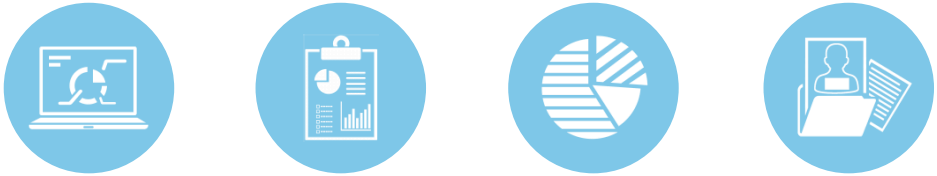
Big Data and Hadoop Developer

Lesson 3—Hadoop Architecture



Objectives

By the end of this lesson, you will be able to:

- Describe the use of Hadoop in commodity hardware
- Explain the various configurations and services of Hadoop
- Differentiate between regular file system and Hadoop Distributed File System (HDFS)
- Explain HDFS architecture

Key Terms

Some key terms used while discussing Hadoop Architecture:

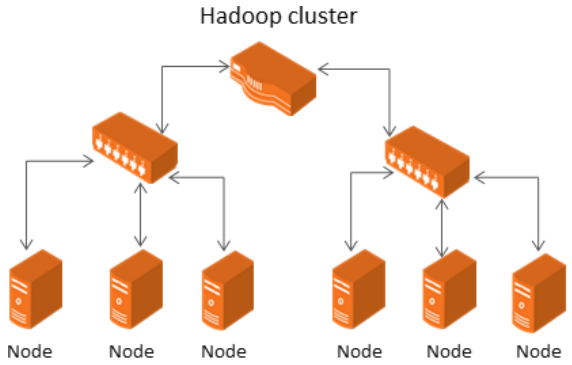
- Commodity hardware: PCs which can be used to make a cluster
- Cluster: Interconnection of systems in a network
- Node: Commodity servers interconnected through a network device

Hadoop Cluster Using Commodity Hardware

Hadoop supports the concept of distributed architecture.

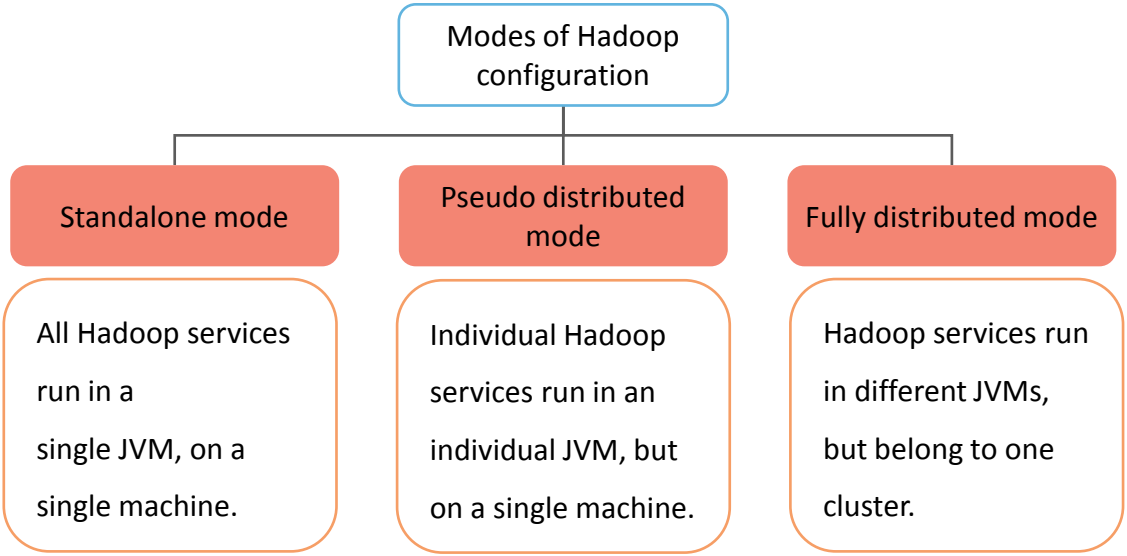
The diagram represents the nodes connected and installed with Hadoop.

