Introduction To Hadoop

BY:

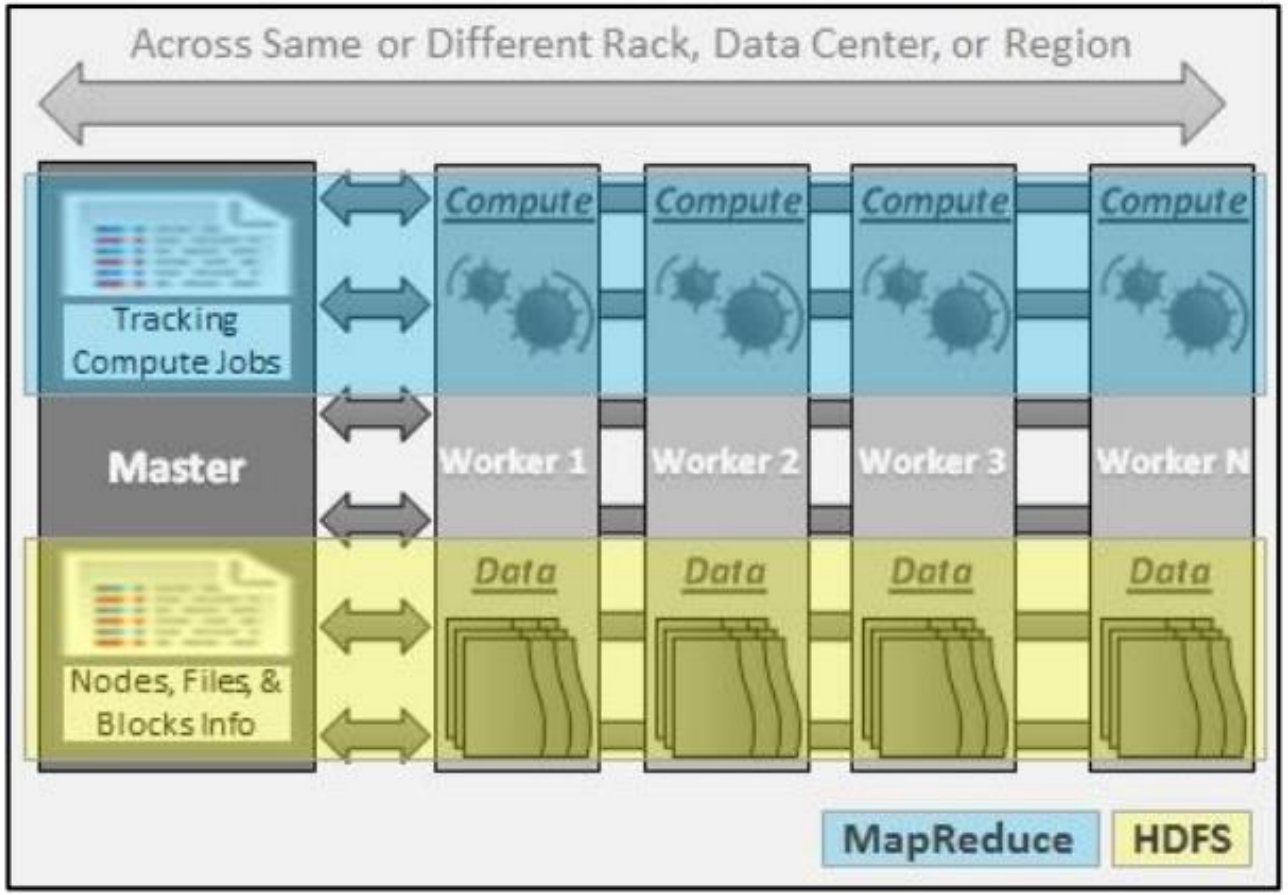
SHAVANTREVVA S BILAKERI

Common Types of Architecture-Multiprocessor

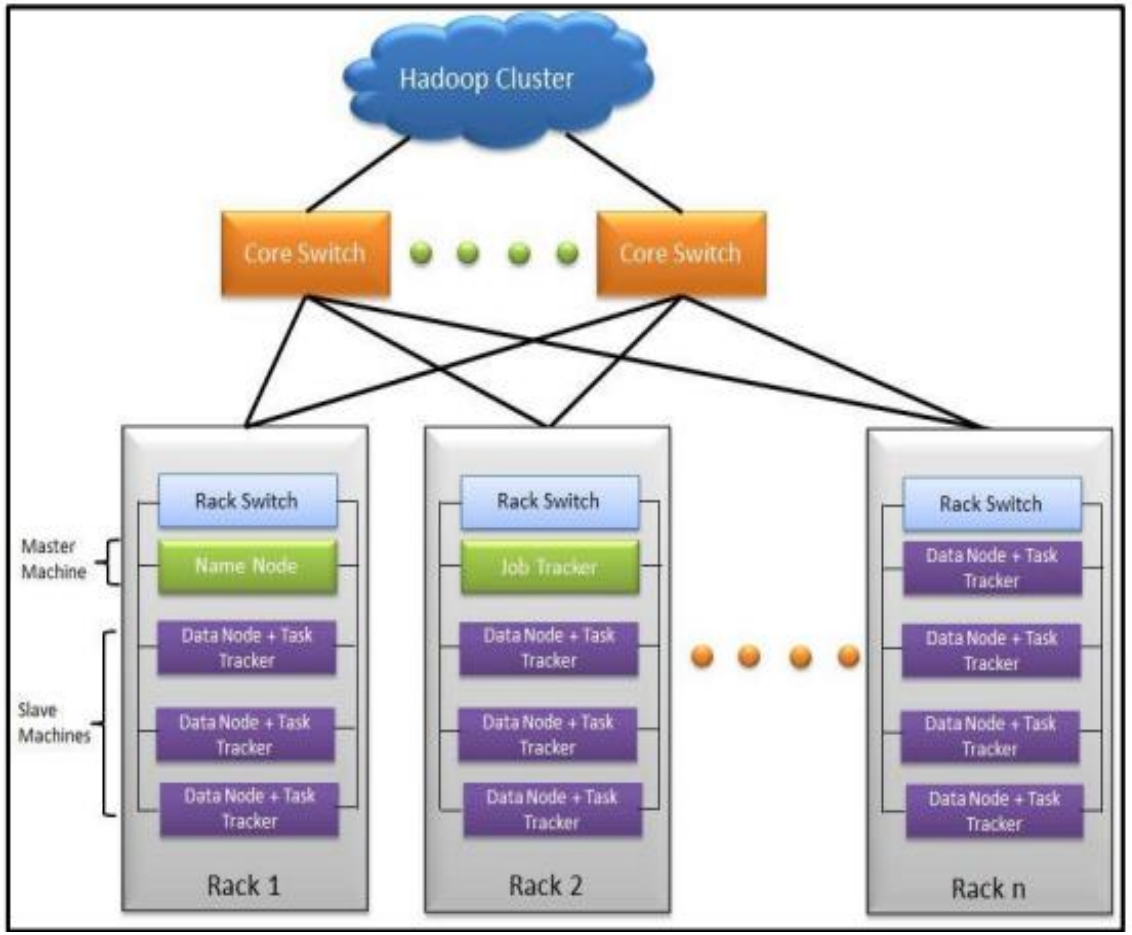
1) Shared Memory (SM): Common Central Memory – Shared by multiple processors
2) Shared Disk (SD) : Multiple Processors - Common Collection of Disks – Own Private Memory
3) Shared Nothing (SN) : Neither memory nor Disk – Shared among multiple processors.

