BIG DATA TOOLS

(1/1 point)

Why is using traditional analysis tools for big data a poor choice?

Top of Form

Storage is becoming less expensive.The time to read from a 1 TB drive is 3 hours.Big data does not fit on a single machine. Big data does not fit on a single machine. - correctCPUs are getting faster and faster every year.

Bottom of Form

**EXPLANATION**

Big data is data that is many 10's of Terabytes to Petabytes in size, which means it does not fit on a single machine.

MODERN BIG DATA HARDWARE

(1/1 point)

Which of the following properties does modern hardware for big data have:

Top of Form

Uses consumer grade hardware, Uses complex software, Easy to add capacity, - correct

Uses premium hardwareUses consumer grade hardware Uses complex hardwareUses complex software Easy to add capacity

Bottom of Form

Note: Make sure you select all of the correct options—there may be more than one!

**EXPLANATION**

Modern big data hardware is based on less expensive, consumer grade hardware which makes it easy to grow capacity. Software is used to handle any problems, instead of hardware.

USING DIVIDE AND CONQUER

(1/1 point)

What are some of the challenges of using divide and conquer:

Top of Form

Moving data is very expensive, Having many machines means having to deal with many failures, Having many machines means having to deal with slow machines, - correct

Moving data is very expensive Using a single machine is faster than multiple machinesHaving many machines means having to deal with many failures Having many machines means having to deal with slow machines Using hash tables for very large documents works well

Bottom of Form

Note: Make sure you select all of the correct options—there may be more than one!

**EXPLANATION**

When using divide and conquer, you have to consider the network and data locality because moving data between machines is expensive. Even with a low per-machine failure rate, using many machines means that several will fail per day. As machines age, they may fail in ways that cause slow performance (e.g., a failing disk drive that retries each read or write operation multiple times before successfully completing).

MAP REDUCE

(1/1 point)

Which of the following problems does a MapReduce implementation handle?

Top of Form

Recovering from machine failures, Shuffling data between the Map and Reduce functions, Running the Map and Reduce functions on many machines, Recovering from slow machines, - correct

Recovering from machine failures Shuffling data between the Map and Reduce functions Running the Map and Reduce functions on many machines Automatically parallelizing an algorithmRecovering from slow machines

Bottom of Form

Note: Make sure you select all of the correct options—there may be more than one!

**EXPLANATION**

Map Reduce handles the execution of Map and Reduce functions on many machines including the shuffling of data between Map and Reduce functions. It also automatically recovers from both machine failures and slow machines.

MAP REDUCE AND SPARK DIFFERENCES

(1/1 point)

Spark is often faster than a traditional MapReduce implementation because:

Top of Form

Results do not need to be written to disk, Results do not need to be serialized, - correct

It sends less data over the networkResults do not need to be written to disk It detects machine failures more quicklyIt replicates the output of each task to recover from failures quicklyResults do not need to be serialized

Bottom of Form

Note: Make sure you select all of the correct options—there may be more than one!

**EXPLANATION**

Spark keeps results in memory so they do not need to be serialized (converted into a format that can be stored on disk) and they do not need to be written to disk.