- The number of nodes in a rack depends on the network speed.
- Uplink from rack to node is 3 to 4 Gb/s.
- Uplink from rack to rack is 1 Gb/s.



Hadoop Configuration

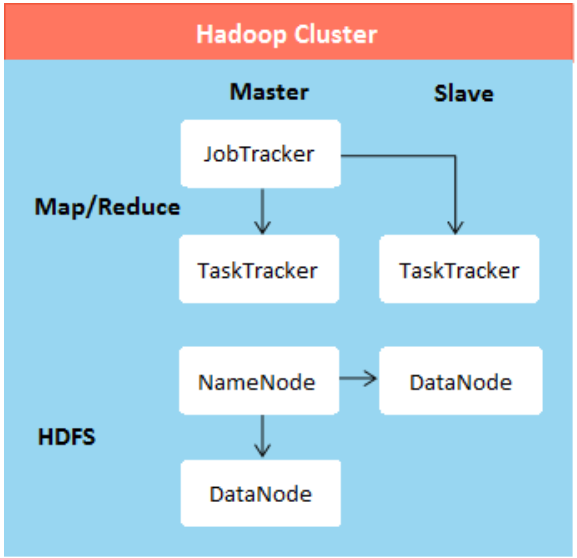
Standalone, pseudo distributed, and fully distributed are three modes of Hadoop configuration.



Hadoop Core Services

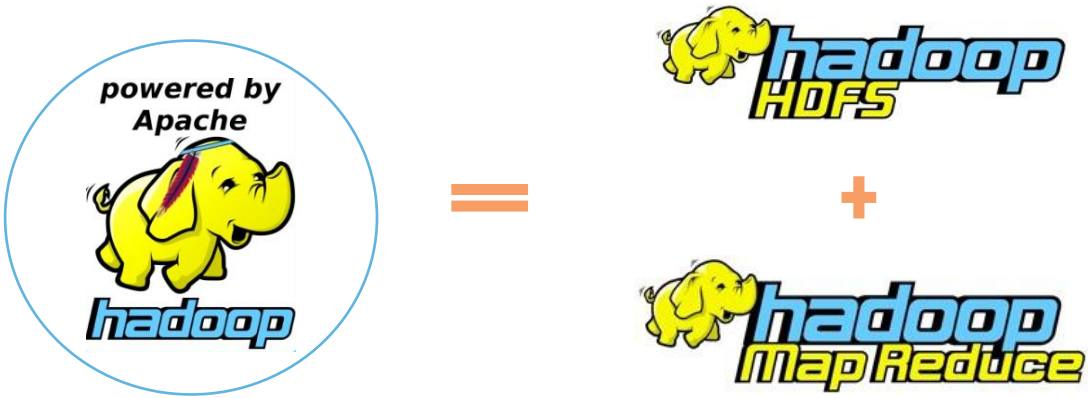
The core services of Hadoop are:

- NameNode
- DataNode
- JobTracker
- TaskTracker
- Secondary NameNode



Apache Hadoop Core Components

Hadoop HDFS and Hadoop MapReduce are the core components of Hadoop.



Hadoop Core Components—HDFS

The key features of Hadoop HDFS are as follows:

- provides high-throughput access to data blocks;
- provides limited interface for managing the file system to allow it to scale; and
- creates multiple replicas of each data block and distributes them on computers throughout the cluster to enable reliable and rapid data access.

The key features of Hadoop MapReduce are as follows:

- performs distributed data processing using the MapReduce programming paradigm;
- allows to possess user-defined map phase, which is a parallel, share-nothing processing of input (MapReduce paradigm); and
- the aggregating the output of the map phase, which is a user-defined, reduces phase after a map process.