Hadoop Cluster

1. The architecture of Hadoop Cluster
2. Core Components of Hadoop Cluster
3. Work-flow of How File is Stored in Hadoo



1. Hadoop Cluster

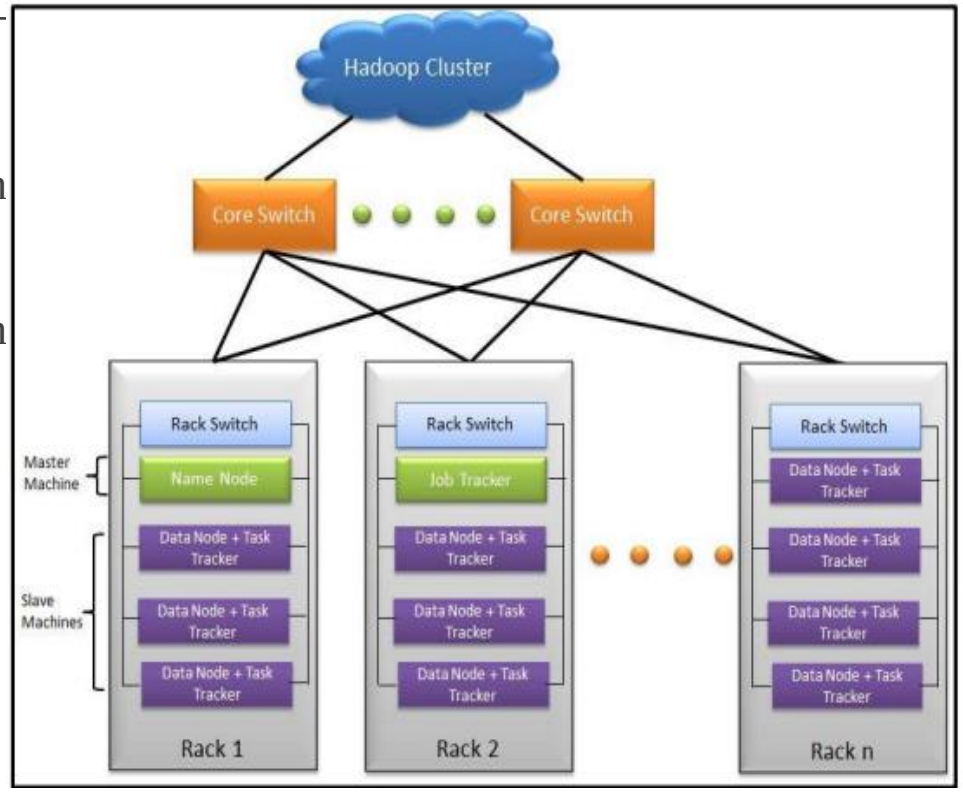


- These clusters run on low cost commodity computers.
- Hadoop clusters are often referred to as "shared nothing" systems because the only thing that is shared between nodes is the network that connects them.
- Large Hadoop Clusters are arranged in several racks.
- Network traffic between different nodes in the same rack is much more desirable than network traffic across the racks.
- Example: Yahoo's Hadoop cluster. They have more than 10,000 machines running Hadoop and nearly 1 petabyte of user data.
- A small Hadoop cluster includes a single master node and multiple worker or slave node.
- As discussed earlier, the entire cluster contains two layers.
- One of the layer of MapReduce Layer and another is of HDFS Layer.
- The master node consists of a JobTracker, NameNode.
- A slave or worker node consists of a DataNode and TaskTracker.
- It is also possible that slave node or worker node is only data or compute node.

