Processing of Big Data

Nelson L. S. da Fonseca

IEEE ComSoc Summer Scool

Albuquerque, July 17-21, 2017

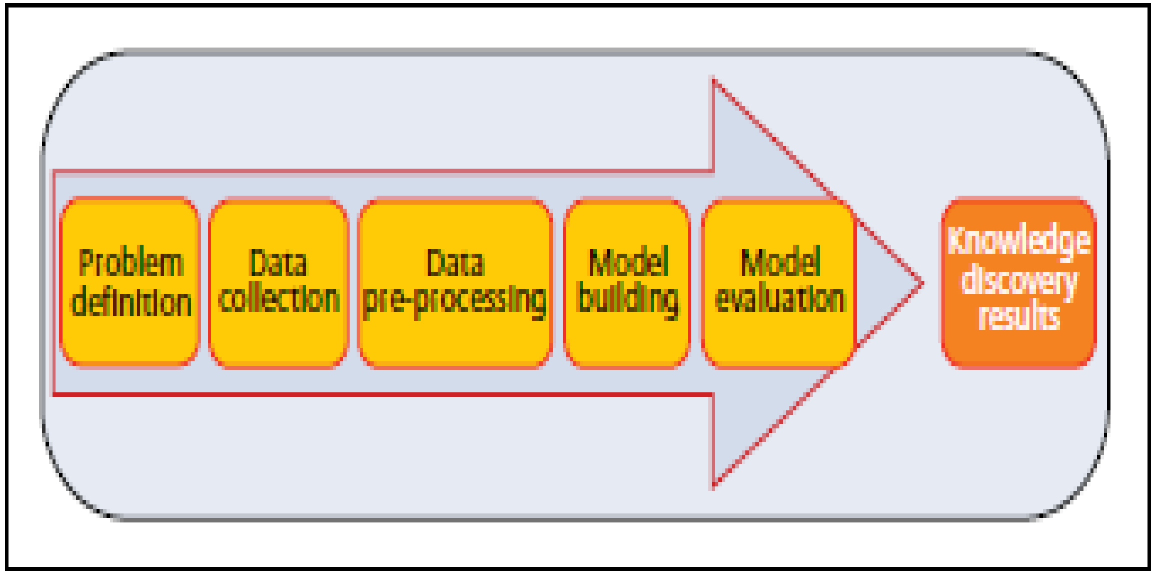
Acknowledgement

- Some slides in this set of slides were provided by EMC Corporation

- Some slides in this set were provided by Sandra Avila, UNICAMP

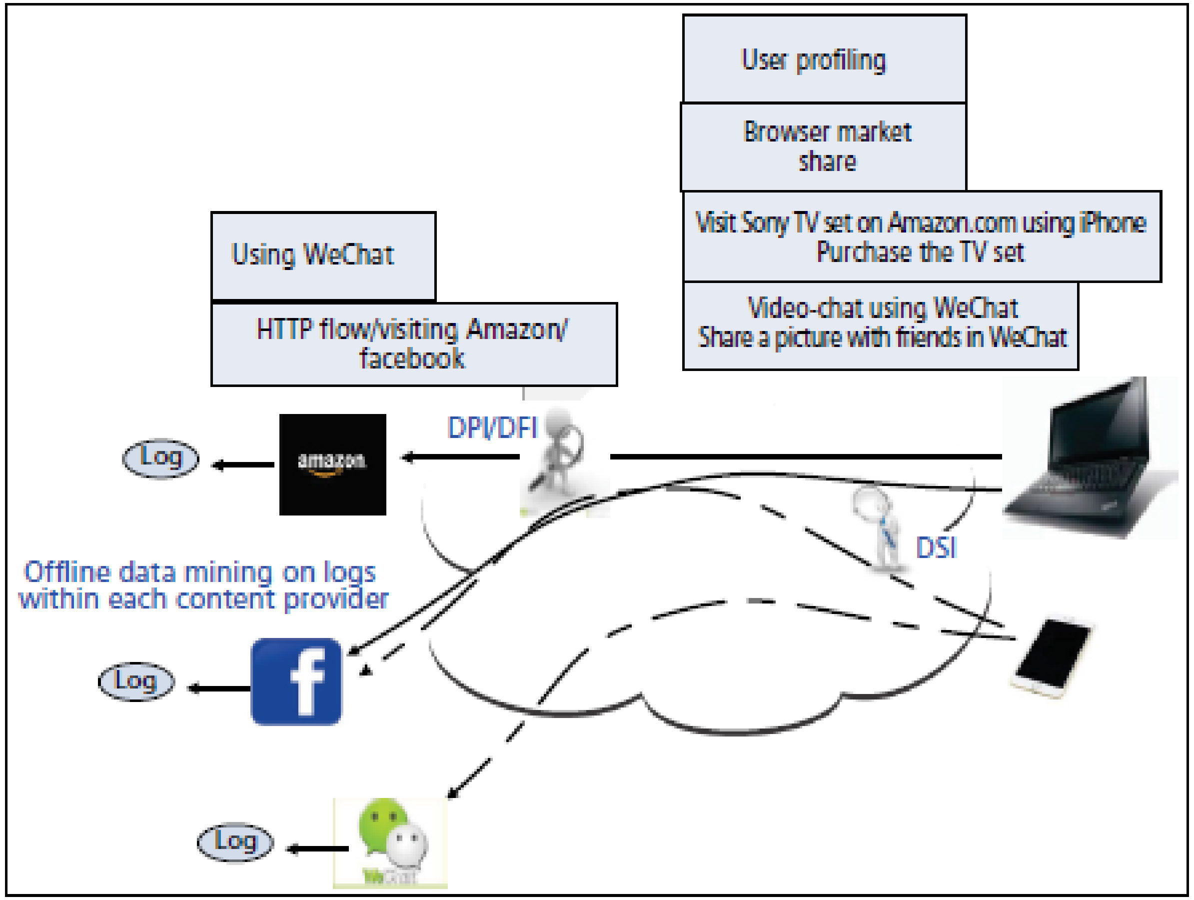
Data Mining

Data Mining Process



M. De Sanctis, I. Bisio, and G. Araniti, Data Mining Algorithms for Communication Networks Control: Concepts, Survey and Guidelines, IEEE Network, pp 24-29, Jan/Feb 2016

Data Collection



Features

$$
X = \begin{array}{c c}
 & \begin{array}{cccc} \mathit{feature1} & \mathit{feature2} & \cdots & \mathit{featureN} \end{array} \\
\begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_M \end{array} &
\begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,N} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{M,1} & x_{M,2} & \cdots & x_{M,N} \end{pmatrix}
\end{array}
$$

Protocol layer	Features
PHY	SNR, RSSI, LQI, CSI, channel variance, BER, channel coding rate, modulation index, number of subcarriers
MAC/LLC	MAC frame length, ARQ mode, FER, number of time slots, number of users
Network	Packet length, packet delay, packet delay jitter, inter-arrival time, network topology
Transport	Source port, destination port, PER, throughput, goodput
Application	Protocol type, bit rate of video coding, QoS level, QoE, SLA, applied tariff, service availability

Table 1. Features of communication networks at different protocol layers.

M. De Sanctis, I. Bisio, and G. Araniti, Data Mining Algorithms for Communication Networks Control: Concepts, Survey and Guidelines, IEEE Network, pp 24-29, Jan/Feb 2016

Features

Continuous

- Packet Error Rate (PER)
- Signal-to-noise ratio (SNR)
- Channel State Information(CSI)

Categorical

- Service Level Agreement (SLA)
- Quality of service (QoS)
- Network topology

Data Mining Methods

Descriptive

- Association Rule Mining
- Clustering (or Segmentation)
- Sequential Pattern Mining

Predictive

- Classification
- Regression
- Anomaly Detection

Map Reduce

What is MapReduce?

