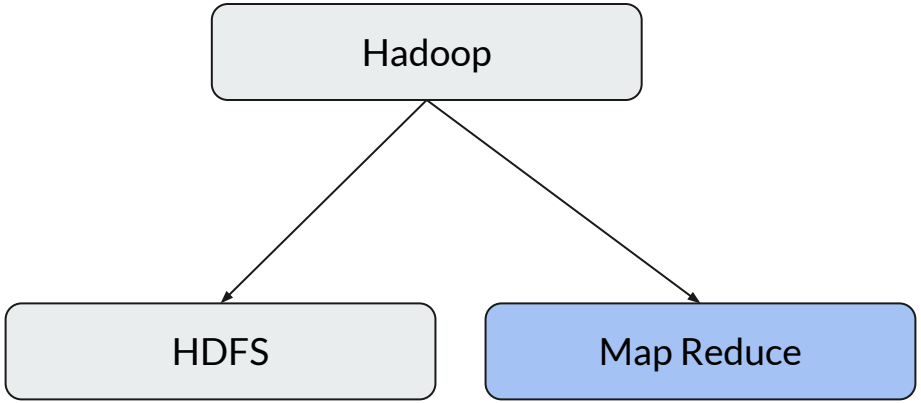

hadoop
Map Reduce

Recall ...

Hadoop Fundamentals: Storage & Processing



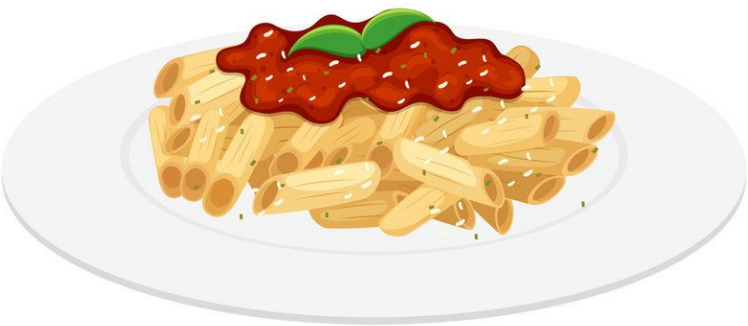
What is MapReduce?

You have 1 hour …

Forgot that you had invited your friends for dinner?

Forgot that you had invited your friends for dinner?

You decide to make `PASTA`

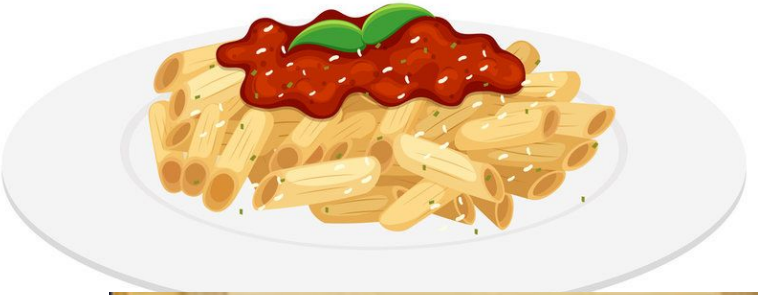


You call your spouse and your teenage kids to action in the kitchen.

Forgot that you had invited your friends for dinner?

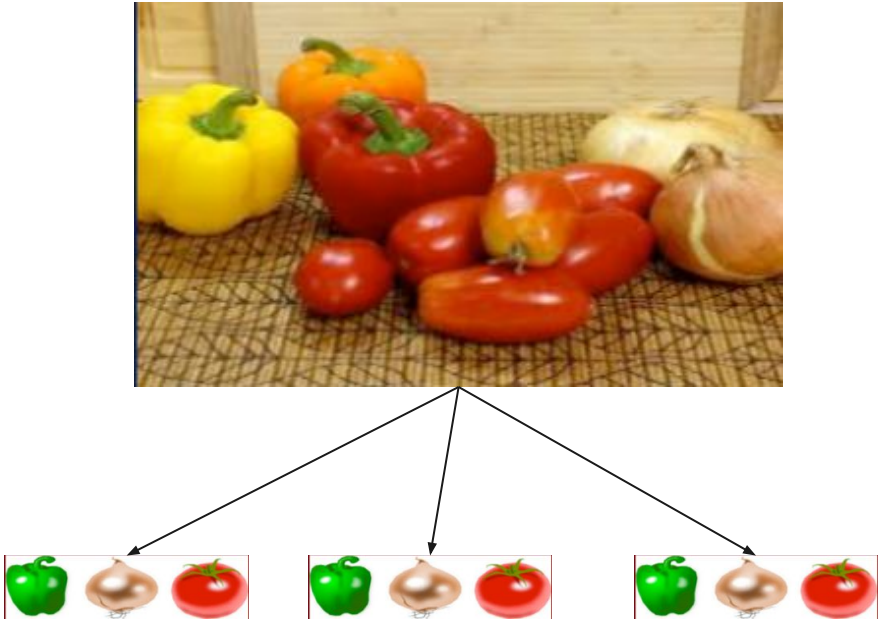
All mixed up!

You decide to make `PASTA`



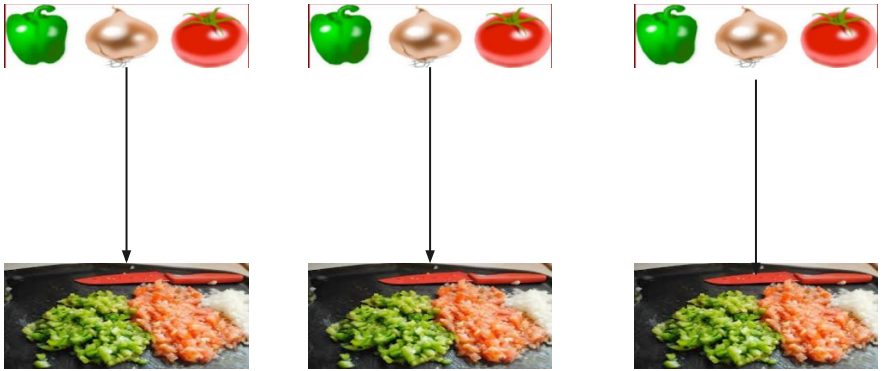
You cal
the kitc

Forgot that you had invited your friends for dinner?



