Big Data and Hadoop

BAS Academy

Agenda

- Big Data
- Hadoop
- Hadoop HDFS
- Hadoop MapReduce
- Use Cases

Big Data

What is BIG DATA

- **Big data** is a term for **data** sets that are so **large** or complex that traditional **data** processing applications are inadequate to deal with them. Challenges include **capture, analysis**, **data** curation, search, sharing, **storage**, transfer, visualization, querying, updating and information privacy.

- Ex:

  Server logs

  Social Media Data

  Stock Exchange Data

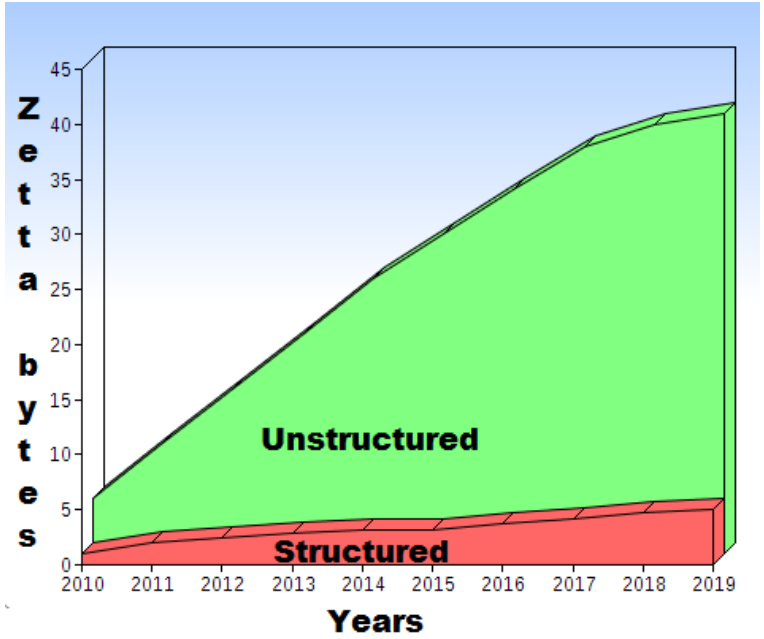
  Transport Data

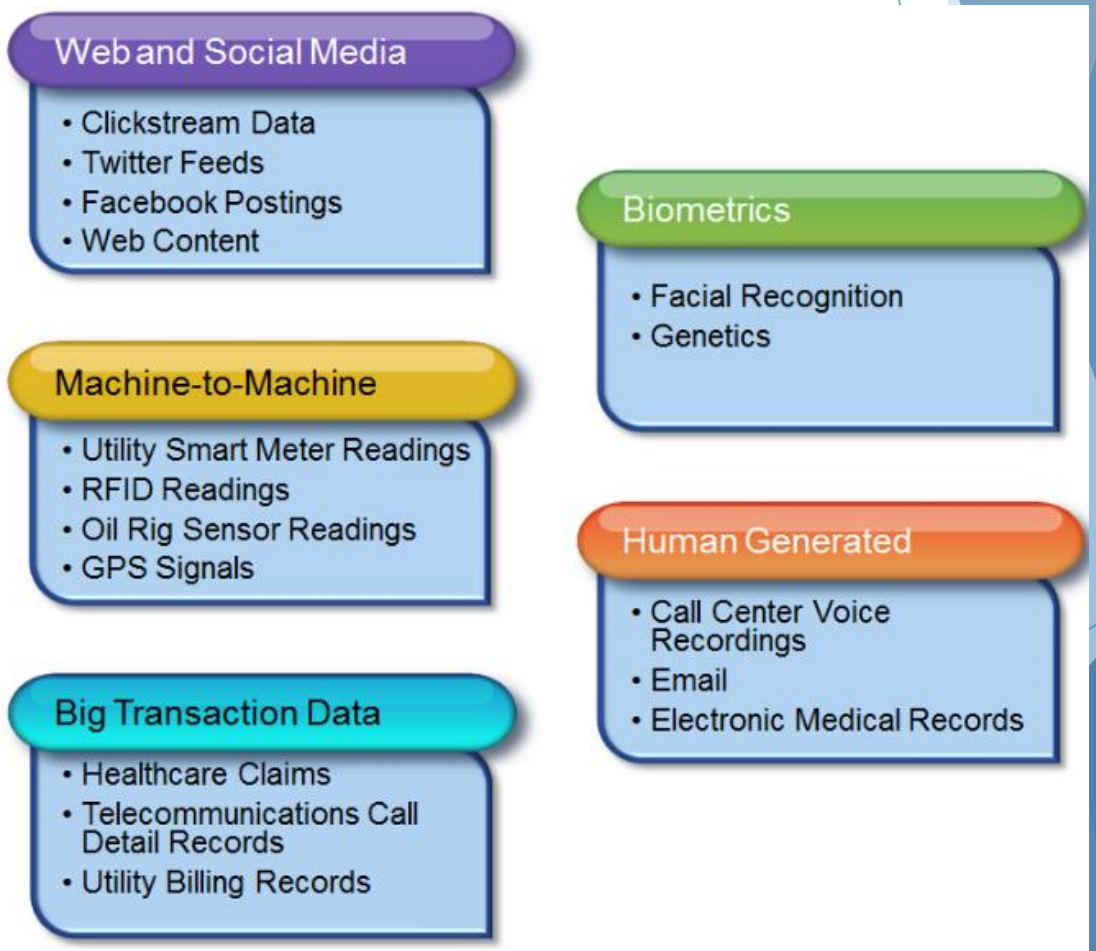
  Search Engine Data

Types of Data

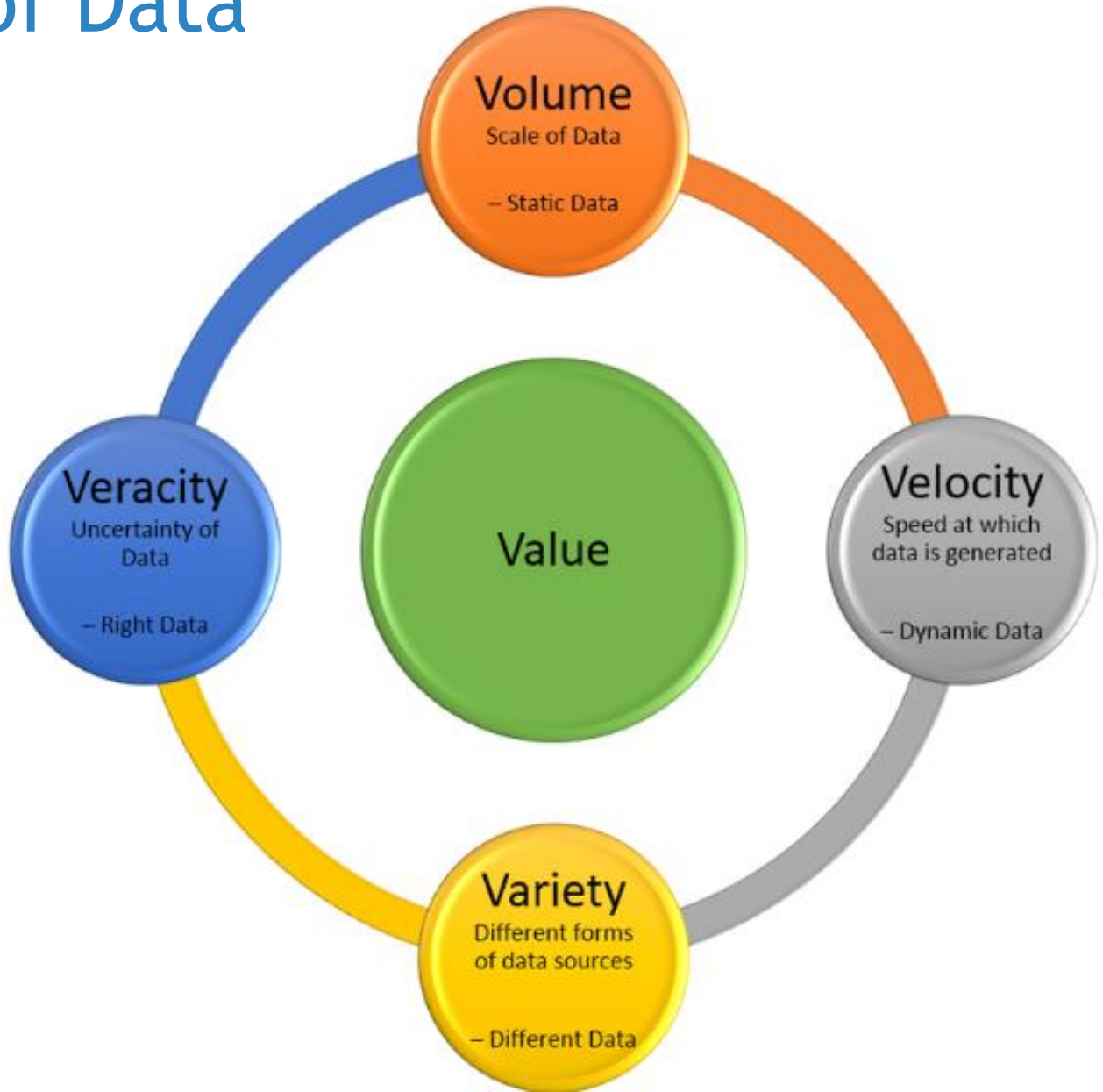
The data in Big Data is of three types.

- **Structured data** : Relational data.
- **Semi Structured data** : XML data.
- **Unstructured data** : Word, PDF, Text, Media Logs

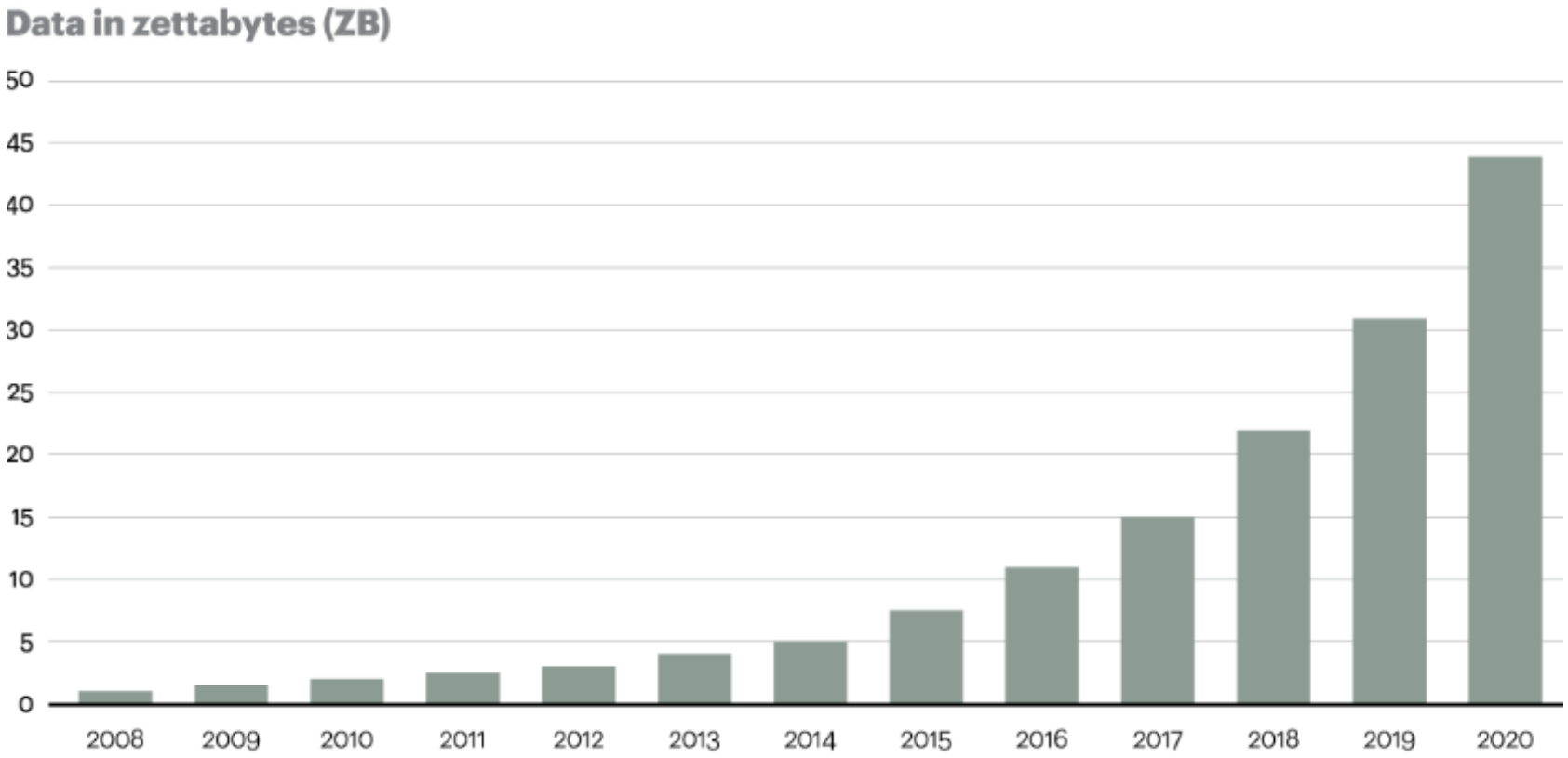


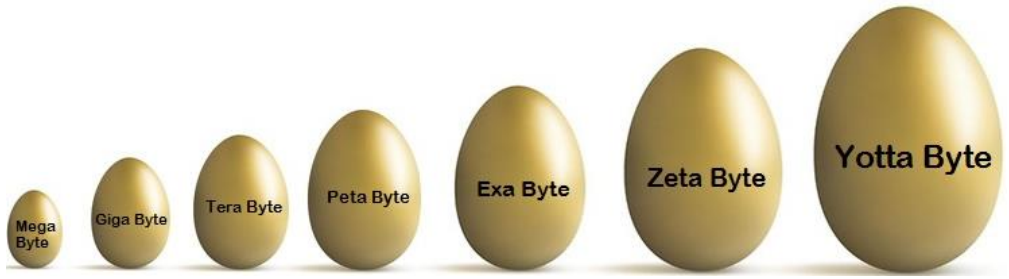


5 Vs of Data


Volume
Scale of Data
– Static Data
Velocity
Speed at which data is generated
– Dynamic Data
Variety
Different forms of data sources
– Different Data
Veracity
Uncertainty of Data
– Right Data
Value