Regular File System vs. HDFS

A simple comparison between regular file system and HDFS is summarized below:

Regular File System	HDFS
• Each block of data is small in size; approximately 51 bytes • Large data access suffers from disk I/O problems; mainly because of multiple seek operation	• Each block of data is very large in size; 64MB by default • Reads huge data sequentially after a single seek

HDFS—Characteristics

The basic characteristics of HDFS that make it popular are:

- High fault-tolerance
- High throughput
- Suitable for applications with large data sets
- Suitable for applications with streaming access to file system data
- Can be built on commodity hardware and heterogeneous platforms

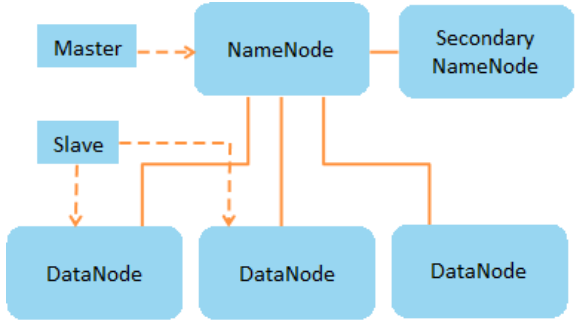
HDFS—Key Features

Some key features of HDFS:

- HDFS creates multiple replicas of each data block and distributes them on computers throughout a cluster to enable reliable and rapid access.
- HDFS is the storage system for both input and output of the MapReduce jobs. Hadoop file URL is to be specified like hdfs://filename.
- Block storage meta data controls the physical location of the block and replication within the cluster.
- Each block is replicated to a small number of physically separate machines.

HDFS architecture can be summarized as follows:

- NameNode and the Secondary NameNode services constitute the master service. DataNode service is the slave service.
- The master service is responsible for accepting a job from clients and ensures that the data required for the operation will be loaded and segregated into chunks of data blocks.
- HDFS exposes a file system namespace and allows user data to be stored in files. A file is split into one or more blocks stored and replicated in DataNodes. The data blocks are then distributed to the DataNode systems within the cluster. This ensures that replicas of the data are maintained.



HDFS Cluster

HDFS Operation Principle

The HDFS components comprise different servers like NameNode, DataNode, and Secondary NameNode.

NameNode Server (single instance)

- Maintains the file system name space
- Manages the files and directories in the file system tree
- Stores information in the namespace image and the edit log
- NameNode knows the data nodes on which all the blocks for a given file exist
- NameNode is a critical one point failure node

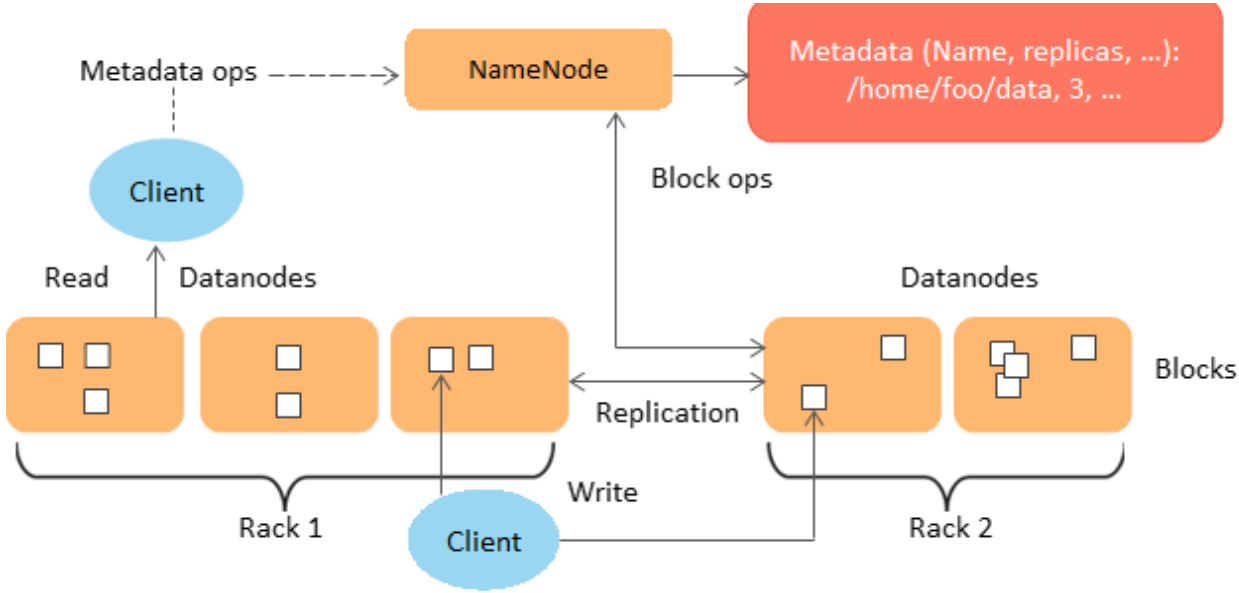
DataNode Server (multiple instances)