Instead of sorting them first, you give everyone a randomly mixed batch.

Forgot that you had invited your friends for dinner?

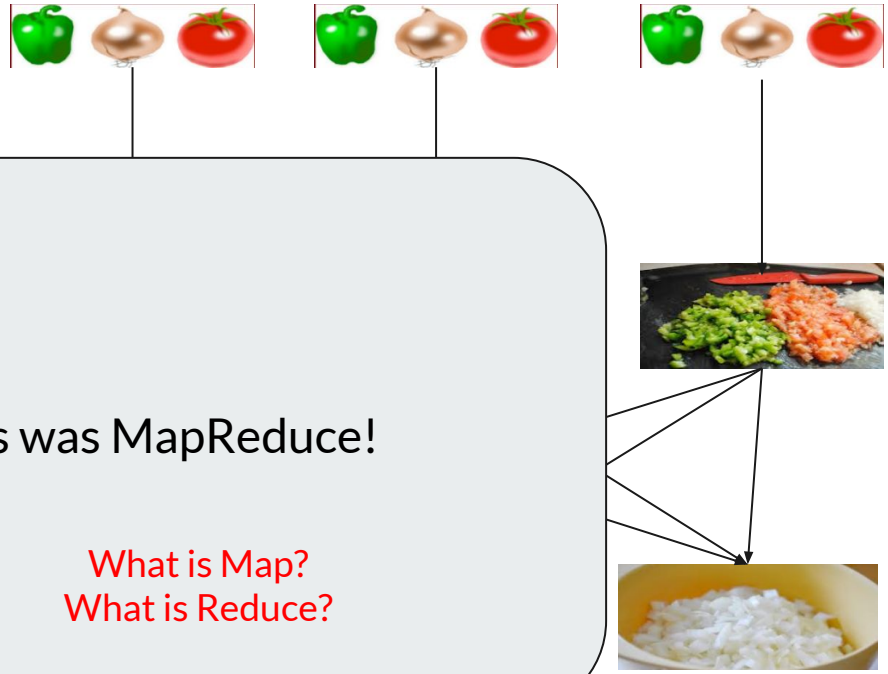


They need to ensure not mix different types of veggies.

Forgot that yo had invited yo friends for dinner?

Collect items of the same type.

Forgot that yo had invited yo friends for dinner?

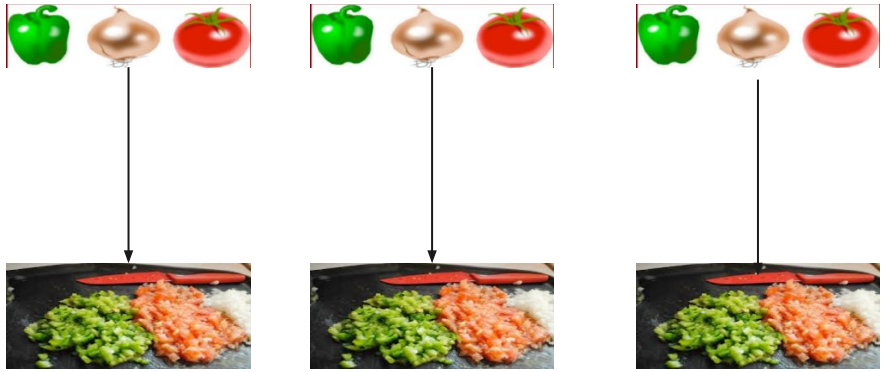


This was MapReduce!

What is Map?
What is Reduce?

Collect items of the same type.

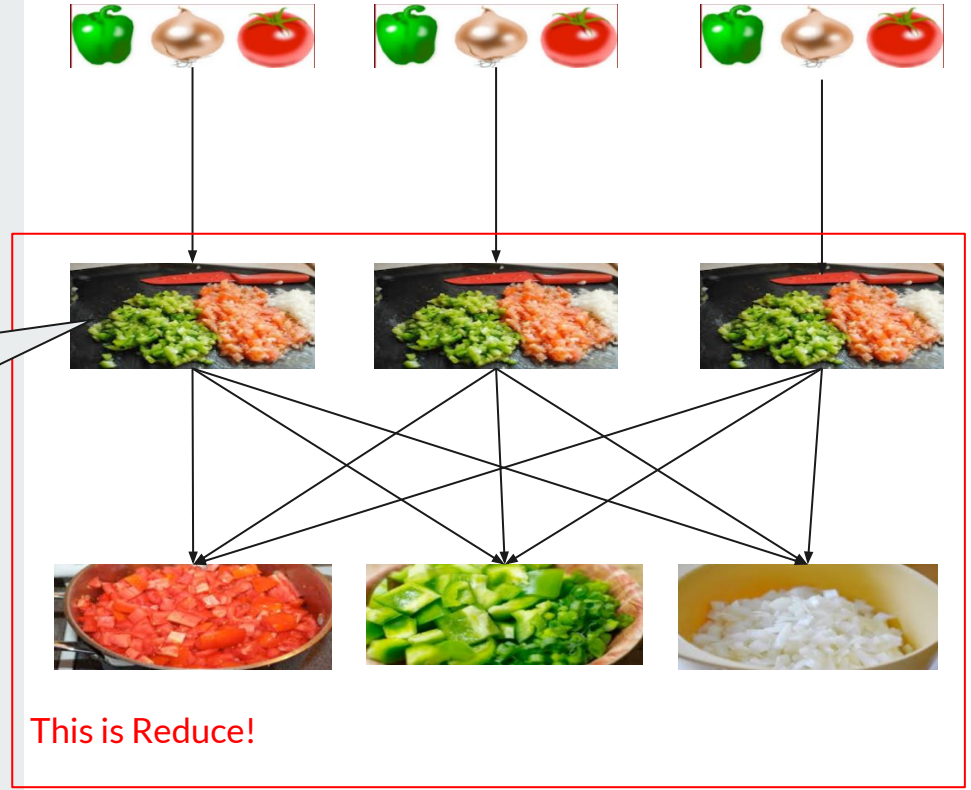
Forgot that you had invited your friends for dinner?