Volume

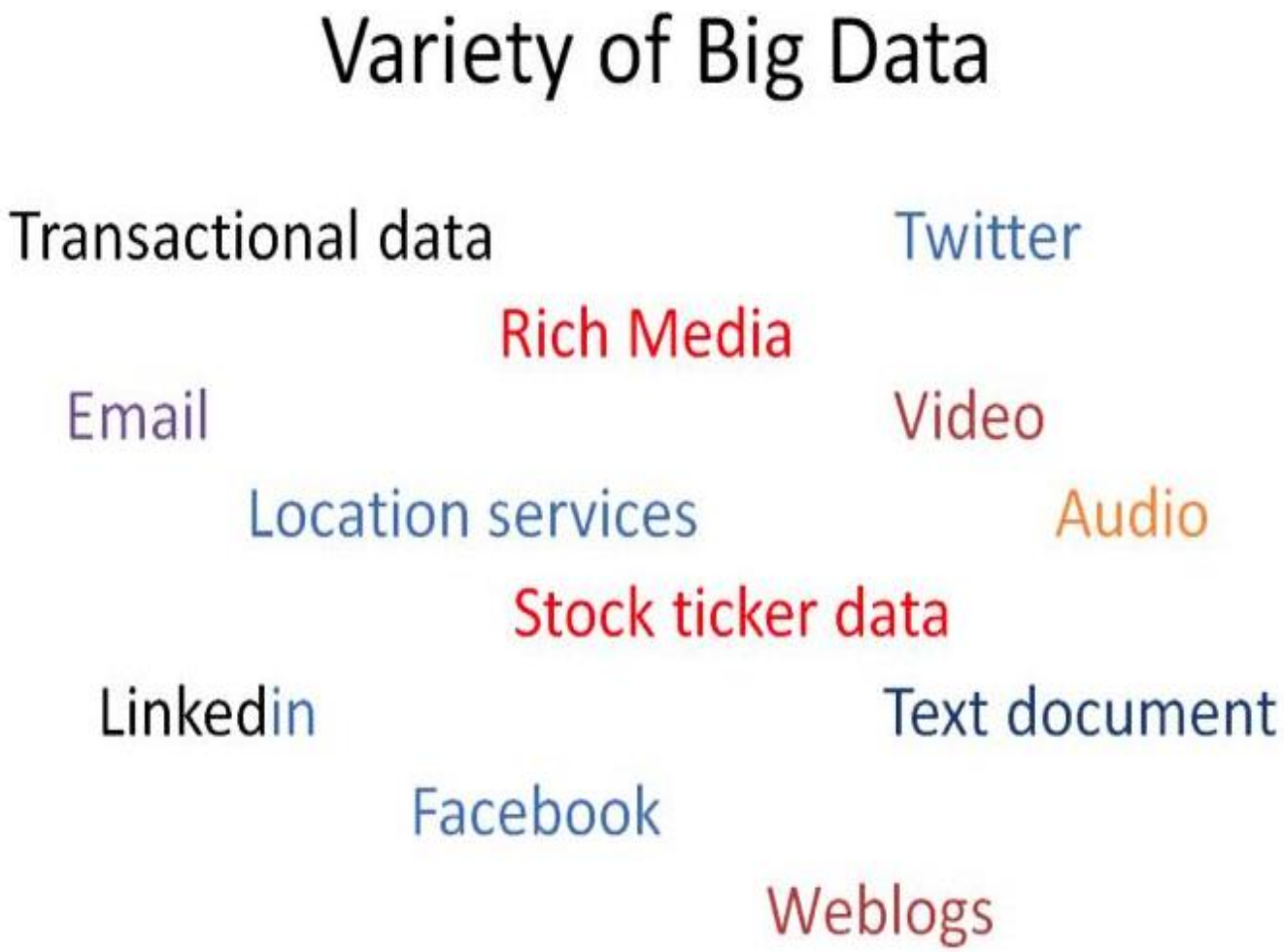

Data in zettabytes (ZB)
50
45
40
35
30
25
20
15
10
5
0
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020


Mega Byte
Giga Byte
Tera Byte
Peta Byte
Exa Byte
Zeta Byte
Yotta Byte

Velocity

Airbus generates 10 TB every 30 minutes
About 640 TB is generated in one flight

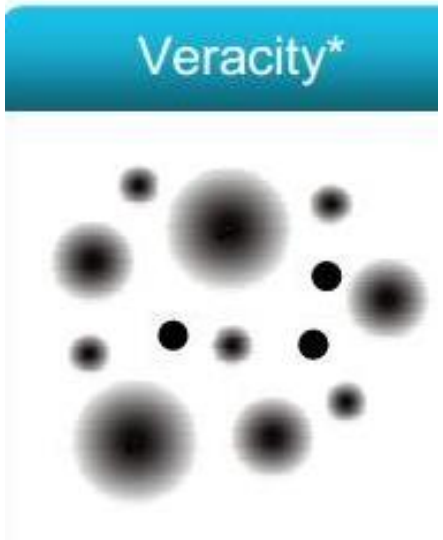
Variety


Variety of Big Data
Transactional data
Twitter
Rich Media
Email
Video
Location services
Audio
Stock ticker data
Linkedin
Text document
Facebook
Weblogs

Veracity

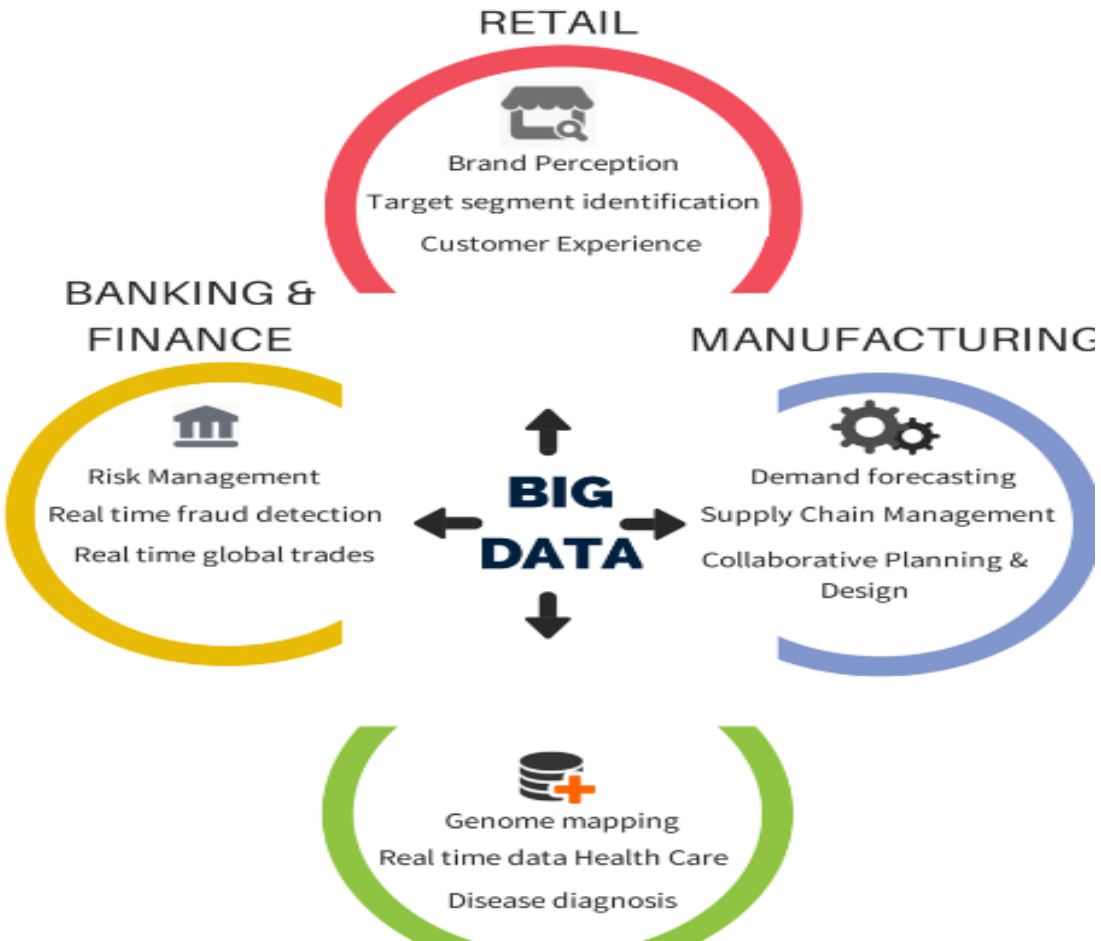
- Measurement error in the case of sensors?
- Should you trust all tweets about a given company?
- Lack of credential in case of social media

- Veracity provides confidence in the trustworthiness of the data

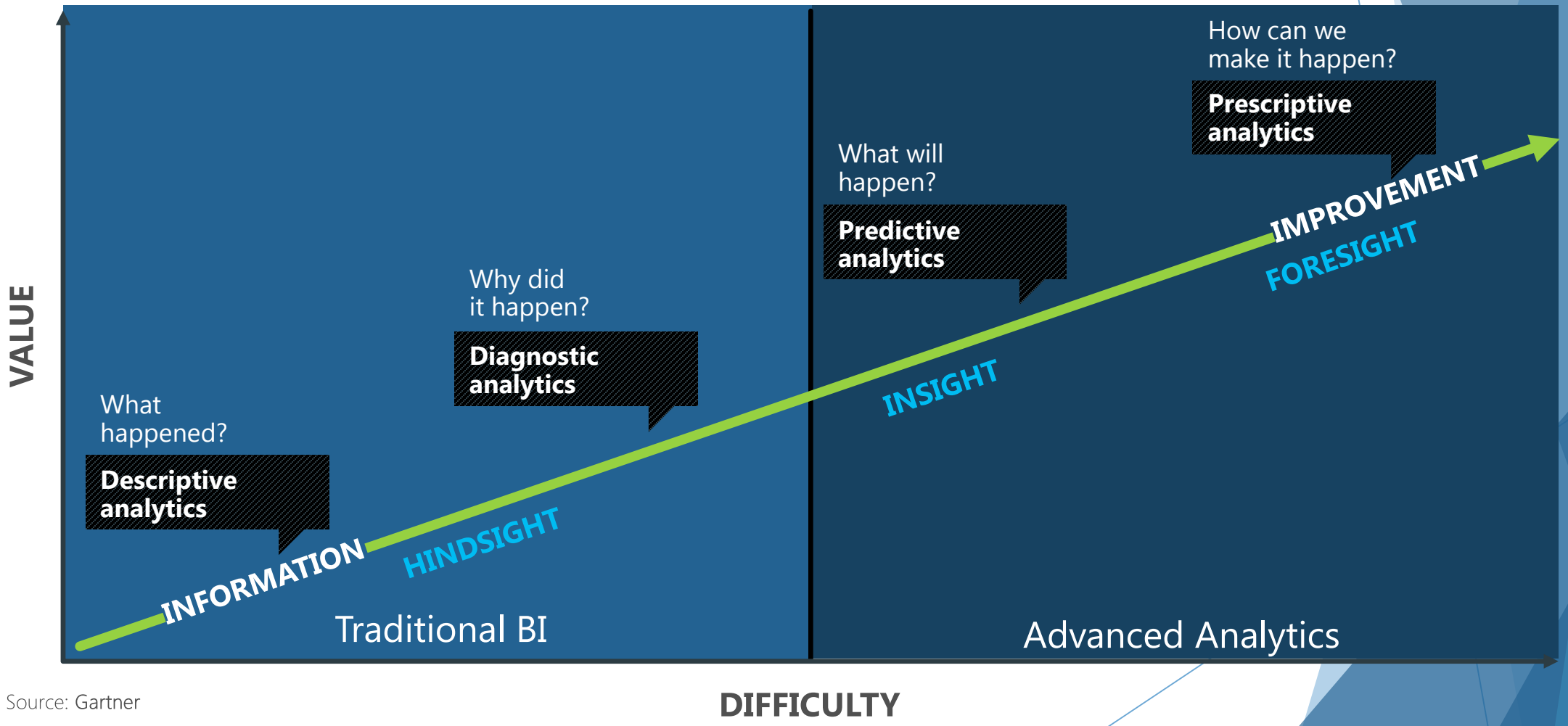


Value

- Extracting business insights and revenue from Big data
- Over 80% of organizations say, Big data is critical to meet strategic objectives



Advanced Analytics
Beyond Business Intelligence



Source: Gartner

Hadoop

Handling Big Data

- Problem – Storing the data into a scalable file system

Solution: Hadoop Distributed File System (HDFS)