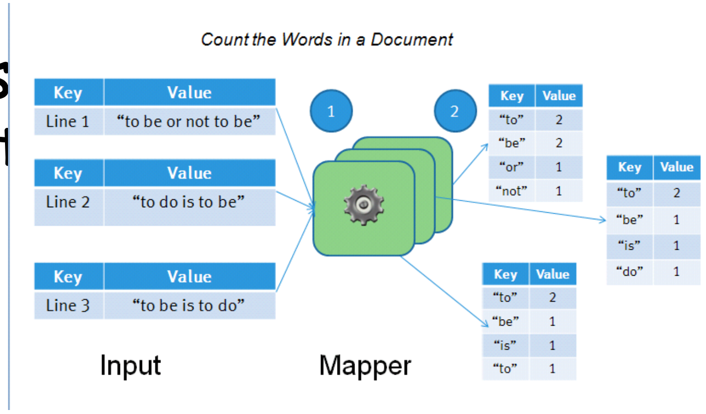
- A parallel programming model suitable for big data processing
  - Split data into distributable chunks (“shards”)
  - Define the steps to process those chunks
  - Run that process in parallel on the chunks

- Scalable by adding more machines to process chunks
  - Leverage commodity hardware to tackle big jobs

- The foundation for Hadoop
  - **MapReduce** is a **parallel programming model**
  - **Hadoop** is a **concrete platform that implements MapReduce**

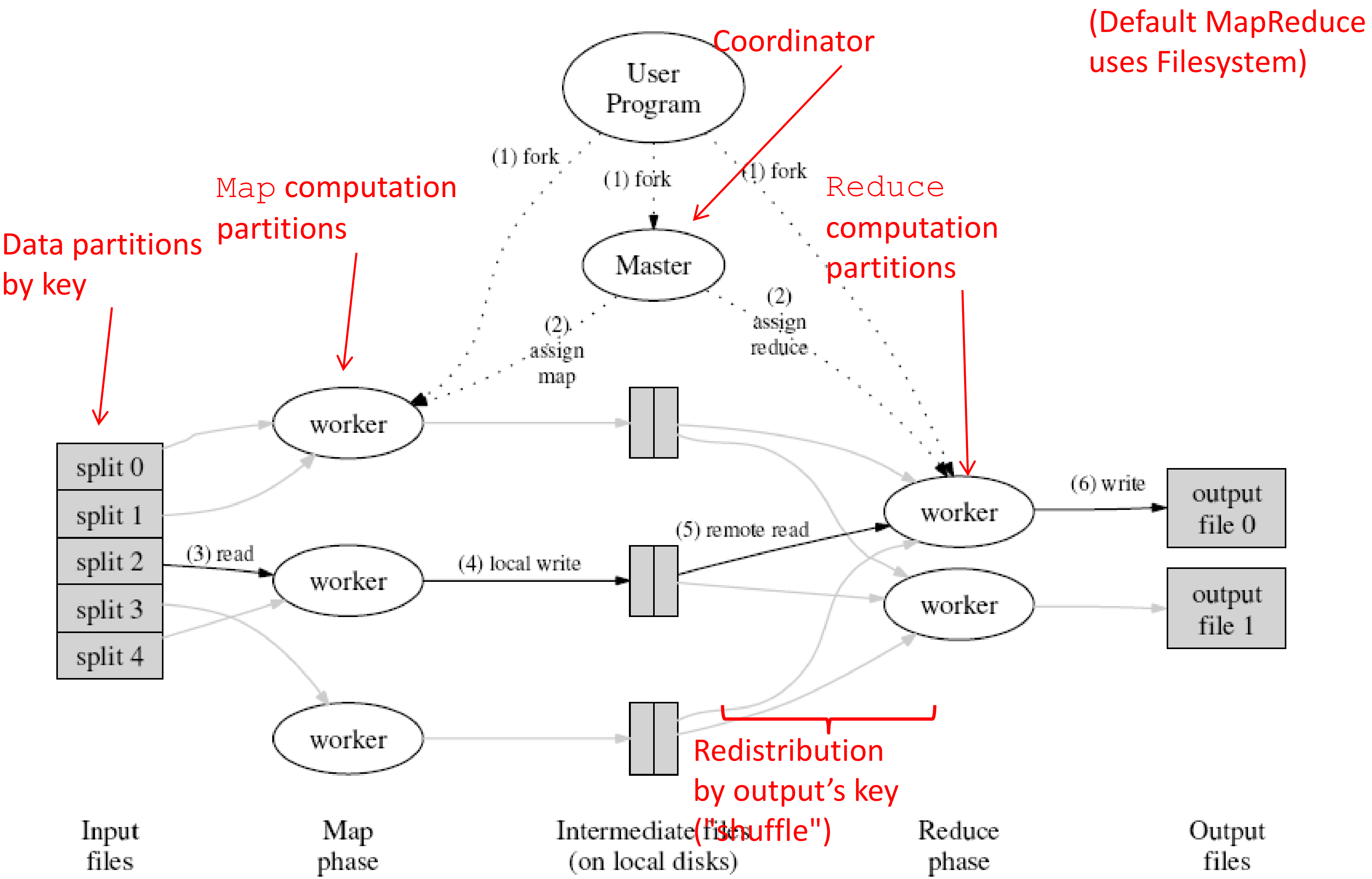
The Map part of MapReduce

- Transform
  - (Map) input values to output values: <k1,v1> → <k2,v2>
- Input – Key/Value Pairs
  - For instance, Key = line number, Value = text string
- Map Function
  - Steps to transform input pairs to output pairs
    - For example, count the different words in the input
- Output – Key/Value Pairs
  - For example, Key = <word>, Value = <count>
- Map output is the input to Reduce

