Understanding & Planning Hadoop Cluster

Cloudwick Technologies

Agenda

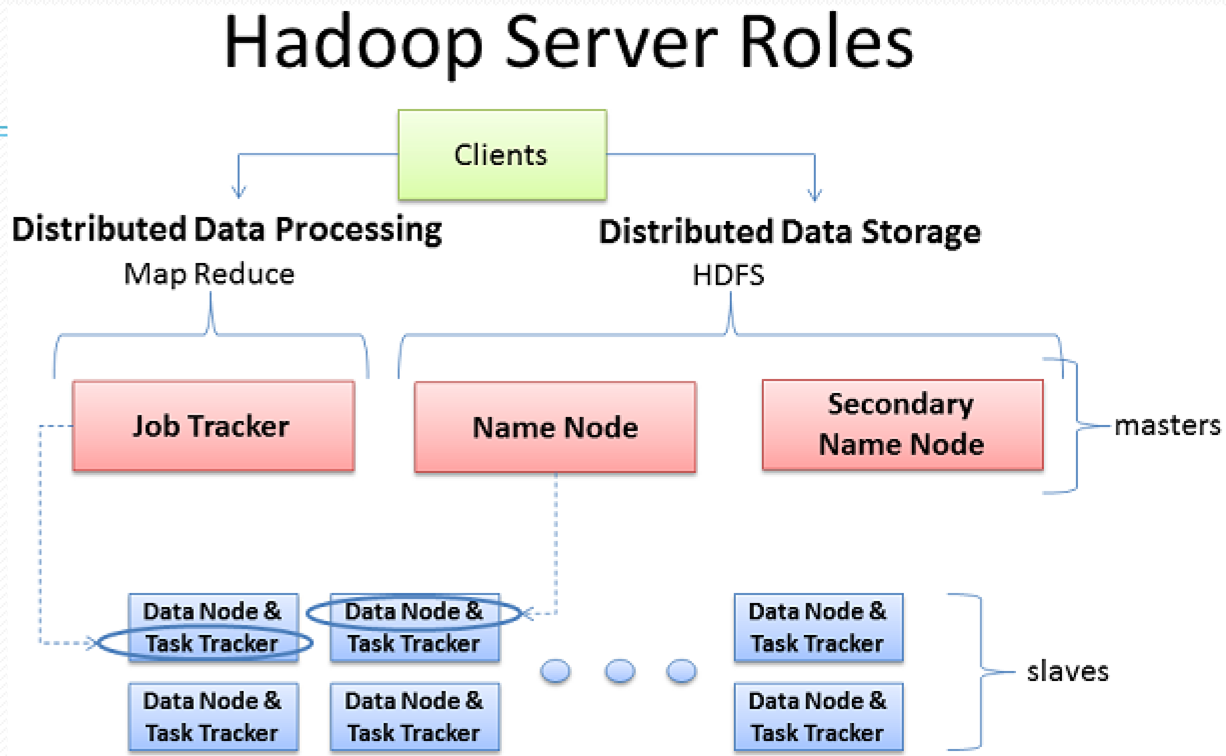
- Typical Workflow
- Writing Files to HDFS
- Reading files from HDFS
- Rack Awareness
- Planning for Cluster
- Choosing Right Hardware
- Choosing Right Hadoop Distribution

Cloudwick Technologies

Understanding Hadoop Cluster

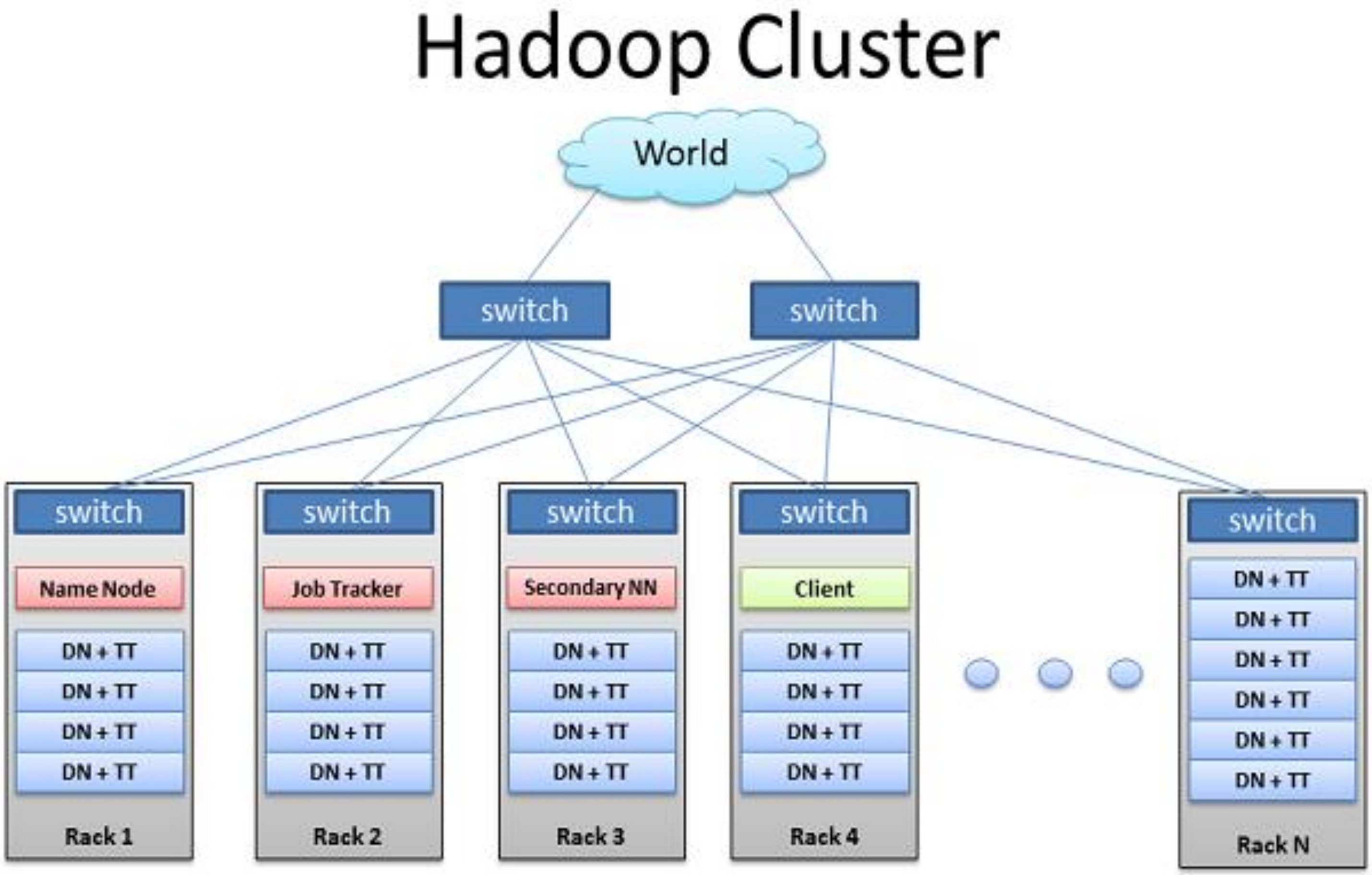
3 Major Categories

- Clients
- Master Nodes
- Slave Nodes



Cloudwick Technologies

Understanding Hadoop Cluster



This is the typical architecture of a 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:

