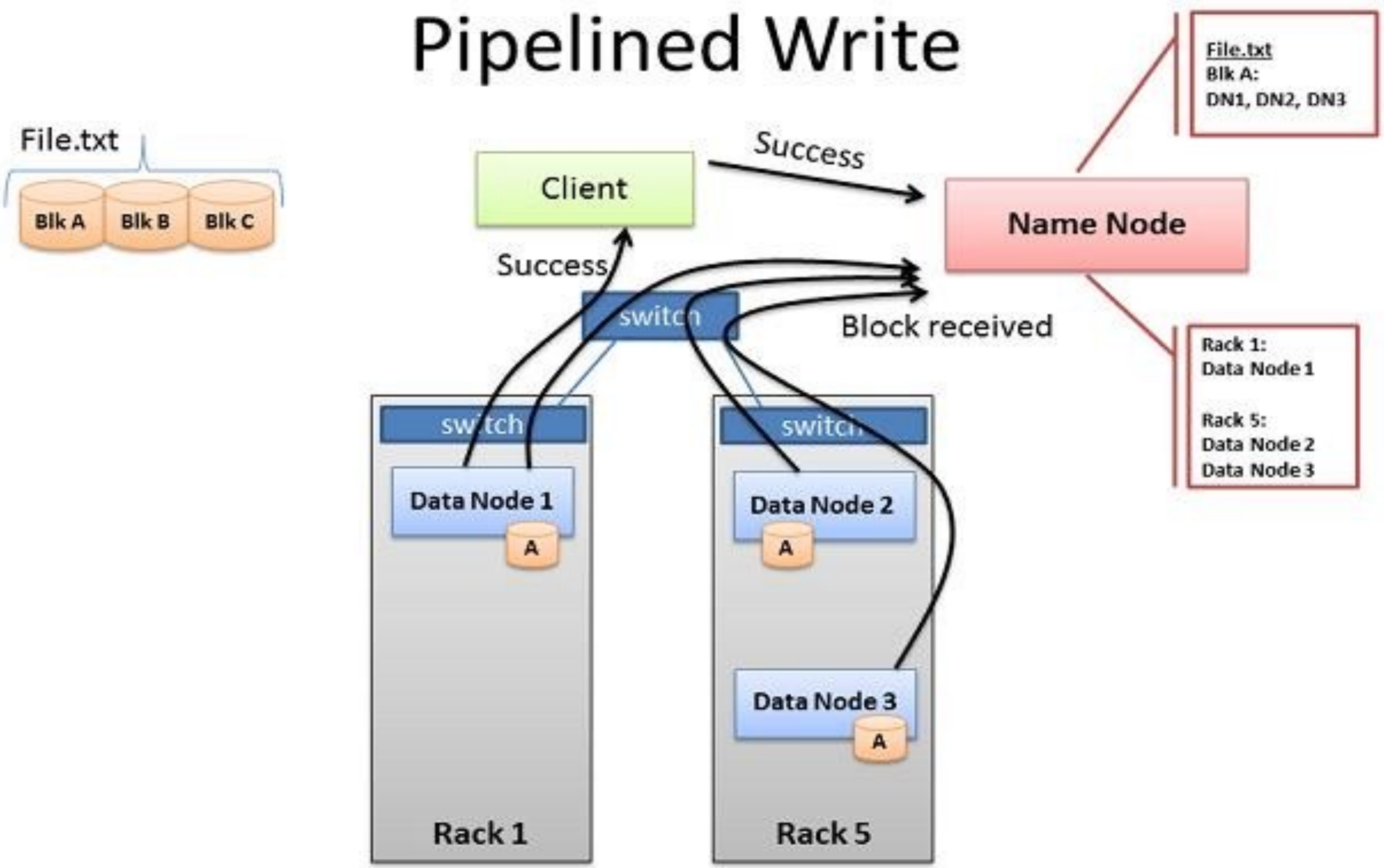


Understanding Hadoop Cluster

Name node Meta Data

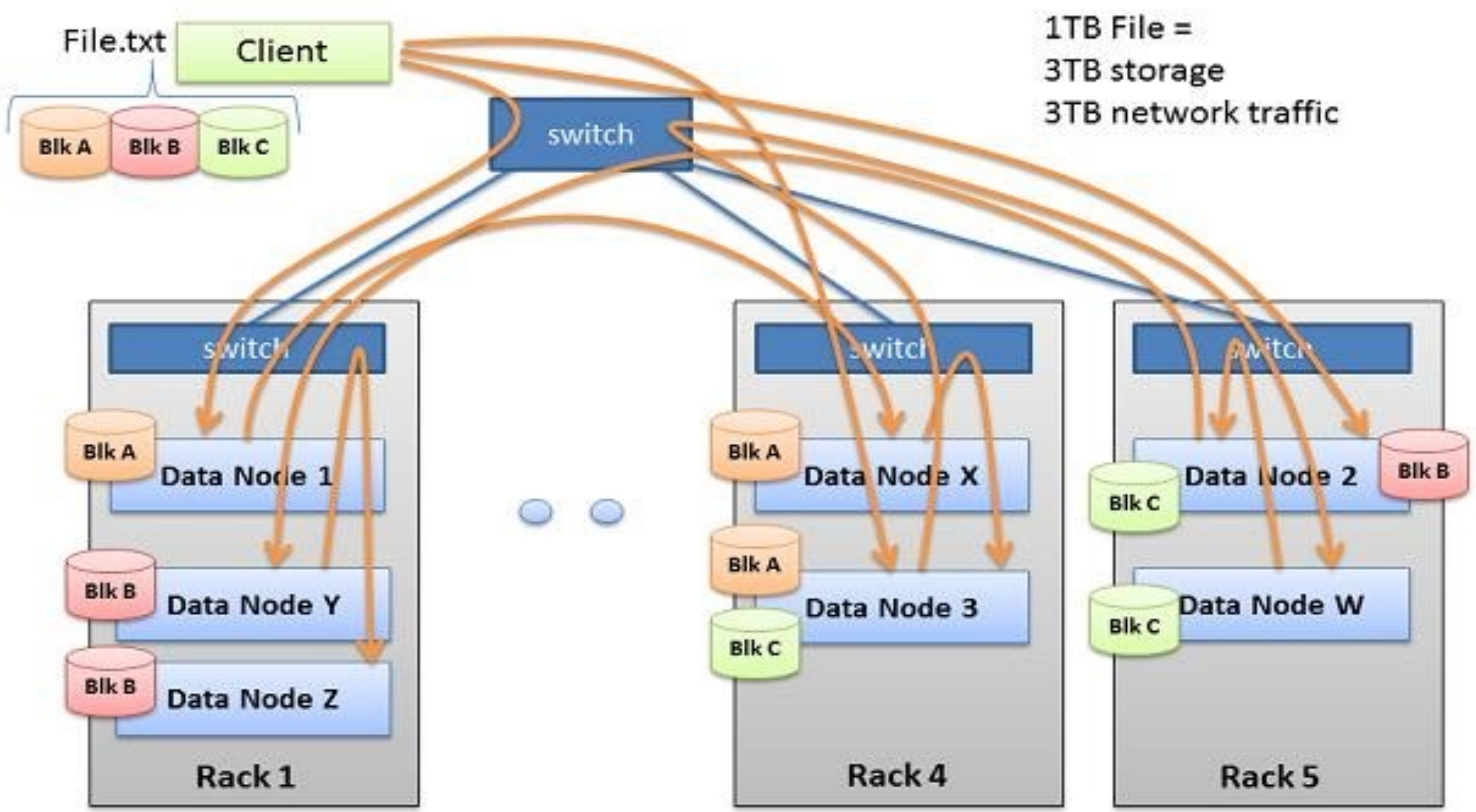
1. File to Block Mapping
   - Ex: File1.txt -→ Block A, Block B, Block C
2. Block To Node Mapping
   - Ex: Block A-→ Node1, Node 5,Node 6
   - Block B-→Node 7, Node 1, Node 2
   - Block C→ Node3, Node 8, Node 9