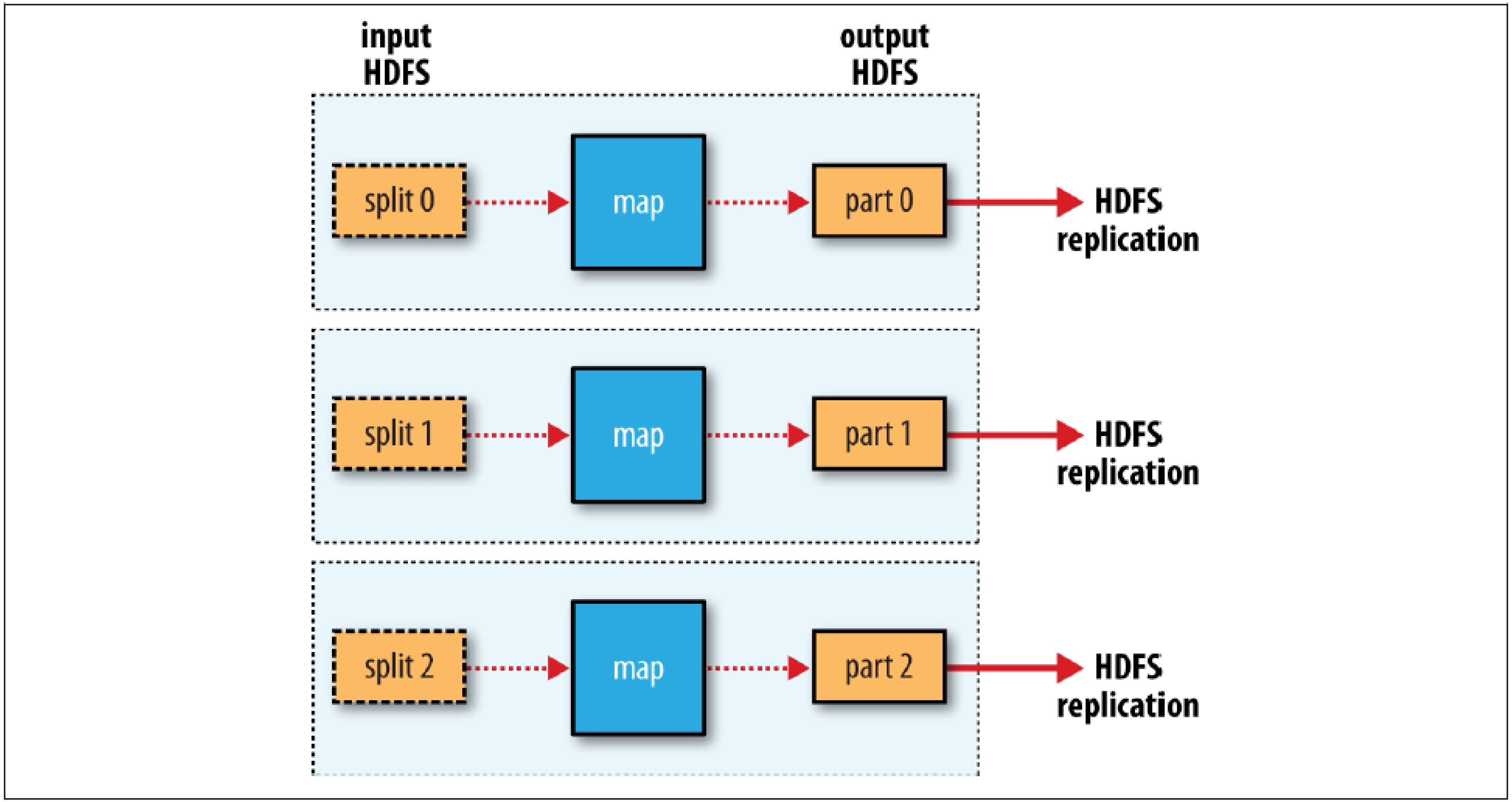


The Reduce Part of MapReduce

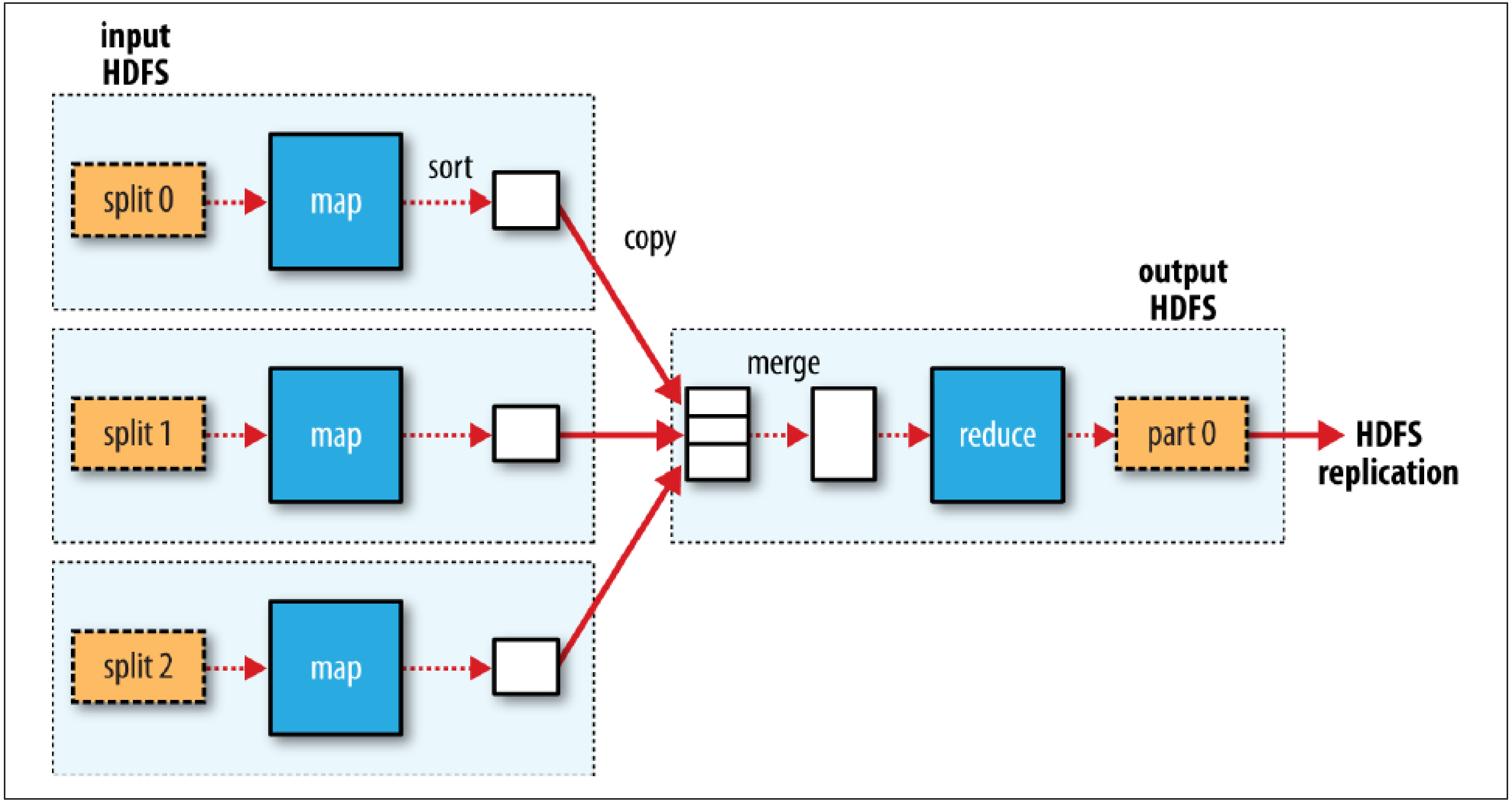
- Merge (Reduce) Values from the Map phase
  - Reduce is optional. Sometimes all the work is done in the Mapper
- Input
  - Values for a given Key from all the Mappers
- Reduce Function
  - Steps to combine (Sum?, Count?, Print?,...) the values
- Output
  - Print values?, load into a DB? send to the next MapReduce job?


(Default MapReduce uses Filesystem)
Coordinator
User Program
(1) fork
(1) fork
(1) fork
Master
Map computation partitions
Reduce computation partitions
Data partitions by key
(2) assign map
(2) assign reduce
worker
worker
worker
worker
worker
split 0
split 1
split 2
split 3
split 4
(3) read
(4) local write
(5) remote read
(6) write
output file 0
output file 1
Redistribution by output's key ("shuffle")
Input files
Map phase
Intermediate files (on local disks)
Reduce phase
Output files

