The Building Blocks of Hadoop - HDFS, MapReduce, and YARN

INTRODUCING HADOOP

Overview

Understand the need for Distributed Computing

Understand the role of Hadoop in a distributed computing setup

Get introduced to core technologies which work on Hadoop

How Much Data Do These Organizations Deal With?

Facebook

NSA

Google

Facebook

Current storage = 300 petabytes

Processed per day = 600 terabytes

Users per month = 1 billion

Likes per day = 2.7 billion

Photos uploaded per day = 300 million

NSA

Current storage = ~5 exabytes

Processed per day = 30 petabytes

NSA touches 1.6% of internet traffic per day

Web searches, websites visited, phone calls, credit/debit card transactions, financial and health information

Google

Current storage = 15 exabytes

Processed per day = 100 petabytes

Number of pages indexed = 60 trillion

Unique search users per month > 1 billion

Searches per second = 2.3 million

Huge Data Set

Facebook

NSA

Google

Huge Data Set

Huge Data Set

Cannot meet big data requirements

Big Data System Requirements

Raw data

Store massive amounts of data

Big Data System Requirements

Store

Extract useful information

Store massive amounts of data

Process it in a timely manner

Big Data System Requirements

Store

Store massive amounts of data

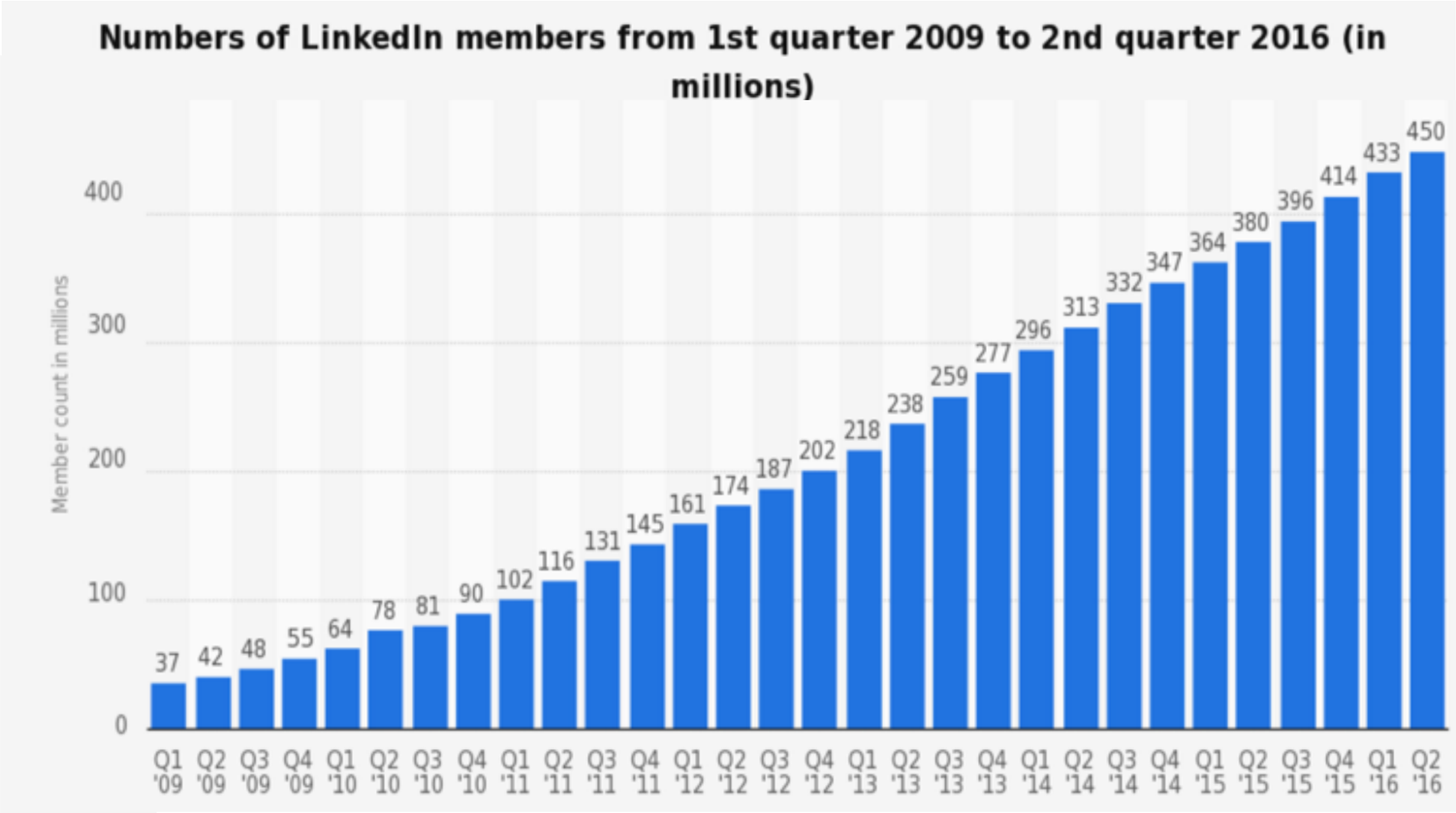
Process

Process it in a timely manner

?

Store Process ?

Growing size of data



Store Process ?