- Associated with data storage places in the file system
- Reports to NameNode periodically with lists of blocks they store
- Stores and retrieves blocks when referred by clients or NameNode
- Servers read, write requests, performs block creation, deletion, and replication upon instruction from NameNode

Secondary NameNode Server (single instance)

- Not exactly a hot backup of the actual NameNode server
- Used for recovery of NameNode in case of NameNode failure
- Keeps namespace image through edit log periodically
- Namespace image lags behind, so total recovery is impossible

HDFS is the place where data is stored and Hadoop operations are performed via HDFS.

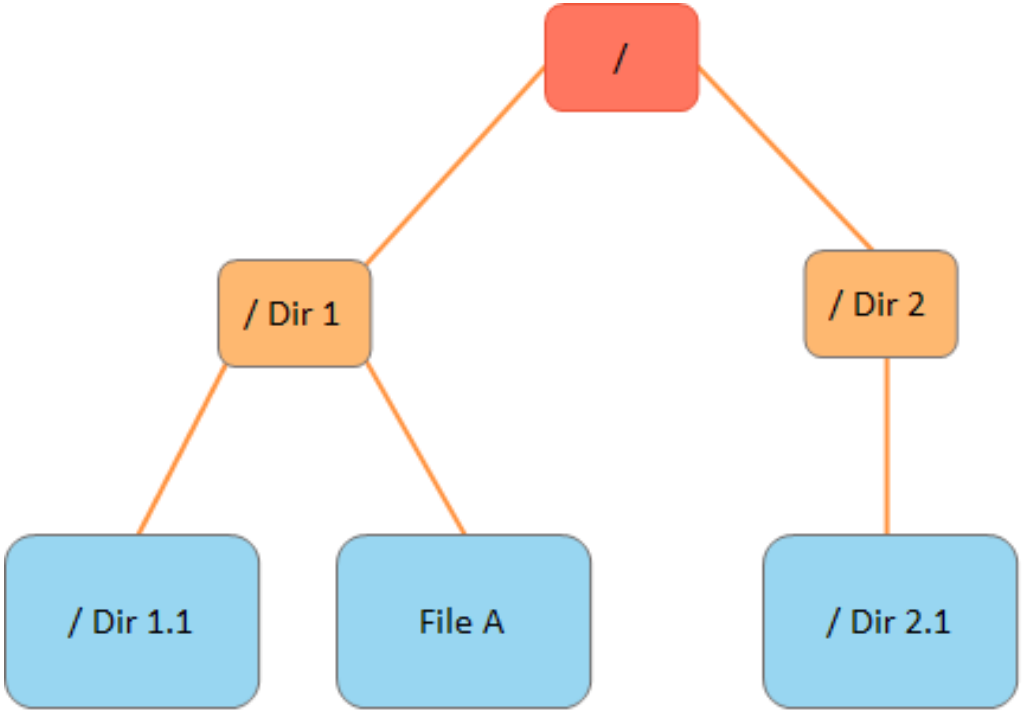


File System Namespace

Some points related to file system

namespace of HDFS:

- HDFS exposes a file system namespace and allows user data to be stored in files.
- HDFS has a Hierarchical file system with directories and files.
- HDFS supports operations like create, remove, move, rename etc.
- The NameNode maintains the file system and records any change to meta information.



NameNode Operation

The HDFS namespace is stored by NameNode.