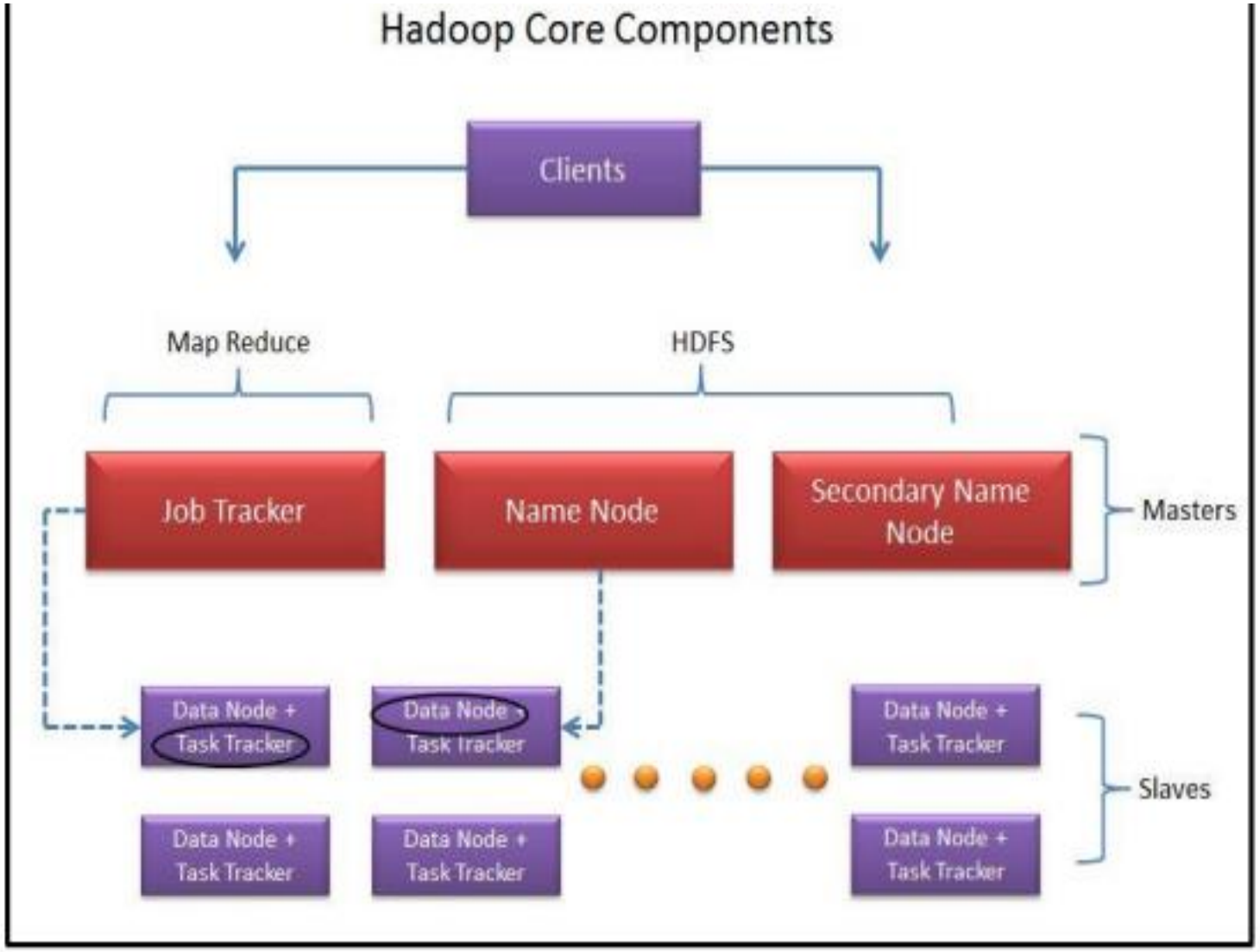
Hadoop Cluster – 3 Components: Client, Master & Slave

Hadoop Cluster would consists of

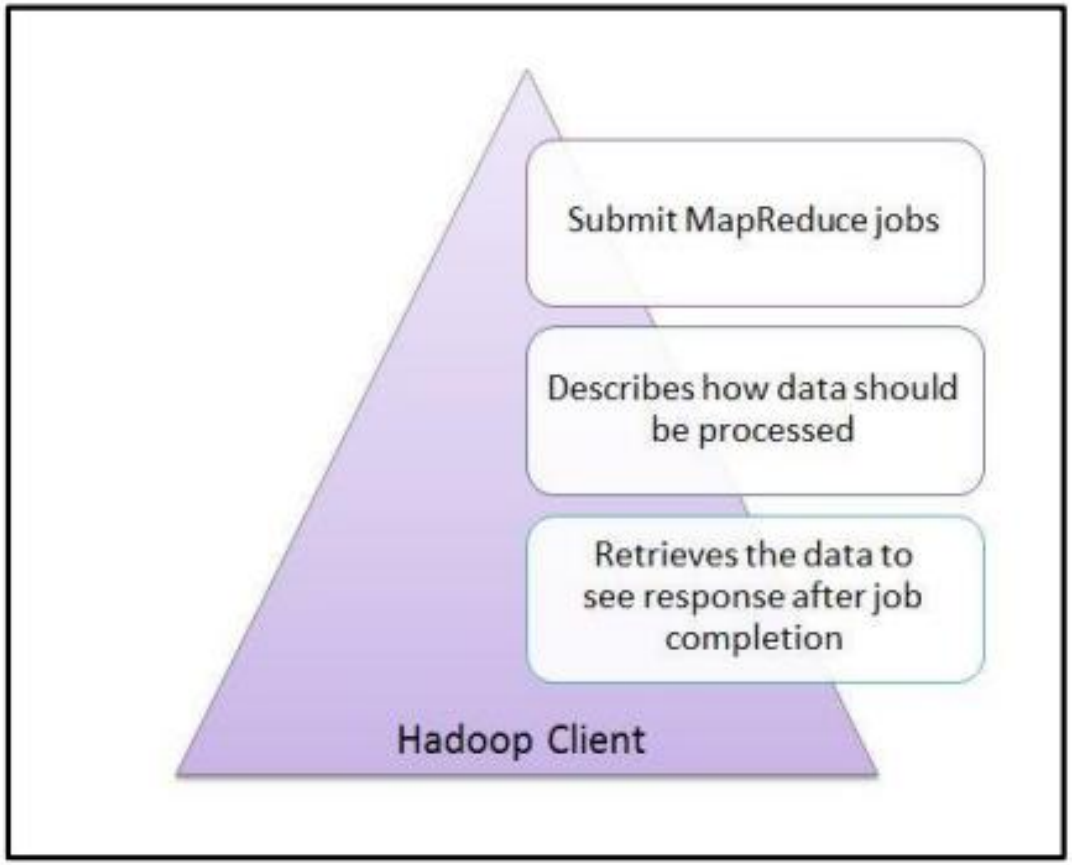
- 110 – Maximum Racks
- Rack - 40 slave machine
- At the top of each rack there is a rack switch
- Each slave machine(rack server in a rack) has cables coming out it from both the ends
- Cables are connected to rack switch at the top which means that top rack switch will have around 80 ports
- Global = 8 core switches
- The rack switch has uplinks connected to core switches and hence connecting all other racks with uniform bandwidth, forming the Cluster
- In the cluster, you have few machines to act as Name node and as JobTracker. They are referred as Masters.