The infrastructure needs to keep up with the growing size of data

Big Data System Requirements

Store

Store massive amounts of data

Process

Process it in a timely manner

Accommodate changing needs

Scale easily as data grows

Big Data System Requirements

Store

Store massive amounts of data

Process

Process it in a timely manner

Scale

Scale easily as data grows

Big Data System Requirements

Traditional data technologies don't cut it anymore

Store **Process** **Scale**

Distributed Computing Frameworks
like Hadoop were developed for
exactly this

Two Ways to Build a System

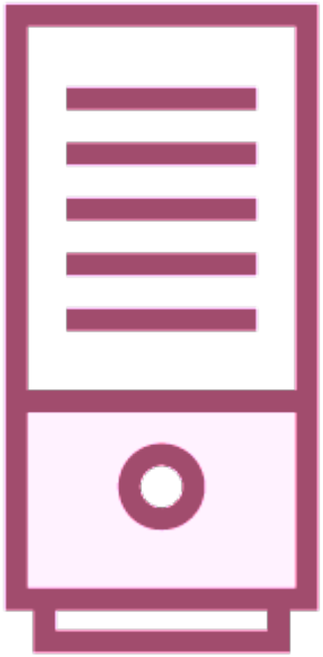
Monolithic

Distributed

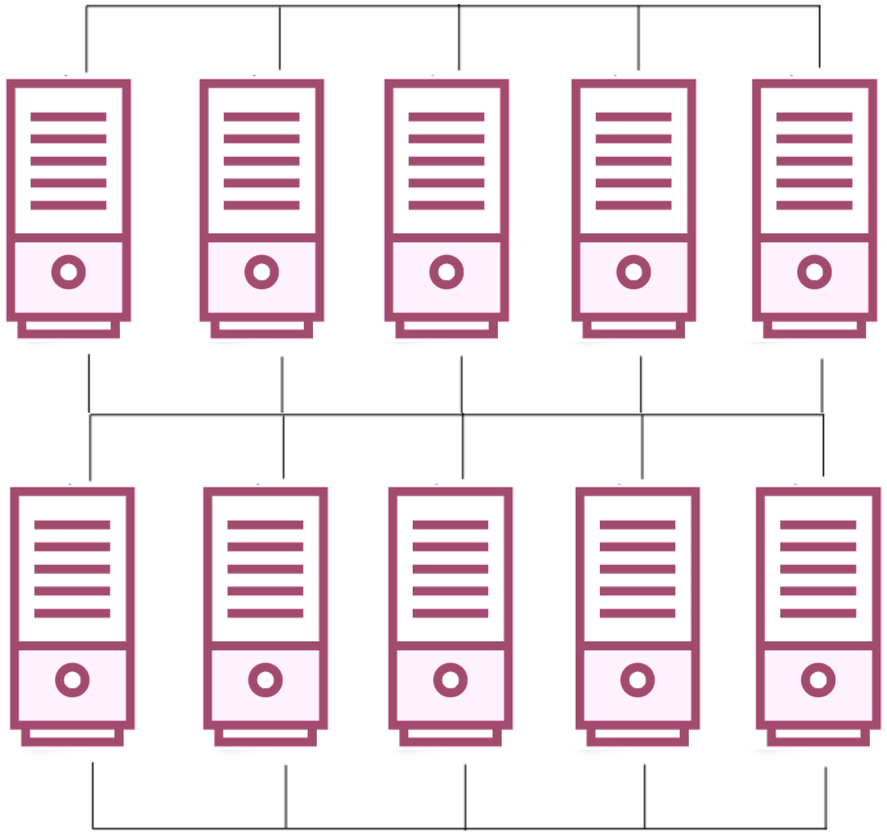