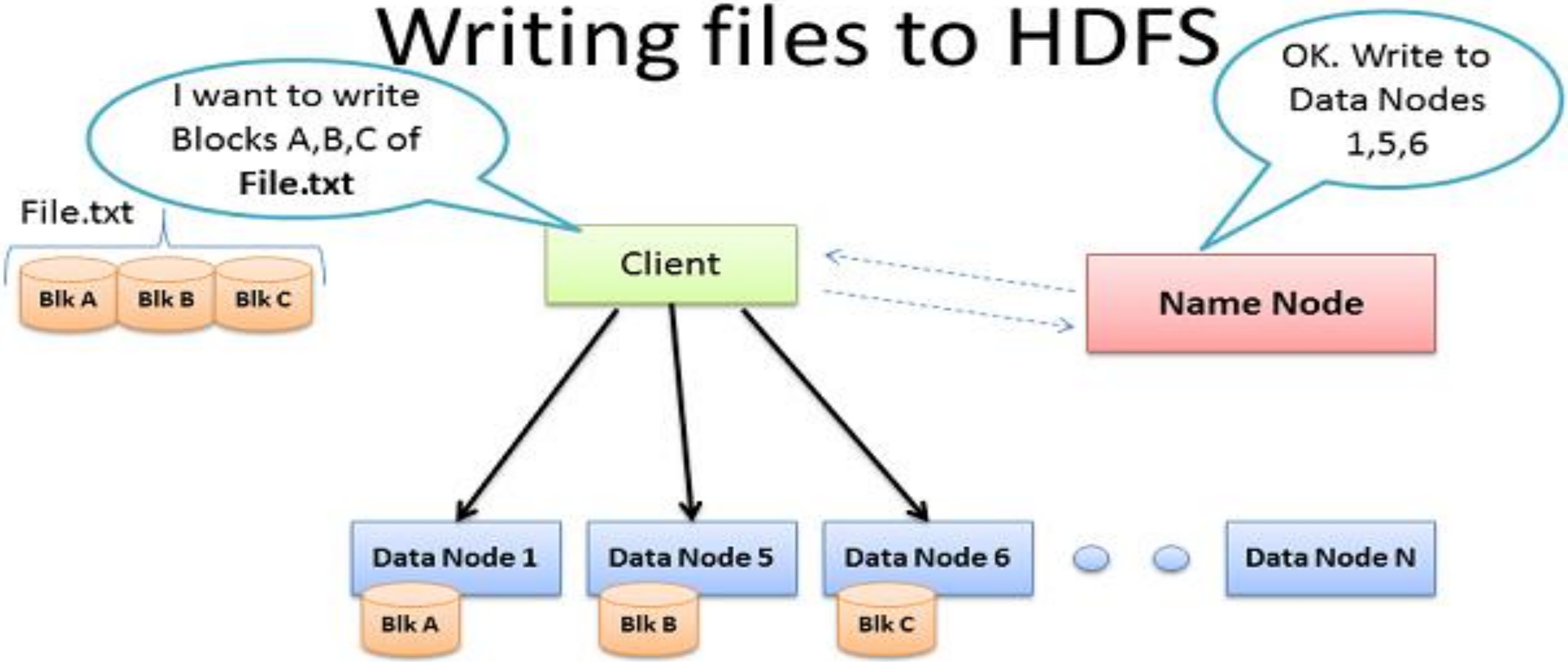
How many times did our customers type the word "**Refund**" into emails sent to customer service?

Huge file containing all emails sent to customer service

File.txt

Understanding Hadoop Cluster

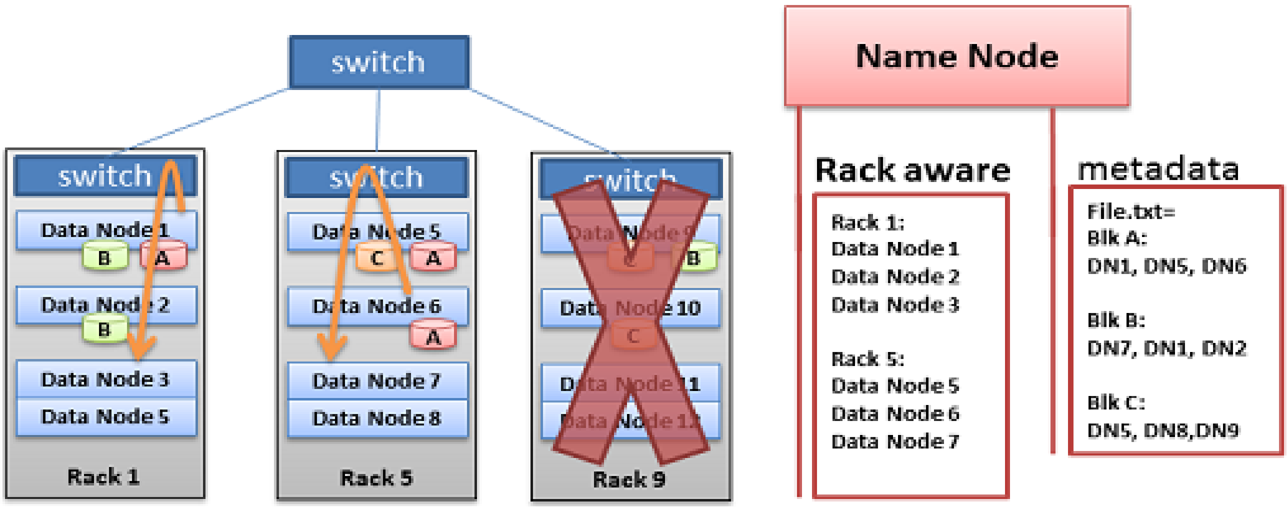
Writing files to HDFS



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

Cloudwick Technologies

Hadoop Rack Awareness – Why?