This is Map!

Forgot that you h
an

Remember - they need to ensure not mix different types of veggies?

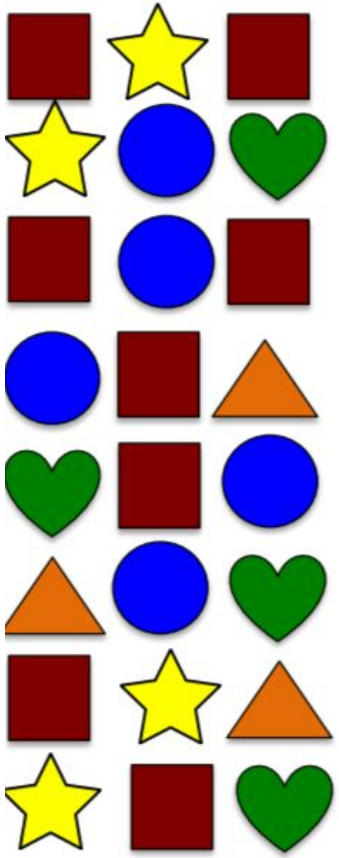


What is MapReduce?

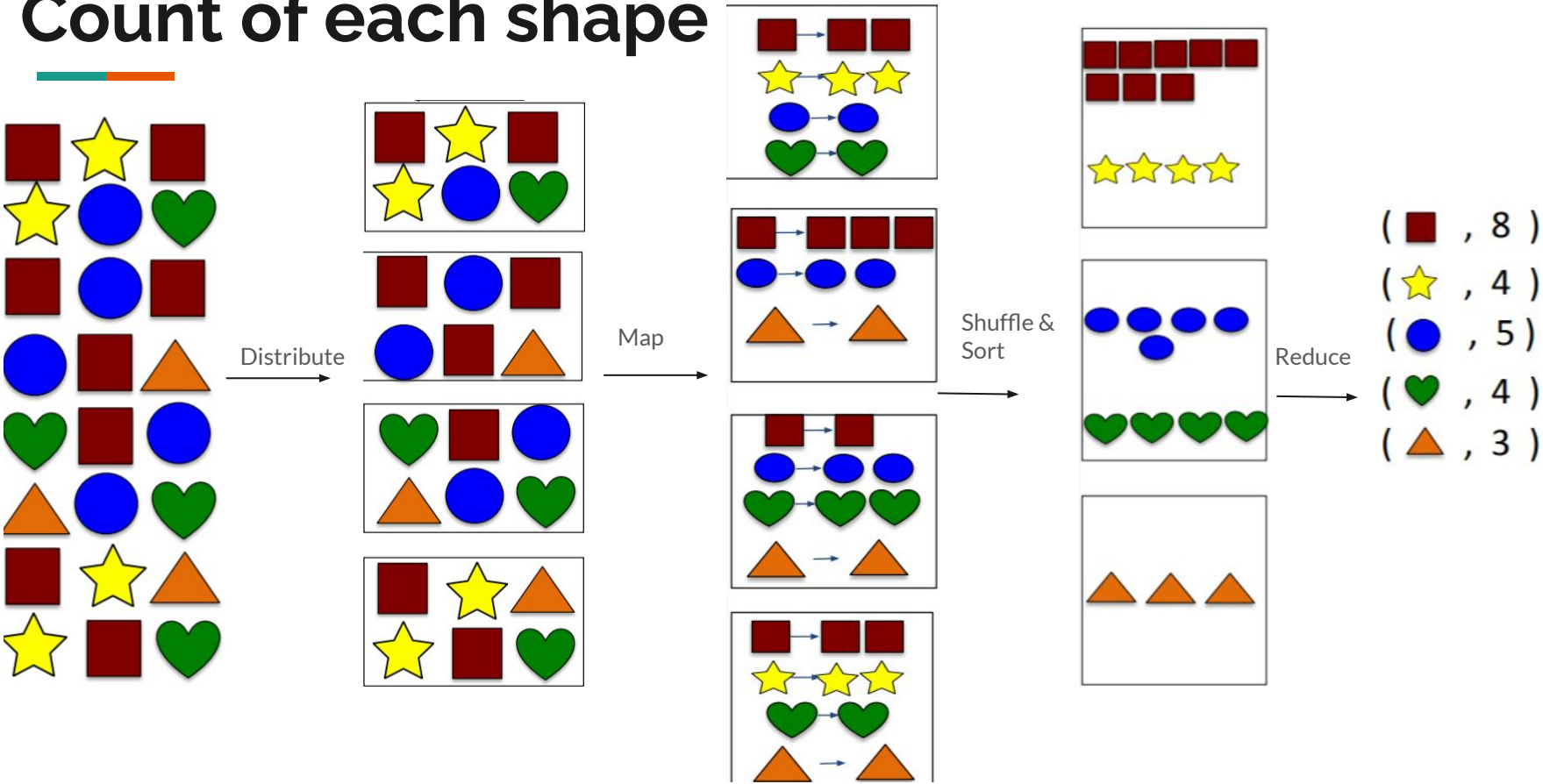
Map - Sort - Reduce

One more example ...