Cluster : Core Components



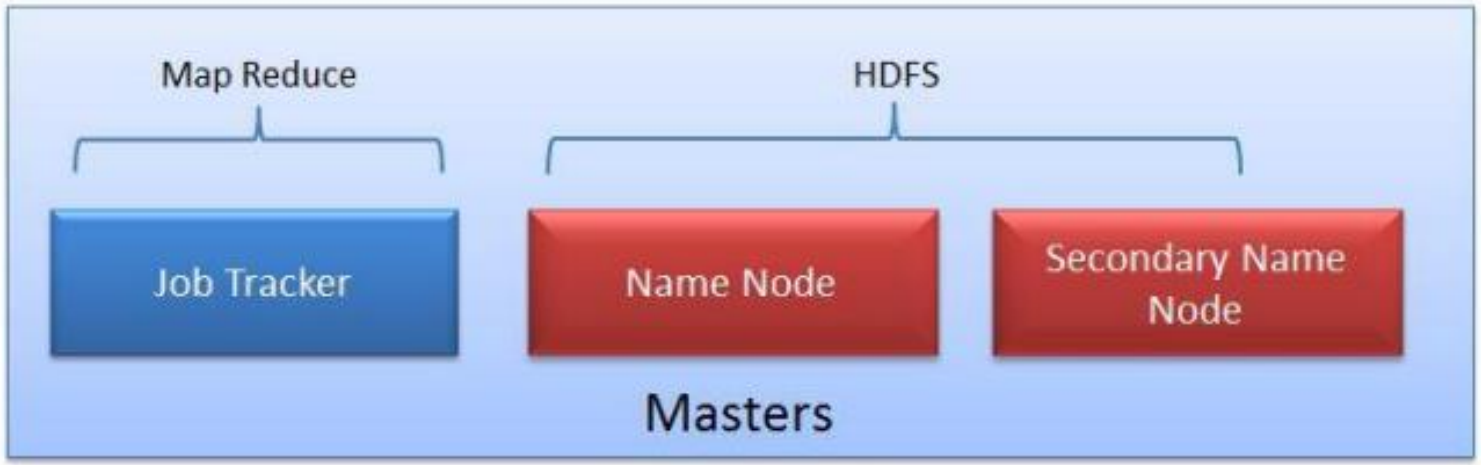
1. Client



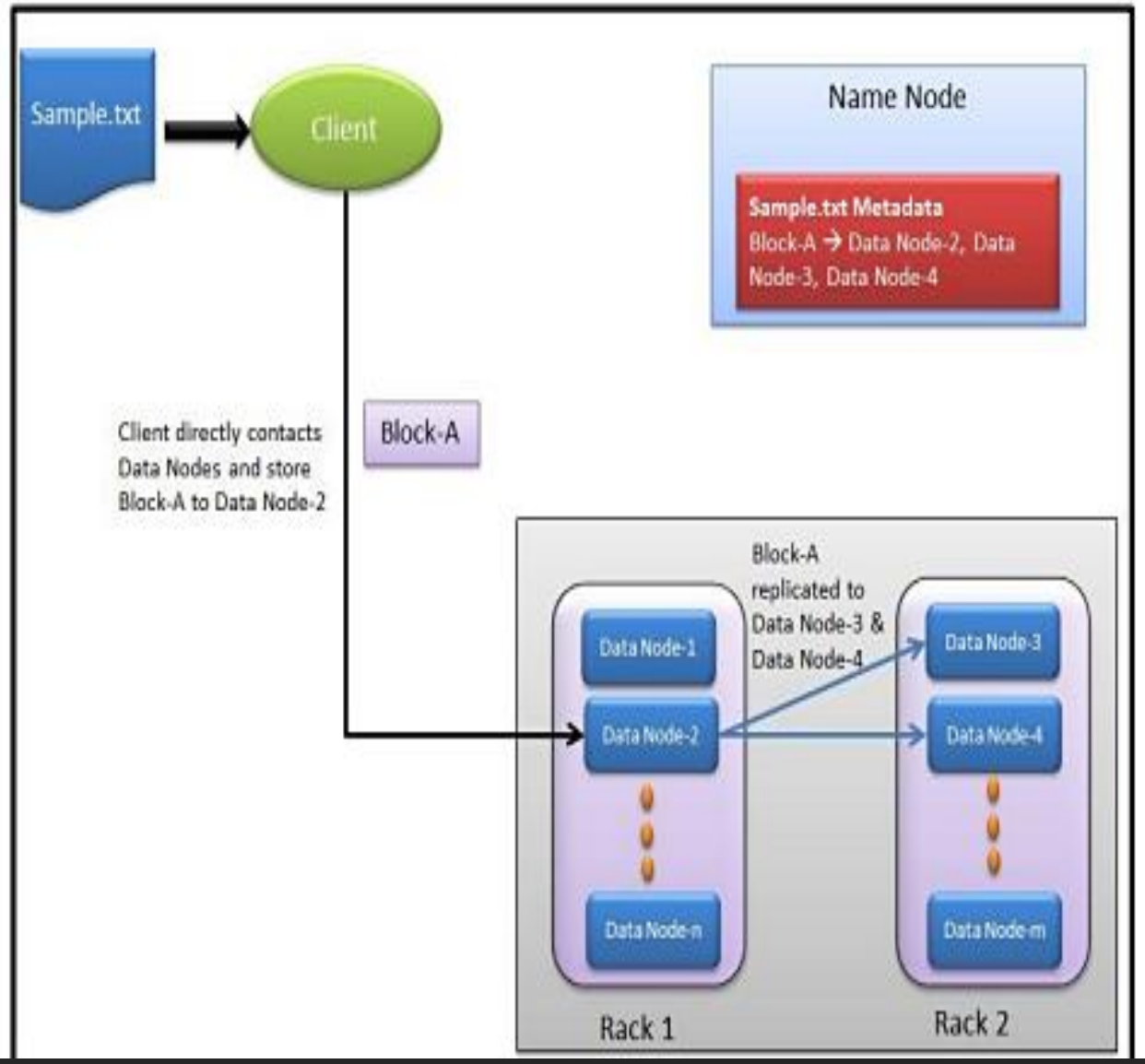
It is neither master nor slave, rather play a role of loading the data into cluster, submit MapReduce jobs describing how the data should be processed and then retrieve the data to see the response after job completion.