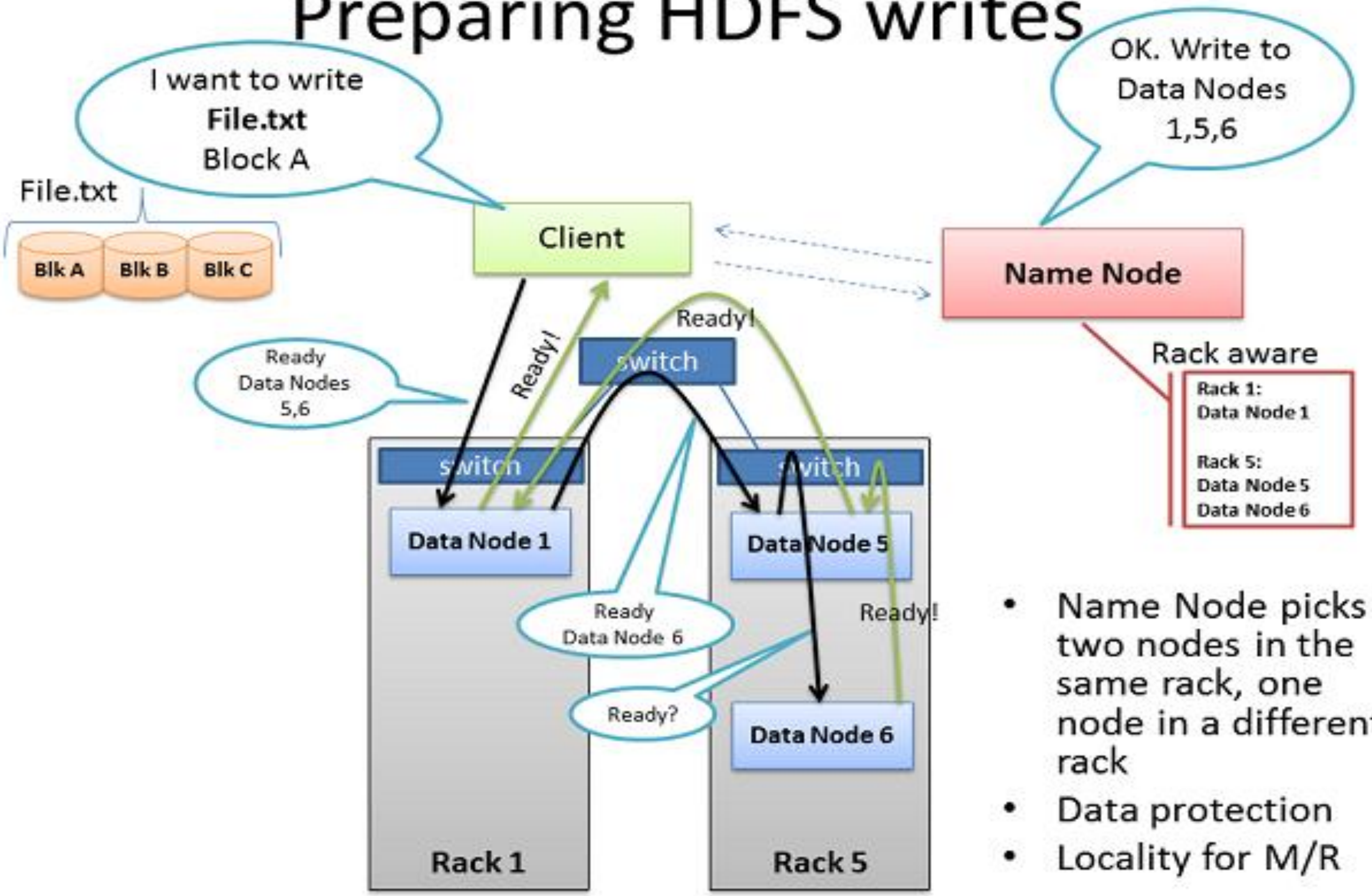


- Never loose all data if entire rack fails
- Keep bulky flows in-rack when possible
- Assumption that in-rack is higher bandwidth, lower latency