Namespace

- Entire filesystem namespace including mapping of blocks to files and file system properties is stored in a file FsImage
- Stored in NameNode's local filesystem

Transaction log

- NameNode uses a transaction log called the EditLog to record every change that occurs to the filesystem meta data
- For example, creating a new file
- Stored in the NameNode's local filesystem

In memory meta data

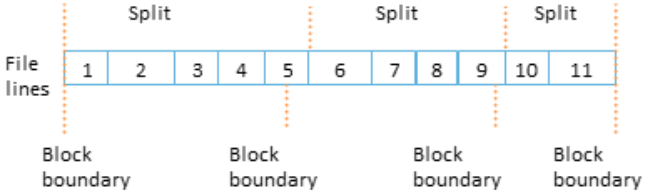
- Centralizes and manages file system metadata in memory
- Maps blocks to DataNodes, filenames, and so on
- Metadata size limited to available RAM of NameNode
- Prefers modest number of large files instead of large number of small files

Data Block Split

Each file is split into one or more blocks stored and replicated in DataNodes.

DataNodes manage names and locations of file blocks, which are 64MB each by default.

- For large files the block size is increased to 128MB to decreases the pressure on the NameNode's memory. This potentially decreases the amount of parallelism that can be achieved, as the number of blocks per file decreases.
- Each map tasks operates on one block, if there are tasks lesser than nodes in the cluster, jobs will run slower than they could. The larger the individual files involved or the more files involved in the average MapReduce job, the less of an issue this is.



Benefits of Data Block Approach

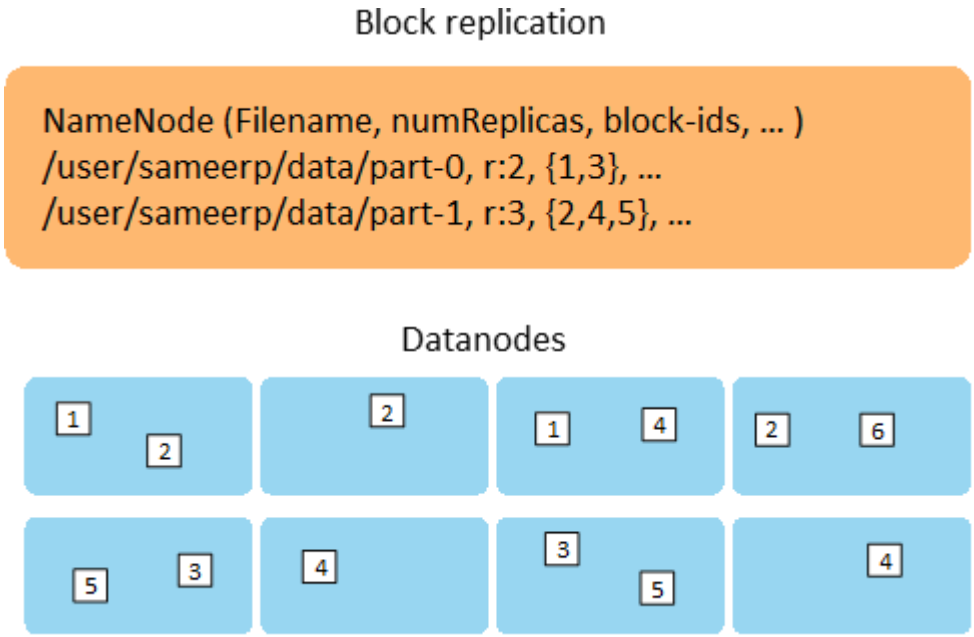
Benefits of data block approach are as follows:

- simplified replication;
- fault tolerance and reliability; and
- shielding users from storage subsystem details.

HDFS—Block Replication Architecture

HDFS represents the unstructured data in the form of data blocks.

- Default size of the blocks is 64 MB; but block size can be reset.
- HDFS performs block replication on multiple data nodes so that if any error on one DataNode Server exists, the JobTracker service will resubmit the job to another DataNode Server.



Replication Method

Each file is split into a sequence of blocks. All blocks in the file except the last are of the same size.

Namespace

- Blocks are replicated for fault tolerance.
- Block replication factor is usually configured at cluster level but can also be configured at file level.

Transaction log

- The NameNode receives a Heartbeat and a BlockReport from each DataNode in the cluster.
- BlockReport contains all the blocks on a DataNode.