Give me count of each different shape



Count of each shape



Map Reduce
Formally ...

Taming Big Data Through Divide and Conquer

- Traditional tools can't handle the scale and complexity.
- **MapReduce as a Solution**: A programming model designed for processing vast amounts of data in parallel across many machines.
- **Core Idea**: Break down a big task into smaller ones, process them independently, then intelligently combine the results.

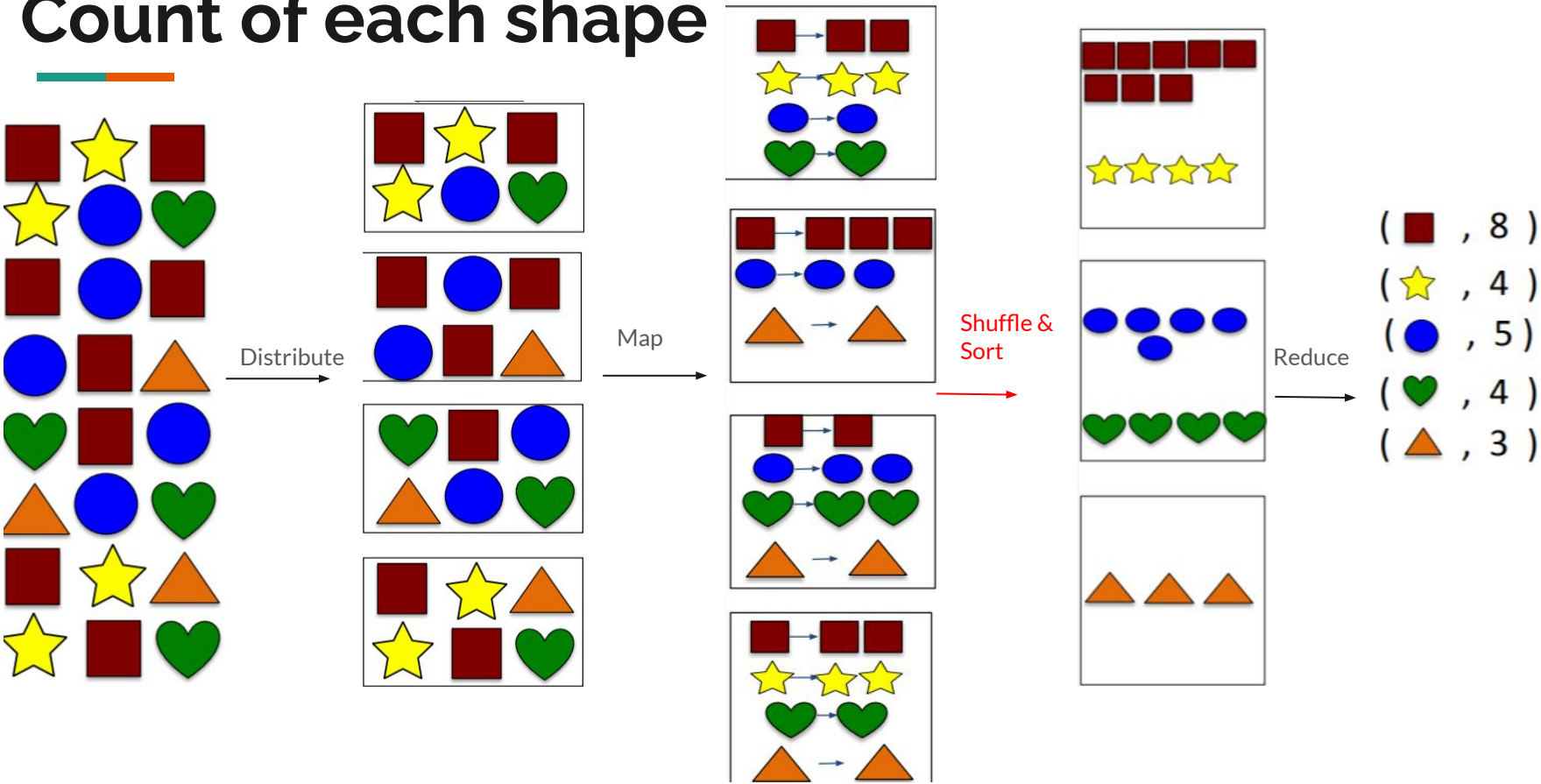
Map: The First Step in Transformation

- **The Mapper's Job:** Each Mapper takes a chunk of input data.
- **Transformation:** The Mapper processes data, potentially filtering, extracting, or calculating new values.
- **Output:** Mappers emit new, intermediate key/value pairs.

Shuffle & Sort: Making Sense of the Chaos

- **The Purpose:** Getting all the right data together for the Reduce phase.

Count of each shape



Shuffle & Sort: Making Sense of the Chaos

- **The Purpose:** Getting all the right data together for the Reduce phase.