Understanding Hadoop Cluster



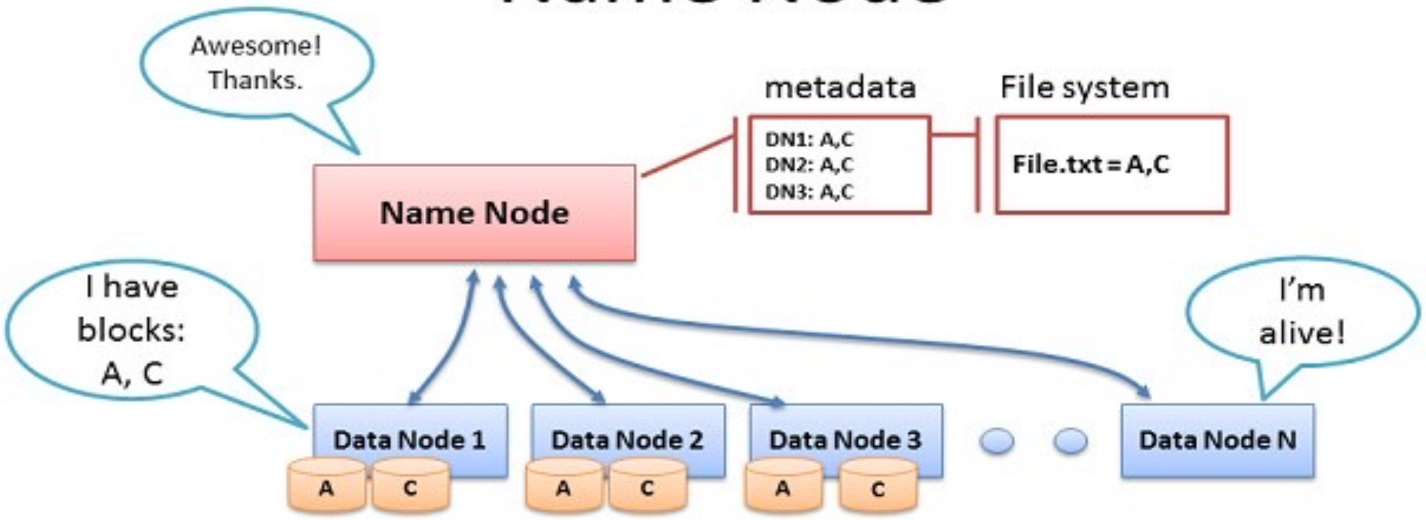
Understanding Hadoop Cluster

Multi-block Replication Pipeline



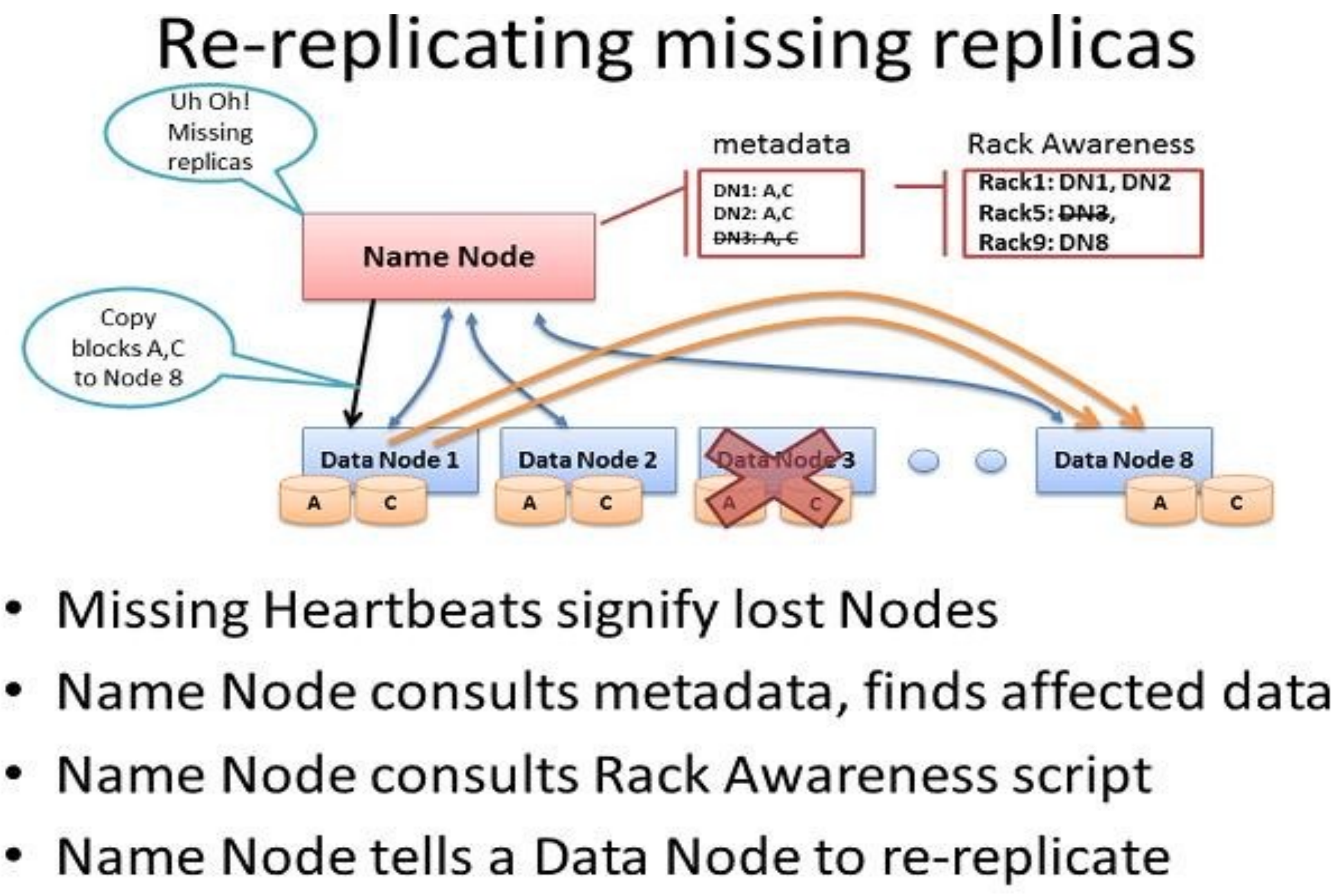
Understanding Hadoop Cluster

Name Node



- Data Node sends Heartbeats
- Every 10th heartbeat is a Block report
- Name Node builds metadata from Block reports
- TCP – every 3 seconds
- If Name Node is down, HDFS is down