Two Ways to Build a Team

A star who dribbles and shoots

A team of good players who know how to pass



A supercomputer



A cluster of decent machines
that know how to parallelize

Two Ways to Build a System

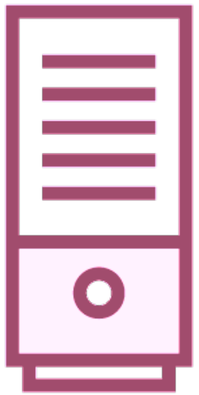
Monolithic

Distributed

Monolithic

One star player

Monolithic



A single powerful server

2x Expense

< 2x Performance

Two Ways to Build a System

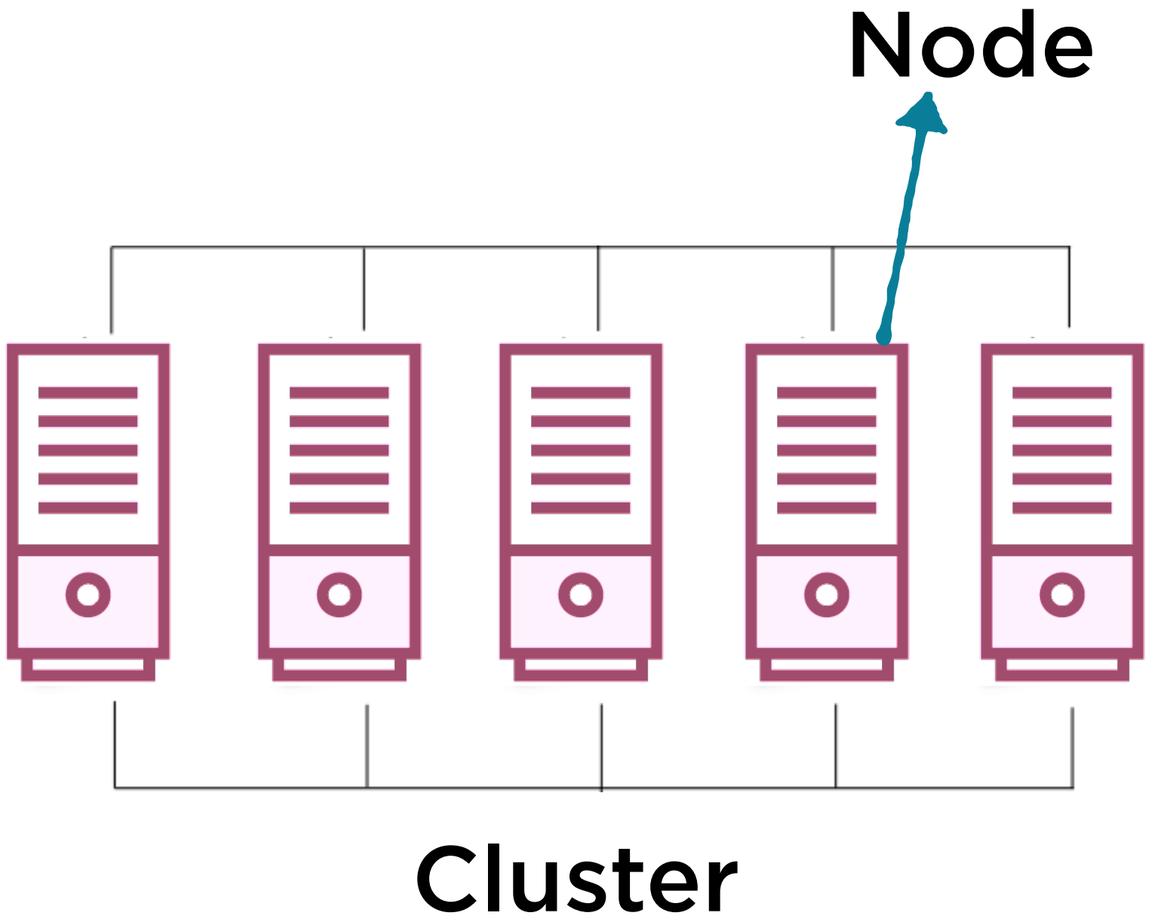
Monolithic

Distributed

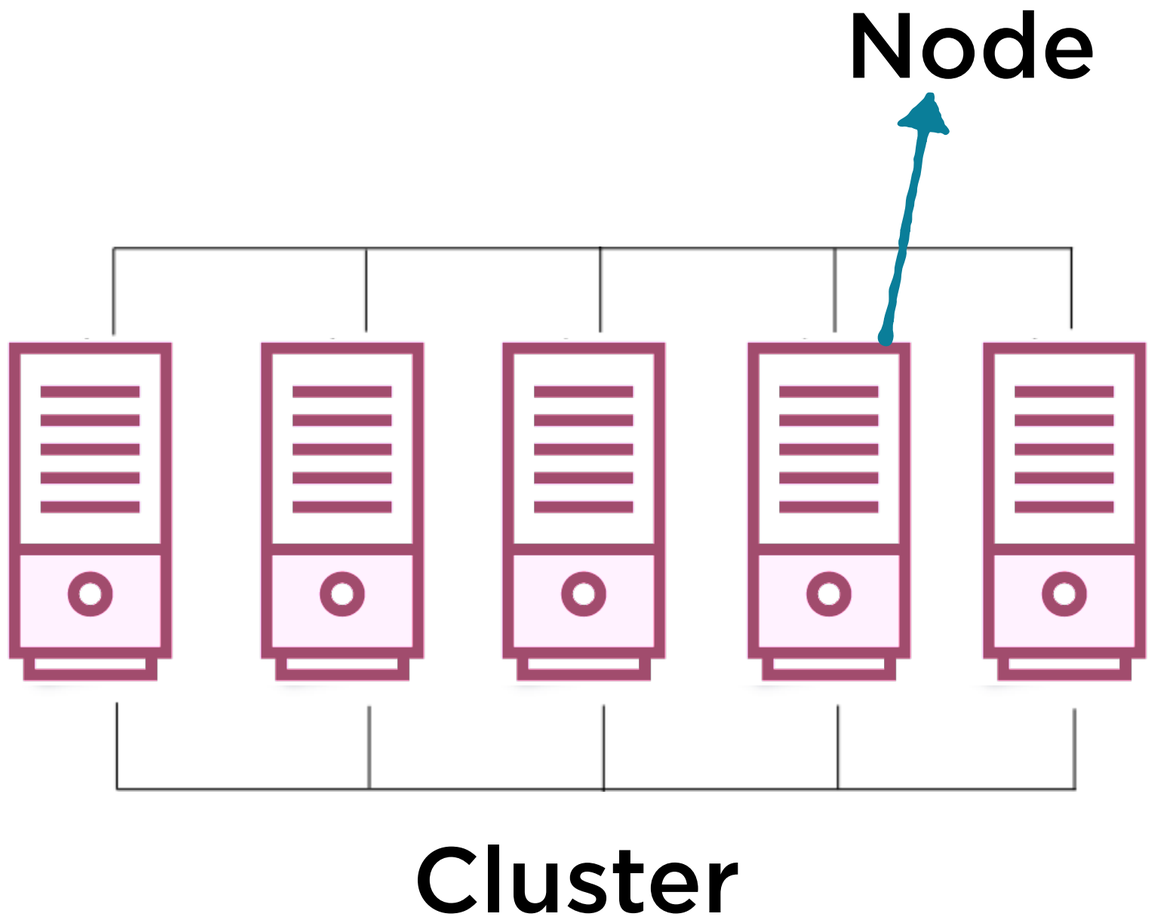
Distributed

A team of good players who know how to pass

Distributed



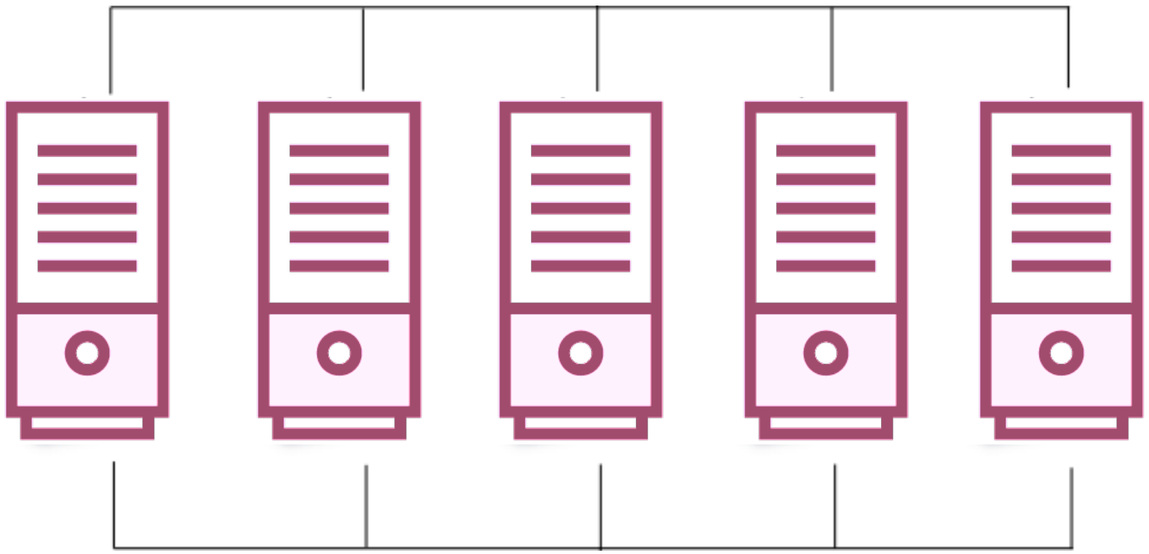
Distributed System



Many small and cheap computers come together...