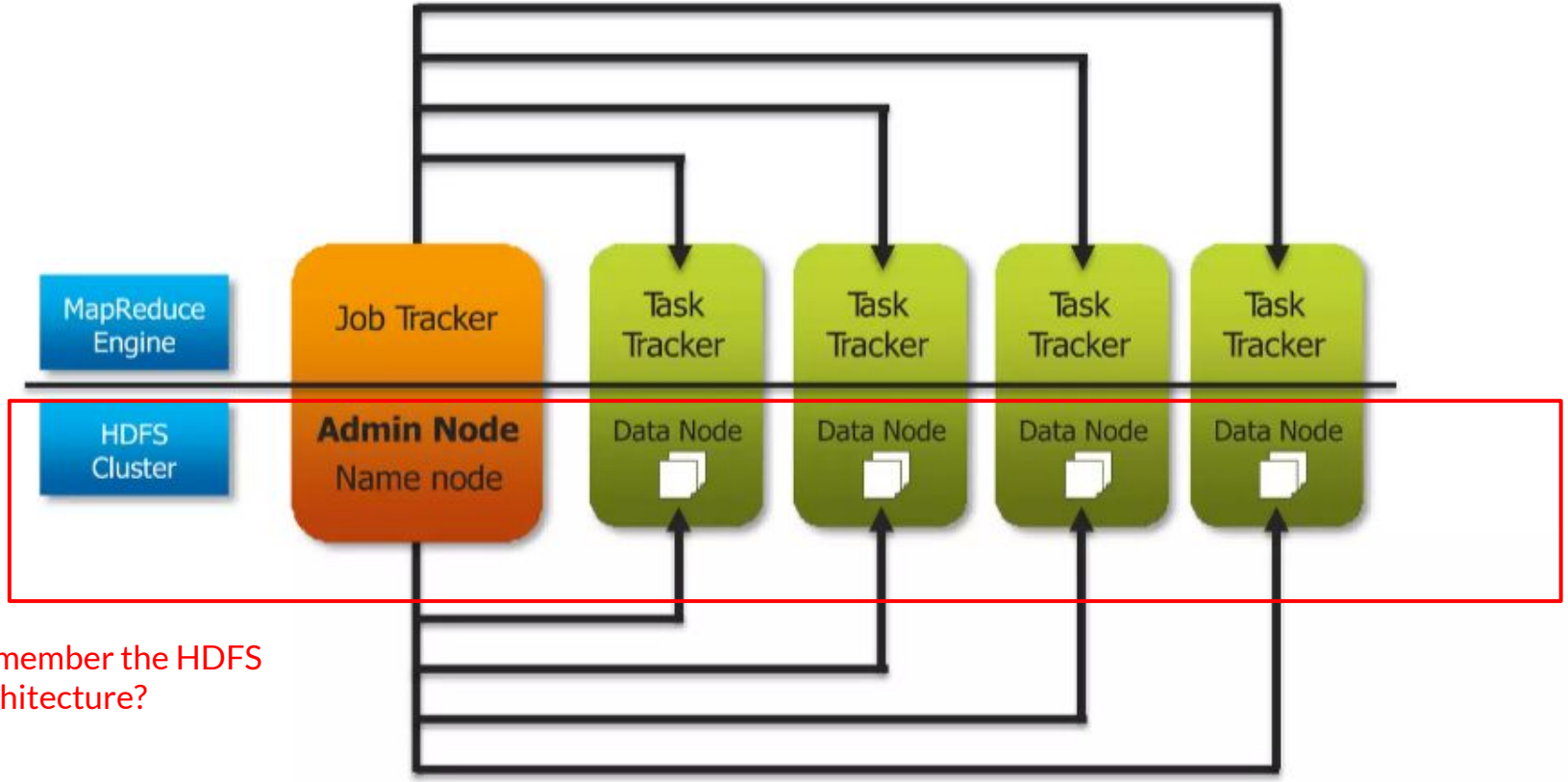
- **What It Does:**
  - **Shuffle:** Data from Mappers is distributed across nodes based on keys.
  - **Sort**: On each node, values with the same key are grouped together.

Reduce: Aggregation and Answers

- **Input:** Reducers receive the output of the "Shuffle & Sort".
- **The Reducer's Job:** Processing those values to produce the final answer related to that key.
- **Types of Operations:** Calculations (sum, average), joining data, filtering/selection.
- **Output:** The result of the Reduce phase is usually the final answer to the original analysis question.

