

# QUESTIONS & ANSWERS

Kill your exam at first Attempt



Cloudera

## CCD-470

*Cloudera Certified Developer for Apache Hadoop CDH4 Upgrade (CCDH)*

<http://killexams.com/exam-detail/CCD-470>



- B. Because combiners perform local aggregation of word counts, thereby reducing the number of mappers that need to run.
- C. Because combiners perform local aggregation of word counts, and then transfer that data to reducers without writing the intermediate data to disk.
- D. Because combiners perform local aggregation of word counts, thereby reducing the number of key-value pairs that need to be snuff let across the network to the reducers.

**Answer:** A

**Explanation:**

\* Simply speaking a combiner can be considered as a “mini reducer” that will be applied potentially several times still during the map phase before to send the new (hopefully reduced) set of key/value pairs to the reducer(s). This is why a combiner must implement the Reducer interface (or extend the Reducer class as of hadoop 0.20).

\* Combiners are used to increase the efficiency of a MapReduce program. They are used to aggregate intermediate map output locally on individual mapper outputs. Combiners can help you reduce the amount of data that needs to be transferred across to the reducers. You can use your reducer code as a combiner if the operation performed is commutative and associative. The execution of combiner is not guaranteed, Hadoop may or may not execute a combiner. Also, if required it may execute it more then 1 times. Therefore your MapReduce jobs should not depend on the combiners execution.

**QUESTION:** 90

Which two 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 Hadoopcluster.
- D. HDFS is a distributed file system that runs on top of native OS filesystems and is well suited to storage of very large data sets.

**Answer:** B, D

**Explanation:**

B: HDFS is designed to support very large files. Applications that are compatible with HDFS are those that deal with large data sets. These applications write their data only once but they read it one or more times and require these reads to be

satisfied at streaming speeds. HDFS supports write-once-read-many semantics on files. D:

\* Hadoop Distributed File System: A distributed file system that provides high-throughput access to application data.

\* DFS is designed to support very large files.

**Reference:**

24 Interview Questions & Answers for Hadoop MapReduce developers

**QUESTION: 91**

You need to create a GUI application to help your company's sales people add and edit customer information. Would HDFS be appropriate for this customer information file?

- A. Yes, because HDFS is optimized for random access writes.
- B. Yes, because HDFS is optimized for fast retrieval of relatively small amounts of data.
- C. No, because HDFS can only be accessed by MapReduce applications.
- D. No, because HDFS is optimized for write-once, streaming access for relatively large files.

**Answer: D**

**Explanation:**

HDFS is designed to support very large files. Applications that are compatible with HDFS are those that deal with large data sets. These applications write their data only once but they read it one or more times and require these reads to be satisfied at streaming speeds. HDFS supports write-once-read-many semantics on files.

**Reference:**

24 Interview Questions & Answers for Hadoop MapReduce developers, What is HDFS ? How it is different from traditional file systems?

**QUESTION: 92**

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 blocklocation(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). The NameNode then queries the DataNodes for block locations. The DataNodes respond to the

NameNode, and the NameNode redirects the client to the DataNode that holds the requested data block(s). The client then reads the data directly off the DataNode.

D. The client contacts the NameNode for the block location(s). 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 communication to HDFS happens using Hadoop HDFS API. Client applications talk to the NameNode whenever they wish to locate a file, or when they want to add/copy/move/delete a file on HDFS. The NameNode responds the successful requests by returning a list of relevant DataNode servers where the data lives. Client applications can talk directly to a DataNode, once the NameNode has provided the location of the data.

**Reference:**

24 Interview Questions & Answers for Hadoop MapReduce developers, How the Client communicates with HDFS?

**QUESTION:** 93

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

A. The file is divided into variable size blocks, which are stored on multiple data nodes. Each block is replicated three times by default.

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

C. 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 datanodes.

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

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

**Answer:** E

**Explanation:**

HDFS is designed to reliably store very large files across machines in a large cluster. It store each file as a sequence of blocks; all blocks in a file except the last block are the same size. The blocks of a file are replicated for fault tolerance. The block size and replication factor are configurable per file. An application can specify the

number of replicas of a file. The replication factor can be specified at file creation time and can be changed later. Files in HDFS are write-once and have strictly one writer at any time. The NameNode makes all decisions regarding replication of blocks. HDFS uses rack-aware replica placement policy. In default configuration there are total 3 copies of a datablock on HDFS, 2 copies are stored on datanodes on same rack and 3rd copy on a different rack.

**Reference:**

24 Interview Questions & Answers for Hadoop MapReduce developers , How the HDFS Blocks are replicated?

**QUESTION: 94**

Your cluster has 10 DataNodes, each with a single 1 TB hard drive. You utilize all your disk capacity for HDFS, reserving none for MapReduce. You implement default replication settings. What is the storage capacity of your Hadoop cluster (assuming no compression)?

- A. about 3 TB
- B. about 5 TB
- C. about 10 TB
- D. about 11 TB

**Answer: A**

**Explanation:**

In default configuration there are total 3 copies of a datablock on HDFS, 2 copies are stored on datanodes on same rack and 3rd copy on a different rack.

Note: HDFS is designed to reliably store very large files across machines in a large cluster. It stores each file as a sequence of blocks; all blocks in a file except the last block are the same size. The blocks of a file are replicated for fault tolerance. The block size and replication factor are configurable per file. An application can specify the number of replicas of a file. The replication factor can be specified at file creation time and can be changed later. Files in HDFS are write-once and have strictly one writer at any time. The NameNode makes all decisions regarding replication of blocks. HDFS uses rack-aware replica placement policy.

**Reference:**

24 Interview Questions & Answers for Hadoop MapReduce developers , How the HDFS Blocks are replicated?

For More exams visit <https://killexams.com/vendors-exam-list>



*Kill your exam at First Attempt....Guaranteed!*