Understanding Hadoop Cluster

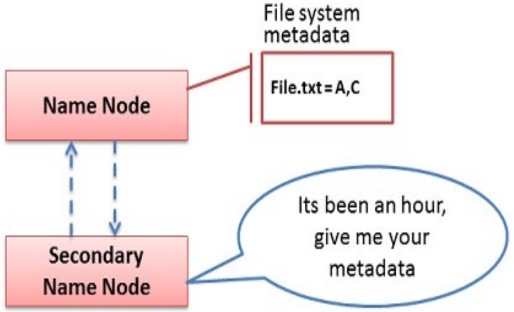


Re-replicating missing replicas

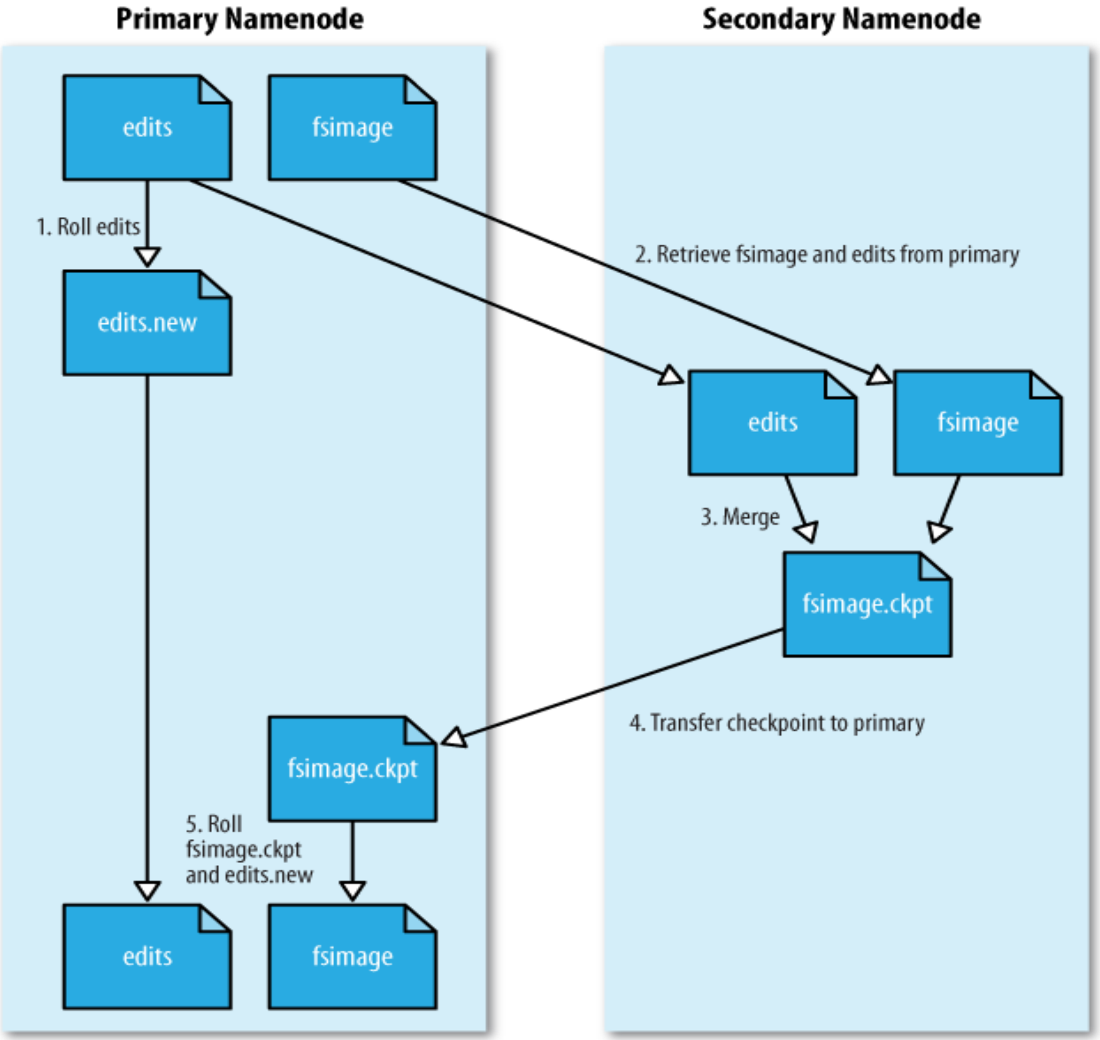
- Missing Heartbeats signify lost Nodes
- Name Node consults metadata, finds affected data
- Name Node consults Rack Awareness script
- Name Node tells a Data Node to re-replicate