- Problem – Using parallel processing on the data

Solution: Hadoop Map-Reduce

- Problem – Storing the data into a database for faster access

Solution: NoSQL Databases like MongoDb, Cassandra, HBase

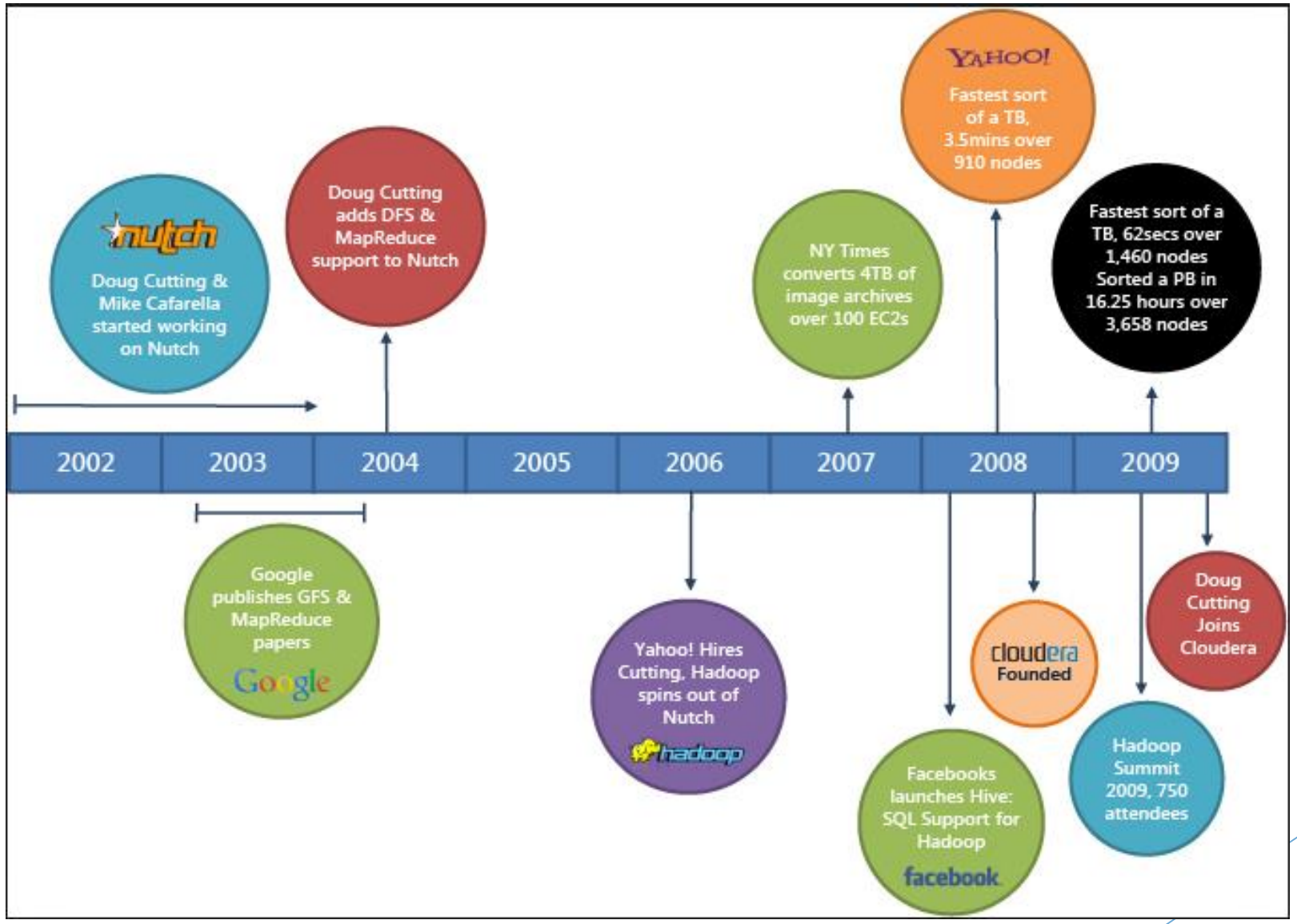
What is Hadoop

- Open source framework for distributed processing to handle Big Data by Apache
- Developed by Doug Cutting with support from Yahoo later
- Inspired by Map Reduce and Google File System where data is stored in commodity hardware with multiple replication to provide High Availability
- Designed to be scalable from a single system to support thousands of nodes

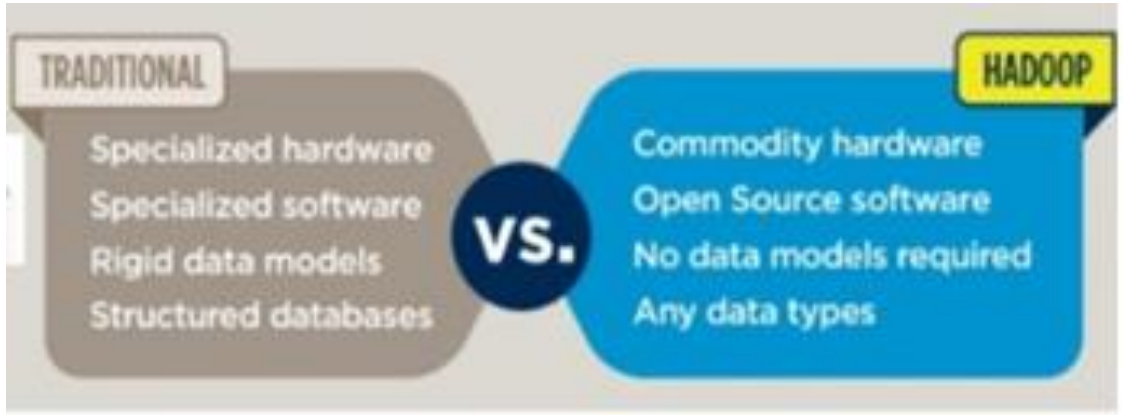
- **Yahoo has 4500 node Hadoop Cluster**
- **Facebook has 1100 node Hadoop Cluster**

The cute li'l yellow elephant is actually Doug's son's toy elephant; Hadoop is named after it!

From Spider to Elephant



Why Hadoop

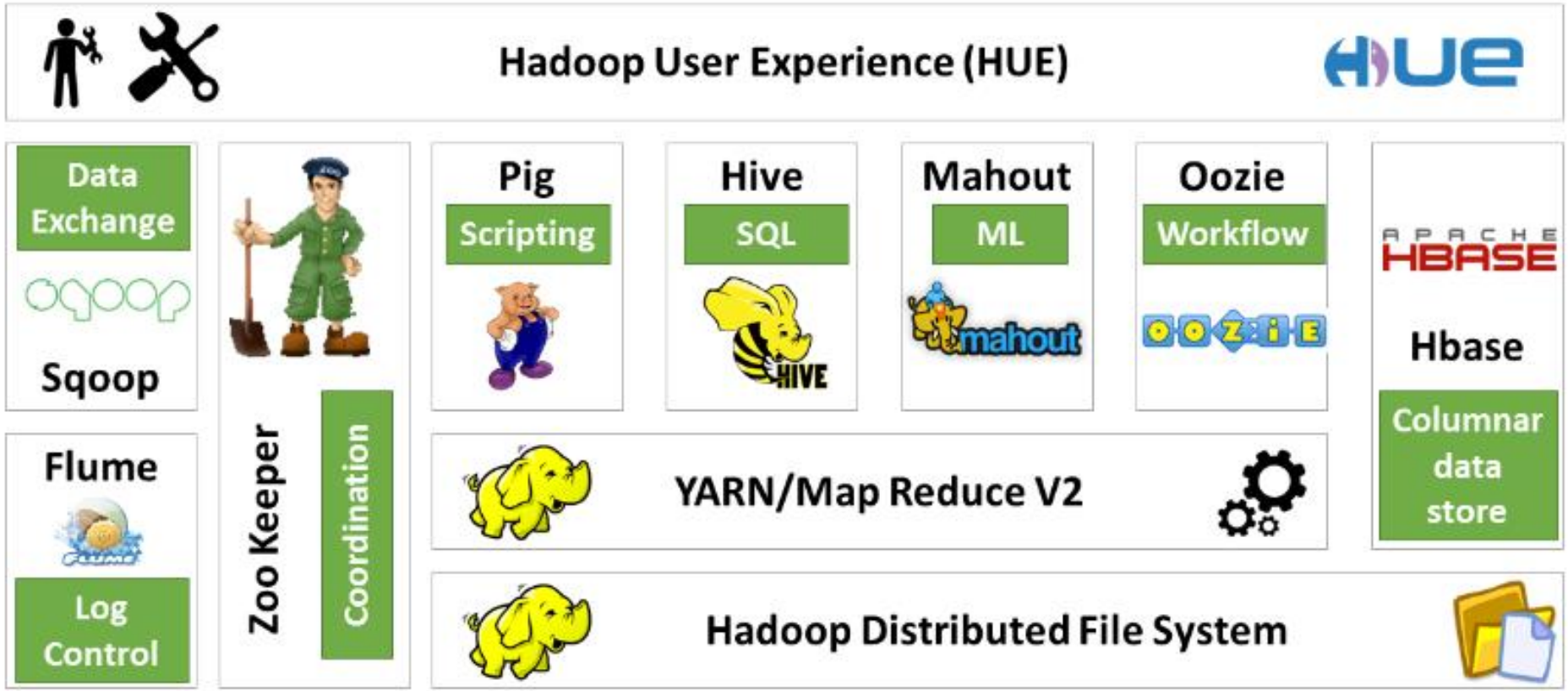


- All the processing in Hadoop will be done in individual data nodes
- Scale Out is much more effective than Scale In
- Hadoop is open source low cost software

Hadoop is :

- Reliable
- Scalable
- Distributed computing

Hadoop Ecosystem



Hadoop Projects

HDFS (Hadoop Distributed File System)

HDFS
Hadoop Distributed File System

- It is a file system designed for storing very large files with streaming data access patterns, running clusters on commodity hardware.

Map Reduce

MapReduce
Distributed Programming Framework

- It is a software framework for easily writing applications which process big amounts of data in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner.

YARN:

YARN
(cluster resource management)

- This is a framework for job scheduling and cluster resource management

Hadoop Projects

Pig

Pig is a high-level platform for creating MapReduce programs used with Hadoop

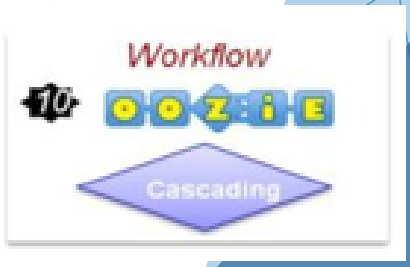
Hive

Hive provides a database query interface to Apache Hadoop

A data warehouse infrastructure that provides data summarization and ad hoc querying

Oozie

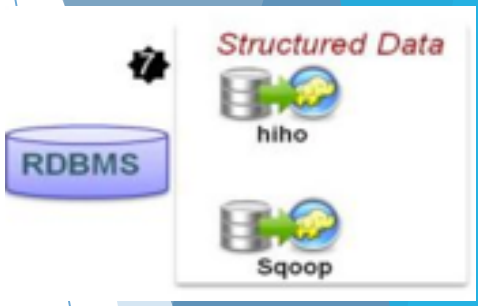
Oozie is a workflow scheduler system to manage Apache Hadoop jobs.



Hadoop Projects

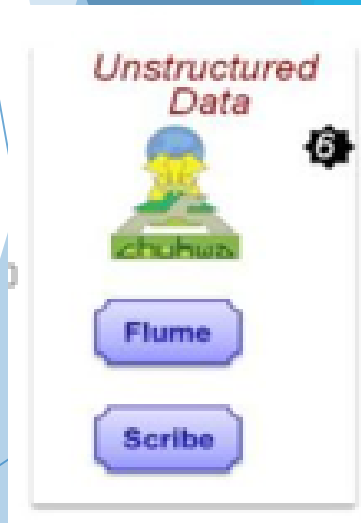
Sqoop 

Apache Sqoop(TM) is a tool designed for efficiently transferring bulk data between Apache Hadoop and structured data stores such as relational databases.



Flume 

A service for streaming logs into **Hadoop**



Hbase 

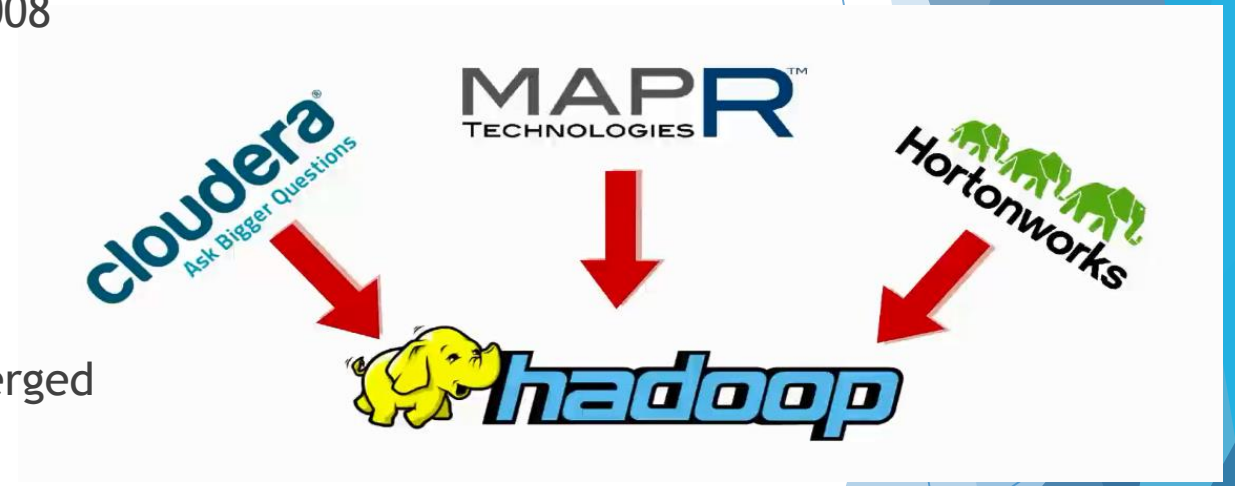
A scalable, distributed database that supports structured data storage for large tables

Hadoop – Distributed Networks

- They provide enterprise-ready Hadoop distributions

- Cloudera Inc. was founded by big data geniuses from Facebook, Google, Oracle and Yahoo in 2008 and is mostly used.
- Cloudera Manager is proprietary management software

  Versions: CDH 4.7.x, CDH 5.1.x



- Hortonworks, founded in 2011, has quickly emerged as one of the leading vendors of Hadoop
- Ambari is open source management software

  Versions: HDP 2.1, HDP 2.2

- MapR develops and sells Apache Hadoop-derived software (Ex: MapRFS in the place of HDFS)

Hadoop Interfaces

- Command Line
- Web Interface (Ex: Hue)
- ODBC
- JDBC
- Thrift Interface(For applications that are not written in Java like Python scripts)
- REST API (For applications that are written in Java like Python scripts)

Hadoop Data Types

1. **Sentiment**
   How your customers feel
2. **Clickstream**
   Website visitors' data
3. **Sensor/Machine**
   Data from remote sensors and machines
4. **Geographic**
   Location-based data
5. **Server Logs**
6. **Text**
   Millions of web pages, emails, and documents

Value

- Sentiment data is unstructured data that represents opinions, emotions, and attitudes contained in sources such as social media posts, blogs, online product reviews, and customer support interactions.
- Clickstream data is an information trail a user leaves behind while visiting a website
- Sensor data is generated from sensors and machines
- Geolocation data gives organizations the ability to track every moving aspect of their business
- Server logs are computer-generated log files that capture network and server operations data.