Cloudwick Technologies

Understanding Hadoop Cluster

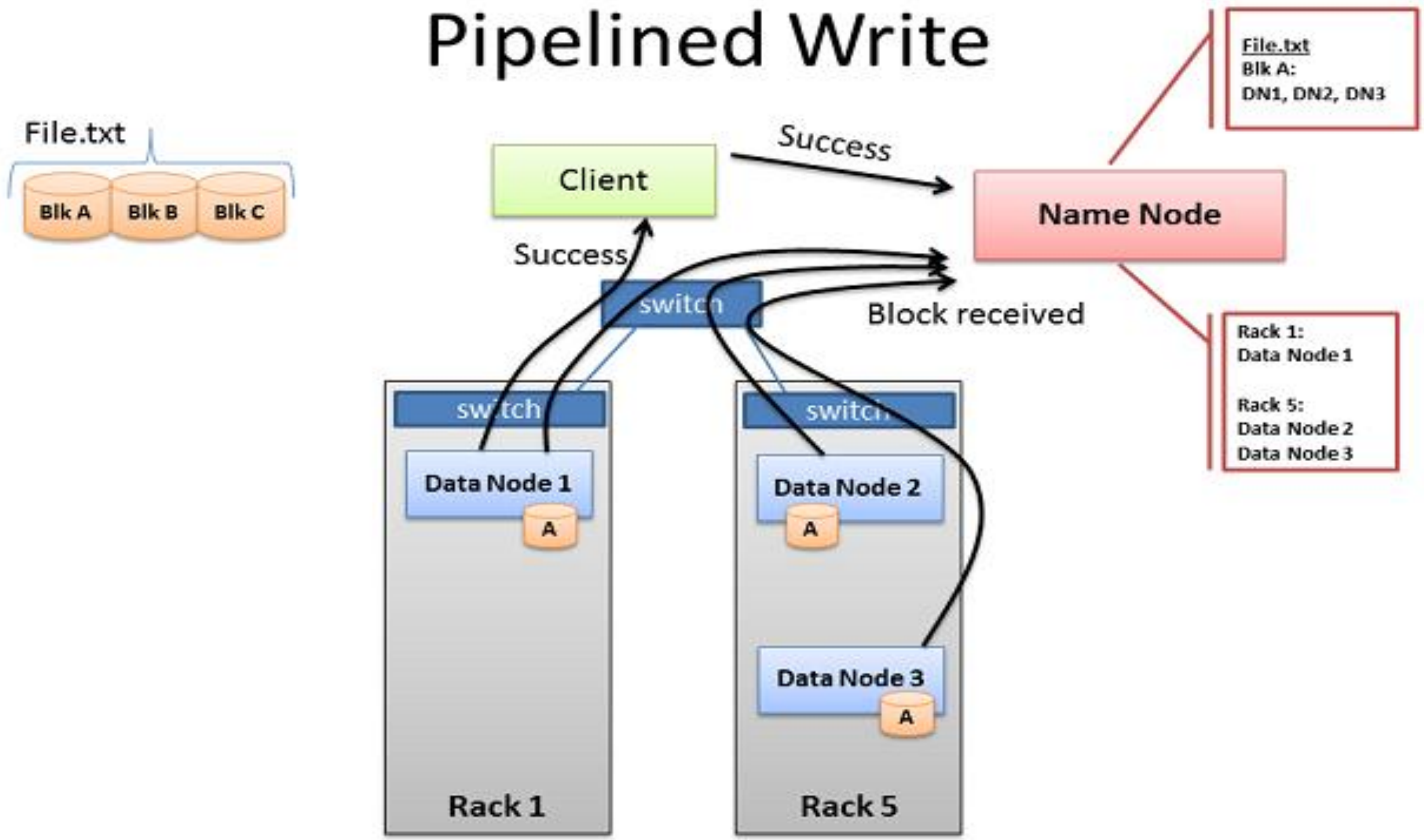
Preparing HDFS writes



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

Cloudwick Technologies

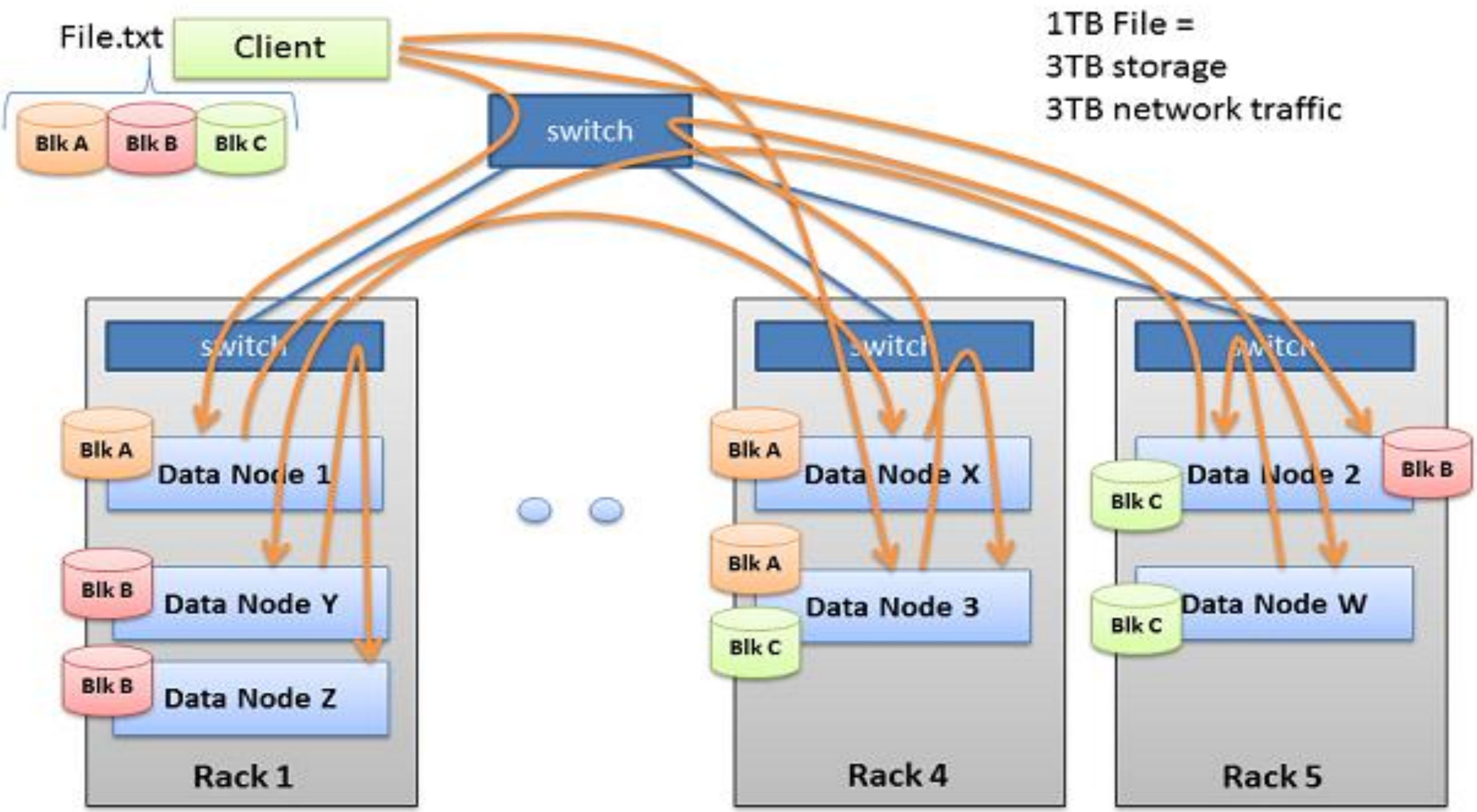
Understanding Hadoop Cluster



Cloudwick Technologies

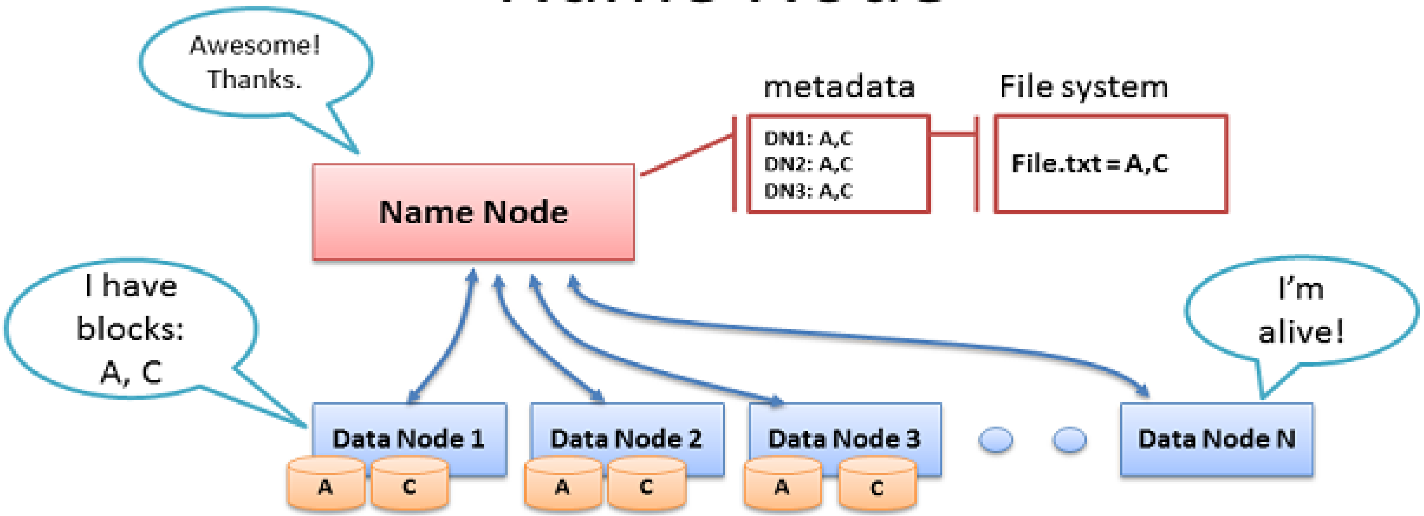
Understanding Hadoop Cluster

Multi-block Replication Pipeline



Understanding Hadoop Cluster

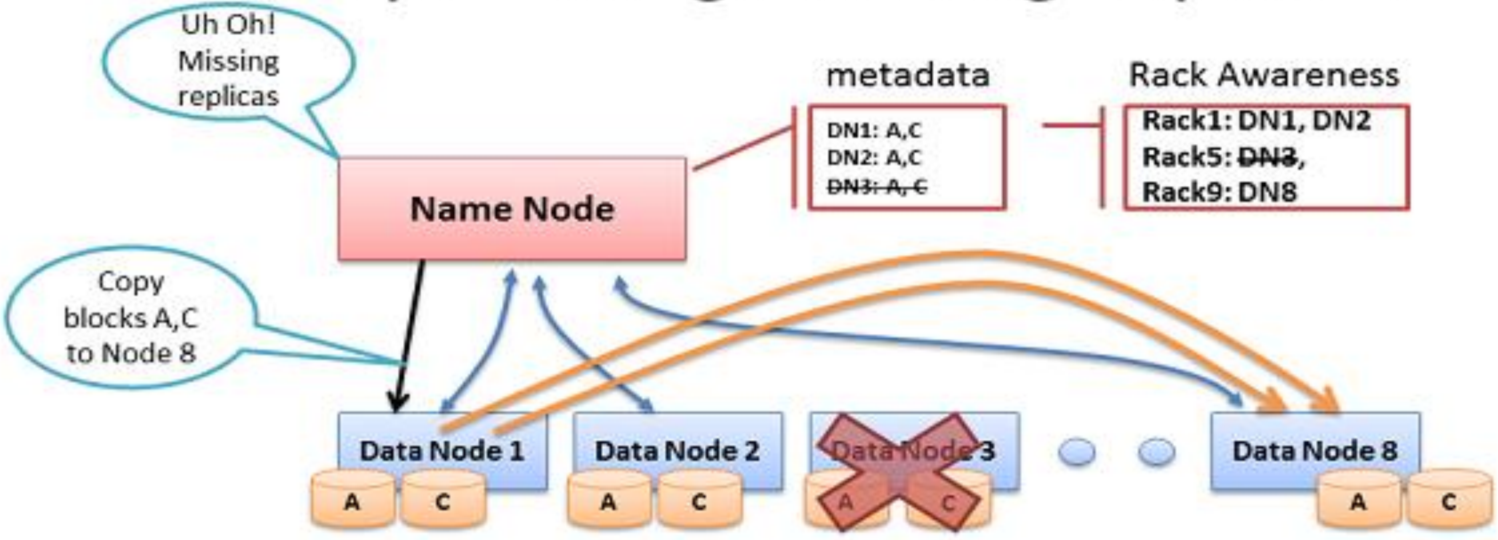
Name Node



- 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

Cloudwick Technologies

Re-replicating missing replicas

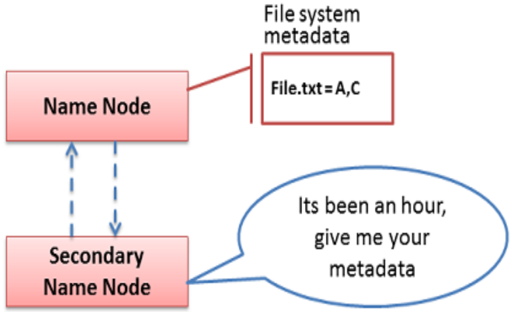


- Missing Heartbeats signify lost Nodes
- Name Node consults metadata, finds affected data
- Name Node consults Rack Awareness script
- Name Node tells a Data Node to re-replicate