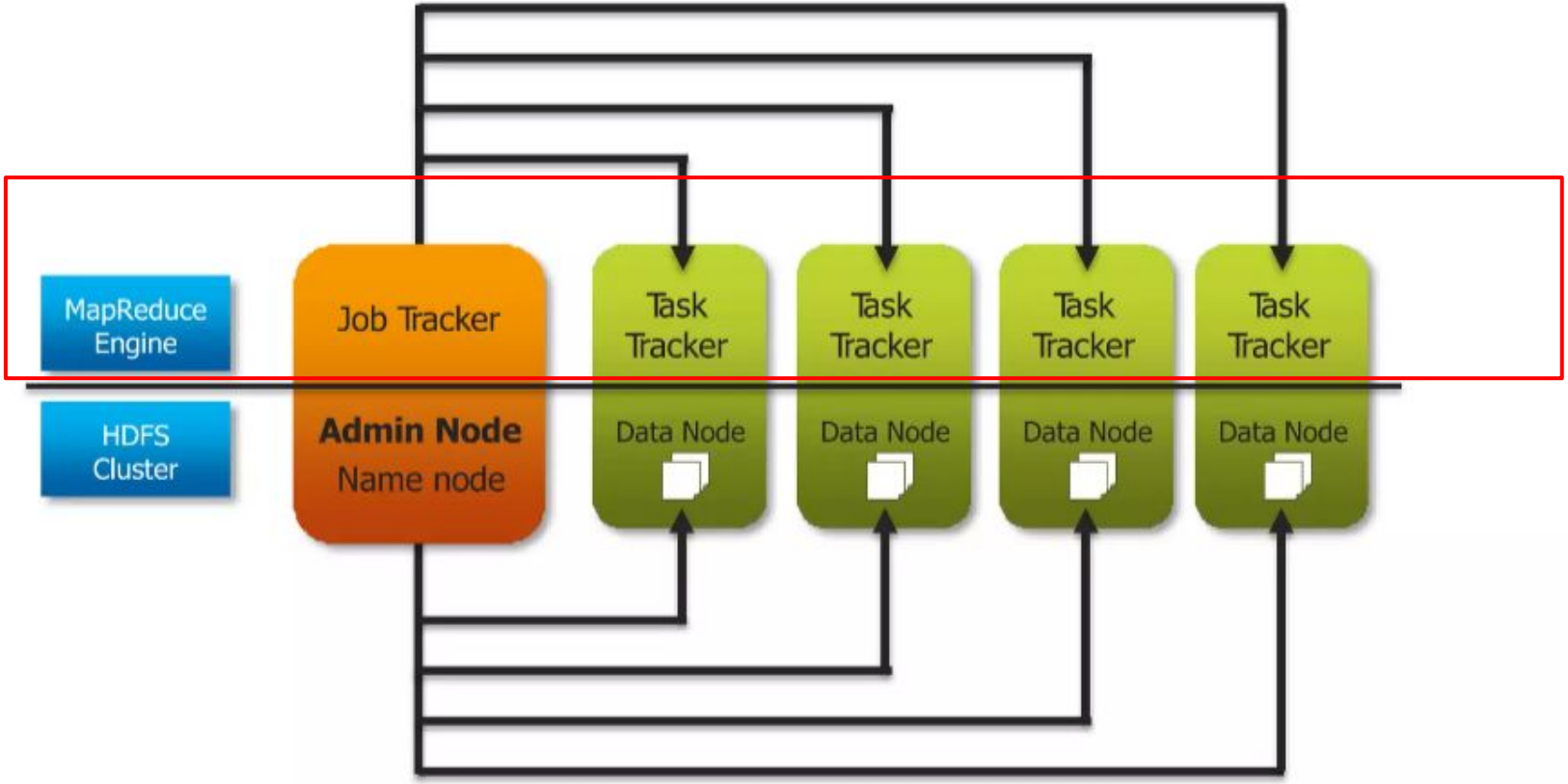
MapReduce in Hadoop

Let's take MapReduce closer to HDFS …

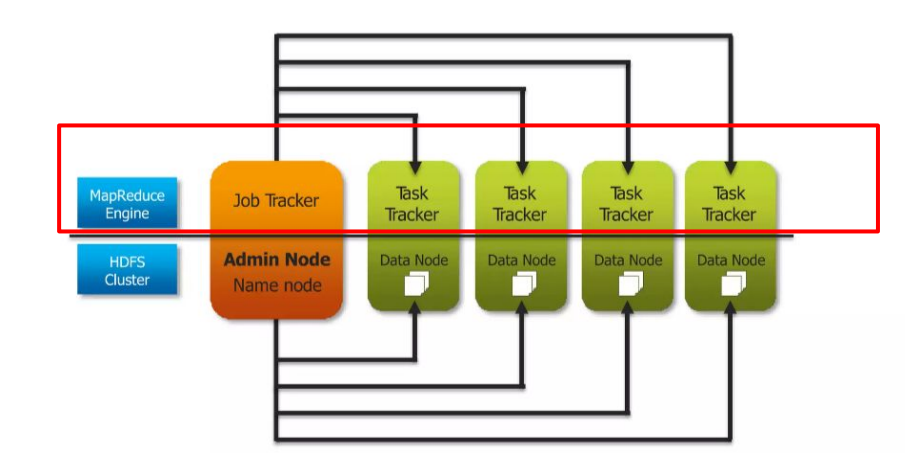


Remember the HDFS architecture?

Let’s take MapReduce closer to HDFS …



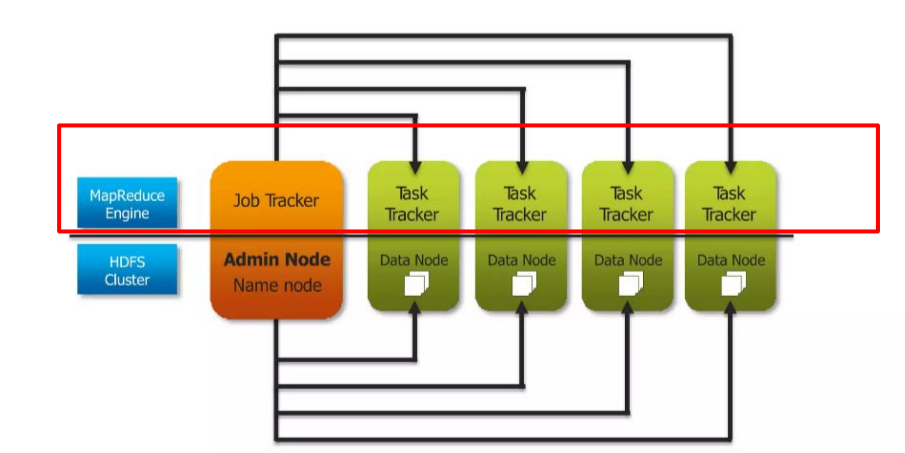
JobTracker



JobTracker: The Maestro of MapReduce

- **Master Node:** A single JobTracker manages the whole Hadoop cluster.
- **Overseer**: Monitors TaskTrackers, re-assigns failed tasks, and provides job status updates to users.
- **Resourcefulness**: Tries to send tasks to nodes where the data they need is already stored.