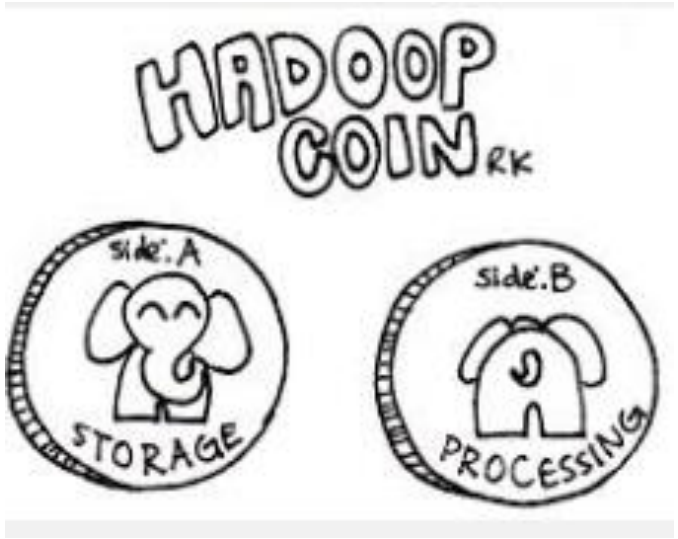
Hadoop Components

HDFS

- Distributed file system
- Responsible to store data in blocks

MapReduce

- Distributed data processing model
- To process the data in massive parallel manner



Hadoop HDFS

About HDFS

- Primary storage system for Hadoop
- It stores large files as small blocks across a cluster of machines
- Designed to be deployed on low-cost hardware
- Designed to scale easily and effectively
- Reliability: data is replicated so that disk failover is handled efficiently

Hadoop Cluster

Cluster is

- a set of connected computers that work together
- For storing and analyzing huge amounts of structured and unstructured data

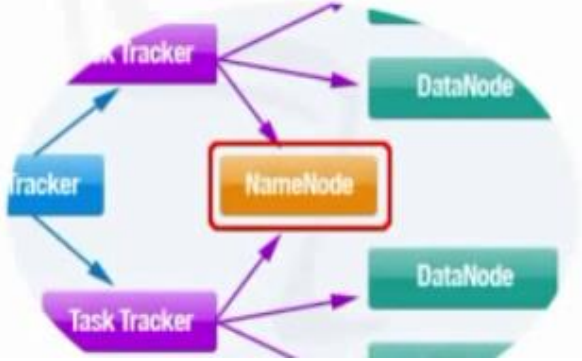
- Each computer is called as Node
- Additional nodes can be added to a cluster

The two components of HDFS cluster are:

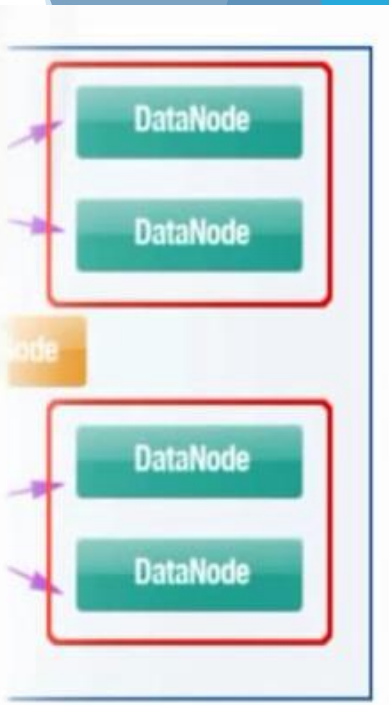
- Name Node (Master)
- Data Node (Slave)

HDFS Components

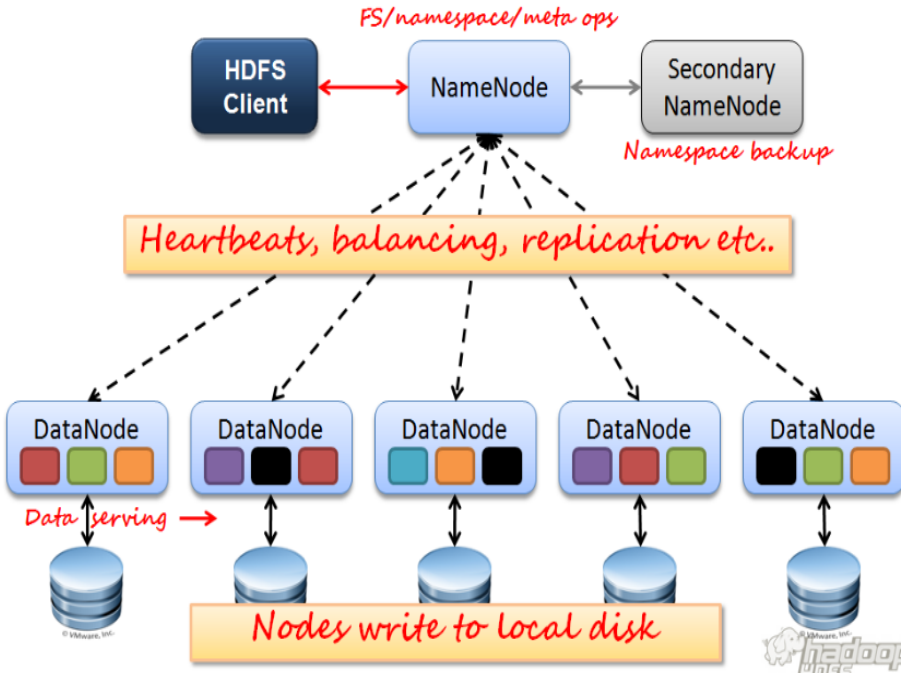
- **NameNode**
  - Only one per Hadoop cluster
  - Manages the filesystem namespace and metadata
  - Single point of failure, but mitigated by writing state to multiple filesystems
  - Single point of failure: Don't use inexpensive commodity hardware for this node, large memory requirements

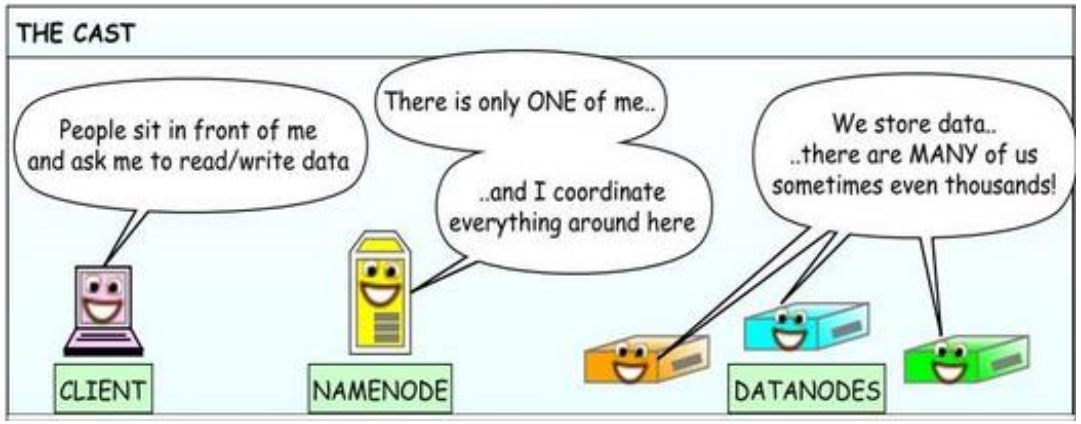


- **DataNode**
  - Many per Hadoop cluster
  - Manages blocks with data and serves them to clients
  - Periodically reports to name node the list of blocks it stores
  - Use inexpensive commodity hardware for this node



HDFS Architecture





- HDFS has a master/slave architecture

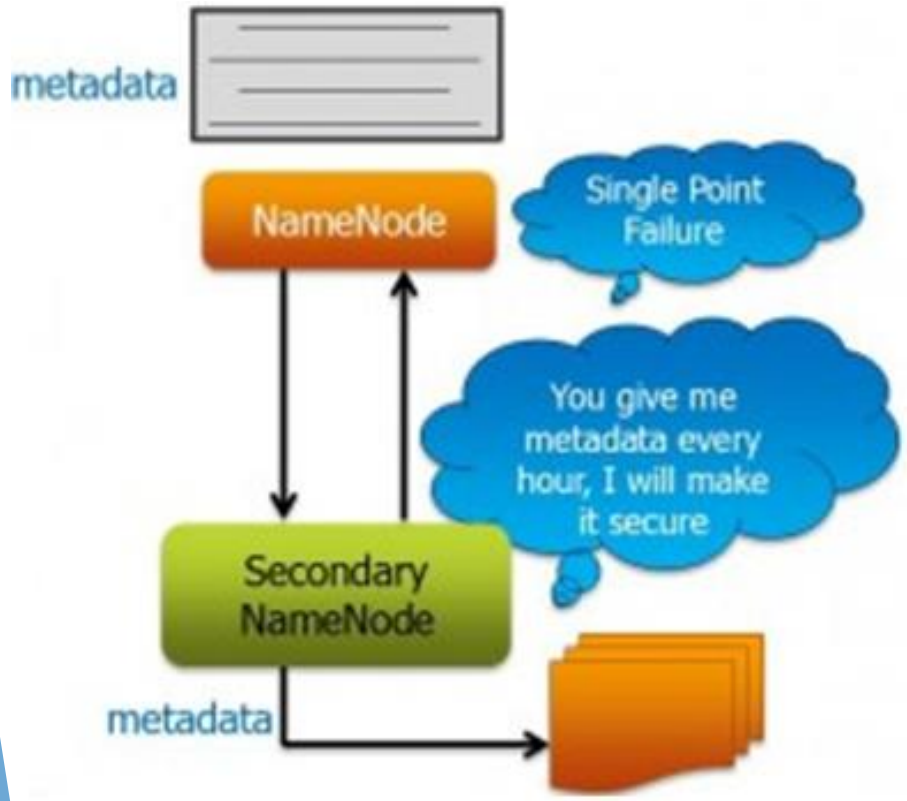
Name Node

- The NameNode determines the mapping of blocks to DataNodes.

Data Node

- Sending heartbeats to the NameNode
- Sending a Blockreport to the NameNode

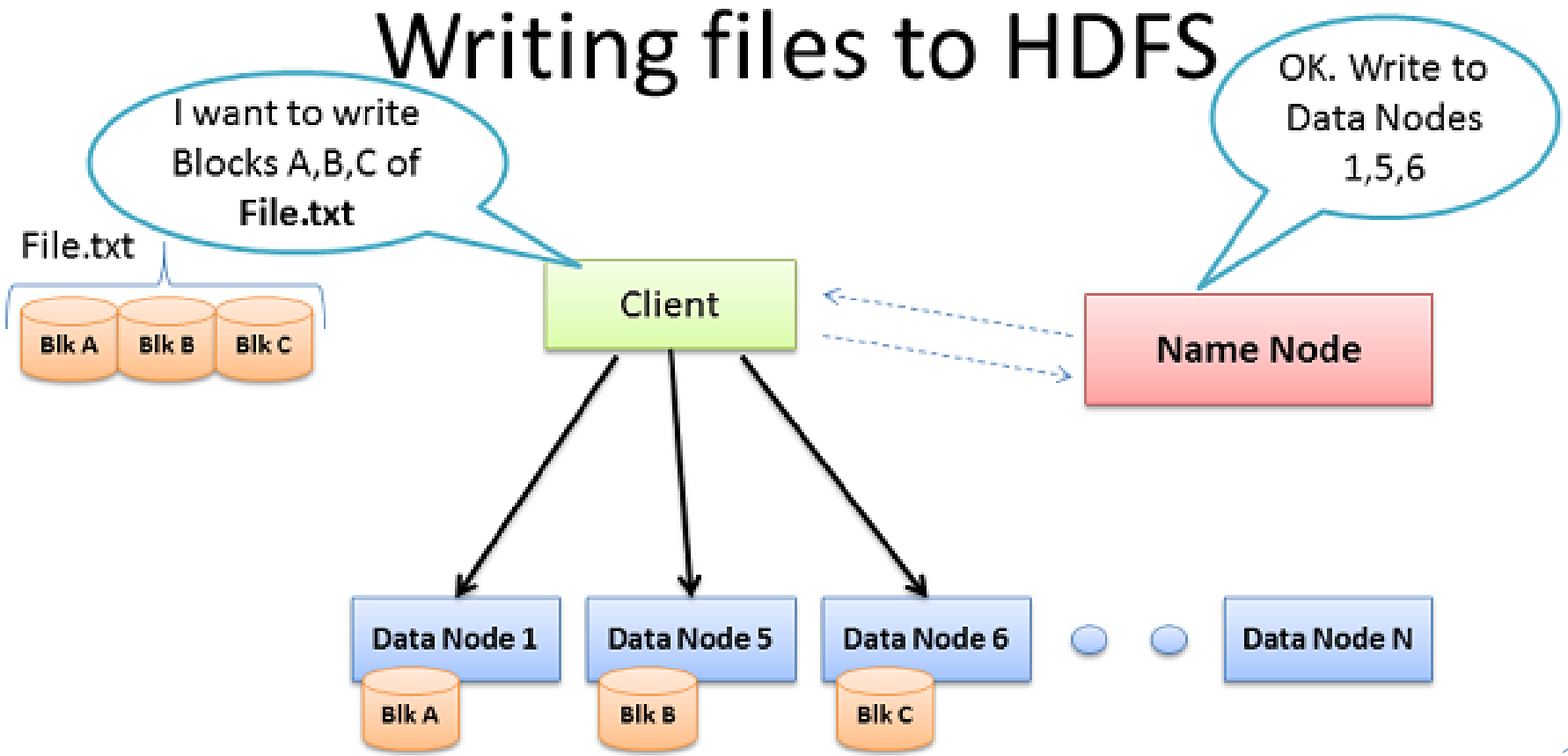
Secondary NameNode



Secondary NameNode:

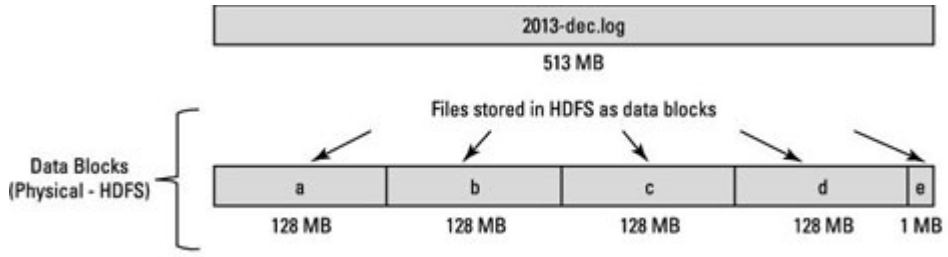
- Connects to NameNode regularly
- Housekeeping, backup of NameNode metadata
- Saved metadata can build a failed NameNode

Writing in HDFS



Blocksize

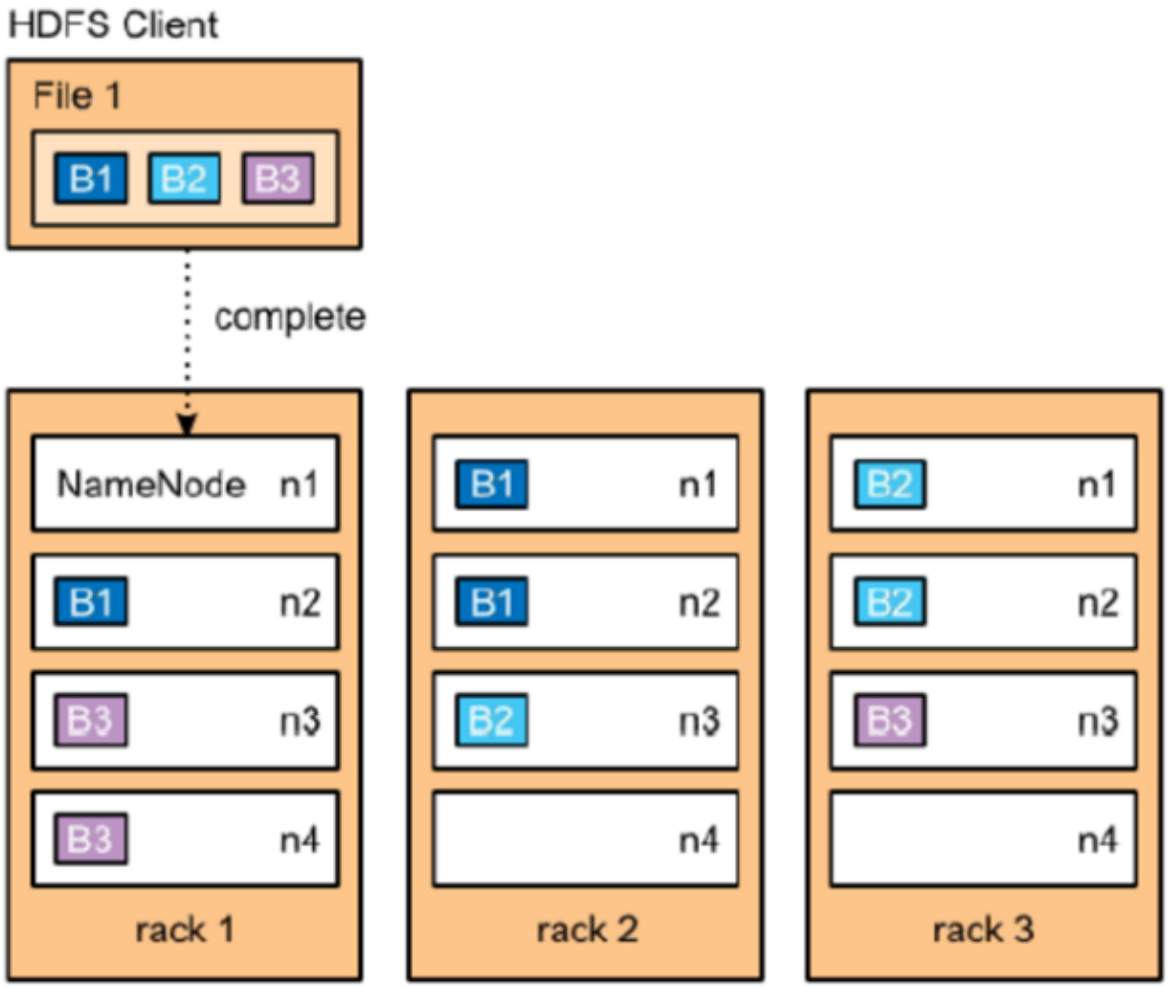
- What is block :: The block is the smallest unit of data that a file system can store. If you store a file that's 1k or 60Mb, it'll take up one block. Once you cross the 128MB boundary, you need a second block.



- Default Block Size = 128 MB
- 514 MB / 128 MB = 4.05 == 5 Blocks
- Replication Factor = 3
- Total Blocks = 5 * 3 = 15
- Total size = 514 * 3 = 1542 MB

- The default block size is 128MB :: In the Hadoop 2.x the default block size is increased to 128 MB from 64MB. **Why?** If block size was set to less than 128MB, there would be a huge number of blocks throughout the cluster, which causes Name Node to manage an enormous amount of metadata.

HDFS Replication

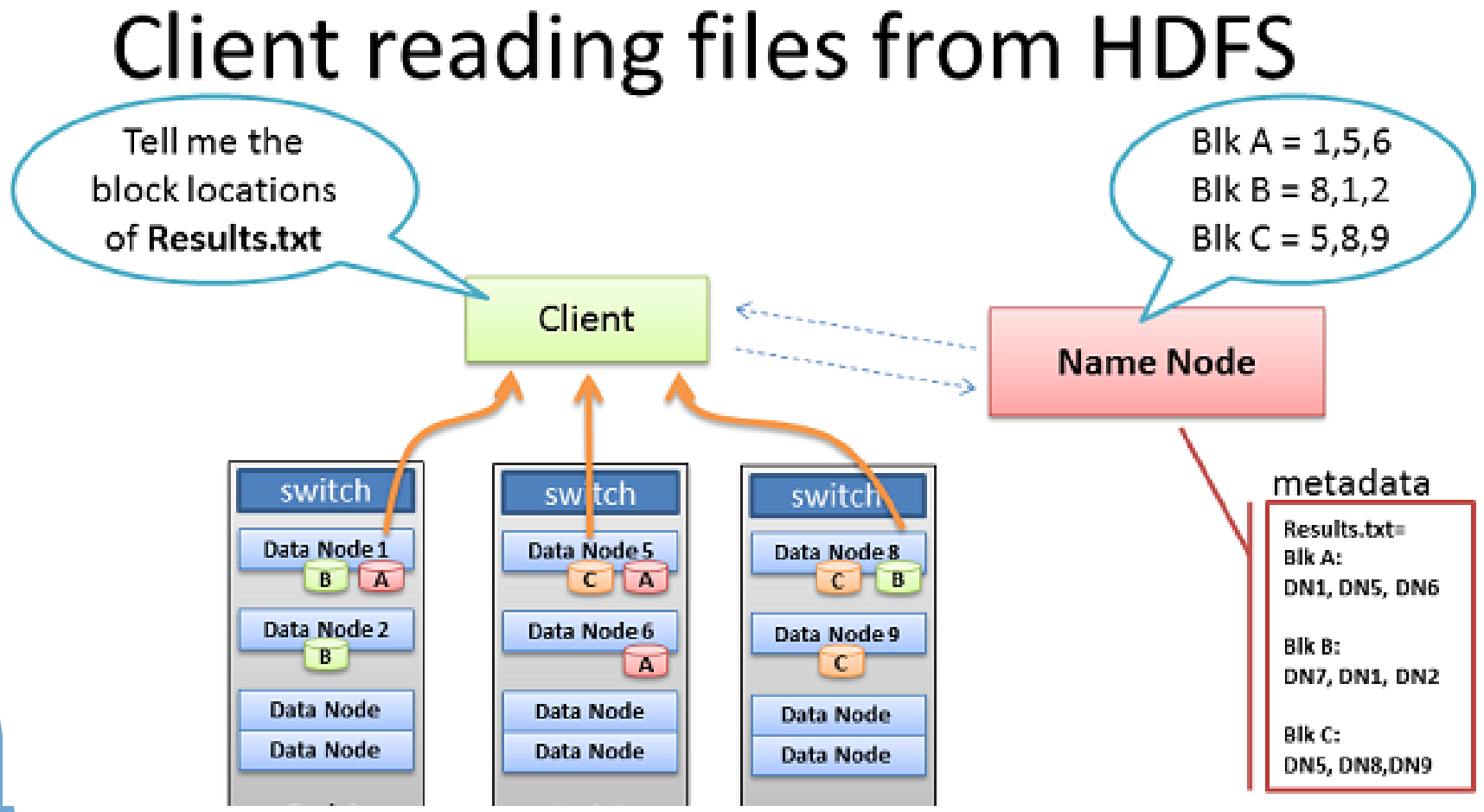


- The default block size is 128MB
- The default replication factor is three

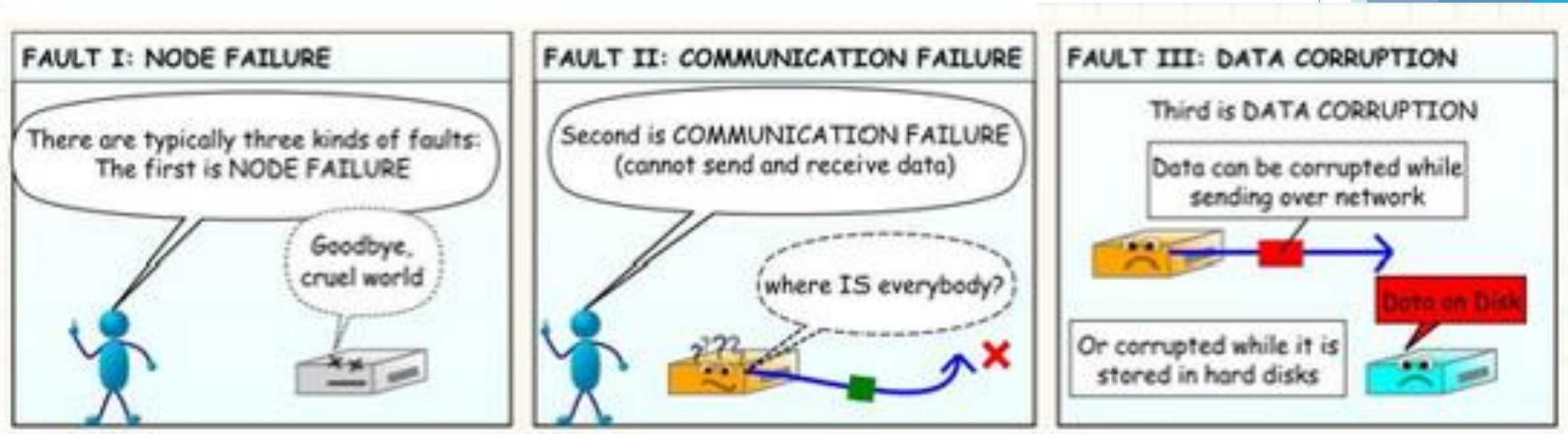
Fault Tolerant

- Replication Factor : As HDFS was designed to be fault-tolerant and to run on commodity hardware, blocks are replicated a number of times to ensure high data availability. The replication factor is a property that can be set in the HDFS configuration file that will allow you to adjust the global replication factor for the entire cluster.
- For each block stored in HDFS, there will be n – 1 duplicated blocks distributed across the cluster. For example, if the replication factor was set to 3 (default value in HDFS) there would be one original block and two replicas.