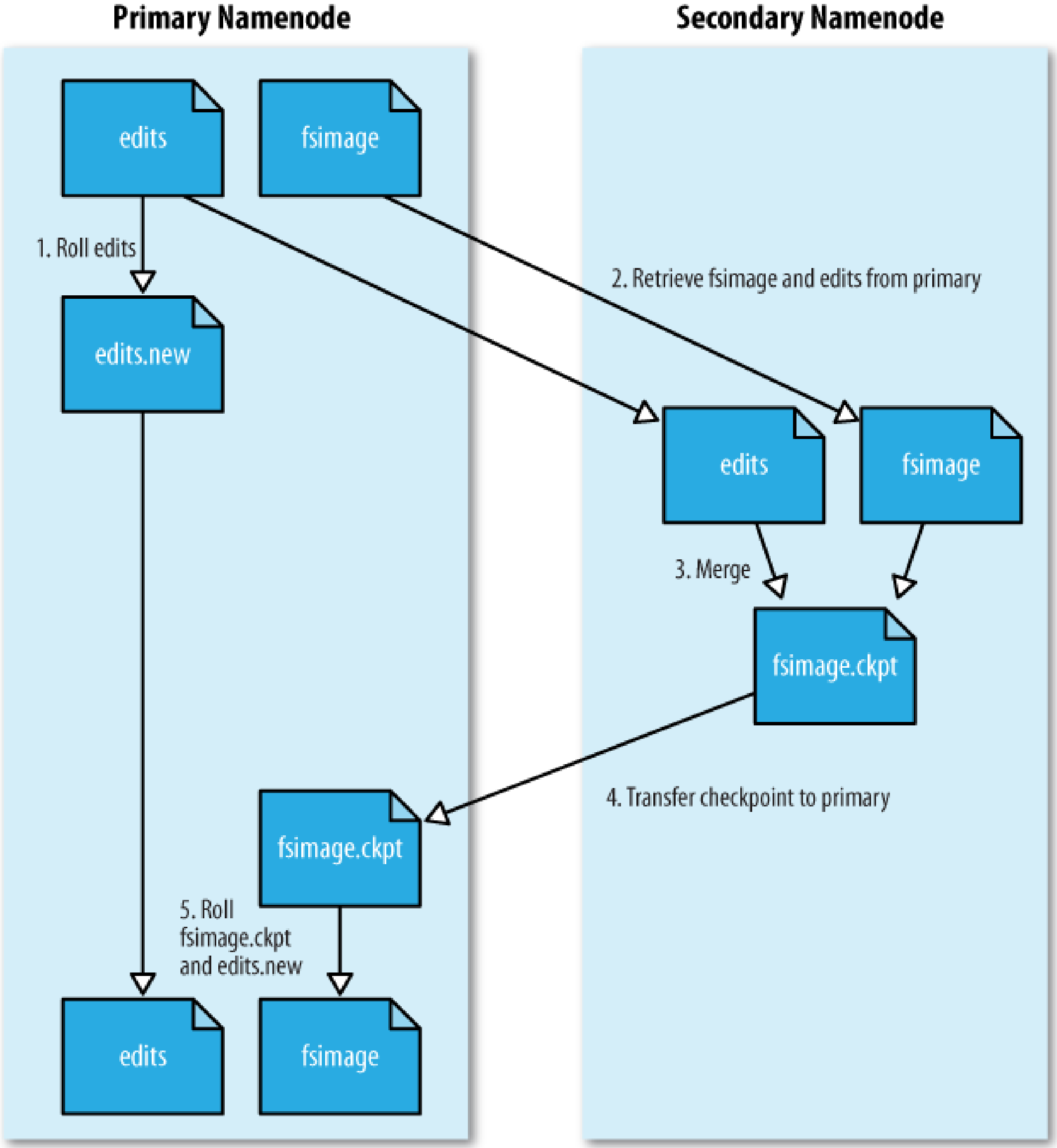
Cloudwick Technologies

Understanding Hadoop Cluster

Secondary Name Node

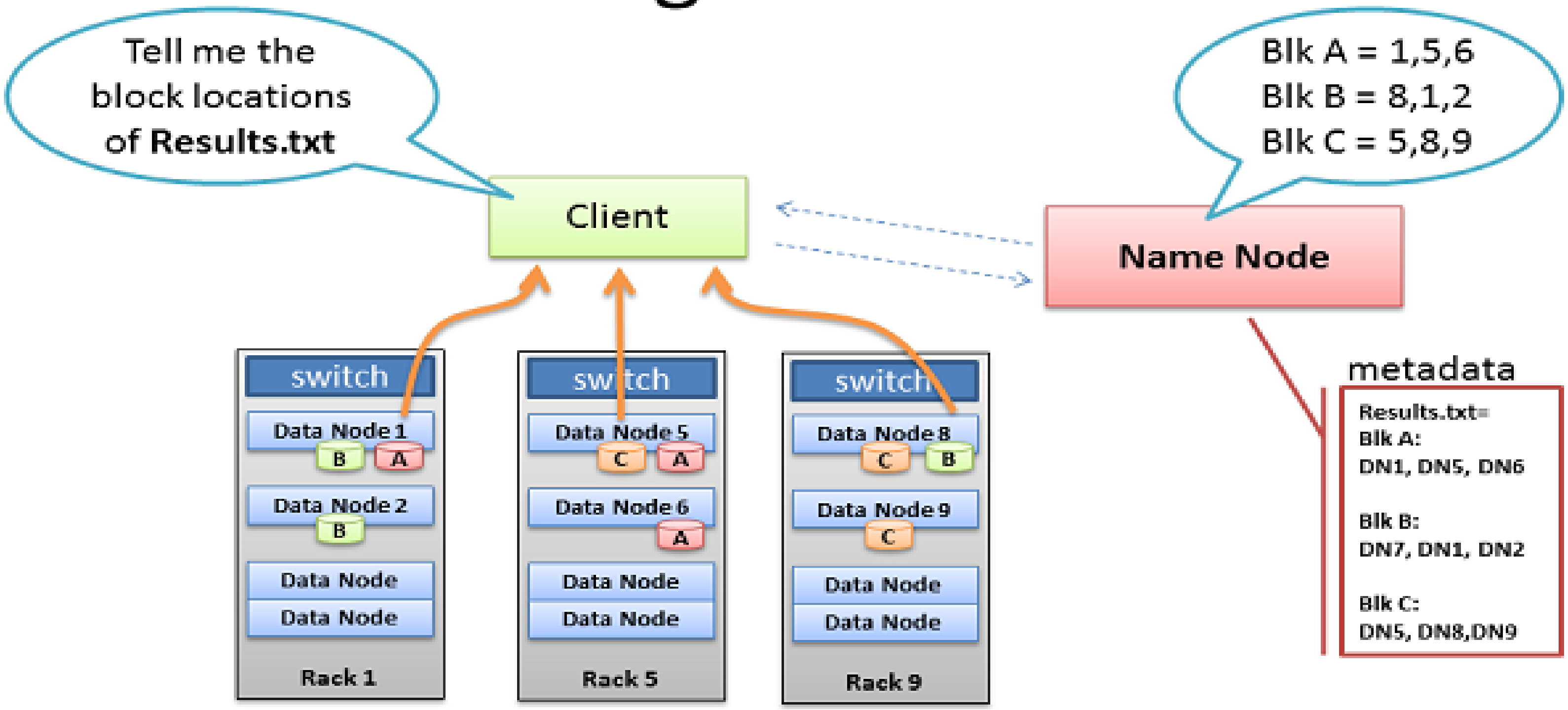


- Not a hot standby for the Name Node
- Connects to Name Node every hour*
- Housekeeping, backup of Name Node metadata
- Saved metadata can rebuild a failed Name Node