...to act as a single entity

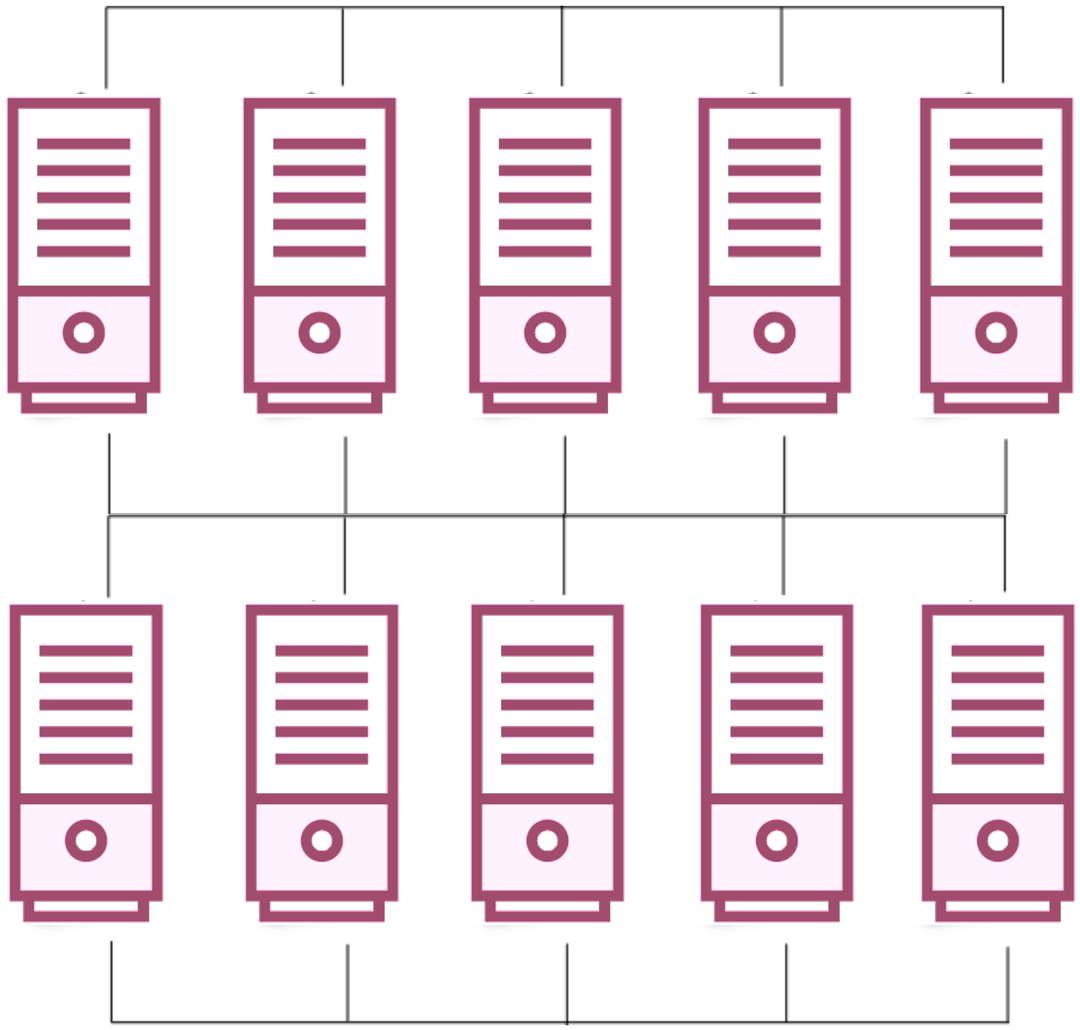
Distributed System



Cluster

Such a system can scale linearly

Distributed System



2x Nodes

2x Storage

~ 2x Speed

Server Farms

Companies like Facebook, Google, Amazon are building vast server farms

Server Farms

These farms have 100s of 1000s of servers working in tandem to process complex data

Server Farms

All of these servers need to be co-ordinated by a single piece of software

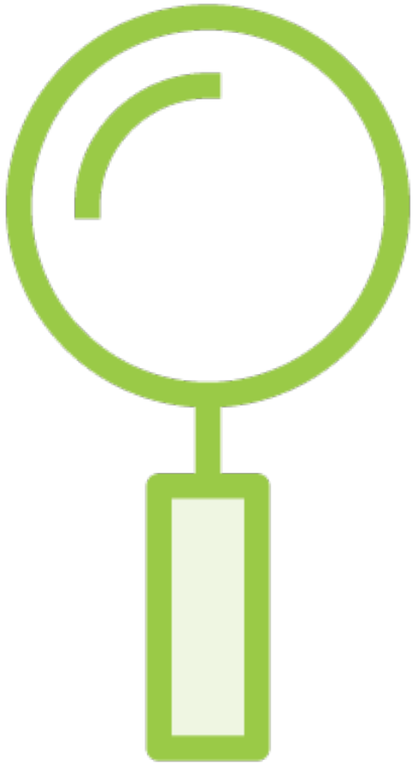
Single Co-ordinating Software

- **Partition data**
- **Co-ordinate computing tasks**
- **Handle fault tolerance and recovery**
- **Allocate capacity to processes**

Store **Process** **Scale**

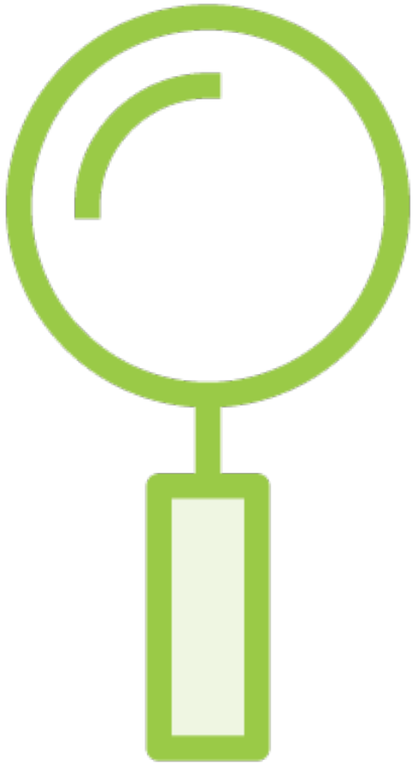
Distributed Computing makes for a lot of complexity

Single Co-ordinating Software



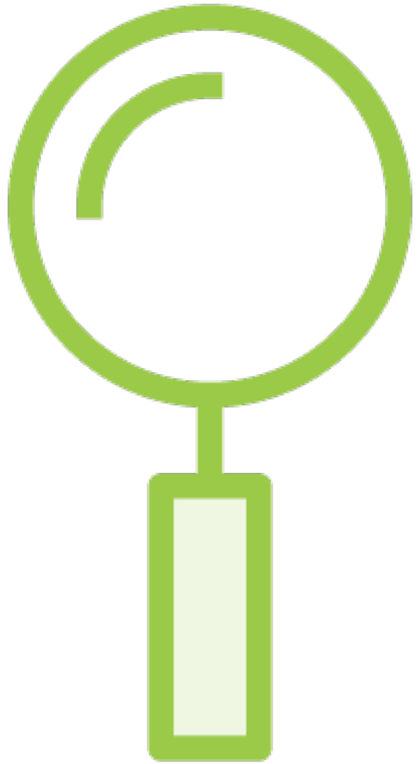
Back in the early 2000s Google realized that web search requires something completely new