Reading from HDFS

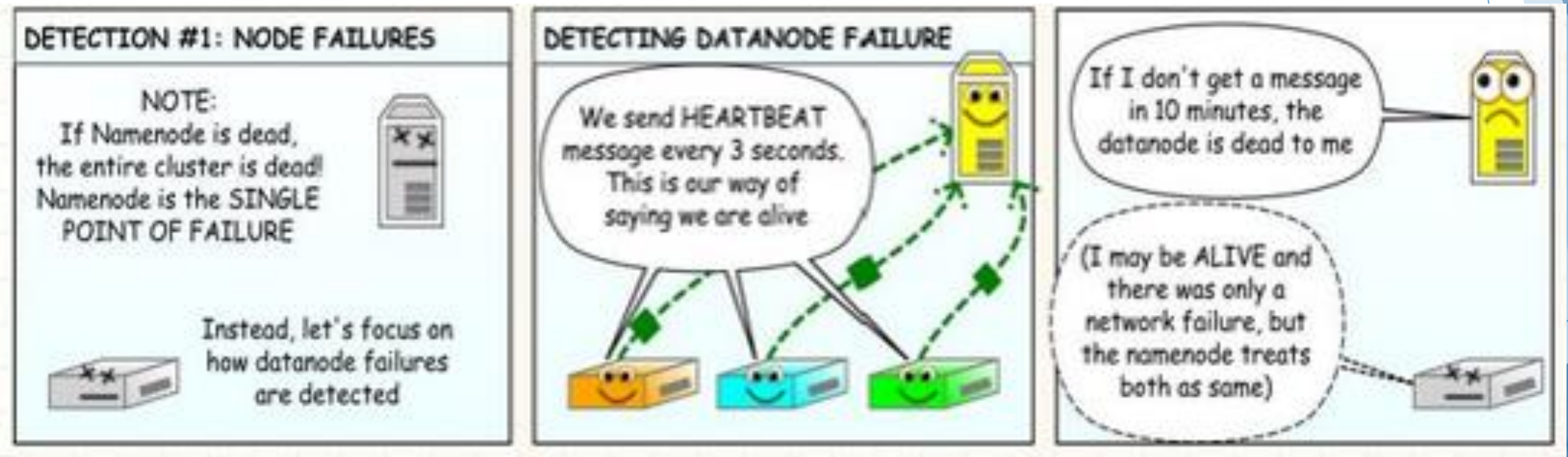


Types of Failures

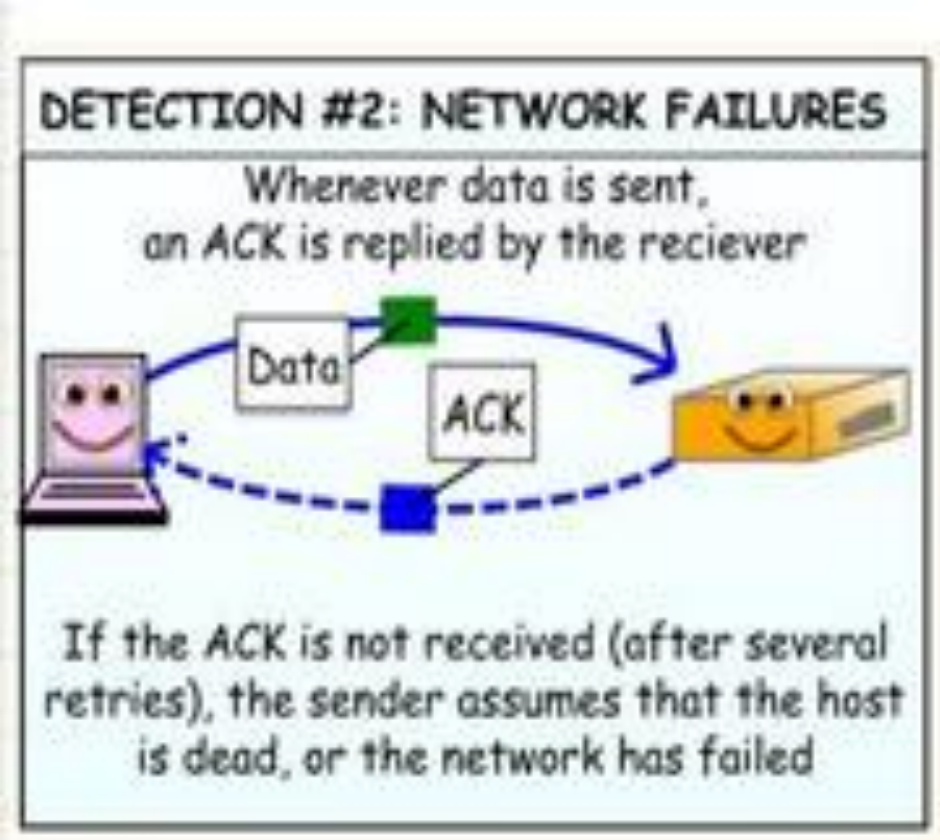


Node Failure

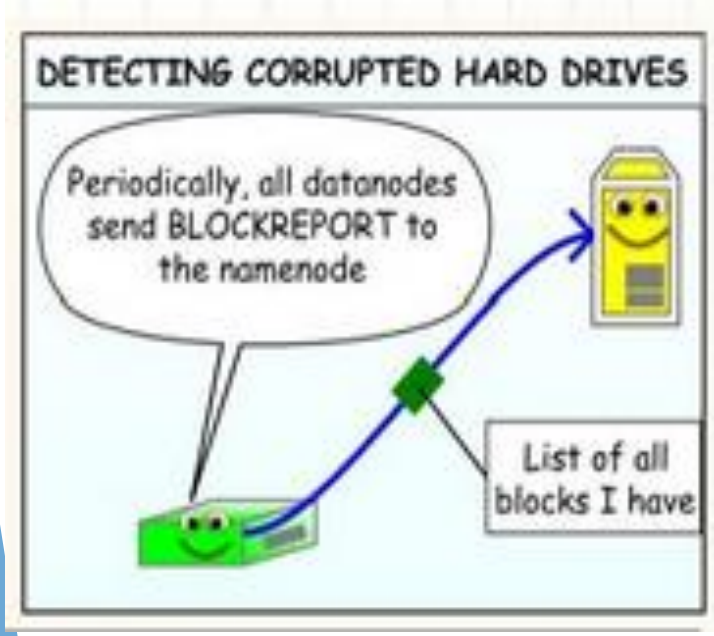
- Name node will mark the data node as dead so that client will not communicate to the data node