Understanding Hadoop Cluster

Secondary Name Node

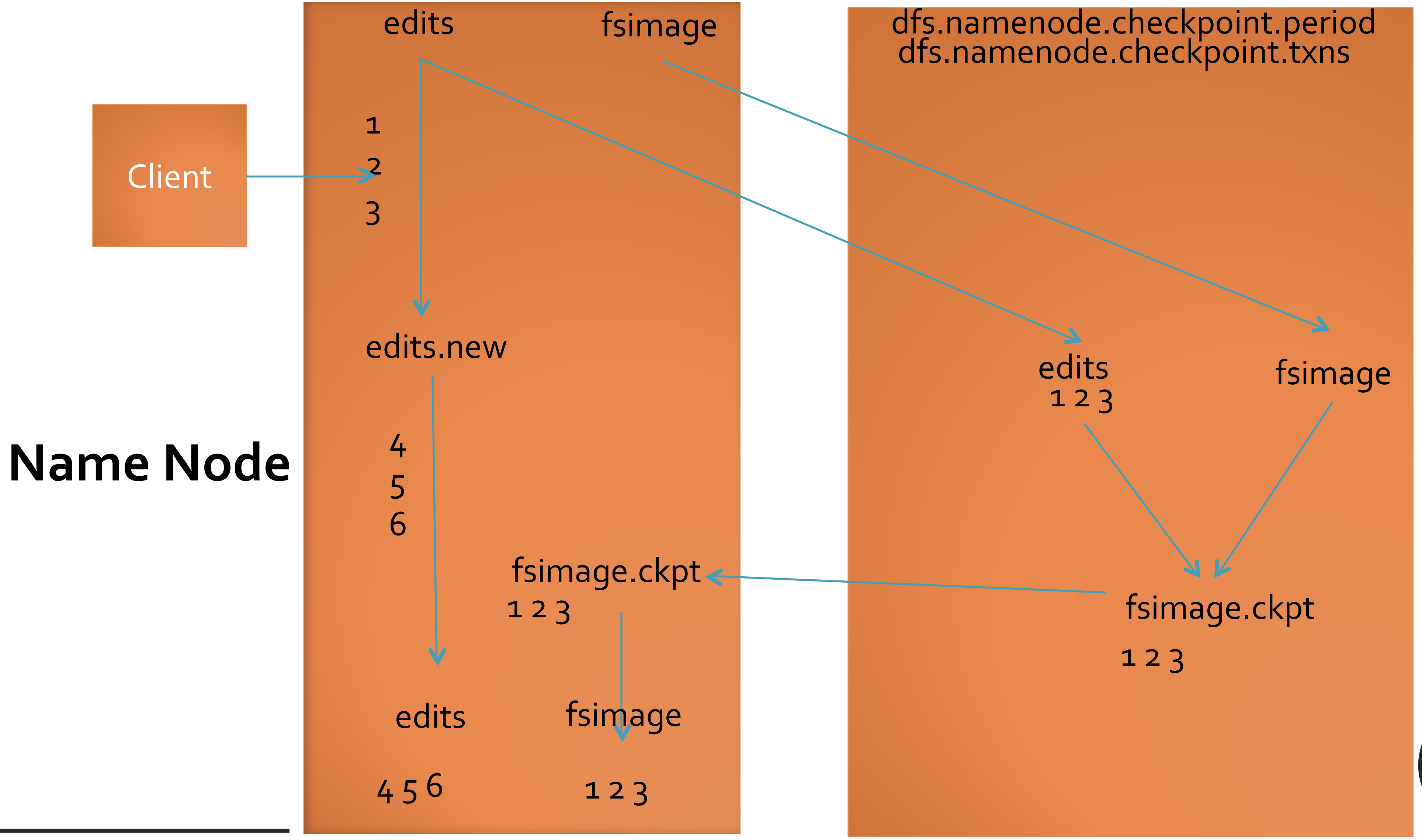


- Not a hot standby for the Name Node
- Connects to Name Node every hour*
- Housekeeping, backup of Name Node metadata
- Saved metadata can rebuild a failed Name Node