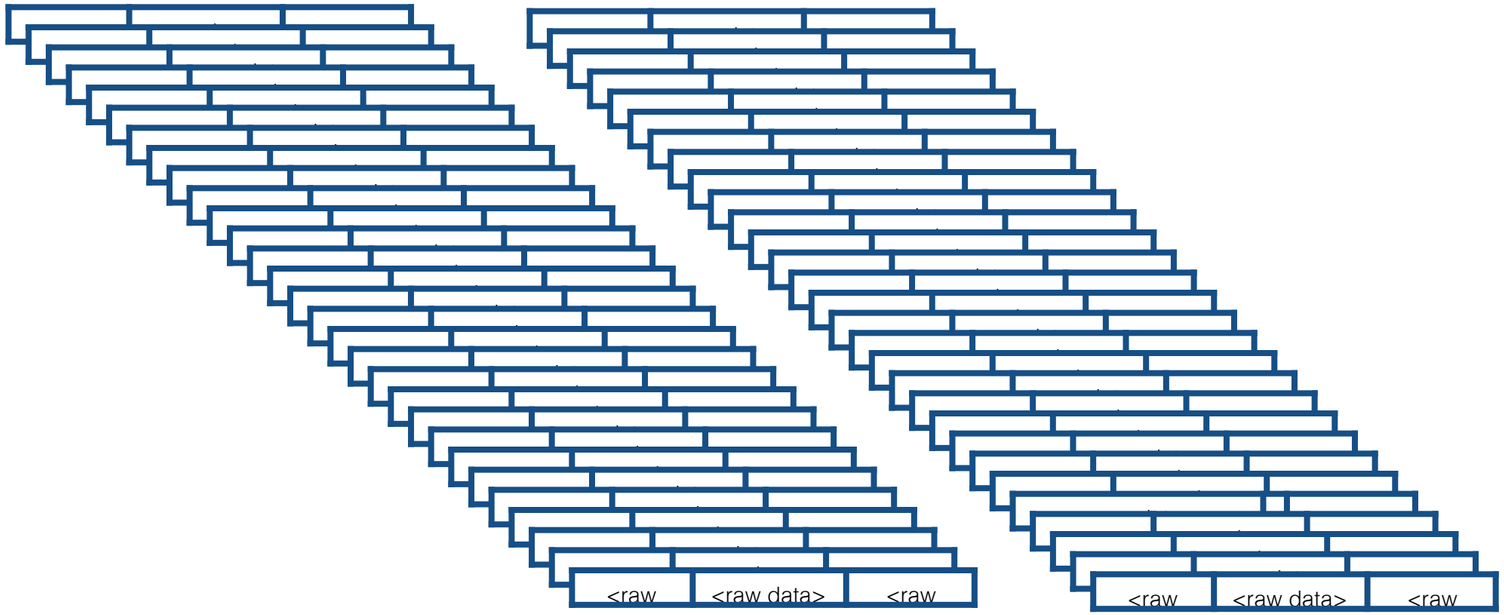
Single Co-ordinating Software



Google developed proprietary software to run on these distributed systems

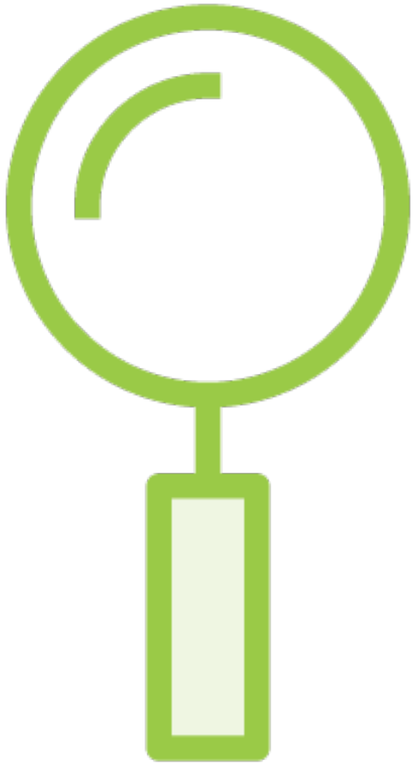
Single Co-ordinating Software

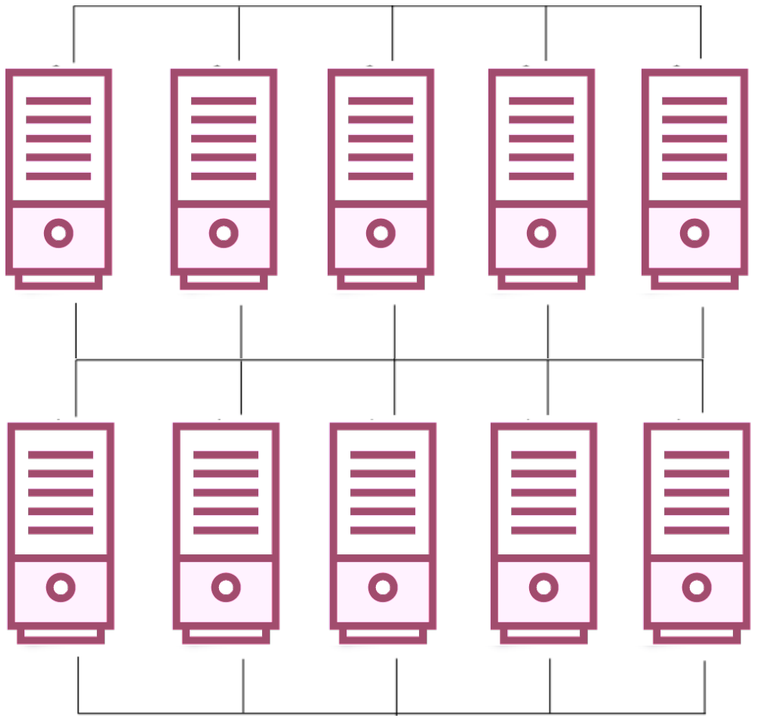




First: store millions of records on multiple machines

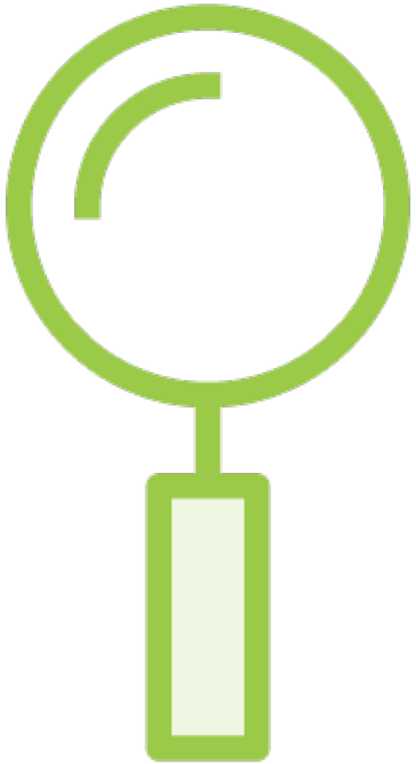
Single Co-ordinating Software





Second: run processes on all these machines to crunch data

Single Co-ordinating Software



Google File System