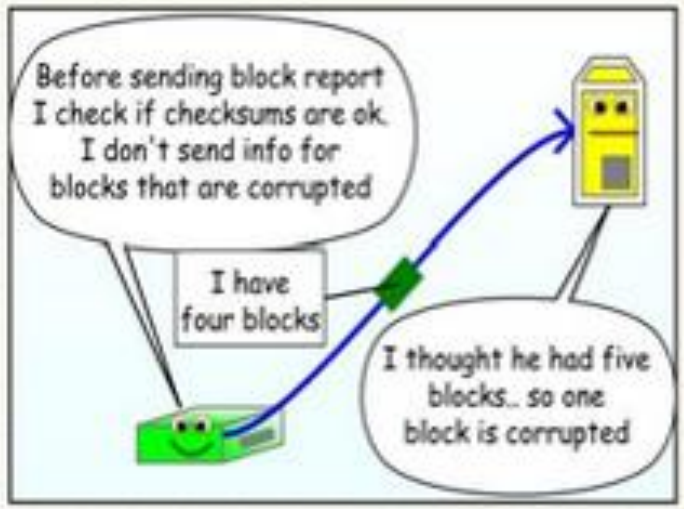


Network Failure

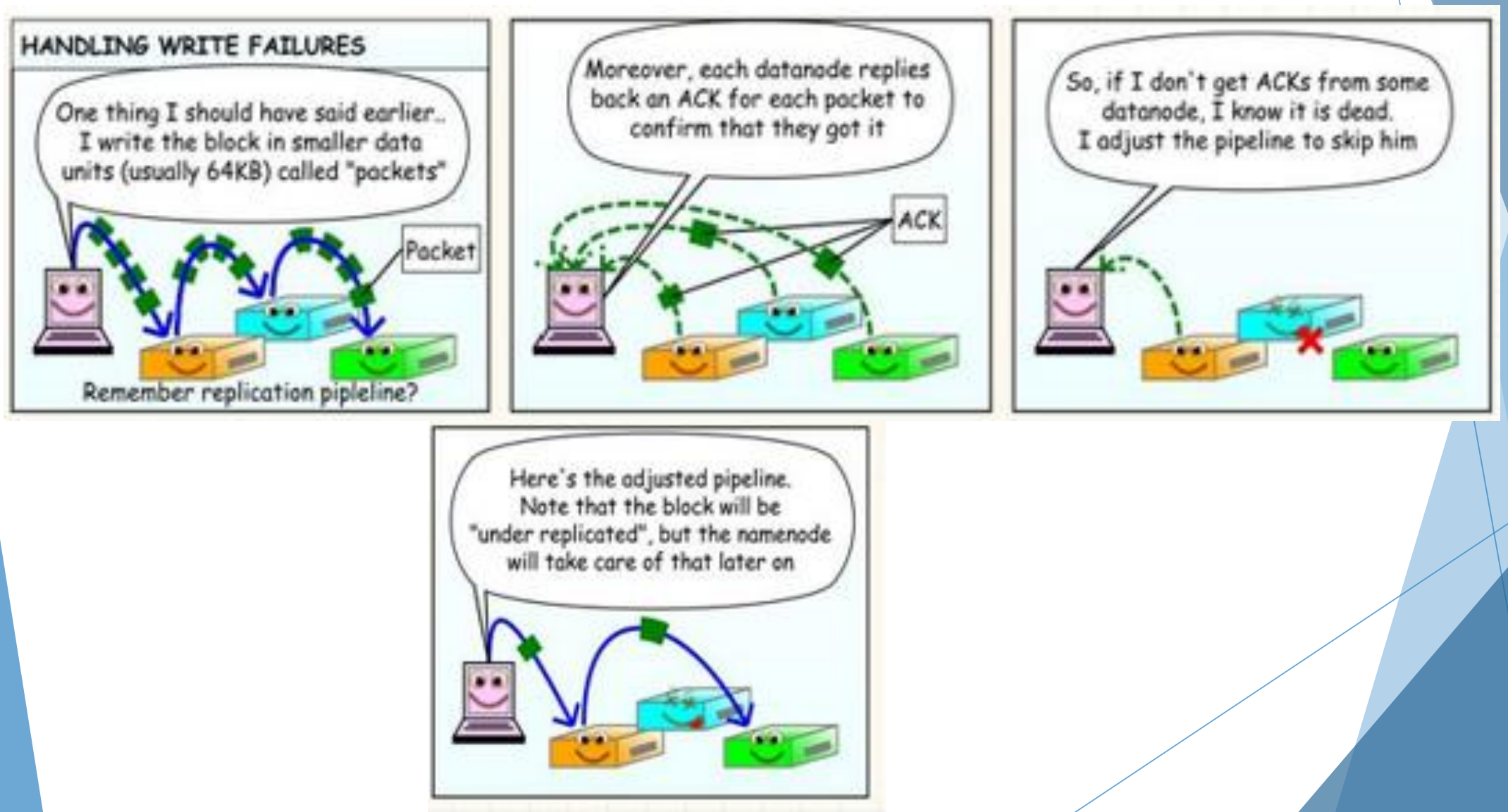


Data Corruption

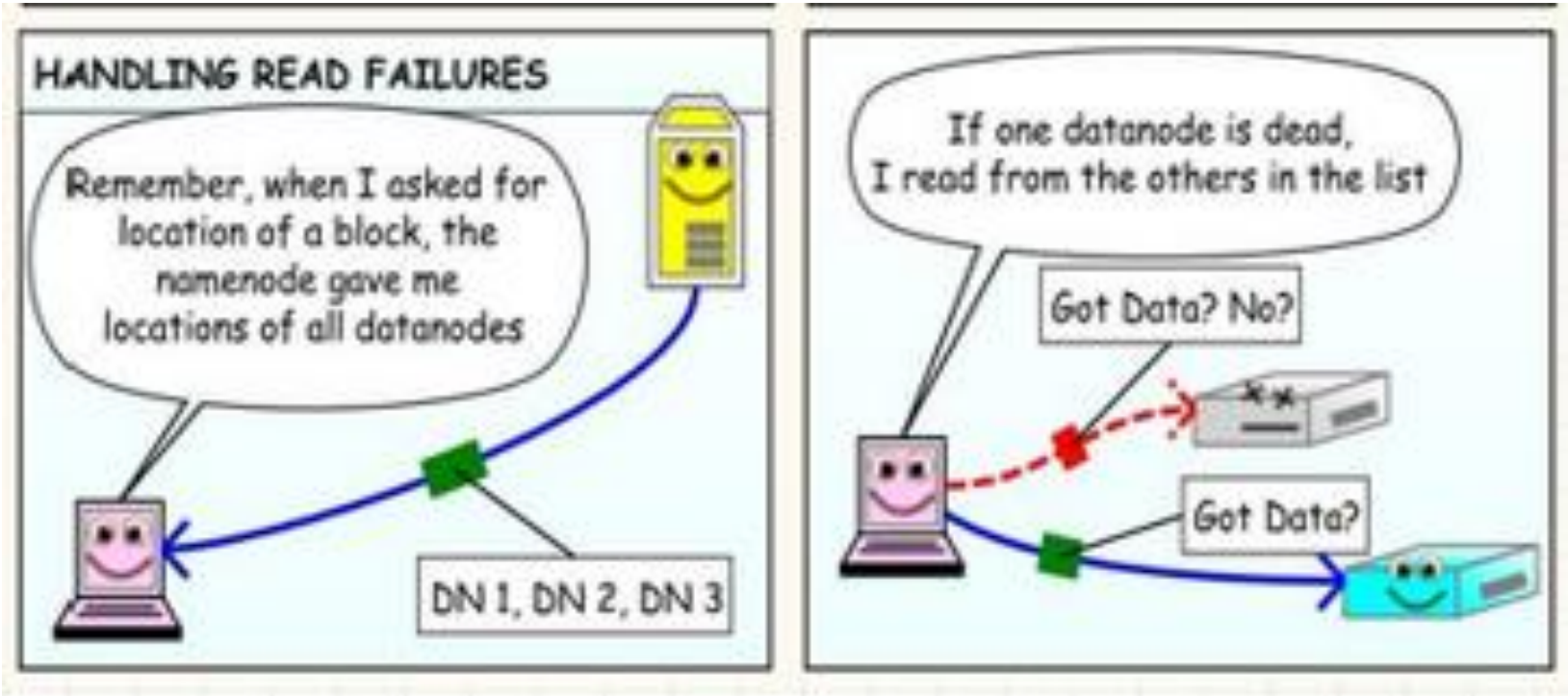




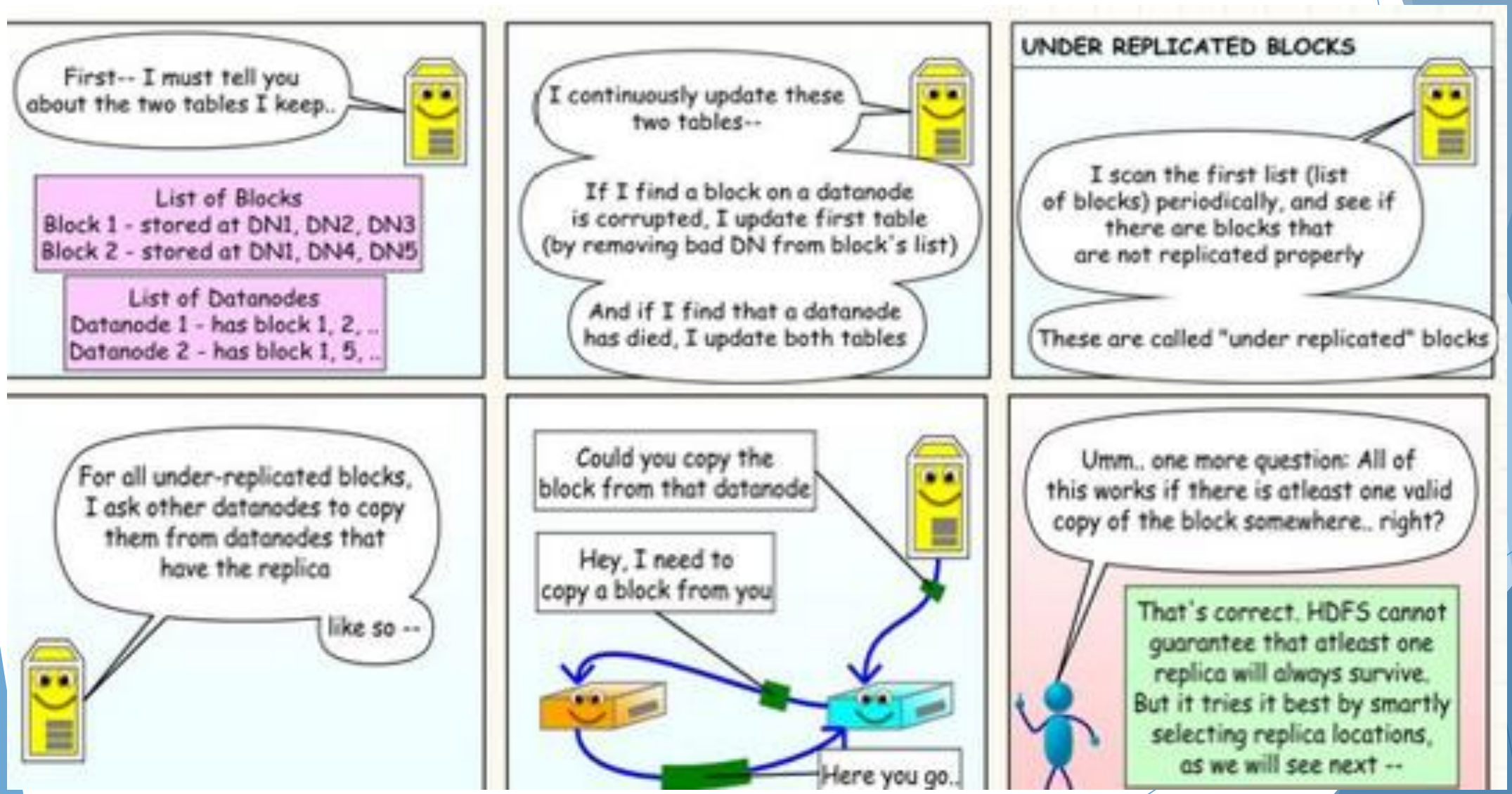
Handling Write Failure



Handling Read Failure



Handling Data Node Failure



HDFS Commands

HDFS – Command Line Interface (hadoop fs command)

Get a directory listing in users home directory in HDFS – (if home directory exists)

- hadoop fs -ls

Get a directory listing of the HDFS root directory

- hadoop fs –ls /

Creating a directory in HDFS

- hadoop fs –mkdir <dirname>

Copying a local file into HDFS

- hadoop fs –copyFromLocal <local fileName> <targetDirectory/targetFileName>
- hadoop fs –put <fileName> <targetDirectory/targetFileName>

Display the contents of a file

- hadoop fs –cat <filename>

Copying a HDFS file into local file system

- hadoop fs –copyToLocal <sourceDirectory/sourceFileName> <local fileName>
- hadoop fs –get <sourceDirectory/sourceFileName> <local fileName>

Removing a directory and its contents from HDFS

- hadoop fs –rmr <targetDirectory>

Hadoop MapReduce

About MapReduce

- To process large amounts of data in parallel across a distributed environment.
- A MapReduce program consists of two main phases:

a map phase and a reduce phase

Map phase :

- Data is input into the mapper, where it is transformed and prepared for the reducer

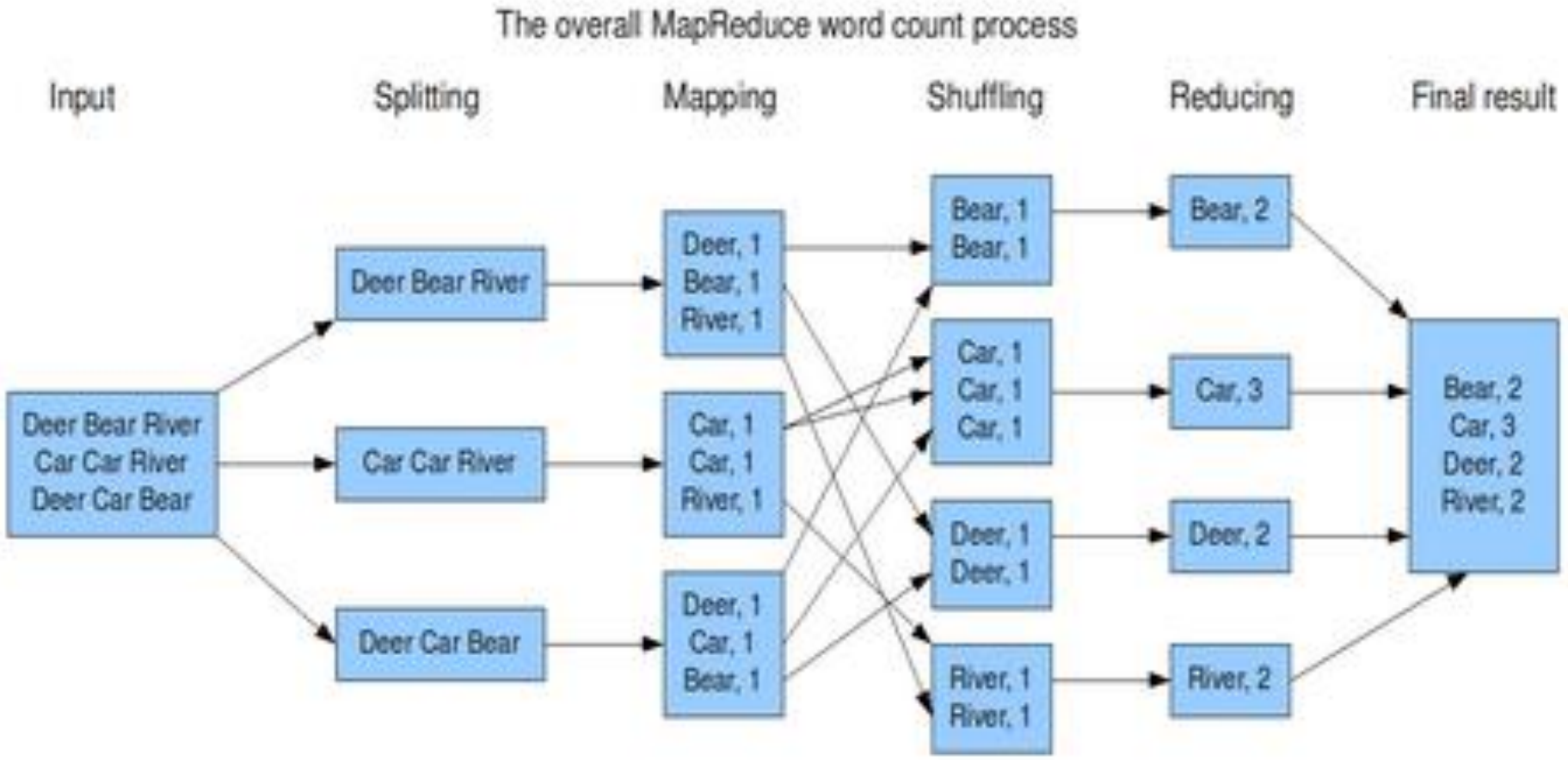
Reduce phase:

- Retrieves the data from the mapper and performs the desired computations or analyses

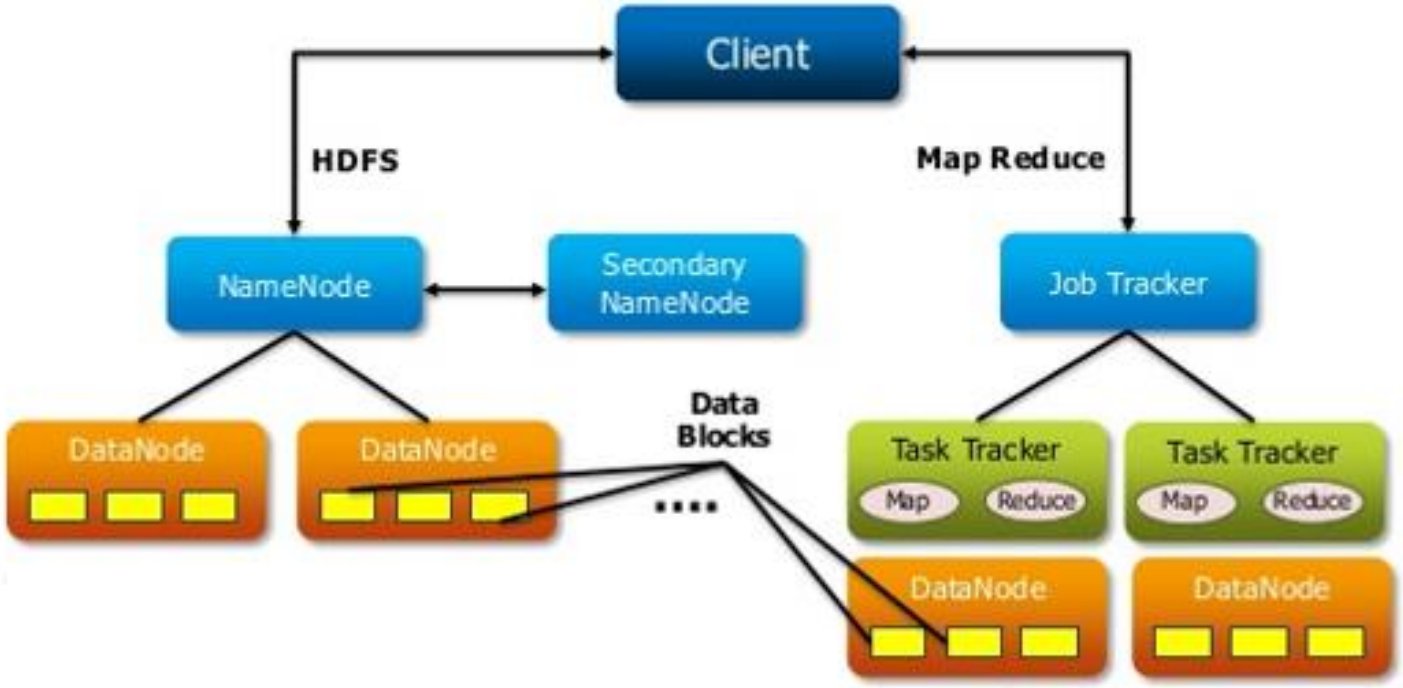
MapReduce Features

- The map and reduce tasks run in their own JVM on the DataNodes
- The mapper inputs key/value pairs from HDFS files and outputs intermediate key/value pairs. The data types of the input and output pairs can be different
- After all of the mappers finish executing, the intermediate key/value pairs go through a shuffle-and-sort phase where all of the values that share a key are combined and sent to the same reducer
- The reducer inputs the intermediate <key, value> pairs and outputs its own <key,value> pairs, which are typically written to HDFS
- The number of mappers is determined by the input format
- The number of reducers is determined by the MapReduce job configuration