MapReduce - 0 Reduce



MapReduce – 1 Reduce



MapReduce – 2 Reduces

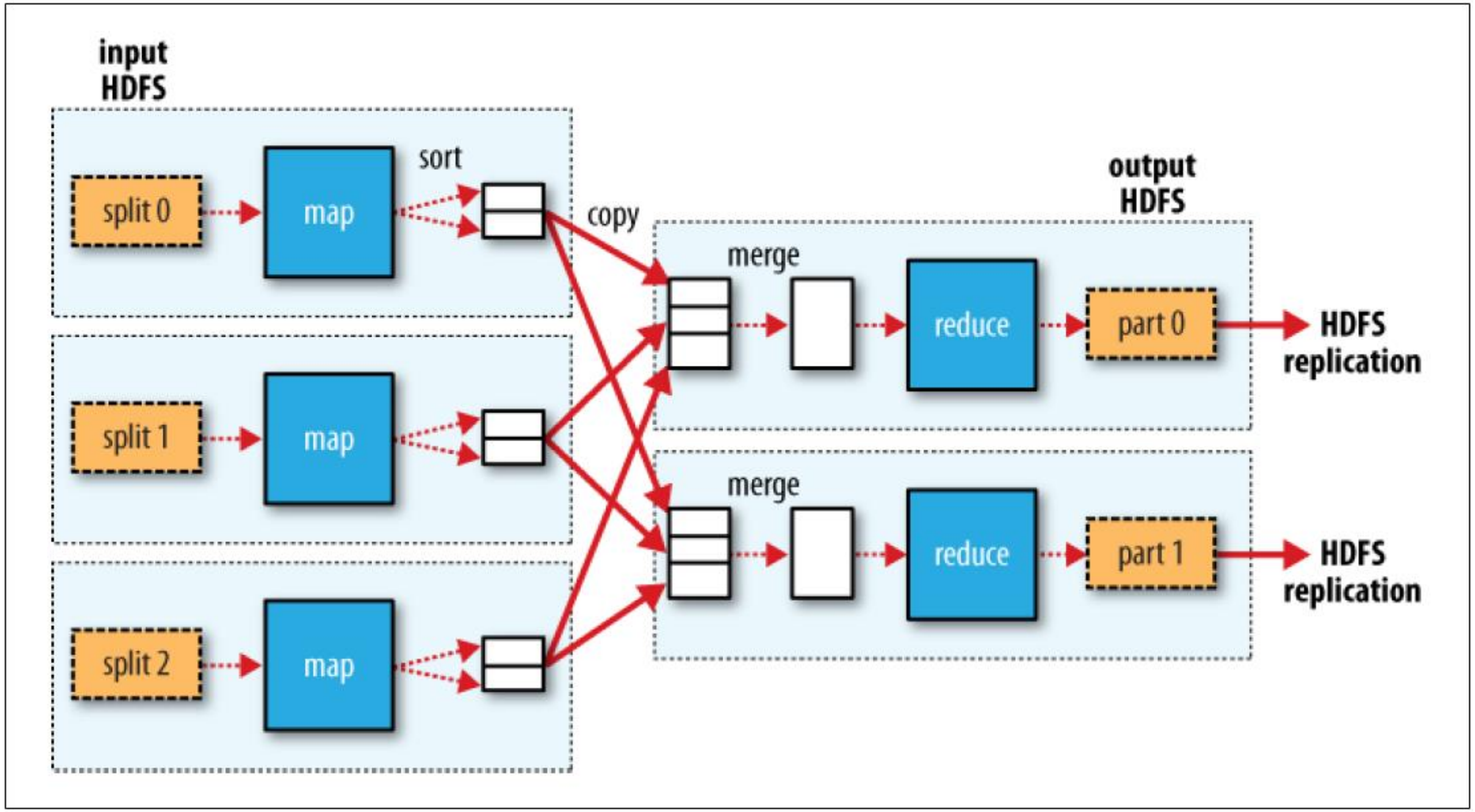
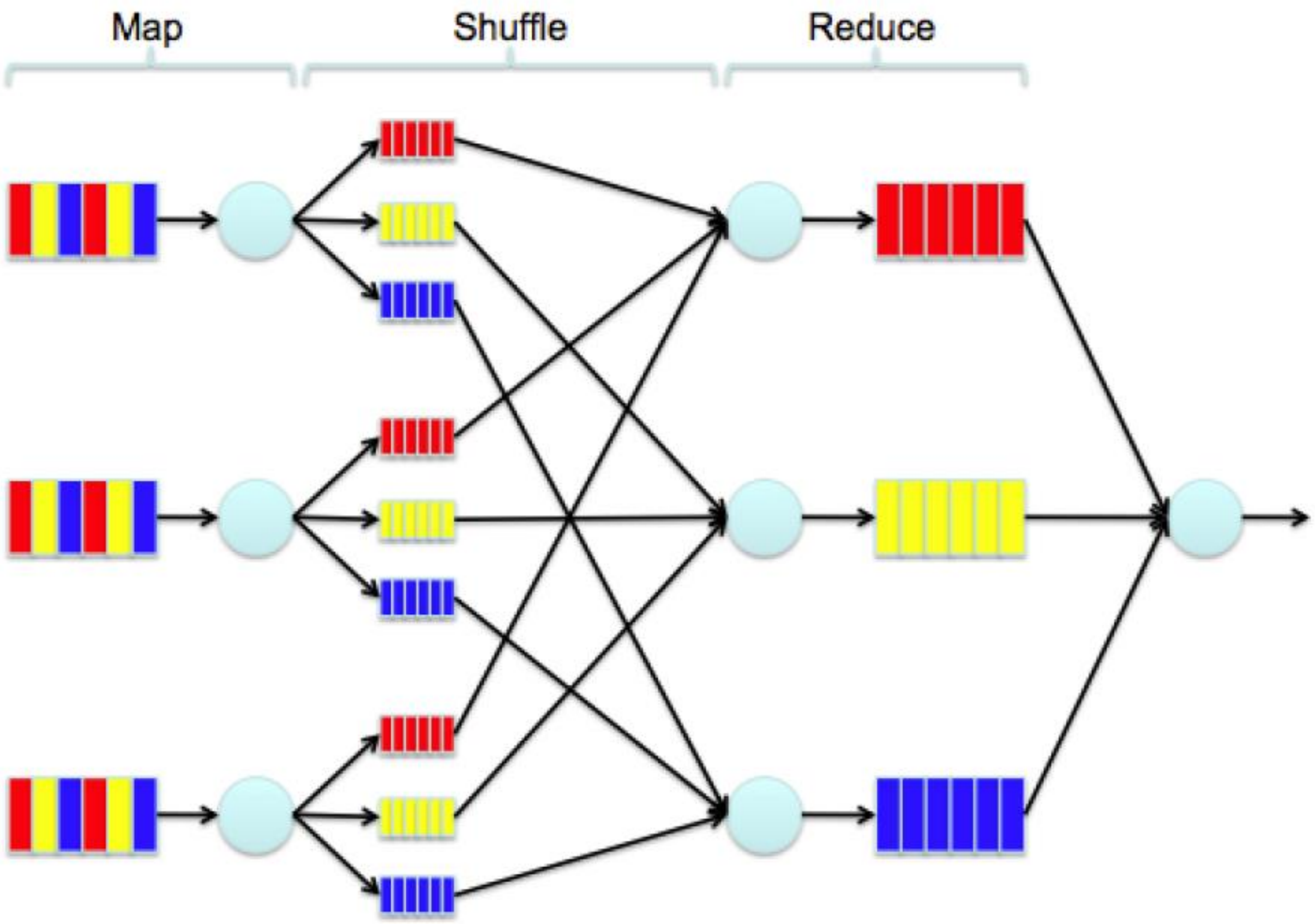
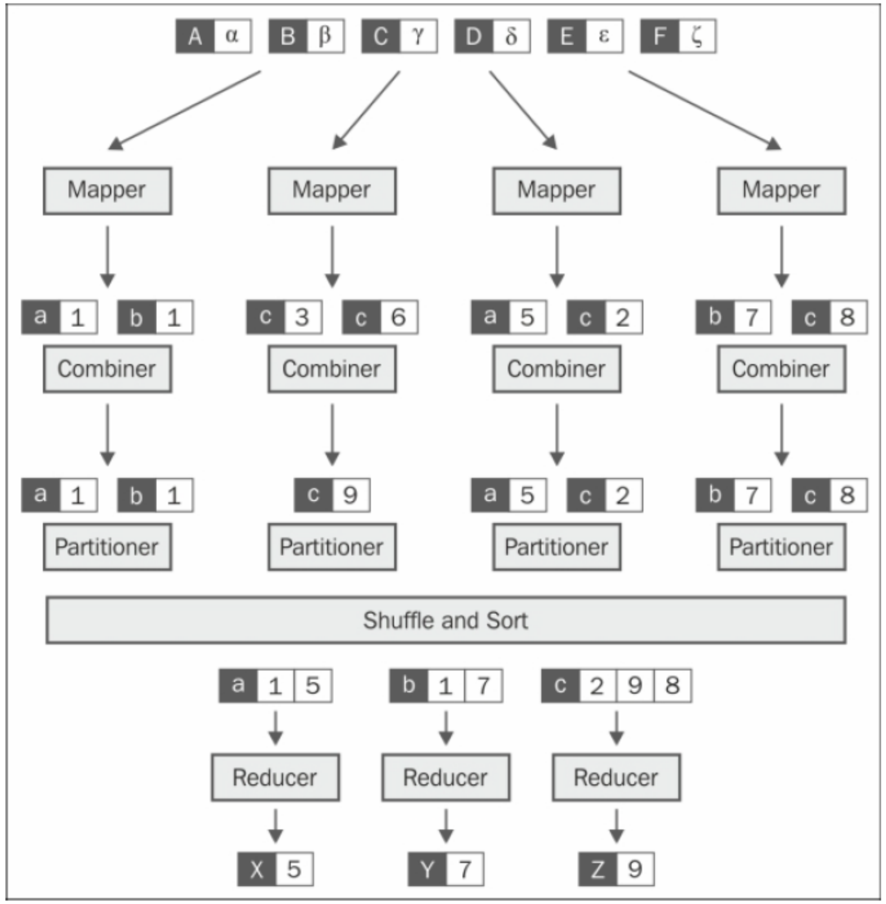