Cloudwick Technologies

Client reading files from HDFS

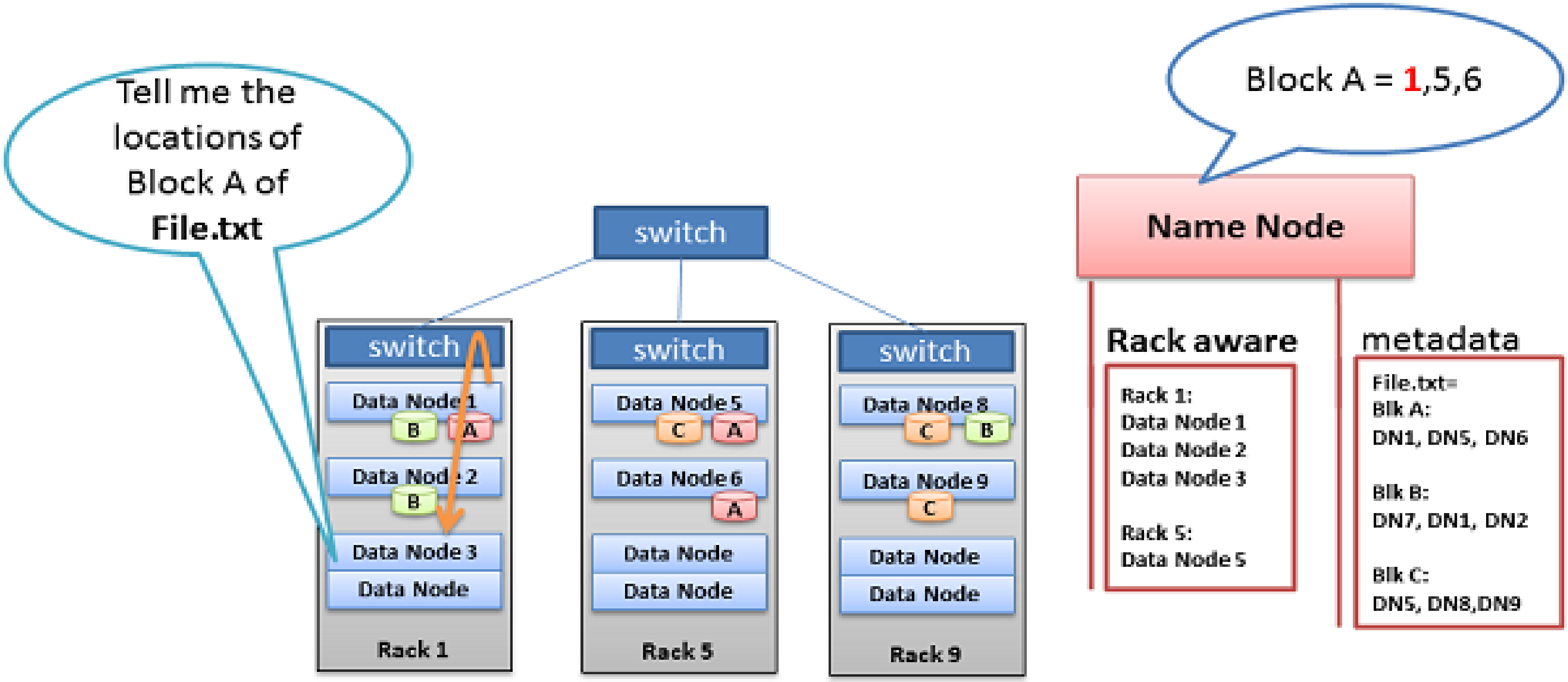


- Client receives Data Node list for each block
- Client picks first Data Node for each block
- Client reads blocks sequentially

Cloudwick Technologies

Understanding Hadoop Cluster

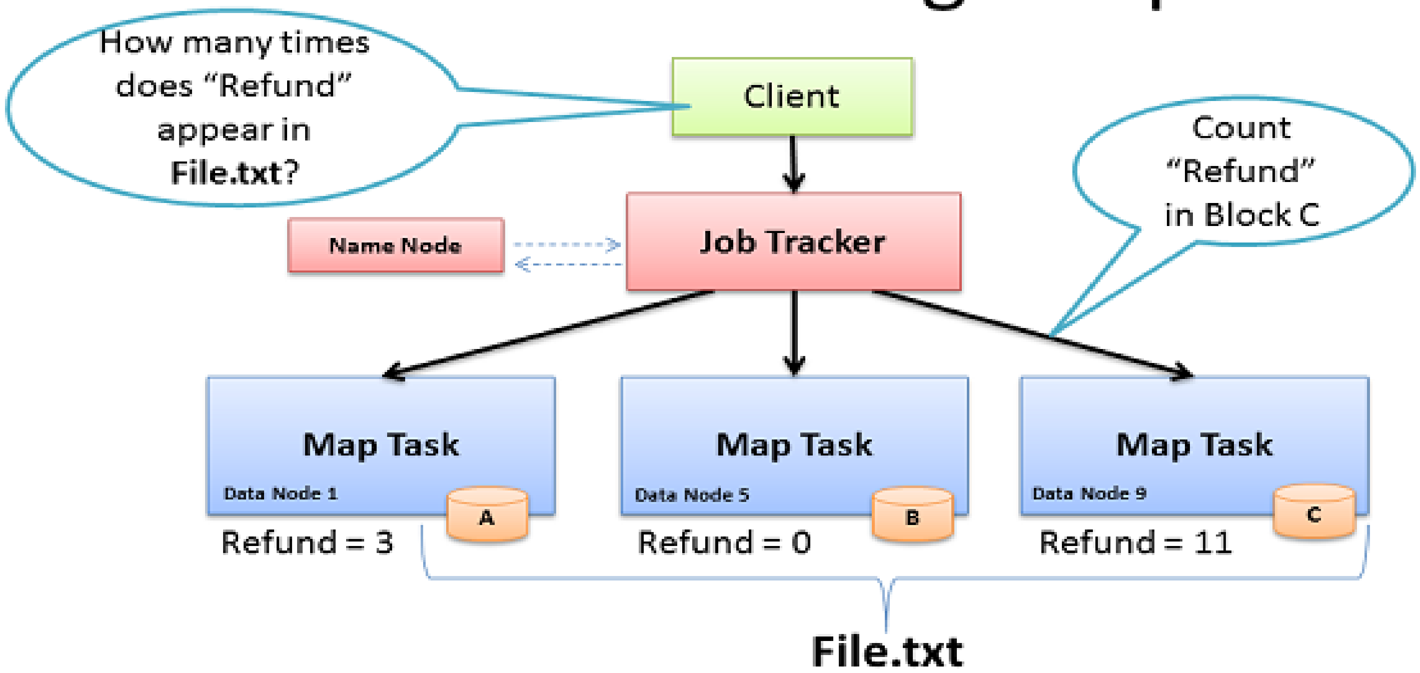
Data Node reading files from HDFS



- Name Node provides rack local Nodes first
- Leverage in-rack bandwidth, single hop