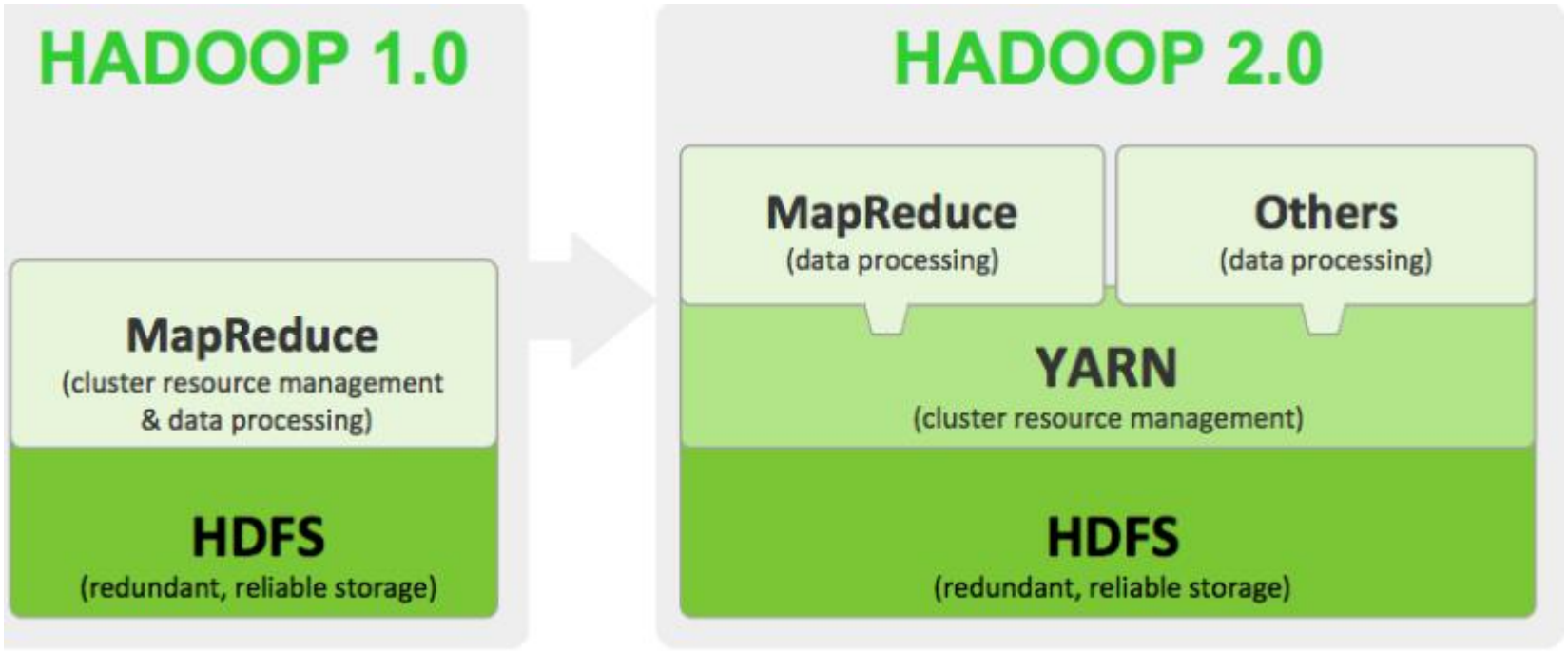
Word Count Example



MapReduce Framework and YARN



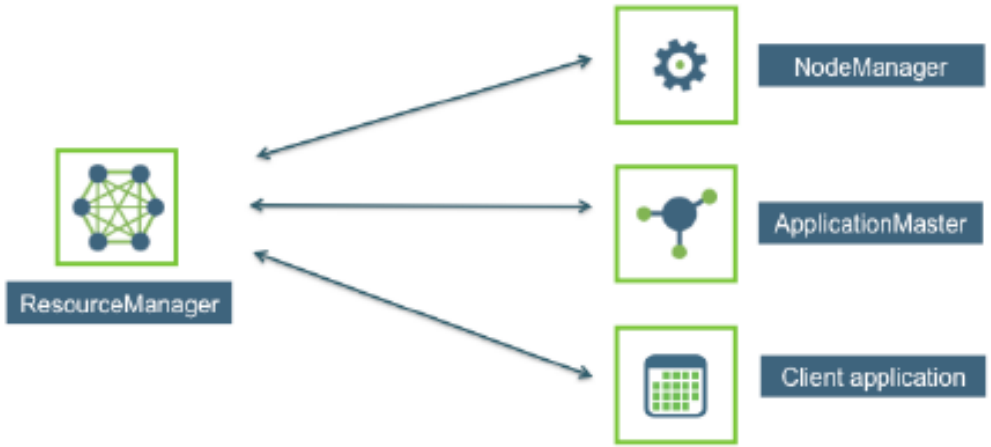
Hadoop Versions



YARN

- YARN is Yet Another Resource Negotiator
- Provides a generic framework that allows for any type of application to execute on the big data across your clusters.
- It replaces job tracker and task tracker
- YARN provides better resource management in Hadoop, resulting in improved cluster efficiency

The Components of YARN



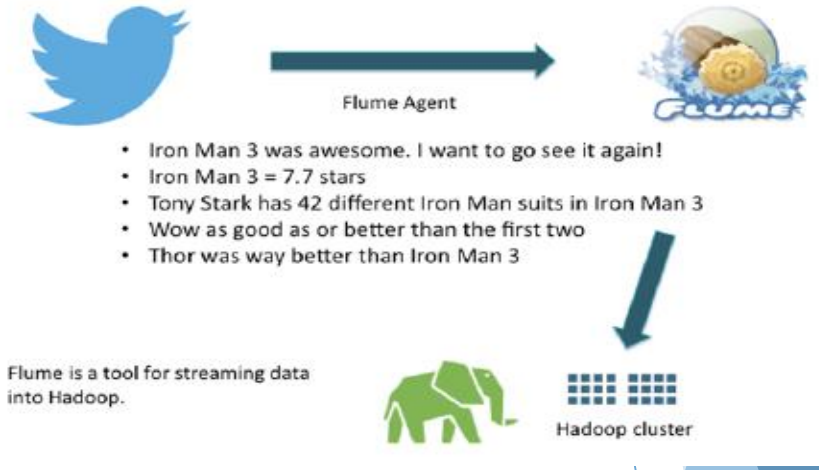
Use Cases

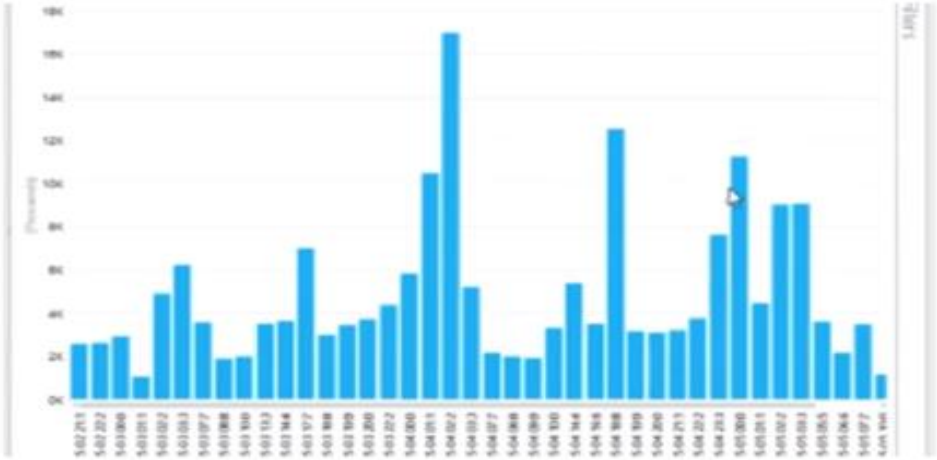
Sentiment Use Case

- Tracking the volume of tweets around the movie's launch

Sentiment Use Case

- Analyze customer sentiment on the days leading up to and following the release of the movie *Iron Man 3*.
- Questions to answer:
  - How did the public feel about the debut?
  - How might the sentiment data have been used to better promote the launch of the movie?

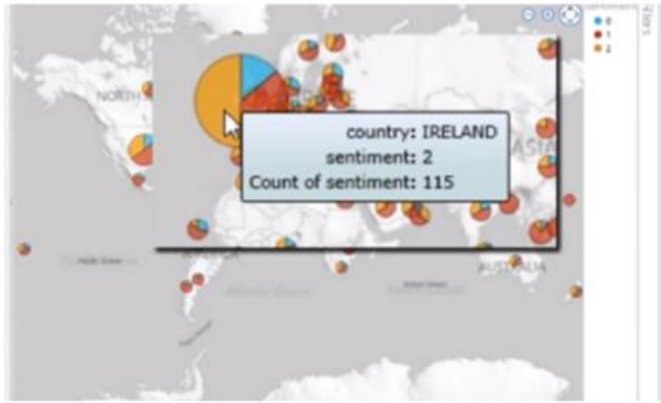




Notice a large spike in tweets around the Thursday midnight opening, and spikes around the Friday evening, Saturday afternoon and Saturday evening showings.

View Spikes in Tweet Volume

The sentiment of the tweets was graphed by country:

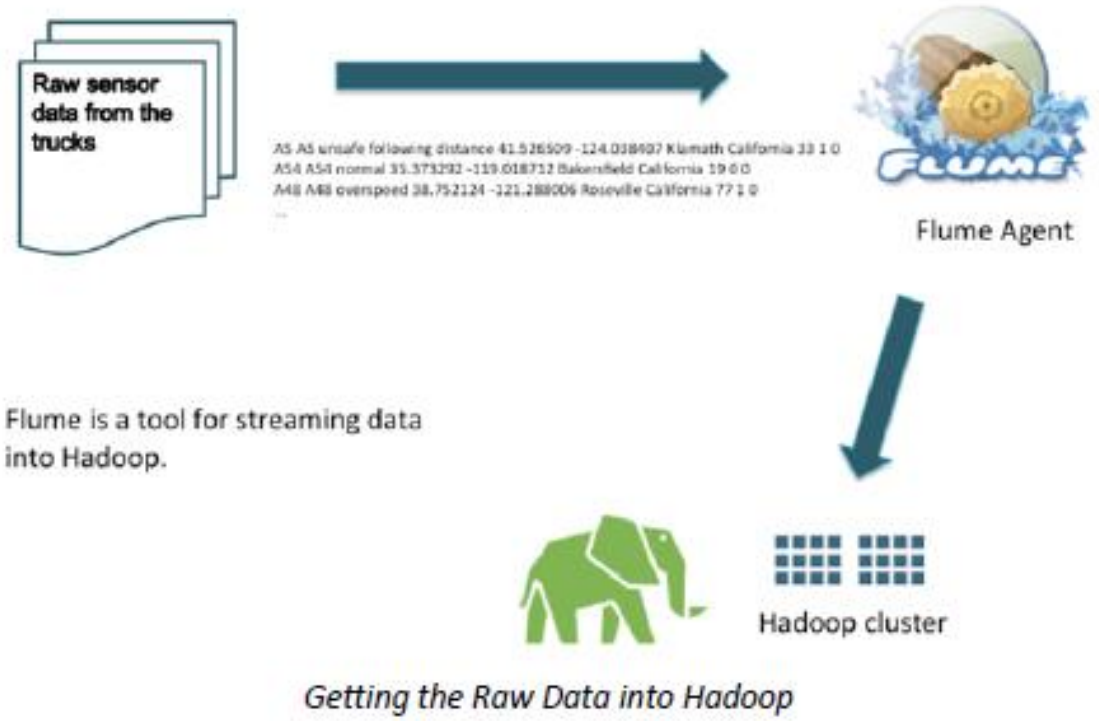


Viewing the tweets on a map shows the sentiment of the movie by country. For example, Ireland had 50% positive tweets, while 67% of tweets from Mexico were neutral.

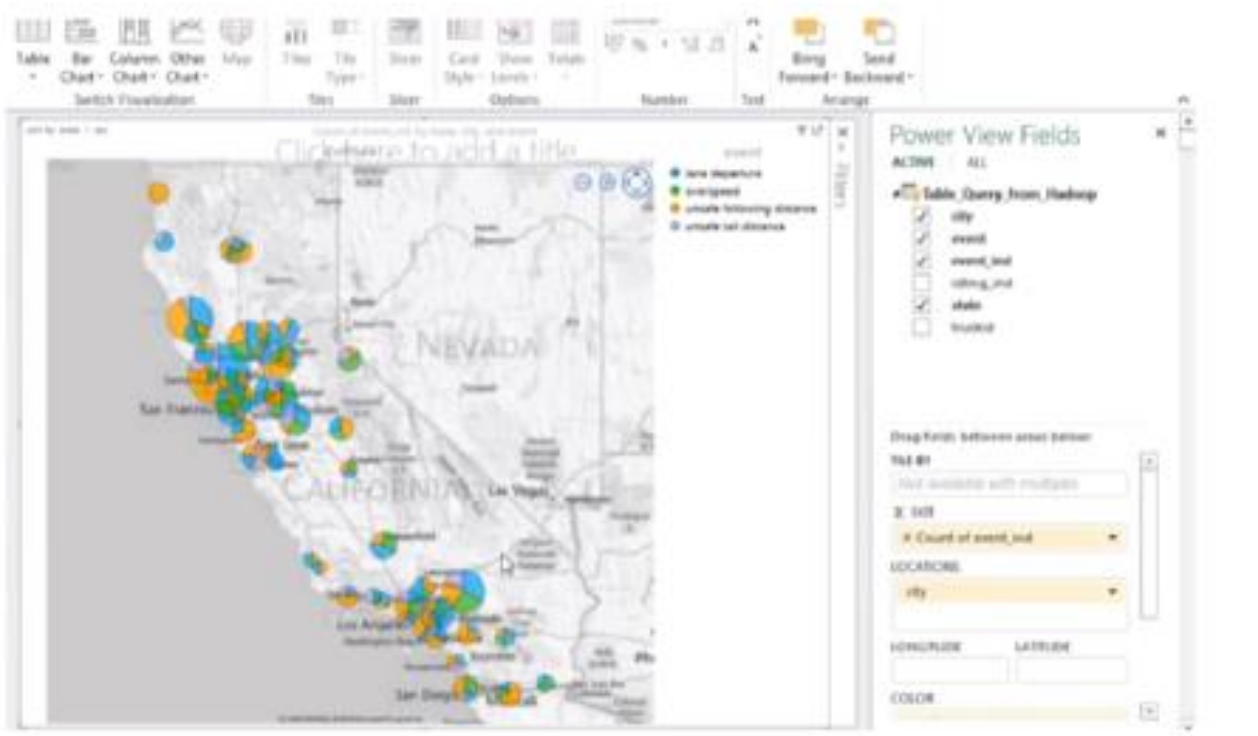
View Sentiment by Country

Geo location Use Case

- A trucking company can analyze geolocation data to reduce fuel costs and improve driver safety.



Getting the Raw Data into Hadoop



Risk Factors Viewed on a Map

Clickstream Use Case

- Amazon's use of real-time, item-based, collaborative filtering (IBCF) to fuel its 'Frequently bought together' and 'Customers who bought this item also bought' features
- Amazon generates about 20% more revenue via this method.

- LinkedIn and Facebook suggesting 'People you may know' or 'Companies you may want to follow'.

Thank You