To solve distributed storage

MapReduce

To solve distributed computing

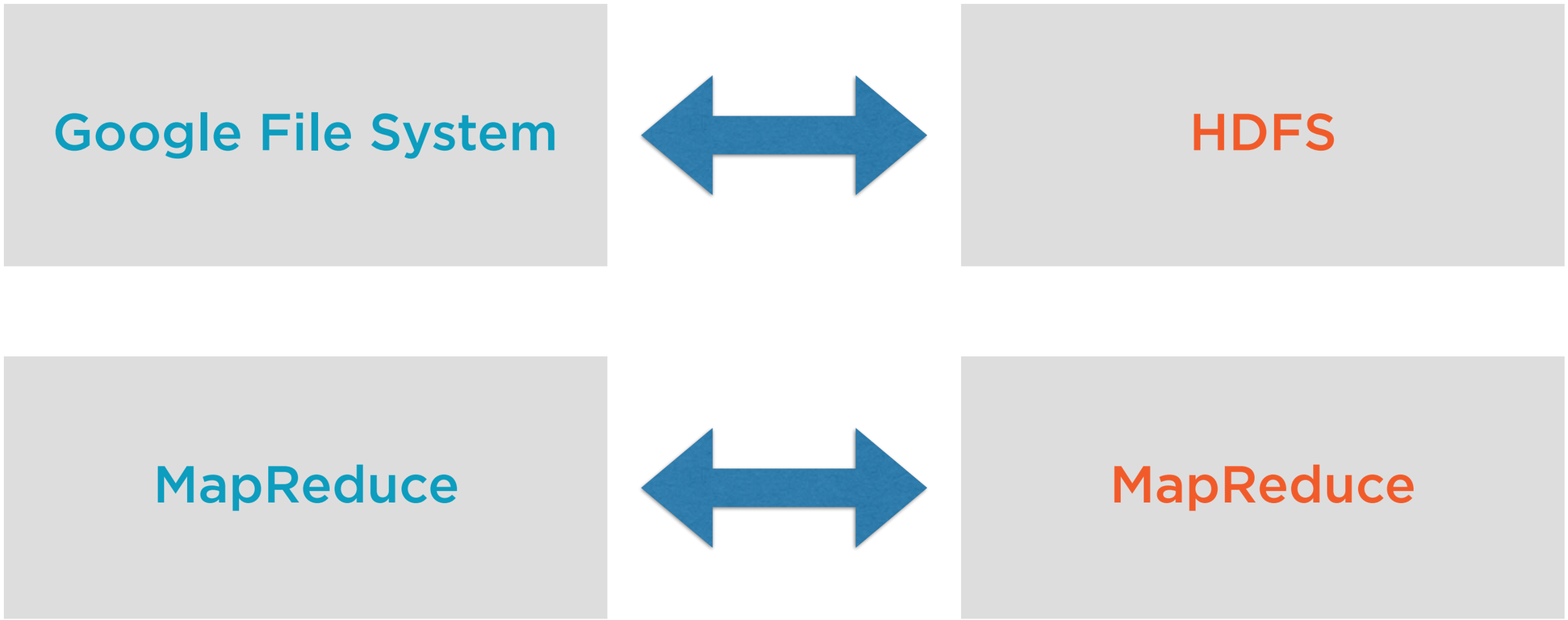
Single Co-ordinating Software

Google File System

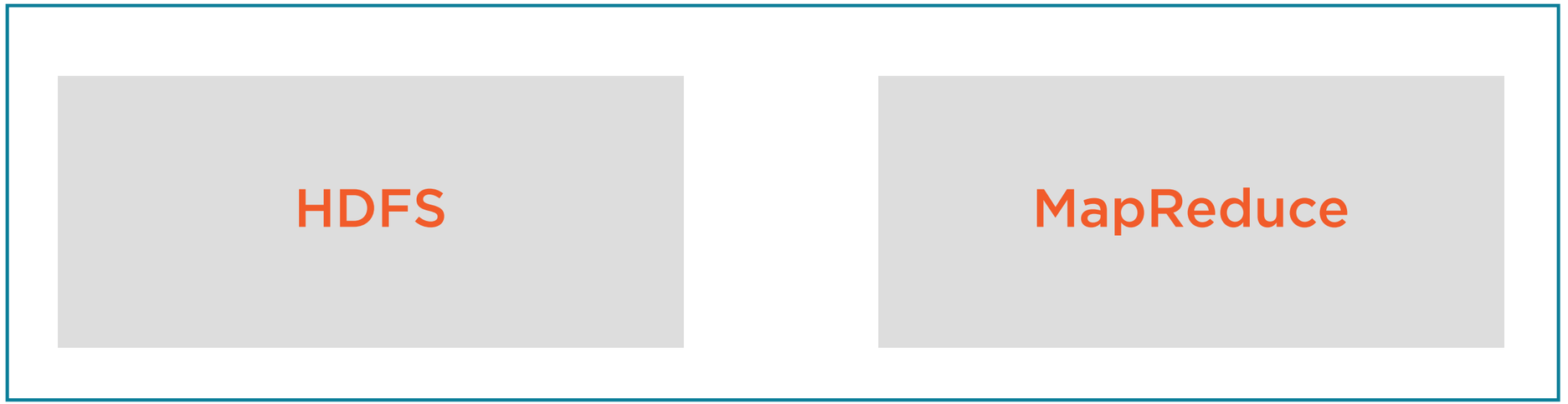
MapReduce

Apache developed open source versions of these technologies

Single Co-ordinating Software



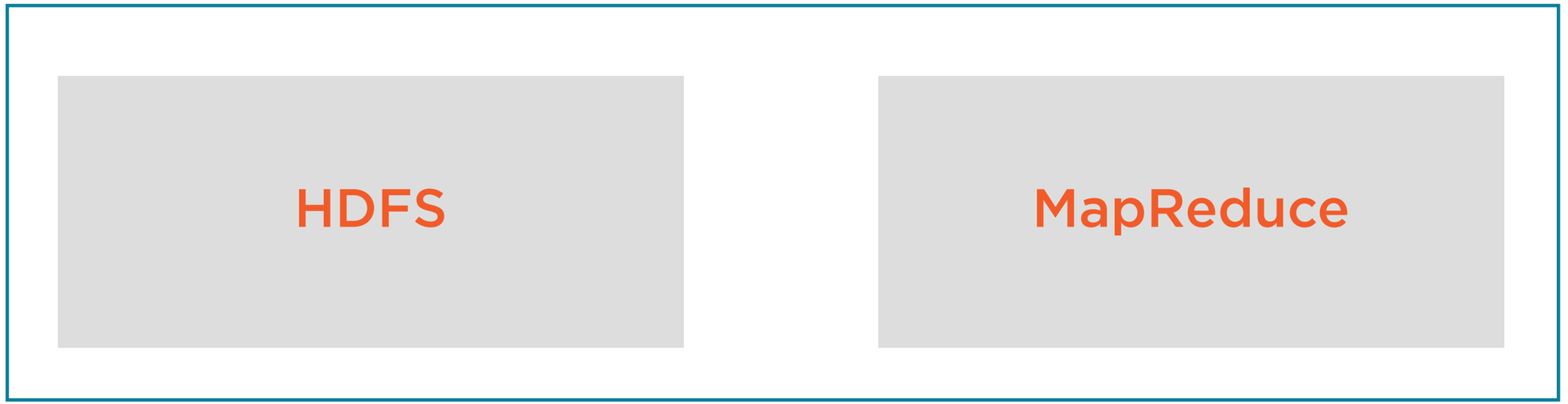
Hadoop



A file system to manage the storage of data

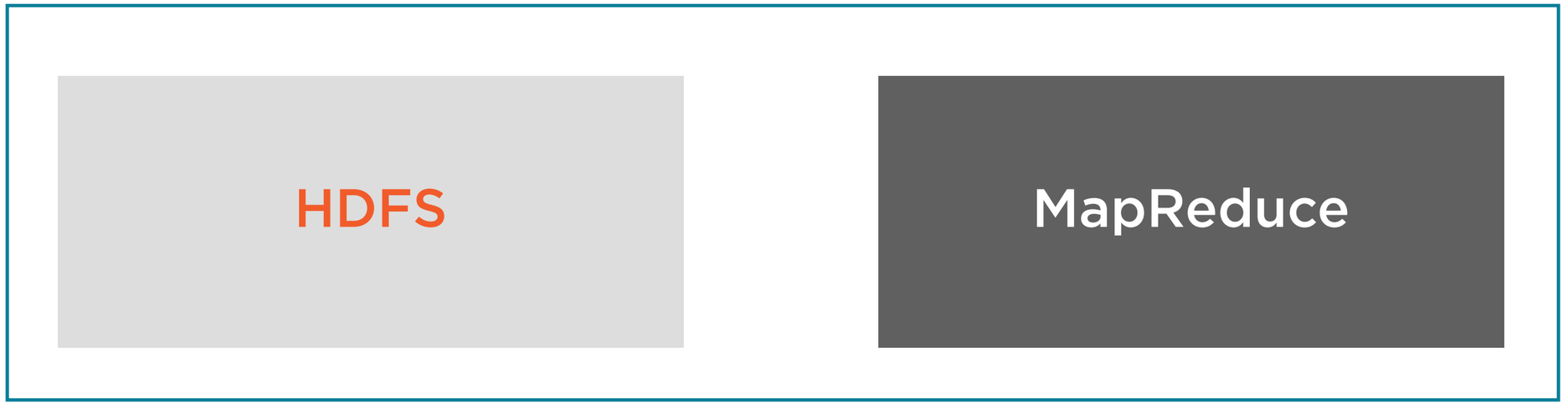
A framework to process data across multiple servers