2. Masters: Name Node, Secondary Node & Job Tracker

The Masters consists of 3 components NameNode, Secondary Node name and JobTracker.



2.1: Name Node:



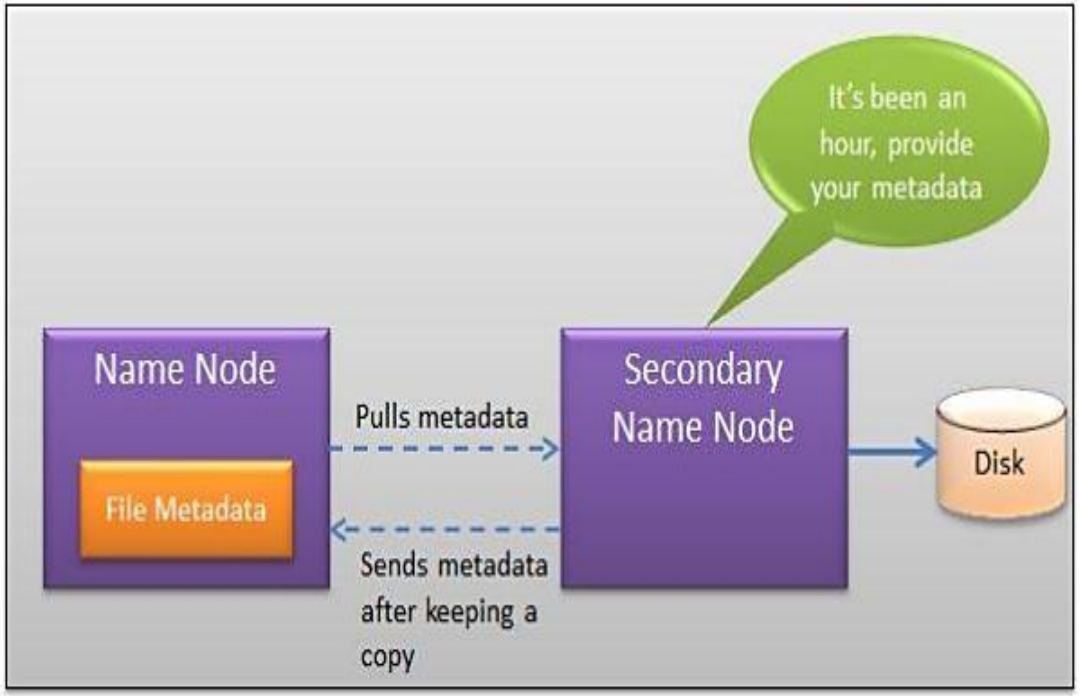
- NameNode oversees the health of DataNode and coordinates access to the data stored in DataNode.

- Name node keeps track of all the file system related information such as:
  - Which section of file is saved in which part of the cluster
  - Last access time for the files
  - User permissions like which user have access to the file

2.2 JobTracker:

JobTracker : Coordinates the parallel processing of data using MapReduce.

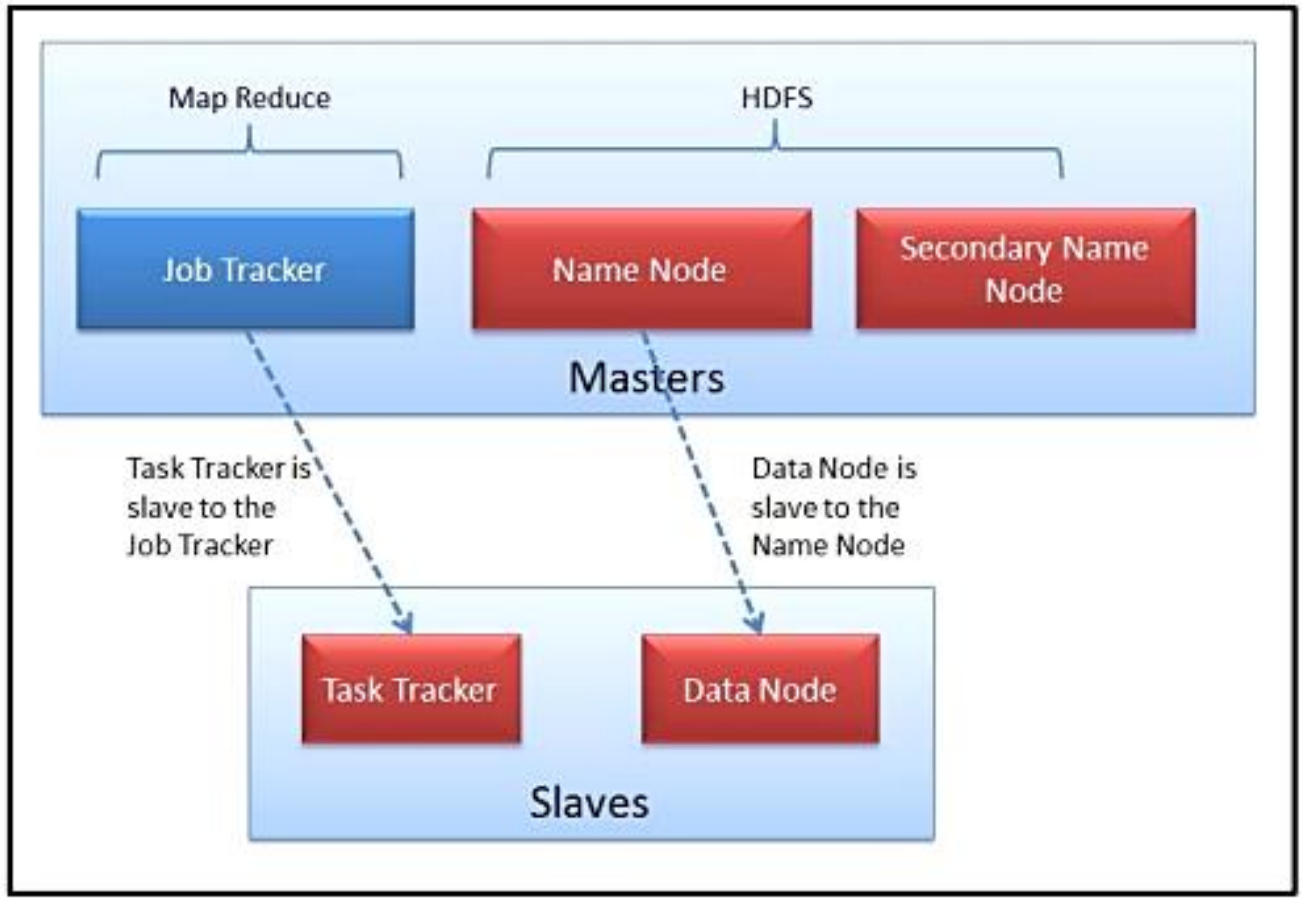
2.3 Secondary Node:



- The job of Secondary Node is to contact NameNode in a periodic manner after certain time interval (by default 1 hour).
- NameNode which keeps all filesystem metadata in RAM has no capability to process that metadata on to disk.

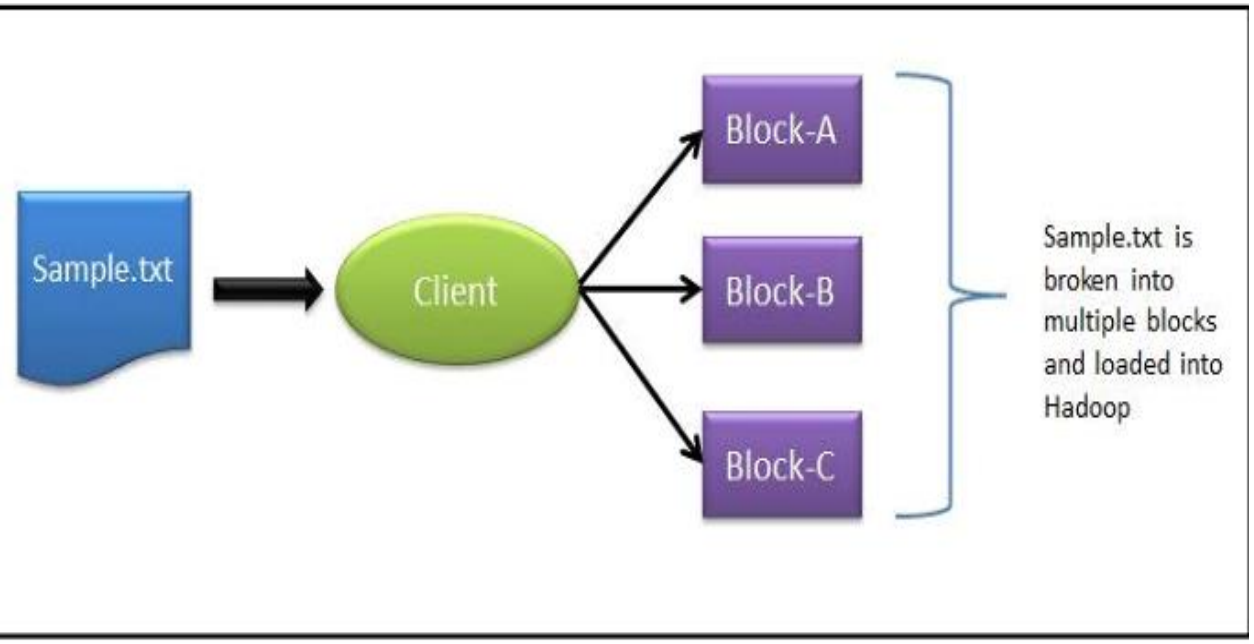
- If NameNode crashes, you lose everything in RAM itself and you don't have any backup of filesystem.
- What secondary node does is it contacts NameNode in an hour and pulls copy of metadata information out of NameNode.
- It shuffle and merge this information into clean file folder and sent to back again to NameNode, while keeping a copy for itself.
- Hence Secondary Node is not the backup rather it does job of housekeeping.
- In case of NameNode failure, saved metadata can rebuild it easily.