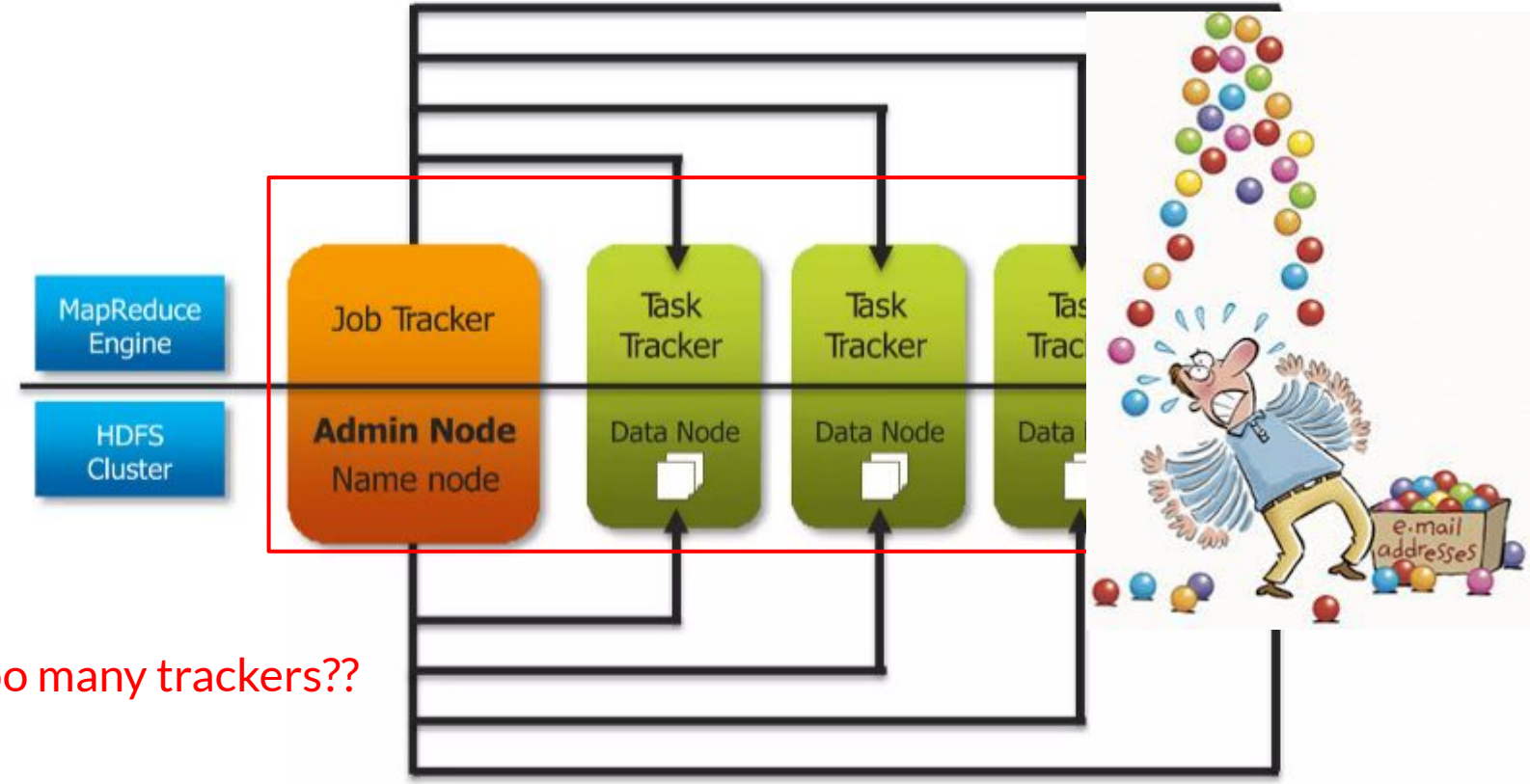
TaskTracker



TaskTracker: The Workhorses of the Cluster

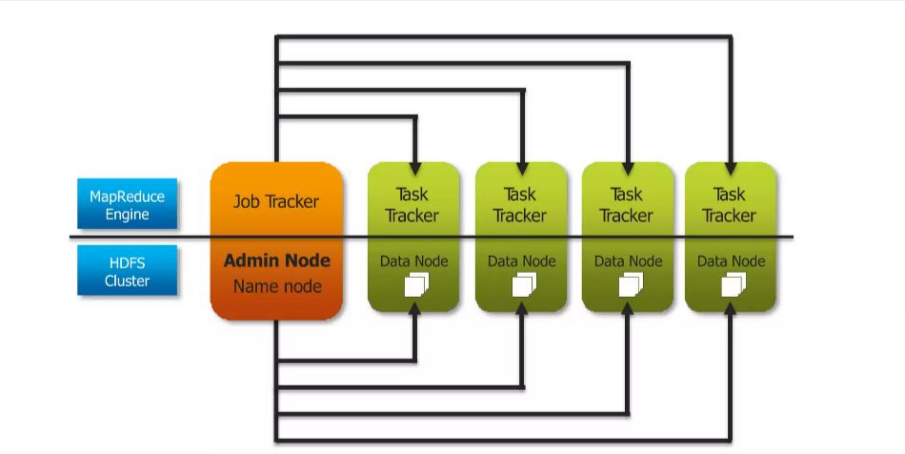
- **Where the Work Happens**
- **Location:** Resides on each DataNode in the cluster.
- **Task Management:** Receives instructions from the JobTracker, launches tasks, and monitors their progress

This works … But …



Too many trackers??

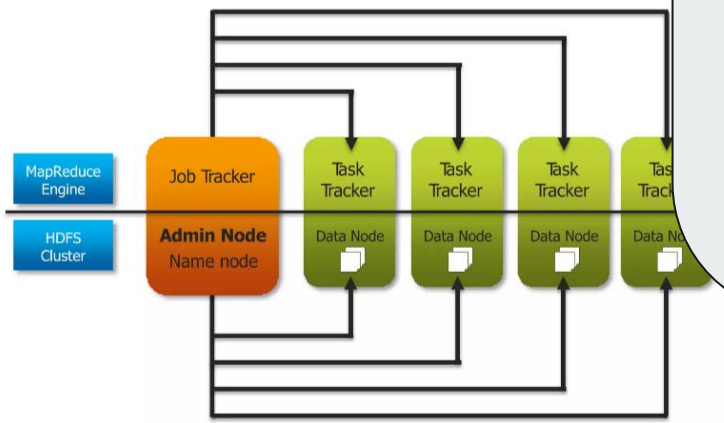
The Challenge of Many Trackers



The Challenge of Many Trackers

- We've got Trackers Everywhere: DataNodes, TaskTrackers, the JobTracker... each has a monitoring role.
- **Coordination Overhead:** Keeping all of them in sync adds complexity to the Hadoop system.
- **Single Point of Failure:** The JobTracker, in particular, is a worry if it goes down.