Hadoop



In 2013, Apache released Hadoop 2.0

Hadoop



MapReduce was broken into two separate parts

Hadoop

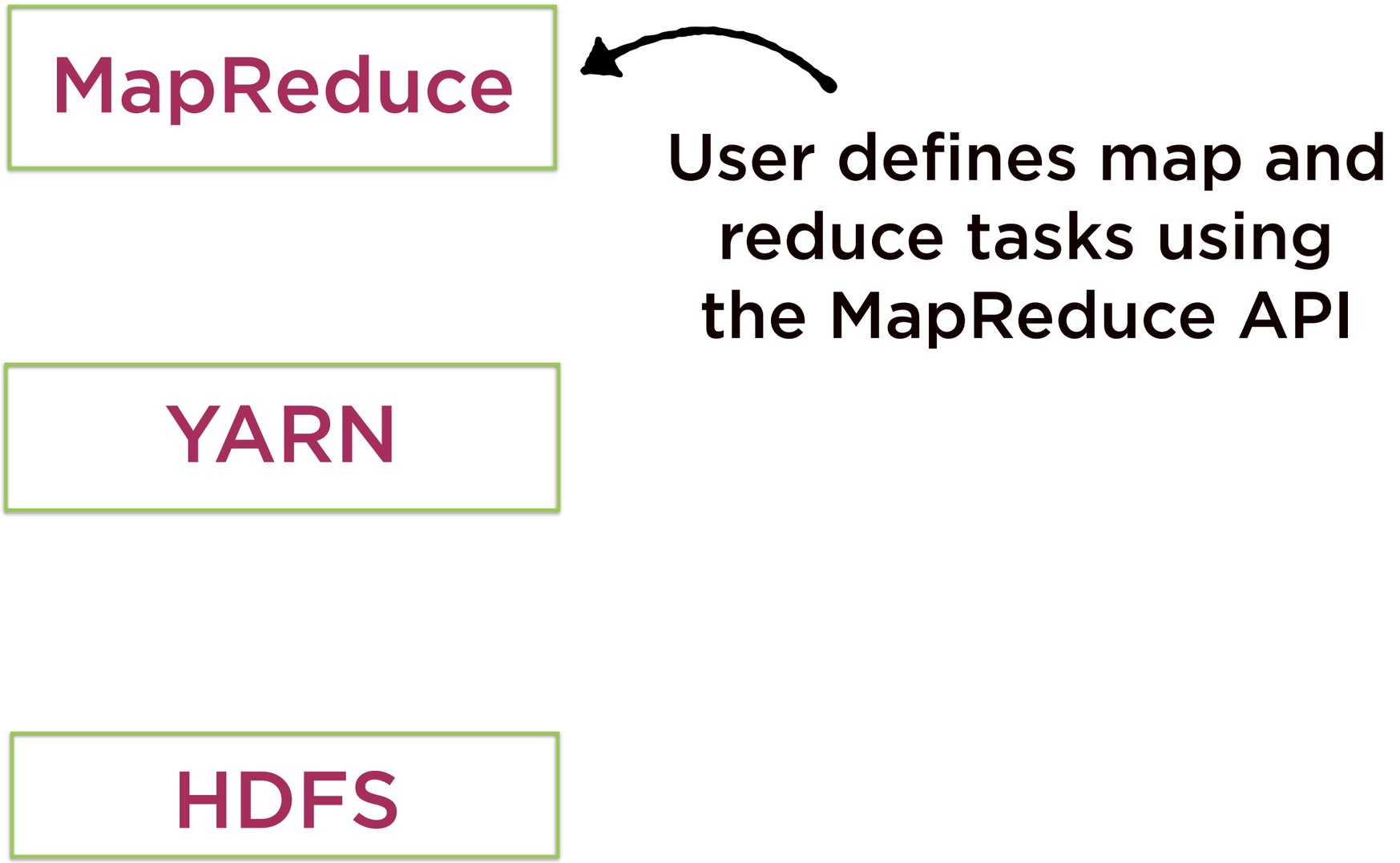
A framework to define a data processing task