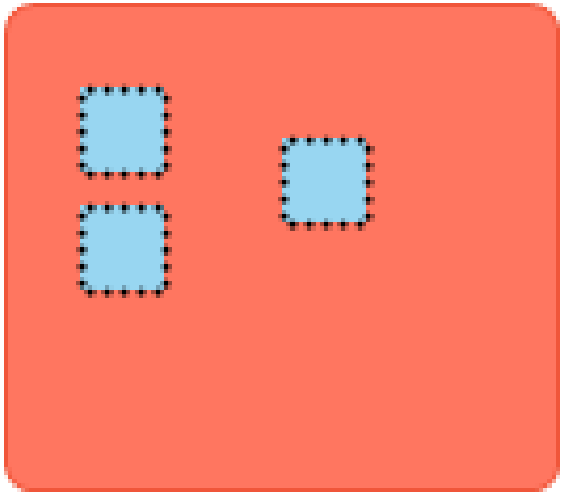
In memory meta data

- An application can specify the number of replicas of the file needed: replication factor of the file. This information is stored in the NameNode.

Data Replication Topology

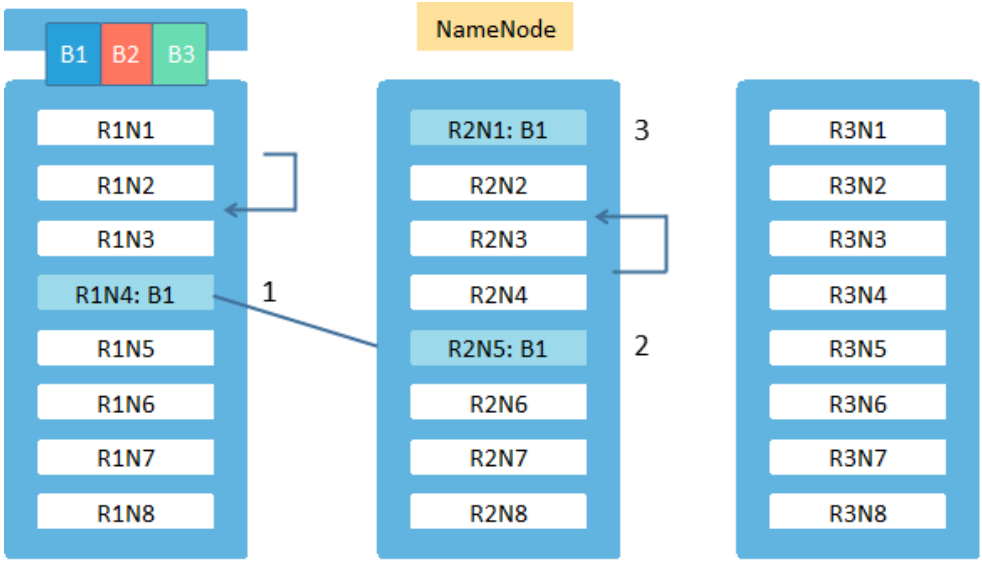
Suggested replication topology

- 1st replica placed on same node as client;
- 2nd replica placed on different rack from 1st rack; and
- 3rd replica placed on same rack as 2nd rack, but on a different node.



Data Replication Representation

Hadoop cluster is represented through the chart:



HDFS Access

HDFS provides following access mechanisms:

- java API for application;
- Python access and C language wrapper for Java API is also available;
- web GUI;
- HTTP browser; and
- FS Shell.

Business Scenario

Olivia Tyler is the EVP of IT operations with Nutri Worldwide Inc. has decided to use HDFS for storing Big Data. She will be using HDFS shell to store the data in Hadoop file system and execute various commands on it.

The demos in the subsequent screens illustrate how to do basic command line operations on HDFS.

Demo

Create a new directory in HDFS. Store a text file in the directory and view its contents

Quiz

QUIZ 1

What are the two major components of Hadoop cluster?

a. Hadoop file system, TaskTracker

b. MapReduce, Hadoop file system

c. JobTracker, MapReduce

d. JobTracker, TaskTracker

QUIZ 1	What are the two major components of Hadoop cluster?

a. Hadoop file system, TaskTracker

b. MapReduce, Hadoop file system

c. JobTracker, MapReduce

d. JobTracker, TaskTracker

Answer: b.