Figure 2-4. MapReduce data flow with multiple reduce tasks

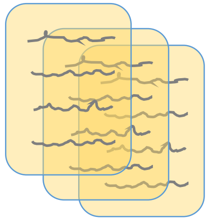
MapReduce - Shuffle



MapReduce - Combiner



Word Count

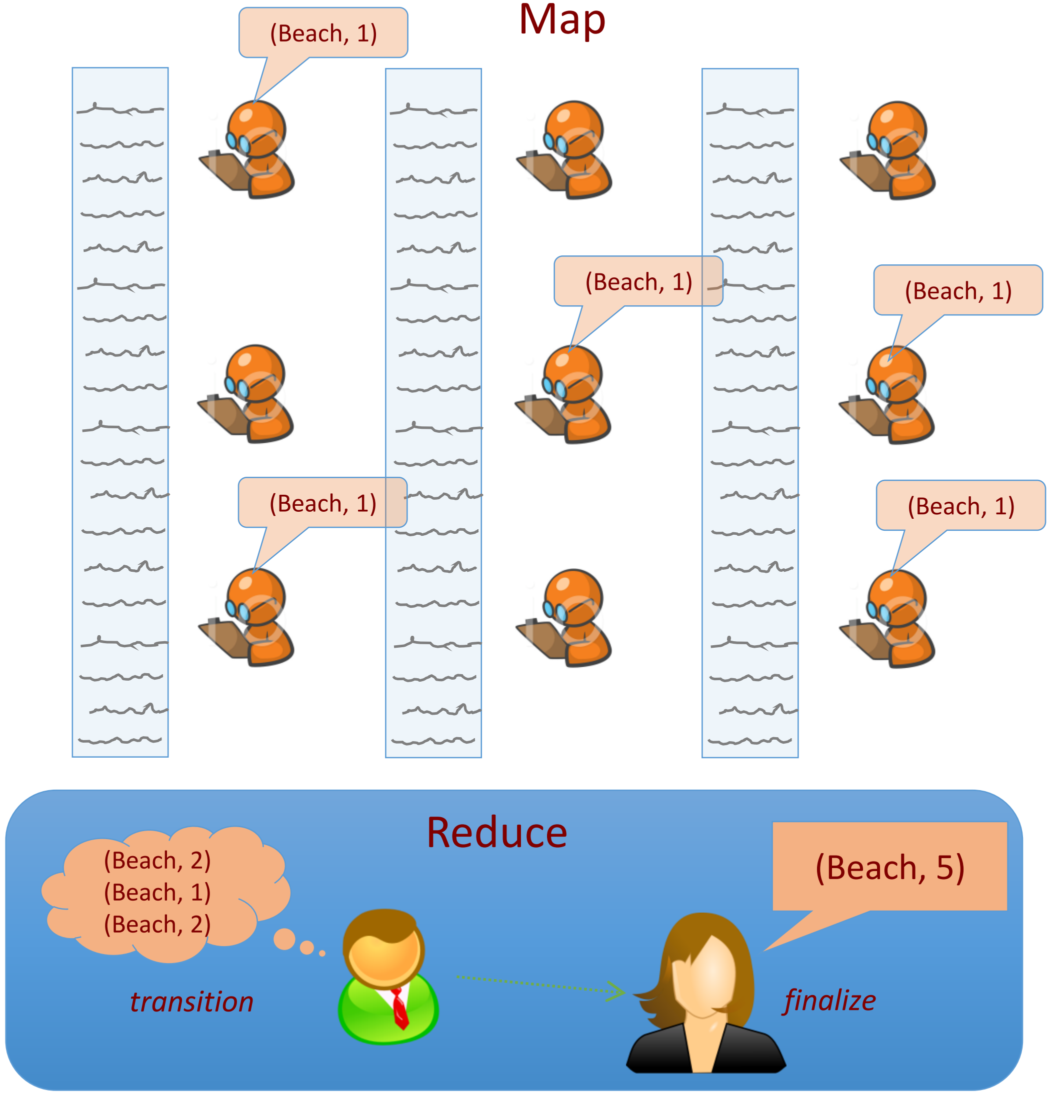


This is the "Hello World" of MapReduce

Distribute the text of millions of documents over hundreds of machines.

MAPPERS can be word-specific. They run through the stacks and shout "One!" every time they see the word "beach"

REDUCERS listen to all the Mappers and total the counts for each word.

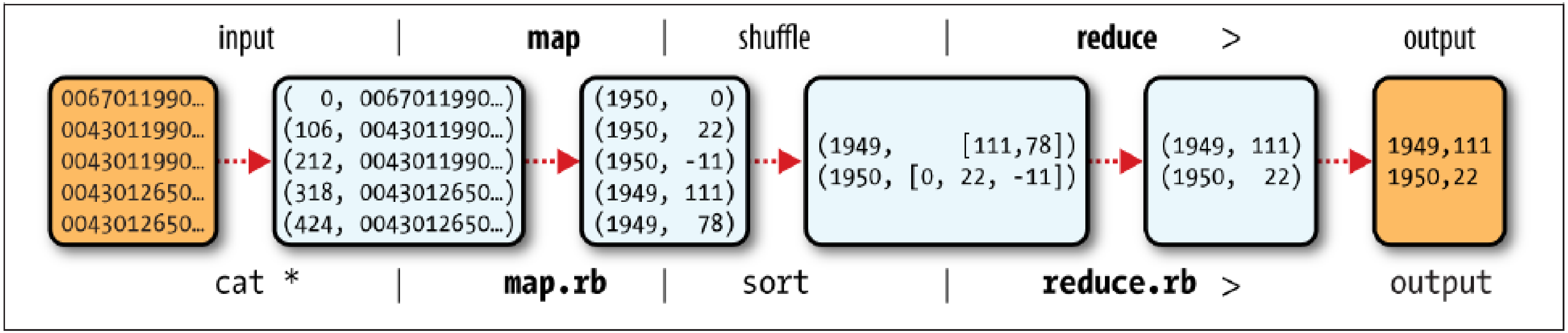


Word Counting

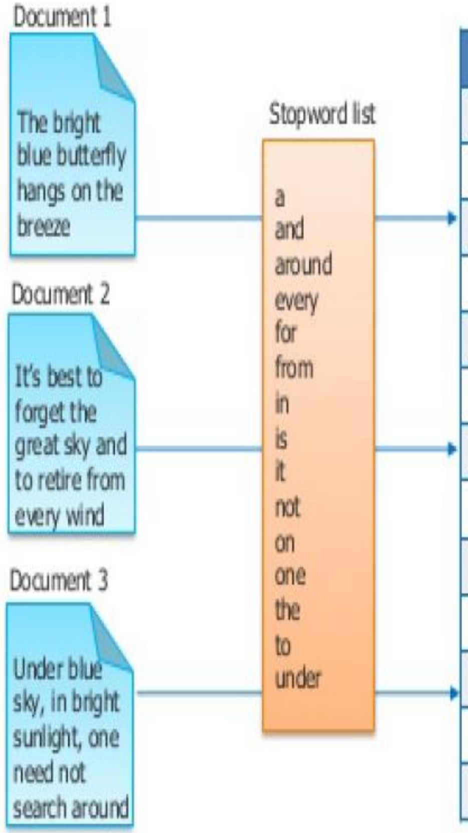
```
//Pseudo-code for "word counting"
map(String key, String value):
  // key: document name,
  // value: document contents
    for each word w in value:
      EmitIntermediate(w, "1");

reduce(String key, Iterator values):
  // key: a word
  // values: a list of counts
    int word_count = 0;
    for each v in values:
      word_count += ParseInt(v);
    Emit(key, AsString(word_count));
```