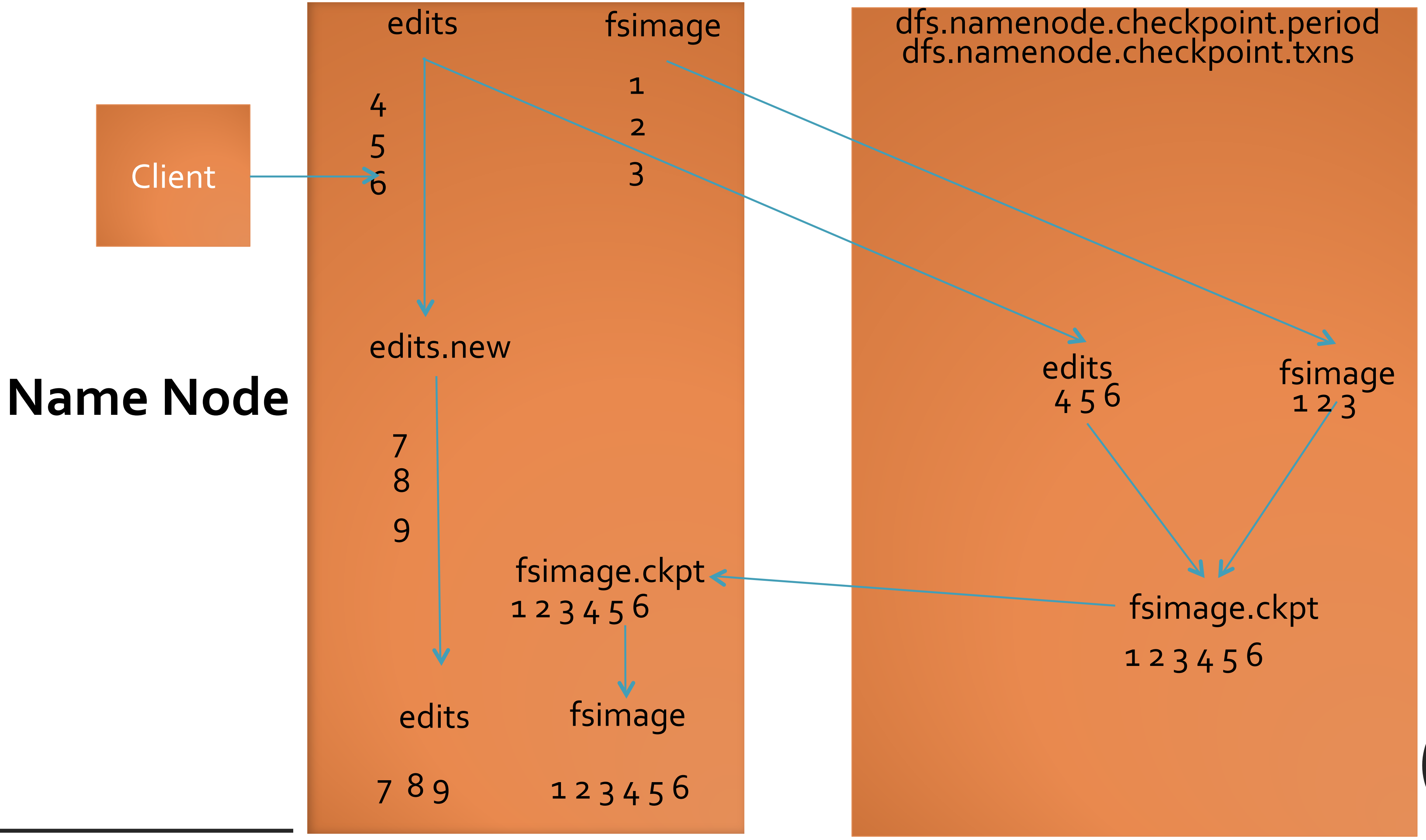
Understanding Hadoop Cluster

Secondary Name Node

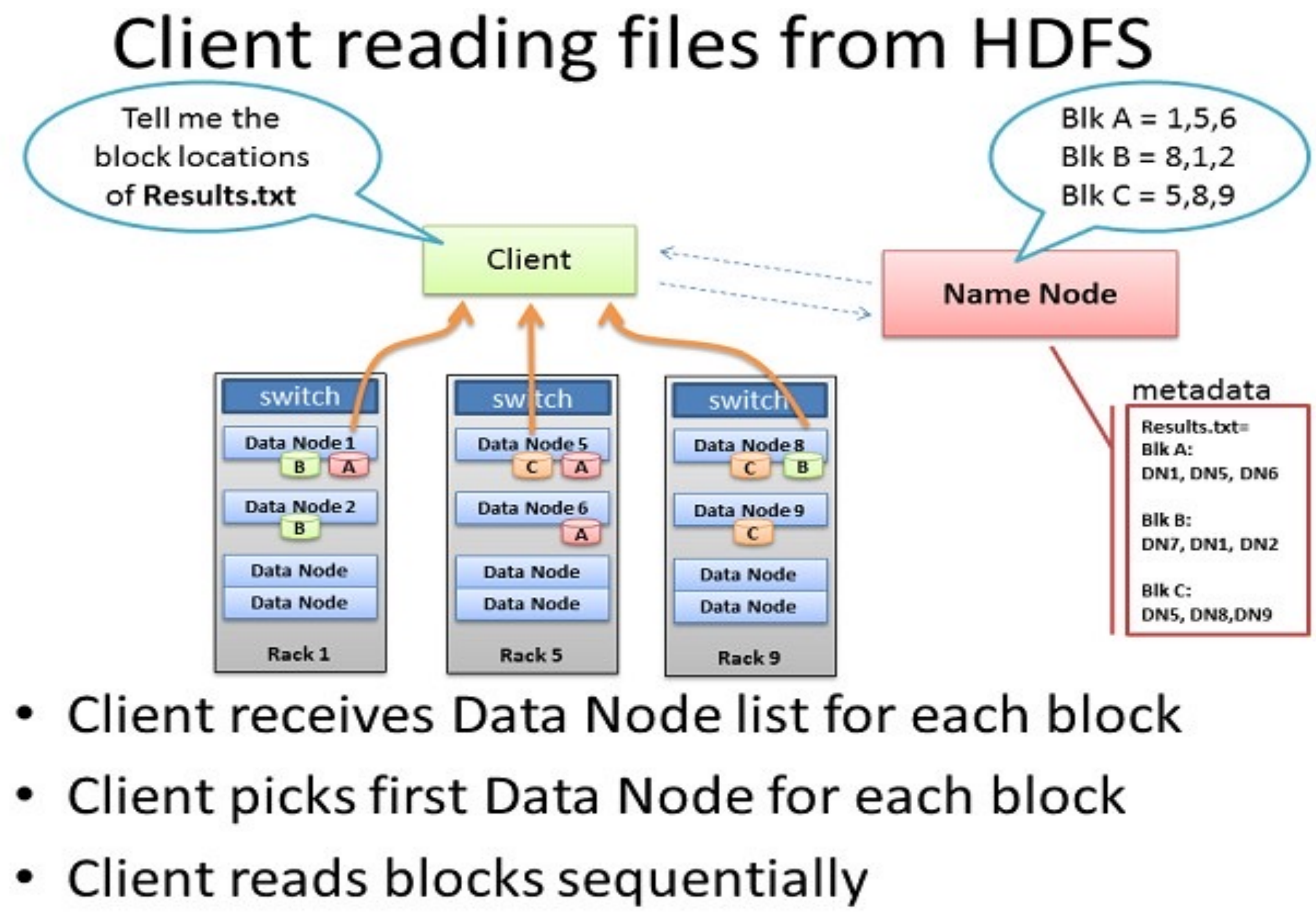


Understanding Hadoop Cluster

Secondary Name Node



Understanding Hadoop Cluster



Client reading files from HDFS

Tell me the block locations of **Results.txt**

Blk A = 1,5,6