Explanation: The two major components of Hadoop cluster are MapReduce and Hadoop File System.

QUIZ 2

Which of the following services run in the Master node of Apache Hadoop in cluster mode (fully distributed mode)?

a. JobTracker

b. TaskTracker

c. JobTracker, MapReduce

d. JobTracker, TaskTracker

QUIZ 2

Which of the following services run in the Master node of Apache Hadoop in cluster mode (fully distributed mode)?

a. JobTracker

b. TaskTracker

c. JobTracker, MapReduce

d. JobTracker, TaskTracker

Answer: a.

Explanation: JobTracker service runs in the Master node of Apache Hadoop in cluster mode.

QUIZ 3

Which are the single instance critical tasks?

a. NameNode, DataNode

b. JobTracker, DataNode

c. JobTracker, NameNode

d. NameNode, Secondary NameNode

QUIZ 3

Which are the single instance critical tasks?

a. NameNode, DataNode

b. JobTracker, DataNode

c. JobTracker, NameNode

d. NameNode, Secondary NameNode

Answer: c.

Explanation: Only one tracker takes care of task trackers and MapReduce jobs and only one NameNode keeps block information like metadata for allocated storage on the data nodes.

QUIZ 4

How is NameNode failure tackled?

a. Secondary NameNode is switched on as NameNode

b. Secondary NameNode automatically starts working as NameNode

c. From Secondary NameNode image backup primary NameNode is recreated

d. Another NameNode in cluster with replication works as main NameNode

QUIZ 4

How is NameNode failure tackled?

a. Secondary NameNode is switched on as NameNode

b. Secondary NameNode automatically starts working as NameNode

c. From Secondary NameNode image backup primary NameNode is recreated

d. Another NameNode in cluster with replication works as main NameNode

Answer: c.

Explanation: NameNode failure is tackled by taking an image backup of the NameNode from Secondary NameNode and incorporating it into a new NameNode machine.

QUIZ 5

Which of the following services are used by MapReduce programs?

a. JobTracker and TaskTracker

b. TaskTracker

c. NameNode

d. Secondary NameNode

QUIZ 5

Which of the following services are used by MapReduce programs?

a. JobTracker and TaskTracker

b. TaskTracker

c. NameNode

d. Secondary NameNode

Answer: a.

Explanation: JobTracker and TaskTracker are used by the MapReduce programs to perform its functionalities.

QUIZ 6

Which of the following statements best describes how a large (100 GB) file is stored in HDFS?

a. The file is replicated three times by default. Each copy of the file is stored on a separate DataNode.

b. The master copy of the file is stored on a single DataNode. The replica copies are divided into fixed-size blocks, which are stored on multiple data nodes.

c. The file is divided into fixed-size blocks, which are stored on multiple data nodes. Each block is replicated three times by default. Multiple blocks from the same file might reside on the same DataNode.

d. The file is divided into fixed-size blocks, which are stored on multiple data nodes. Each block is replicated three times by default. HDFS guarantees that different blocks from the same file are never on the same DataNode.

QUIZ 6

Which of the following statements best describes how a large (100 GB) file is stored in HDFS?

a. The file is replicated three times by default. Each copy of the file is stored on a separate DataNode.

b. The master copy of the file is stored on a single DataNode. The replica copies are divided into fixed-size blocks, which are stored on multiple data nodes.

c. The file is divided into fixed-size blocks, which are stored on multiple data nodes. Each block is replicated three times by default. Multiple blocks from the same file might reside on the same DataNode.

d. The file is divided into fixed-size blocks, which are stored on multiple data nodes. Each block is replicated three times by default. HDFS guarantees that different blocks from the same file are never on the same DataNode.

Answer: d.

Explanation: The file is divided into fixed-size blocks, which are stored on multiple data nodes. Each block is replicated three times by default. HDFS guarantees that different blocks from the same file are never on the same DataNode.