MapReduce

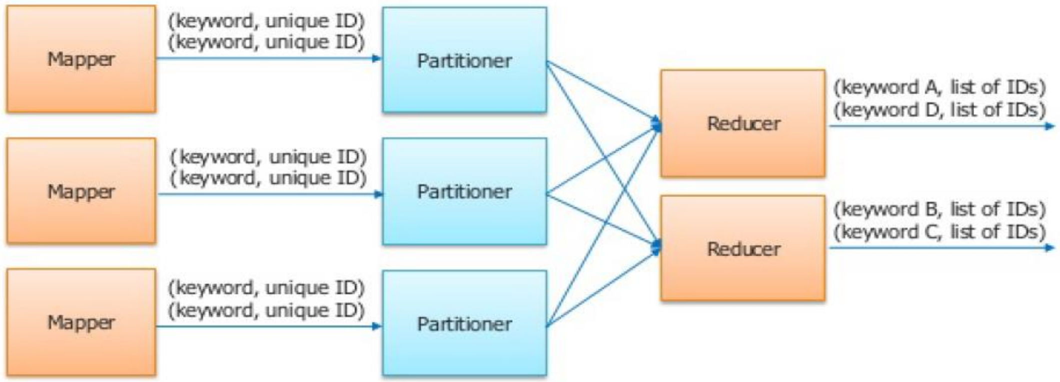


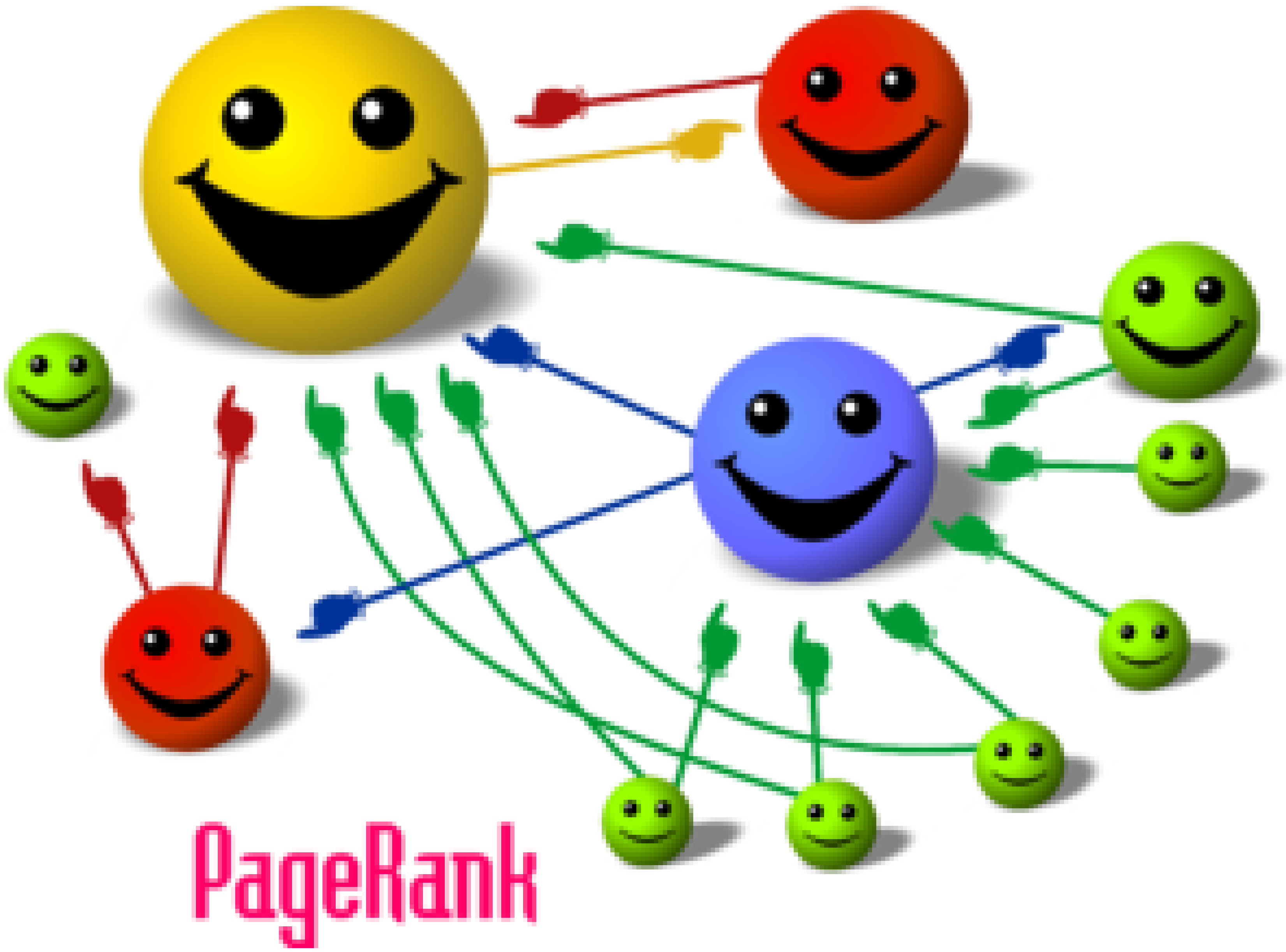
Fonte: Hadoop—The Definitive Guide, *Tom White*



Inverted Index

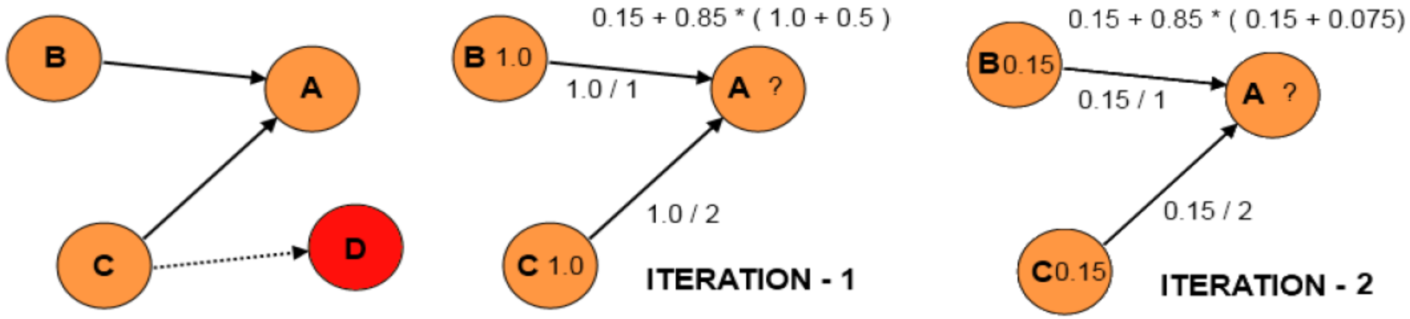
ID	Term	Document
1	best	2
2	blue	1,3
3	bright	1,3
4	butterfly	1
5	breeze	1
6	forget	2
7	great	2
8	hangs	1
9	need	3
10	retire	2
11	search	3
12	sky	2,3
13	wind	2




PageRank

PageRank

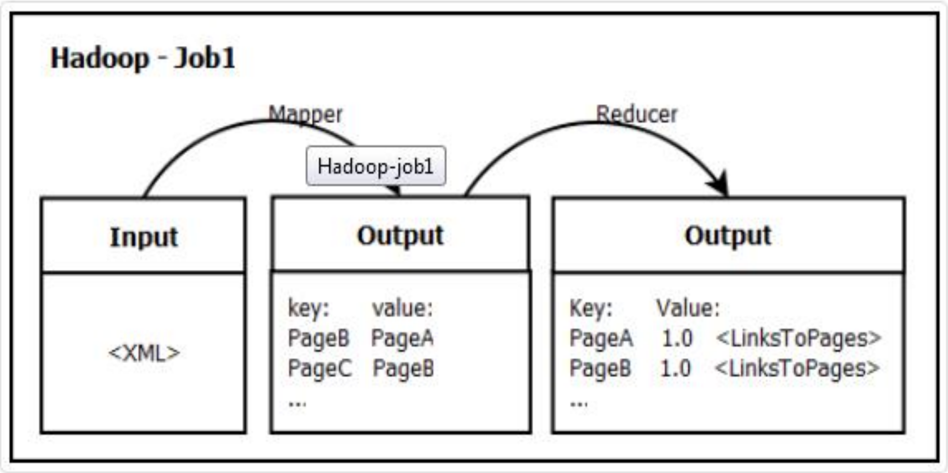
$$PR(x) = (1 - d) + d \sum_{i=1}^{N} PR(t_i)/L(t_i)$$

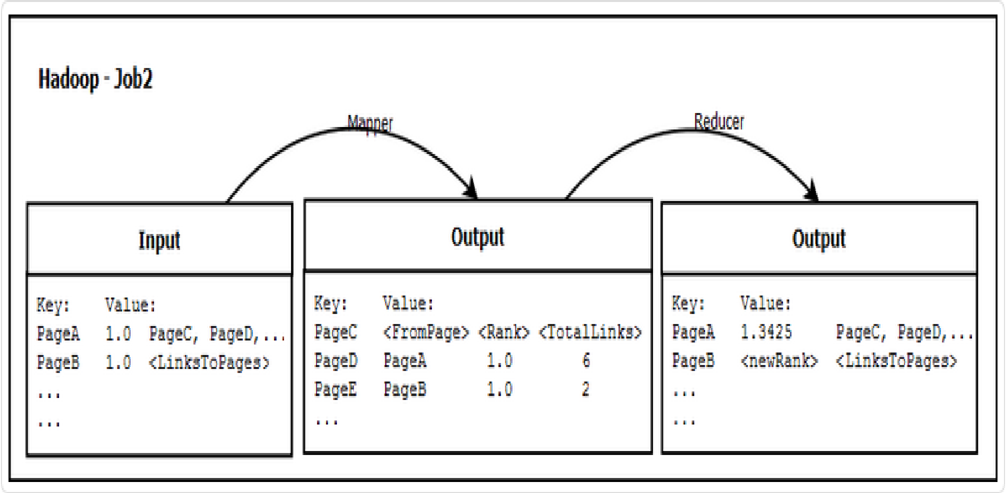


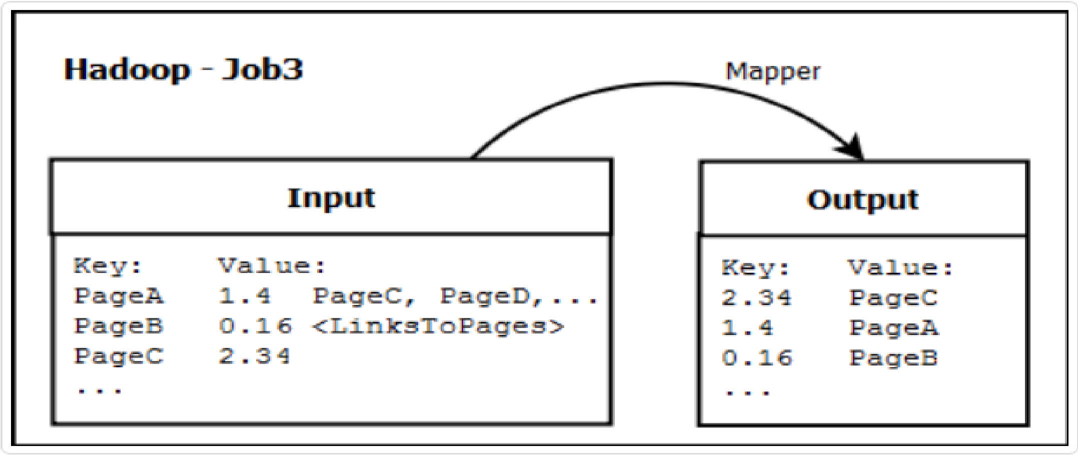
PR(A): 1.0	PR(A): 1.425	PR(A): 0.34125
PR(B): 1.0	PR(B): 0.15	PR(B): 0.15
PR(C): 1.0	PR(C): 0.15	PR(C): 0.15

http://blog.xebia.com/wiki-pagerank-with-hadoop/

PageRank


Hadoop - Job1
Mapper
Reducer
Hadoop-job1
Input
<XML>
Output
key: value:
PageB PageA
PageC PageB
...
Output
Key: Value:
PageA 1.0 <LinksToPages>
PageB 1.0 <LinksToPages>
...