The Challenge
Many Trackers



This is where YARN enters the picture!

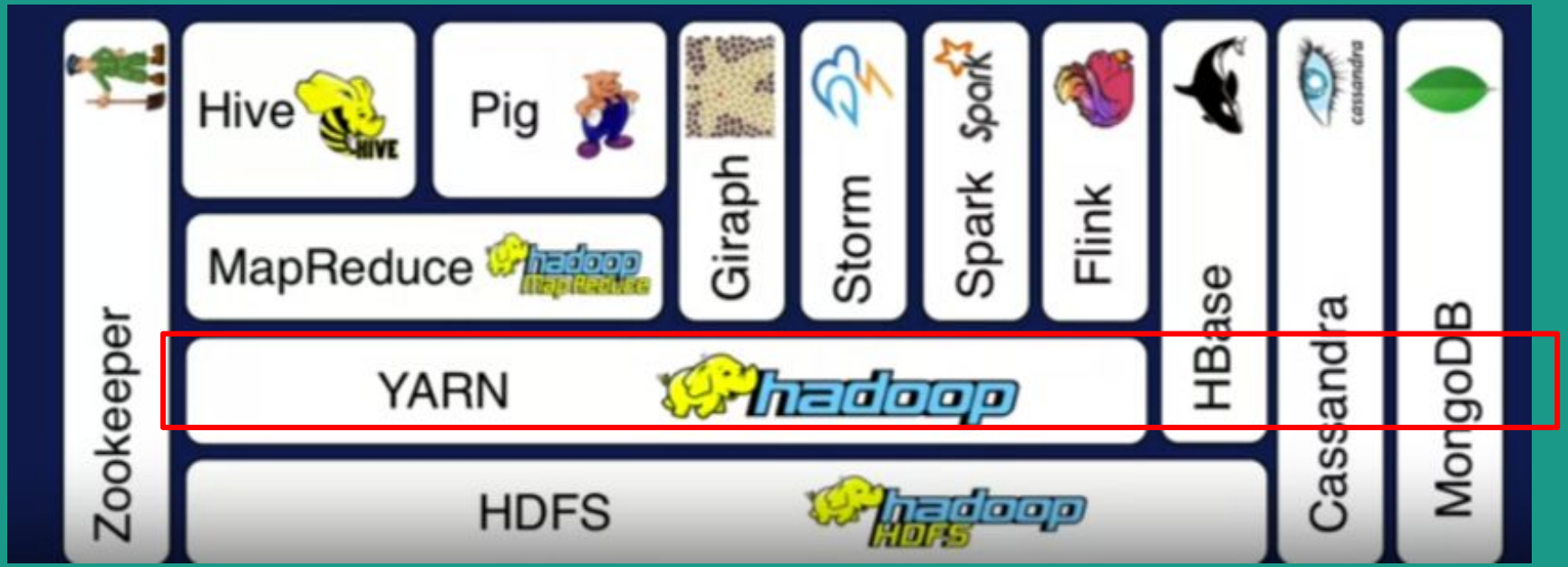
ny Trackers

here:
he
nitoring role.

eeping all of
ty to the

a worry if it

YARN - Yet Another Resource Negotiator



YARN replaces the need for JobTracker & TaskTrackers with a more manageable structure.

Back to Map Reduce …

How to run a custom MapReduce program in Hadoop?

```
hadoop jar hadoop-streaming-2.7.3.jar \

-input <input_file> \

-output <output_location> \

-mapper mapper.py \

-reducer reducer.py
```

How to run a custom MapReduce program in Hadoop?

```
hadoop jar hadoop-streaming-2.7.3.jar \
-input <input_file> \
-output <output_location> \
-mapper mapper.py \
-reducer reducer.py
```

```
[20] !/usr/local/hadoop-3.4.0/bin/hadoop jar /usr/local/hadoop-3.4.0/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.4.0.jar grep ~/input ~/grep_example 'allowed*'

2024-04-09 08:36:40,889 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2024-04-09 08:36:41,262 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2024-04-09 08:36:41,262 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2024-04-09 08:36:41,799 INFO input.FileInputFormat: Total input files to process : 10
```

Limitations of MapReduce

Limitations of MapReduce

Where MapReduce Isn't the Best Tool

- **Iteration is Awkward**: Algorithms that need multiple stages where the output of one MapReduce job feeds into the next. This can get clunky.
- **Unsuitable for interactive applications**: Where the results must be presented to the user very quickly, expecting a return from the user.
- **Not All Problems Fit:** Data isn't easily splittable into Key/Value pairs, or the analysis is inherently graph-like.

Quick Quiz

Which of these is a defining characteristic of Big Data?

(A) It's always stored in the cloud.
(B) It fits easily into traditional databases.
(C) It's too large or complex for traditional tools.
(D) It's only about text data.

HDFS is designed to:

(A) Process data at lightning speed.
(B) Store data reliably across many machines.
(C) Automatically analyze data for insights
(D) Handle real-time data streams.

The NameNode in HDFS does what?

(A) Stores copies of actual data blocks.
(B) Executes MapReduce tasks.
(C) Keeps track of file locations and metadata.
(D) Translates user queries into HDFS operations.

What is the core purpose of the "Map" phase?

(A) Producing the final output.
(B) Distributing data to different nodes.
(C) Transforming input data in preparation for analysis.
(D) Sorting data into a specific order.

Which term best describes Key/Value pairs?

(A) Programming languages used in Hadoop.
(B) Components of the HDFS architecture.
(C) Data structures used in MapReduce.
(D) Types of analysis Hadoop is used for.

What is a Reducer's primary function?

(A) Splitting input data into chunks.
(B) Aggregating data associated with a single key.
(C) Choosing the optimal nodes for Map tasks.
(D) Communicating with the NameNode.