Cloudwick Technologies

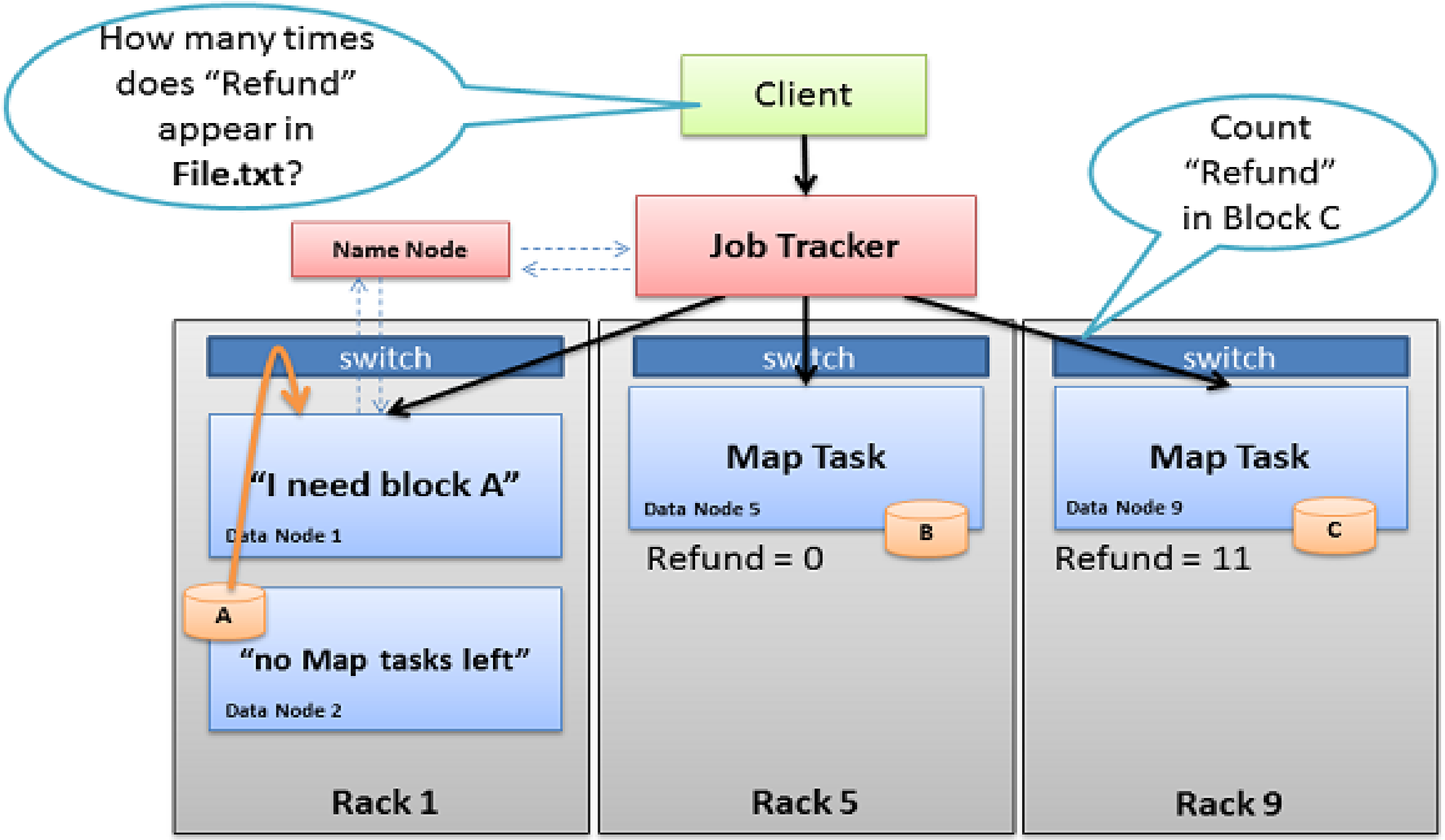
Data Processing: Map



- **Map:** "Run this computation on your local data"
- Job Tracker delivers Java code to Nodes with local data

Understanding Hadoop Cluster

What if data isn't local?

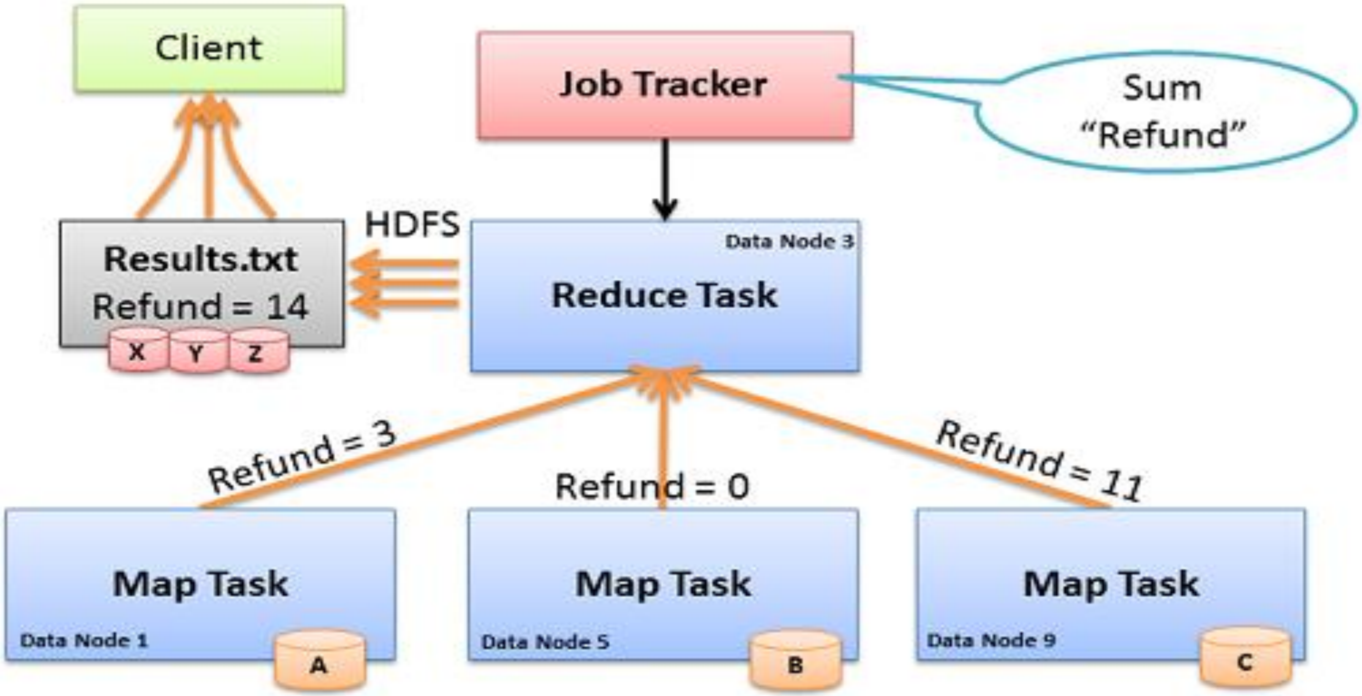


- Job Tracker tries to select Node in same rack as data
- Name Node rack awareness