Hadoop - Job2
Mapper
Reducer
Input
Key: Value:
PageA 1.0 PageC, PageD,...
PageB 1.0 <LinksToPages>
...
...
Output
Key: Value:
PageC <FromPage> <Rank> <TotalLinks>
PageD PageA 1.0 6
PageE PageB 1.0 2
...
Output
Key: Value:
PageA 1.3425 PageC, PageD,...
PageB <newRank> <LinksToPages>
...
...


Hadoop - Job3
Mapper
Input
Key: Value:
PageA 1.4 PageC, PageD,...
PageB 0.16 <LinksToPages>
PageC 2.34
...
Output
Key: Value:
2.34 PageC
1.4 PageA
0.16 PageB
...

Example: UFOs Attack

July 15th, 2010. Raytown, Missouri

When I fist noticed it, I wanted to freak out. There it was an object floating in on a direct path, It didn't move side to side or volley up and down. It moved as if though it had a mission or purpose. I was nervous, and scared, So afraid in fact that I could feel my knees buckling. I guess because I didn't know what to expect and I wanted to act non aggressive. I though that I was either going to be taken, blasted into nothing, or…

Q: What is the witness describing?

A: An encounter with a UFO.

Q: What is the emotional state of the witness?

A: Frightened, ready to flee.

Source: http://www.infochimps.com/datasets/60000-documented-ufo-sightings-with-text-descriptions-and-metada

Example: UFOs Attack

If we really are on the cusp of a major alien invasion, eyewitness testimony is the key to our survival as a species.

Strangely, the computer finds this account ***unreliable!***

When I ***fist*** *noticed it, I wanted to freak out. Th… …oating in on a di…ct path, It* ***didn't*** *move side to side or volley up and down. It … if though it had a mission or purpose. I was nervous, and scared,* ***So*** *…fact that I could feel my knees buckling. I guess because I didn't know what to expect and I wanted to act non agg… …t I was either goi… …hing, or…*

Machine error

Typo

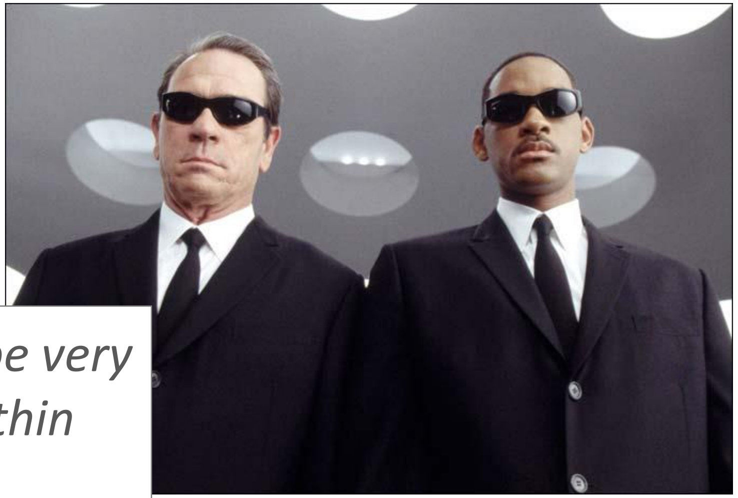
Turn of phrase

Ambiguous meaning

“UFO” keyword missing

Source: http://www.infochimps.com/datasets/60000-documented-ufo-sightings-with-text-descriptions-and-metada

Example: UFOs Attack



Investigators need to…

Search

for keywords and phrases, but your topic may be very complicated or keywords may be misspelled within the document

Manage

document meta-data like time, location and author. Later retrieval may be key to identifying this meta-data early, and the document may be amenable to structure.

Understand

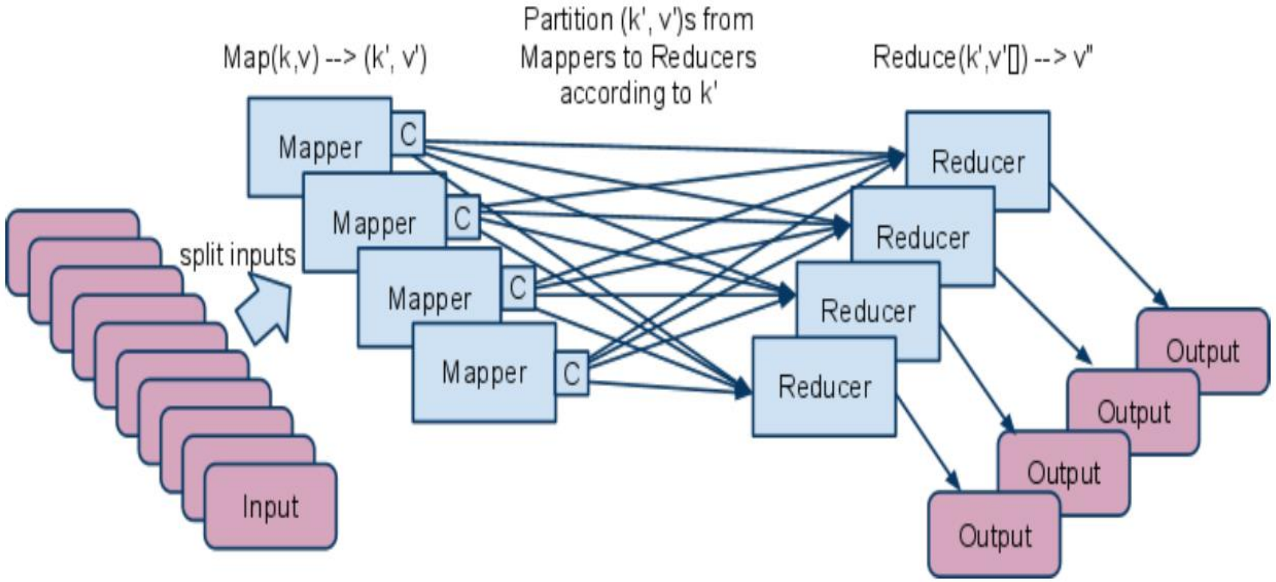
content via sentiment analysis, custom dictionaries, natural language processing, clustering, classification and good ol' domain expertise.