Blk B = 8,1,2
Blk C = 5,8,9

Client

Name Node

metadata

Results.txt=
Blk A:
DN1, DN5, DN6

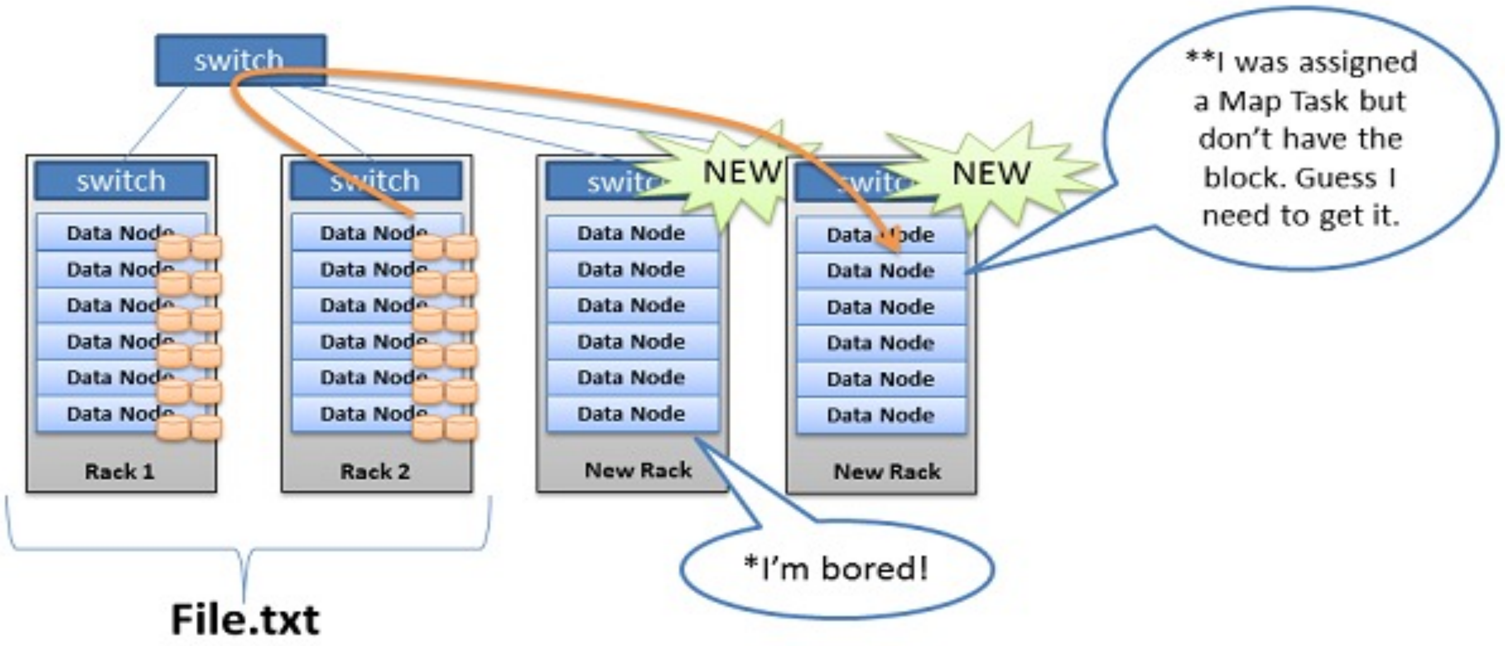
Blk B:
DN7, DN1, DN2

Blk C:
DN5, DN8,DN9

- Client receives Data Node list for each block
- Client picks first Data Node for each block
- Client reads blocks sequentially