A framework to run the data processing task

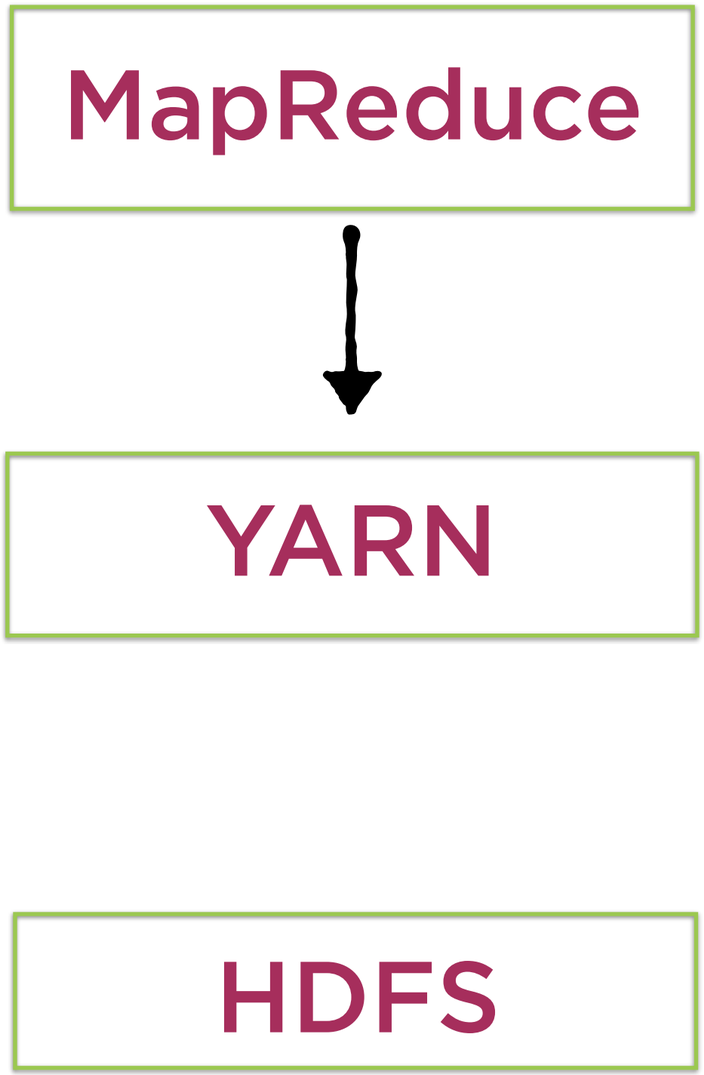
Hadoop

Each of these components have corresponding configuration files

Co-ordination Between Hadoop Blocks

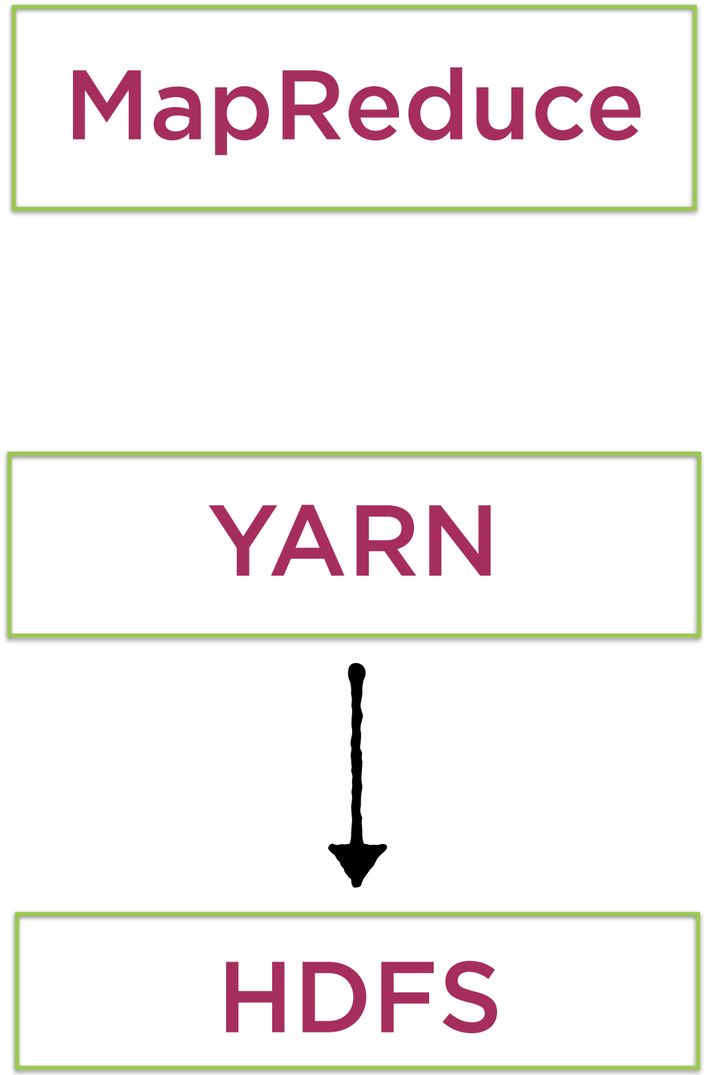


Co-ordination Between Hadoop Blocks



A job is triggered on the cluster

Co-ordination Between Hadoop Blocks

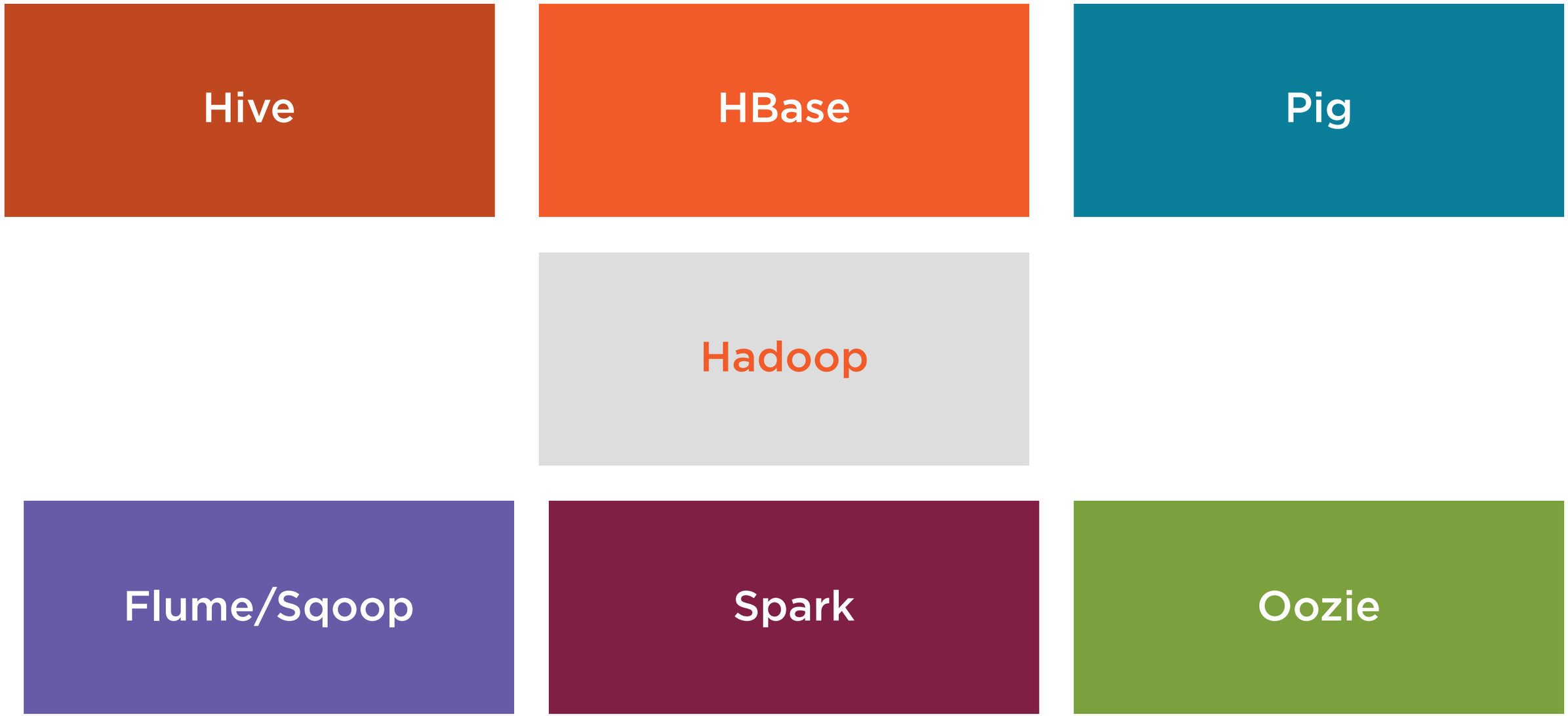


YARN figures out where and how to run the job, and stores the result in HDFS

Hadoop Ecosystem

An ecosystem of tools have sprung up around this core piece of software

Hadoop Ecosystem



Hadoop Ecosystem

Hive

HBase

Pig

Flume/Sqoop

Spark

Oozie

Hive

Provides an SQL interface to Hadoop

The bridge to Hadoop for folks who don’t have exposure to OOP in Java

HBase

A database management system on top of Hadoop

Integrates with your application just like a traditional database

Pig

A data manipulation language

Transforms unstructured data into a structured format

Query this structured data using interfaces like Hive

Spark

A distributed computing engine used along with Hadoop

Interactive shell to quickly process datasets

Has a bunch of built in libraries for machine learning, stream processing, graph processing etc.

Oozie

A tool to schedule workflows on all the Hadoop ecosystem technologies

Flume/Sqoop

Tools to transfer data between other systems and Hadoop

Summary

Understood the need for Distributed Computing

Understood the role of Hadoop in a distributed computing setup

Overview of basic technologies which exist in the Hadoop eco-system