QUIZ 7

Which of the following describes how a client reads a file from HDFS?

a. The client queries the NameNode for the block location(s). The NameNode returns the block location(s) to the client. The client reads the data directly off the DataNode(s).

b. The client queries all DataNodes in parallel. The DataNode that contains the requested data responds directly to the client. The client reads the data directly off the DataNode.

c. The client contacts the NameNode for the block location(s).

d. The NameNode contacts the DataNode that holds the requested data block. Data is transferred from the DataNode to the NameNode, and then from the NameNode to the client.

QUIZ 7

Which of the following describes how a client reads a file from HDFS?

a. The client queries the NameNode for the block location(s). The NameNode returns the block location(s) to the client. The client reads the data directly off the DataNode(s).

b. The client queries all DataNodes in parallel. The DataNode that contains the requested data responds directly to the client. The client reads the data directly off the DataNode.

c. The client contacts the NameNode for the block location(s).

d. The NameNode contacts the DataNode that holds the requested data block. Data is transferred from the DataNode to the NameNode, and then from the NameNode to the client.

Answer: c.

Explanation: The client contacts the NameNode for the block location(s). NameNode then queries the data nodes for block locations. The data nodes respond to the NameNode, and NameNode redirects the client to the DataNode that holds the requested data block(s).The client then reads the data directly off the DataNode.

QUIZ 8

Which of the following are valid statements? (Choose two)

a. HDFS is optimized for storing a large number of files smaller than the HDFS block size.

b. HDFS has the characteristic of supporting a "write once, read many" data access model.

c. HDFS is a distributed file system that replaces ext3 or ext4 on Linux nodes in a Hadoop cluster.

d. HDFS is a distributed file system that runs on top of native OS file systems and is well-suited for storage of very large data sets.

QUIZ 8

Which of the following are valid statements? (Choose two)

a. HDFS is optimized for storing a large number of files smaller than the HDFS block size.

b. HDFS has the characteristic of supporting a "write once, read many" data access model.

c. HDFS is a distributed file system that replaces ext3 or ext4 on Linux nodes in a Hadoop cluster.

d. HDFS is a distributed file system that runs on top of native OS file systems and is well-suited for storage of very large data sets.

Answer: b and d.

Explanation: HDFS has a characteristic of supporting a “Write once, read many” data access model and a distributed file system that runs on top of native OS well suited to storage of very large datasets.

QUIZ 9

The NameNode uses RAM for the following purpose:

a. To store the file contents in HDFS

b. To store filenames, list of blocks, and other meta information

c. To store the edits log that keeps track of changes in HDFS

d. To manage distributed read and write locks on files in HDFS

QUIZ 9

The NameNode uses RAM for the following purpose:

a. To store the file contents in HDFS

b. To store filenames, list of blocks, and other meta information

c. To store the edits log that keeps track of changes in HDFS

d. To manage distributed read and write locks on files in HDFS

Answer: b.

Explanation: NameNode uses RAM to store filenames, list of blocks, and other meta information.

QUIZ 10	You need to move a file titled weblogs into HDFS. You are not allowed to copy the file. You know you have ample space on your data nodes. Which action should you take to relieve this situation and store more files in HDFS?

a. Increase the block size on all current files in HDFS

b. Increase the block size on your remaining files

c. Decrease the block size on your remaining files

d. Increase the amount of memory for the NameNode

QUIZ 10	You need to move a file titled weblogs into HDFS. You are not allowed to copy the file. You know you have ample space on your data nodes. Which action should you take to relieve this situation and store more files in HDFS?

a. Increase the block size on all current files in HDFS

b. Increase the block size on your remaining files

c. Decrease the block size on your remaining files

d. Increase the amount of memory for the NameNode

Answer: c.

Explanation: It is recommended to decrease the block size on your remaining files.

Summary

Let us summarize the topics covered in this lesson:

- Hadoop works on three configurations, namely, standalone mode, pseudo-distributed mode, and fully distributed mode.
- The two core components of Apache Hadoop are HDFS and MapReduce.
- The HDFS layer contains NameNode, Secondary NameNode, and DataNode.
- HDFS can be used to handle to handle big-data sets with its inherent features of fault tolerance, high throughout and streaming data access.

Thank You