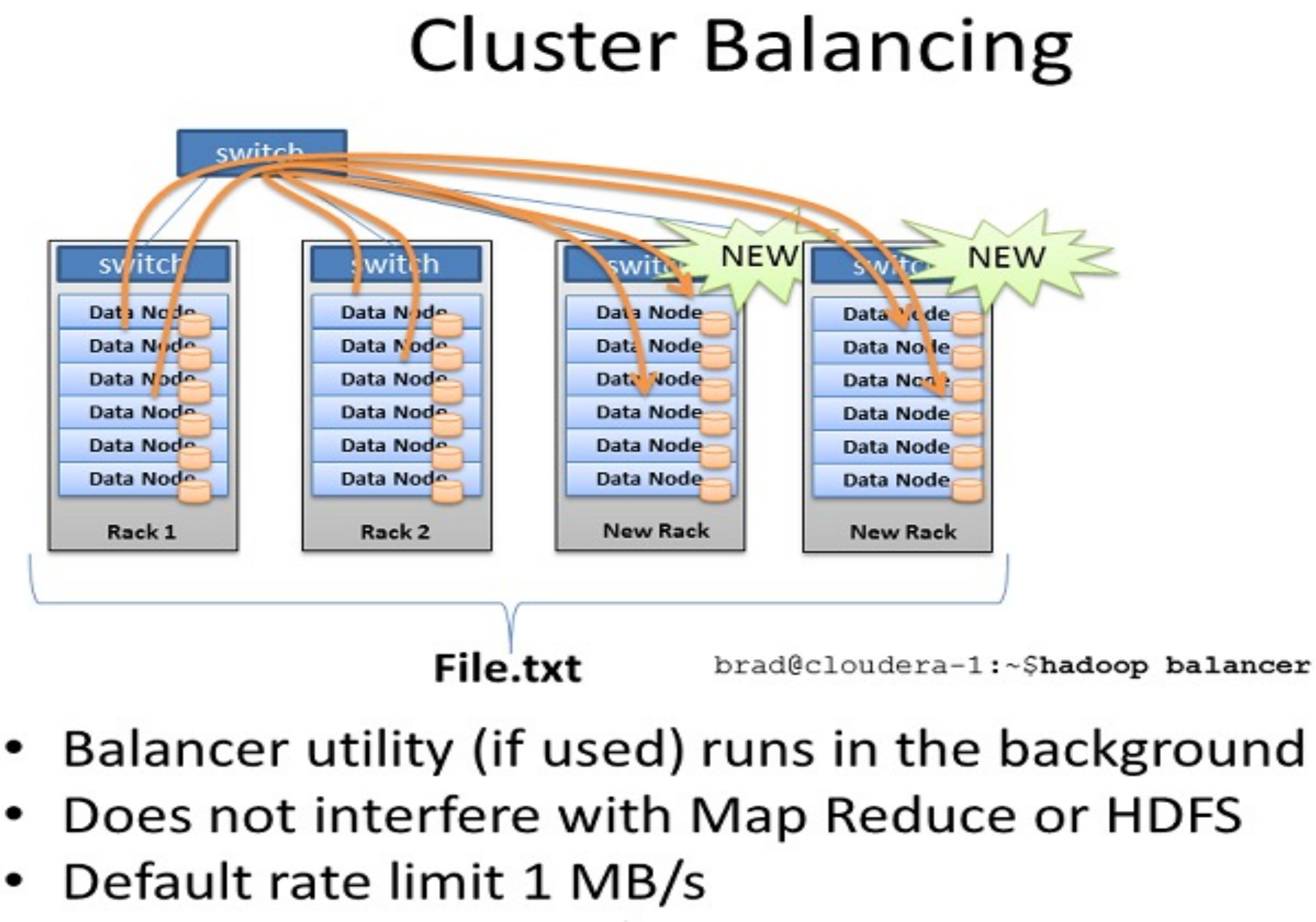
Understanding Hadoop Cluster

Unbalanced Cluster



- Hadoop prefers local processing if possible
- New servers underutilized for Map Reduce, HDFS*
- More network bandwidth, slower job times**

Understanding Hadoop Cluster



Cluster Balancing

```
brad@cloudera-1:~$hadoop balancer
```

- Balancer utility (if used) runs in the background
- Does not interfere with Map Reduce or HDFS
- Default rate limit 1 MB/s

General Planning Considerations

Handle
More
Data

At
Lower
Cost

In
Less
Time

With
Less
Power

Best Practices

Start with small cluster (4 to 10 nodes) and grow as and when required. Cluster can be grown whenever there is a

- Increase in computation power needed
- Increase in data to be stored
- Increase in amount of memory to process tasks
- Increase in data transfer between data nodes

Cluster Growth based on Storage Capacity:

Data Growth TB/Week	Replication Factor	Intermediate & Log Files	Overall Space needed per week
2	3	30%	7.8

Two Machines with 1X4TB are needed.

Where to Optimize

Hardware

Software

Choosing Right Hardware

Master Node:

- Single Point of Failure
- 32 GB RAM
- Dual Xeon E5600 or better (Quad core)
- Dual Power supply for Redundancy
- 4 x 500 GB 7200 rpm SATA drives
- Dual 1 Gb Ethernet cards

Data Nodes:

- 4 1TB hard disks in a JBOD (Just a Bunch Of Disks) configuration. No RAID.
- 2 quad core CPUs, running at least 2-2.5GHz
- 16-24GBs of RAM (24-32GBs if you're considering HBase)
- Gigabit Ethernet

- Master Node:
- No Commodity Hardware