3: Slave



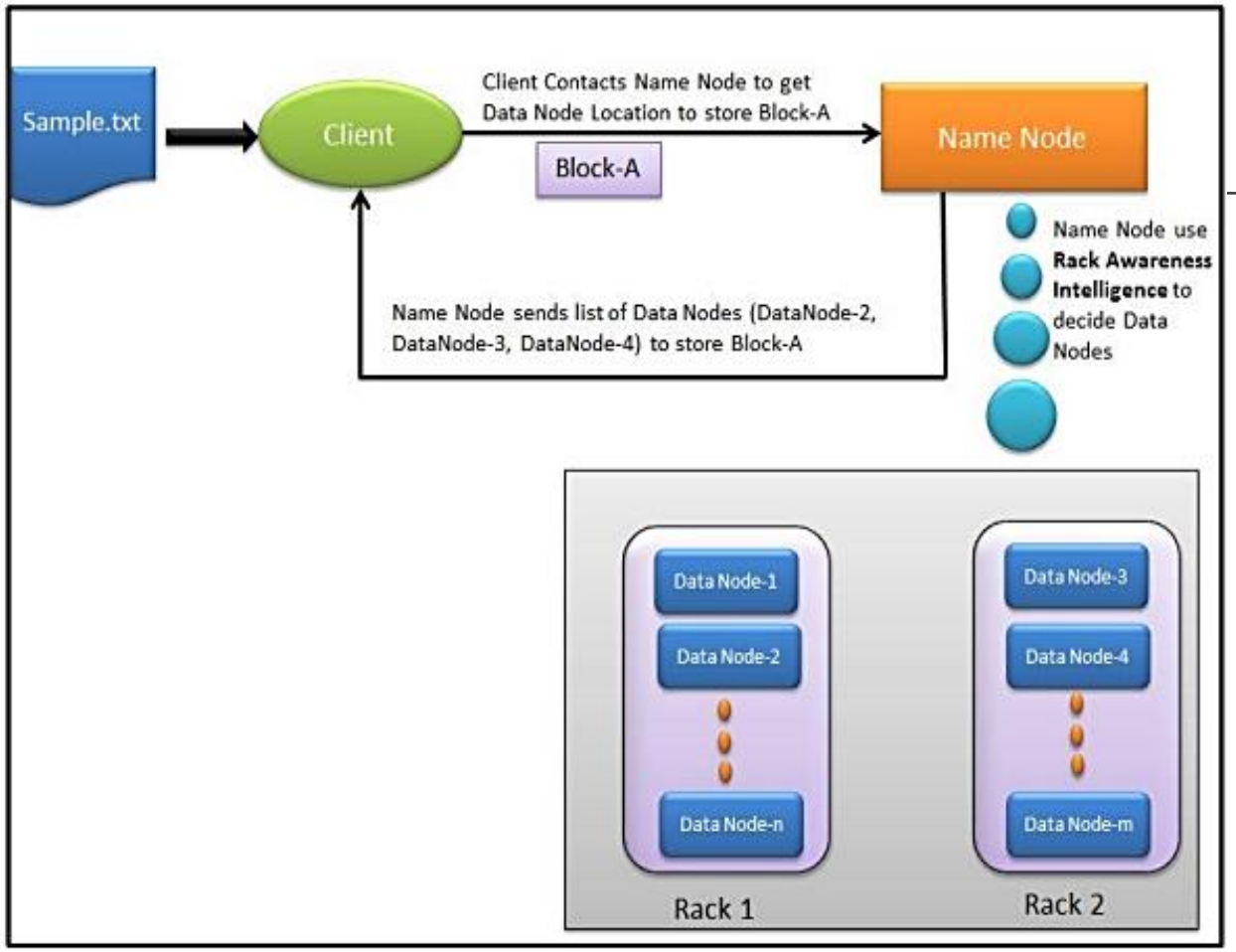
- Slave nodes are the majority of machines in Hadoop Cluster and are responsible to :
  - Store the data
  - Process the computation
- Each slave runs both a DataNode and Task Tracker daemon which communicates to their masters.
- The Task Tracker daemon is a slave to the Job Tracker
- DataNode daemon a slave to the NameNode

Loading File In Hadoop Cluster



- Client machine does this step and loads the Sample.txt into cluster.
- It breaks the sample.txt into smaller chunks which are known as "Blocks" in Hadoop context.
- Client put these blocks on different machines (data nodes) throughout the cluster.

Client knows that to which data nodes load the blocks?

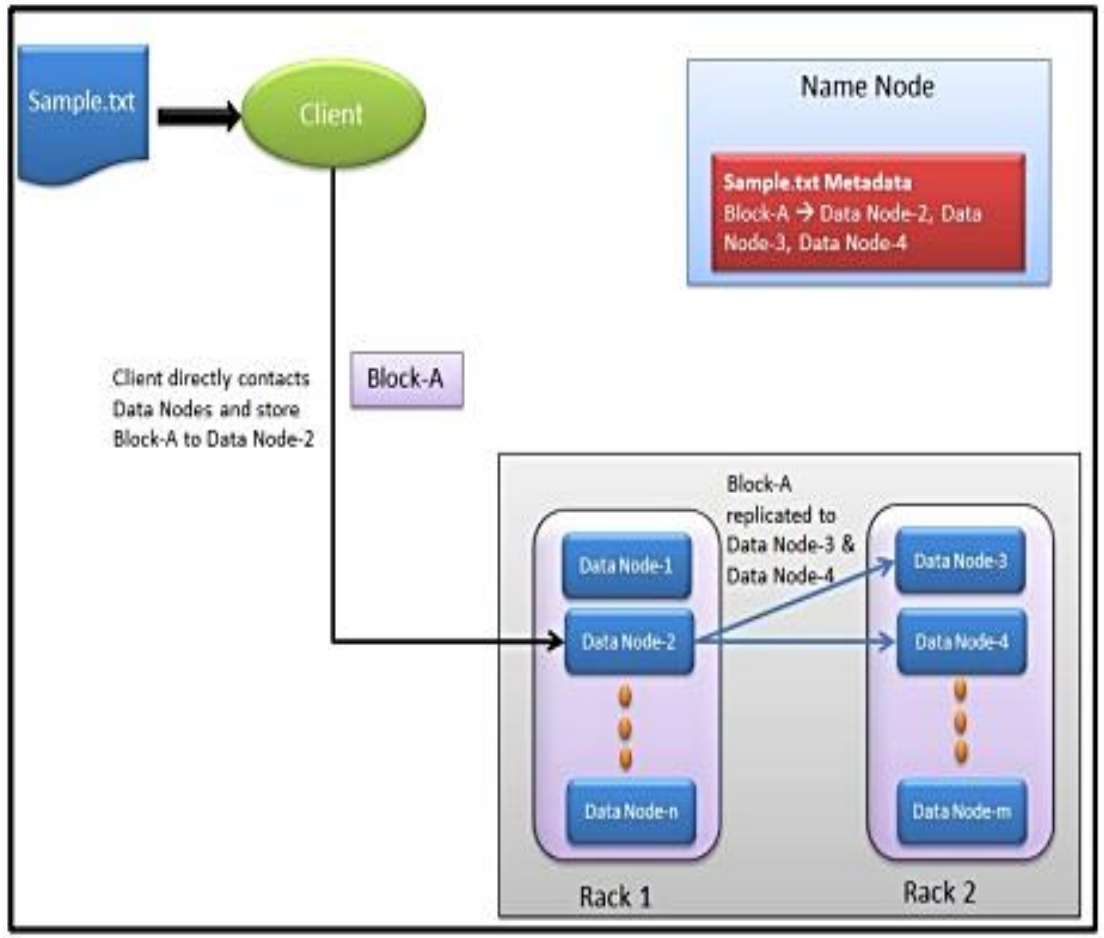


1) Now NameNode comes into picture.
2) The NameNode used its Rack Awareness intelligence to decide on which DataNode to provide.
3) For each of the data block (in this case Block-A, Block-B and Block-C), Client contacts NameNode and in response NameNode sends an ordered list of 3 DataNodes.