Cloudwick Technologies

Understanding Hadoop Cluster

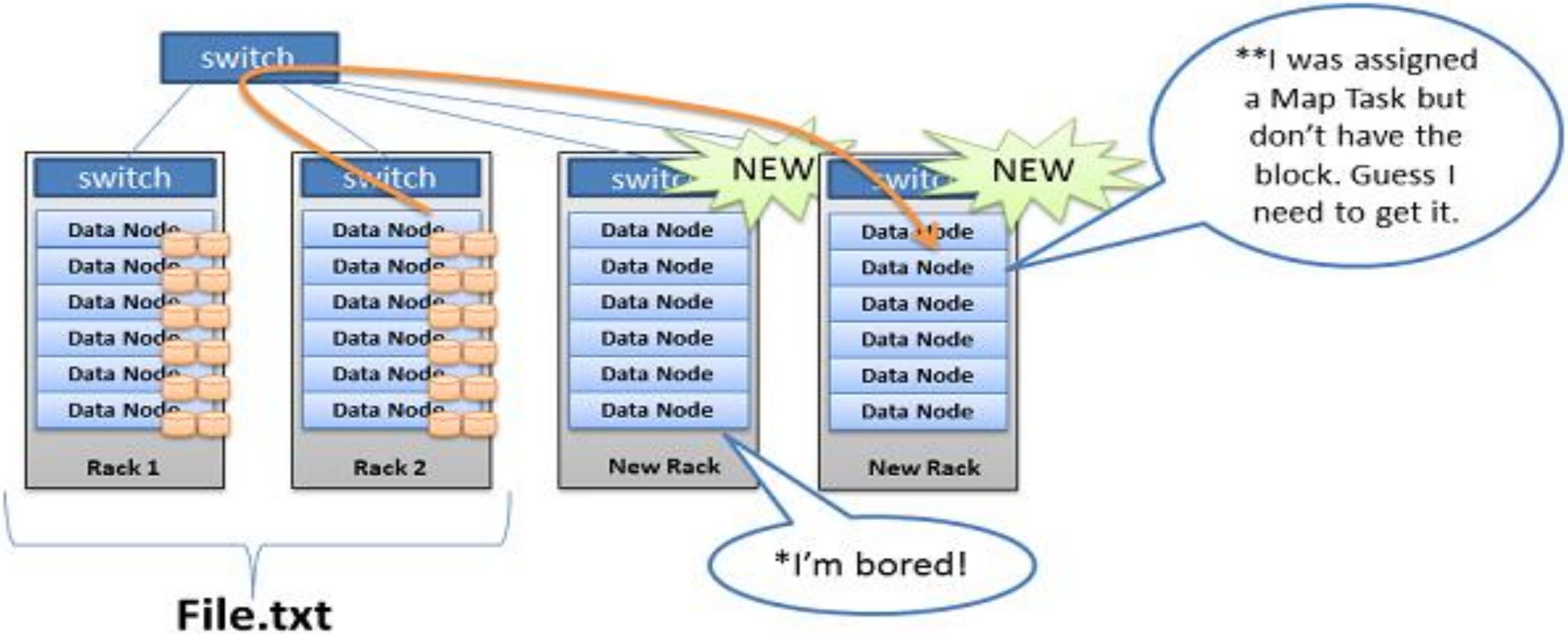
Data Processing: Reduce



- **Reduce:** "Run this computation across Map results"
- Map Tasks send output data to Reducer over the network
- Reduce Task data output written to and read from HDFS

Cloudwick Technologies

Unbalanced Cluster

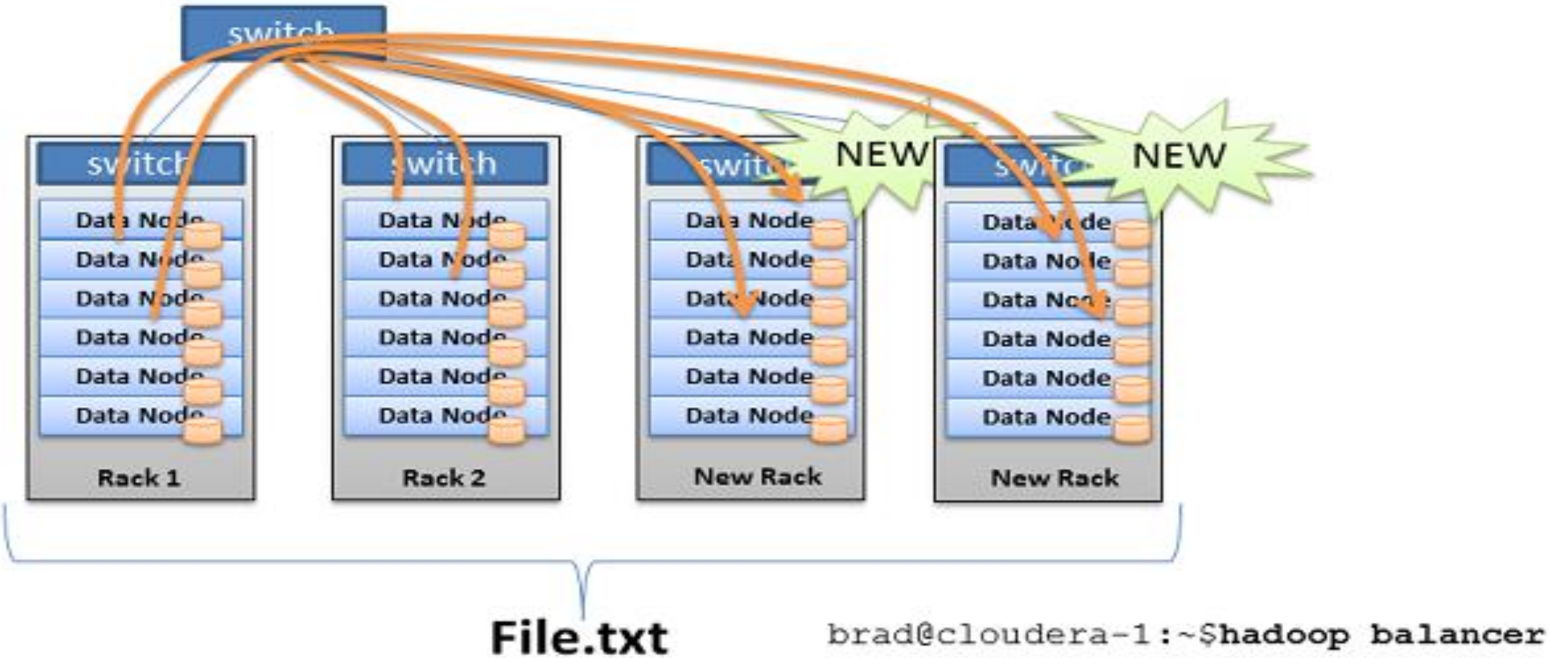


- Hadoop prefers local processing <u>if possible</u>
- New servers underutilized for Map Reduce, HDFS*
- More network bandwidth, slower job times**

Cloudwick Technologies

Understanding Hadoop Cluster

Cluster Balancing



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

Cloudwick Technologies

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.

Cloudwick Technologies

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

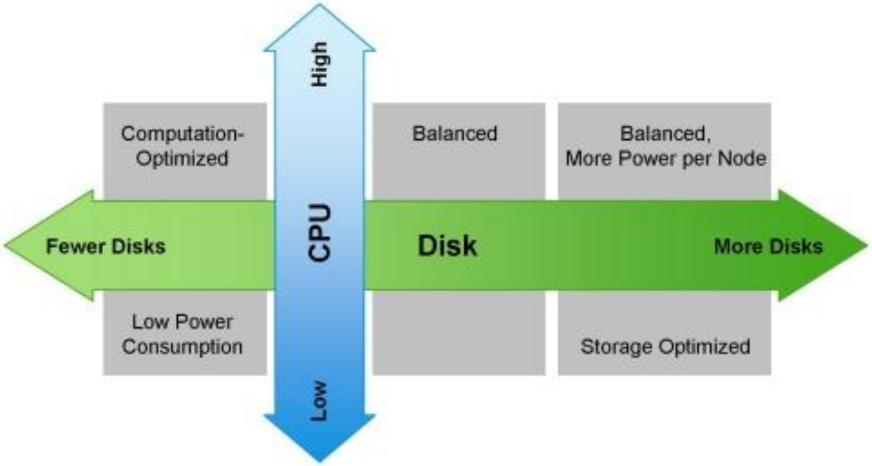
Master Node:
• No Commodity Hardware
•RAIDed hard drives
• Backup Metadata to an NFS Mount
• RAM Thumg rule: 1 GB per 1 million blocks of data. 32GB for 100 nodes.
• If Metadata is lost, whole cluster is lost. Use expensive Name Node.

# 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 Red task
RAM for Hbase Region Server: 0.01 x <dataset size>/<number of slaves>

Cloudwick Technologies

Use Bare Metal Servers Vs Virtualized servers

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.

Cloudwick Technologies

Choosing Right Software

- Using Linux distribution based on Kernel version 2.6.30 or later recommended
- Java 6u14 is or later is recommonded
- Default Linux open file descriptor limit is set to 1024 which is too low for Hadoop daemons. This should be increased to 64000 using /etc/security/limits.conf
- If Linux Kernel 2.6.28 is used, default open ePoll file descriptor limit is 128 which is too low for Hadoop. This should be increased to 4096 in /etc/sysctl.conf
- Adopt a Packaged Hadoop Distribution to Reduce Technical Risk and Increase the Speed of Implementation
- Be Selective About Which Hadoop Projects to Implement
- Use Hadoop in the Cloud for Proof of Concept

Choosing right HADOOP Distro

Cloudwick Technologies

Questions?