- RAIDed hard drives
- Backup Metadata to an NFS Mount
- RAM Thumb rule: 1 GB per 1 million blocks of data. 32GB for 100 nodes.
- If Metadata is lost, whole cluster is lost. Use expensive Name Node.

- # of Tasks per Core:
- 2 Cores - Datanode and Tasktracker
- Thumb Rule – 1 Core can run 1.5 Mappers or Reducers
- Amount of RAM:
- Thumb Rule: 1G per Map or Reduce task

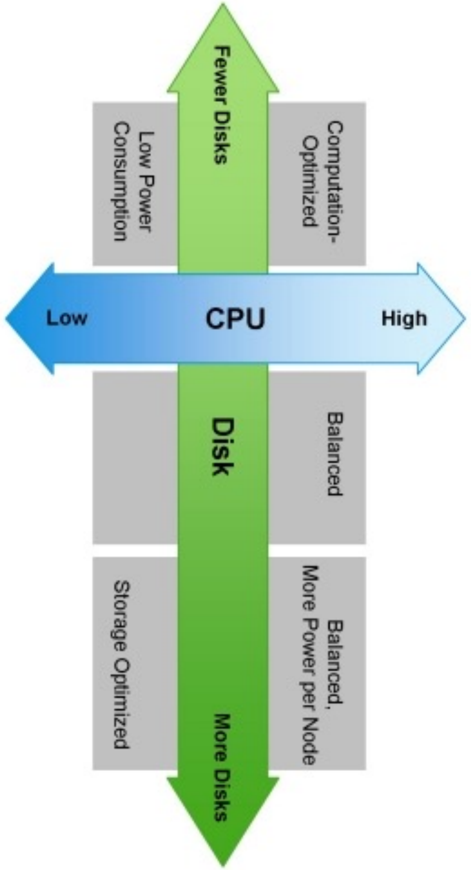
Choosing Right Hardware Based on Different workloads

Light Processing Configuration (1U/machine): Two quad core CPUs, 8GB memory, and 4 disk drives (1TB or 2TB). Note that CPU-intensive work such as natural language processing involves loading large models into RAM before processing data and should be configured with 2GB RAM/core instead of 1GB RAM/core.

Balanced Compute Configuration (1U/machine): Two quad core CPUs, 16 to 24GB memory, and 4 disk drives (1TB or 2TB) directly attached using the motherboard controller. These are often available as twins with two motherboards and 8 drives in a single 2U cabinet.

Storage Heavy Configuration (2U/machine): Two quad core CPUs, 16 to 24GB memory, and 12 disk drives (1TB or 2TB). The power consumption for this type of machine starts around ~200W in idle state and can go as high as ~350W when active.

Compute Intensive Configuration (2U/machine): Two quad core CPUs, 48-72GB memory, and 8 disk drives (1TB or 2TB). These are often used when a combination of large in-memory models and heavy reference data caching is required.



Next Session

Hadoop HDFS Commands