For example in response to Block-A request, Node Name may send DataNode-2, DataNode-3 and DataNode-4.

- ✓ Block-B DataNodes list DataNode-1, DataNode-3, DataNode-4 and for Block C data node list DataNode-1, DataNode-2, DataNode-3. Hence
  - ❖ Block A gets stored in DataNode-2, DataNode-3, DataNode-4
  - ❖ Block B gets stored in DataNode-1, DataNode-3, DataNode-4
  - ❖ Block C gets stored in DataNode-1, DataNode-2, DataNode-3
- ✓ Every block is replicated to more than 1 data nodes to ensure the data recovery on the time of machine failures. That's why NameNode send 3 DataNodes list for each individual block

Block replication



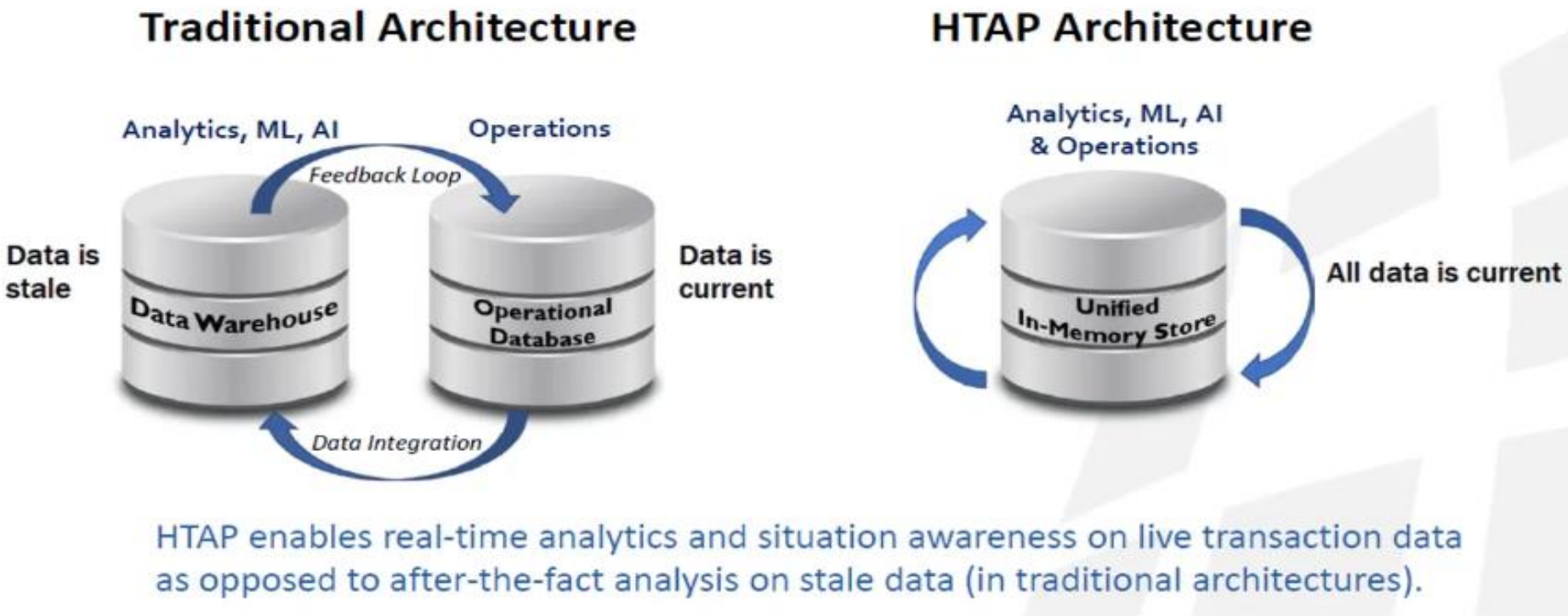
1. Client write the data block directly to one DataNode.
2. DataNodes then replicate the block to other Data nodes.
3. When 1 block gets written in all 3 DataNode then only cycle repeats for next block.
4. In Hadoop Gen 1 there is only one NameNode.
5. In Hadoop Gen2 there is active passive model in NameNode where one more node "Passive Node" comes in picture.
6. The default setting for Hadoop is to have 3 copies of each block in the cluster.
7. This setting can be configured with "dfs.replication" parameter of hdfs-site.xml file.
8. Keep note that Client directly writes the block to the DataNode without any intervention of NameNode in this process.

Parallel Computing vs Distributed Computing

S.NO	Parallel Computing	Distributed Computing
1.	Many operations are performed simultaneously	System components are located at different locations
2.	Single computer is required	Uses multiple computers
3.	Multiple processors perform multiple operations	Multiple computers perform multiple operations
4.	It may have shared or distributed memory	It have only distributed memory
5.	Processors communicate with each other through bus	Computer communicate with each other through message passing.
6.	Improves the system performance	Improves system scalability, fault tolerance and resource sharing capabilities

Digital Transformation is Driving Hybrid Transaction/Analytical Processing (HTAP)

"IMC-enabled HTAP can have a transformational impact on the business." — Gartner 2/17



In-memory computing is said to enable HTAP (Hybrid Transcation/Analytical Processing), which brings benefits in terms of unified architecture and quick access to data and insights. Image: GridGain