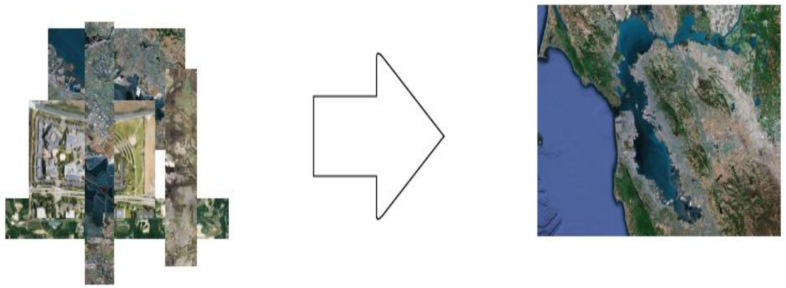
…with computer-aided text mining

Map Reduce

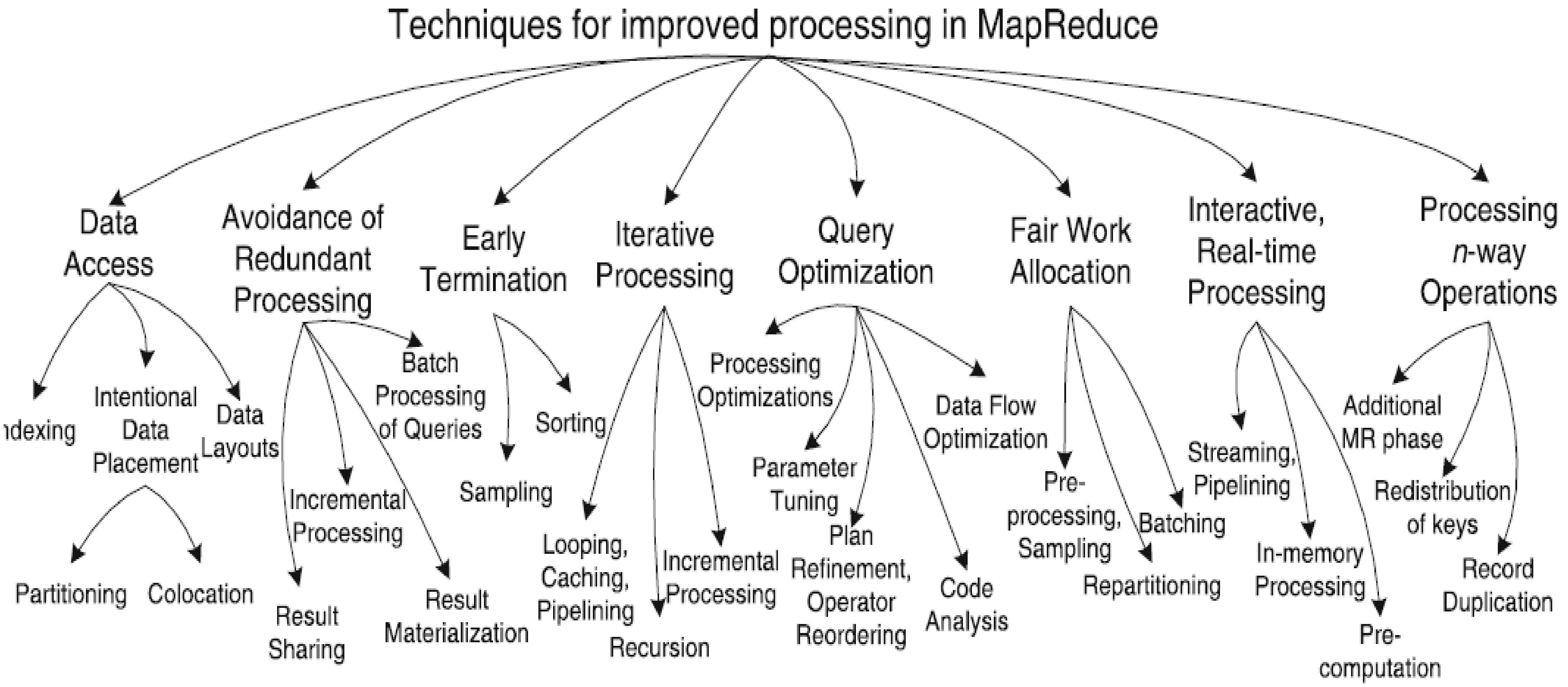
Weakness	Technique
Access to input data	Indexing and data layouts
High communication cost	Partitioning and colocation
Redundant and wasteful processing	Result sharing, batch processing of queries and incremental processing
Recomputation	Materialization
Lack of early termination	Sampling and sorting
Lack of iteration	Loop-aware processing, caching, pipelining, recursion, incremental processing
Quick retrieval of approximate results	Data summarization and sampling
Load balancing	Pre-processing, approximation of the data distribution and repartitioning
Lack of interactive or real-time processing	In-memory processing, pipelining, streaming and pre-computation
Lack of support for n-way operations	Additional MR phase(s), re-distribution of keys and record duplication

Map-Reduce





Facebook Trace analysis: 30% to 50% of running time took up by communication phase



Techniques for improved processing in MapReduce
Data Access
Avoidance of Redundant Processing
Early Termination
Iterative Processing
Query Optimization
Fair Work Allocation
Interactive, Real-time Processing
Processing *n*-way Operations
ndexing
Intentional Data Placement
Data Layouts
Partitioning
Colocation
Batch Processing of Queries
Incremental Processing
Result Sharing
Result Materialization
Sorting
Sampling
Looping, Caching, Pipelining
Recursion
Incremental Processing
Processing Optimizations
Parameter Tuning
Plan Refinement, Operator Reordering
Code Analysis
Data Flow Optimization
Pre-processing, Sampling
Repartitioning
Batching
Streaming, Pipelining
In-memory Processing
Pre-computation
Additional MR phase
Redistribution of keys
Record Duplication

Big Data Processing Environment

What is …………….hadoop?

People use “Hadoop” to mean one of four things:

(ideas)	(actual Hadoop)
> MapReduce paradigm.	> Java Classes for HDFS types and MapReduce job management.
> Massive unstructured data storage on commodity hardware.	> HDFS: The Hadoop distributed file system.

With Hadoop, you can do MapReduce jobs quickly and efficiently.

https://hadoop.apache.org/

What do we Mean by Hadoop

- A framework for performing big data analytics
  - An implementation of the MapReduce paradigm
  - Hadoop glues the storage and analytics together and provides reliability, scalability, and management

Two Main Components

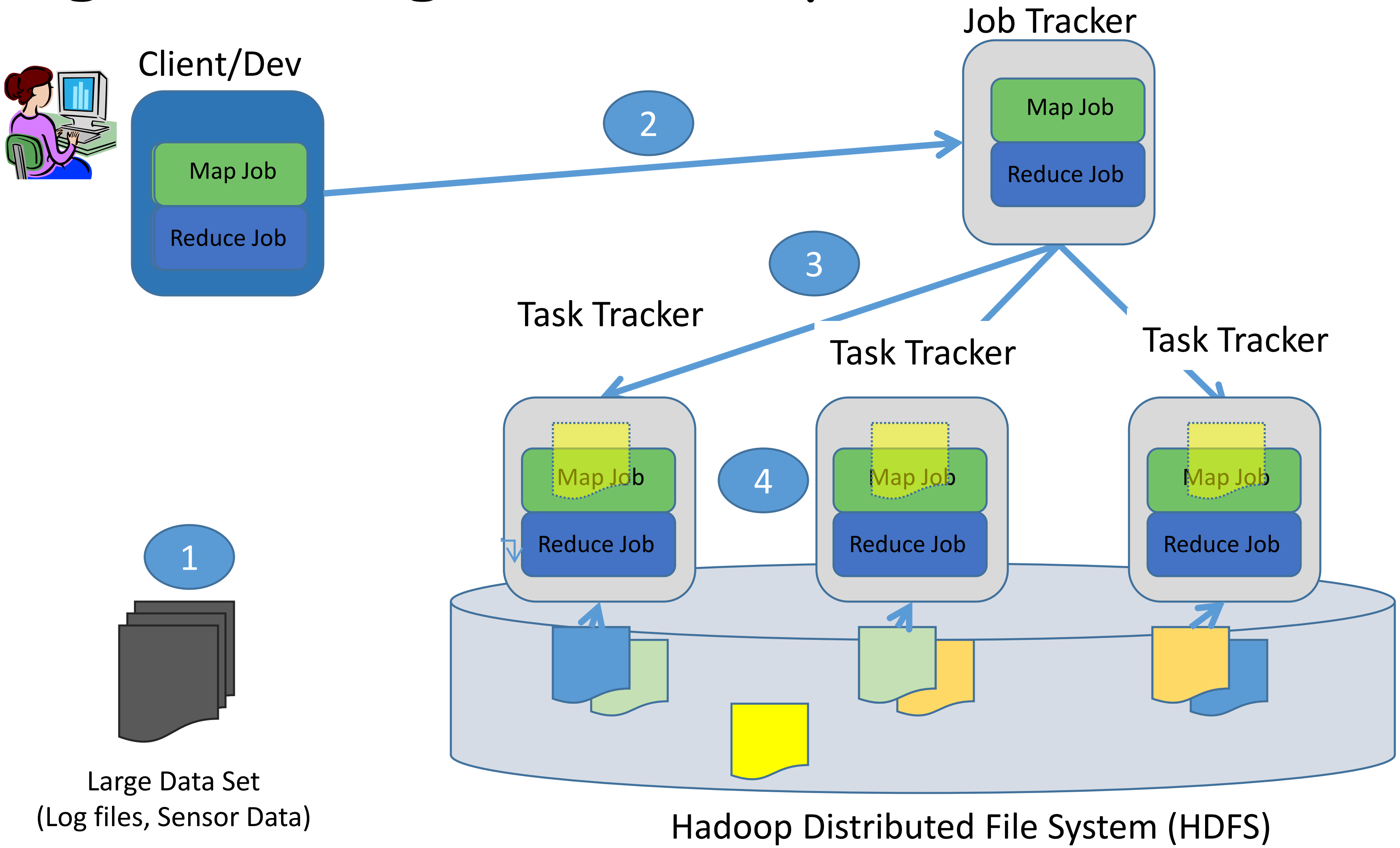
Storage (Big Data)

- HDFS – Hadoop Distributed File System
- Reliable, redundant, distributed file system optimized for large files

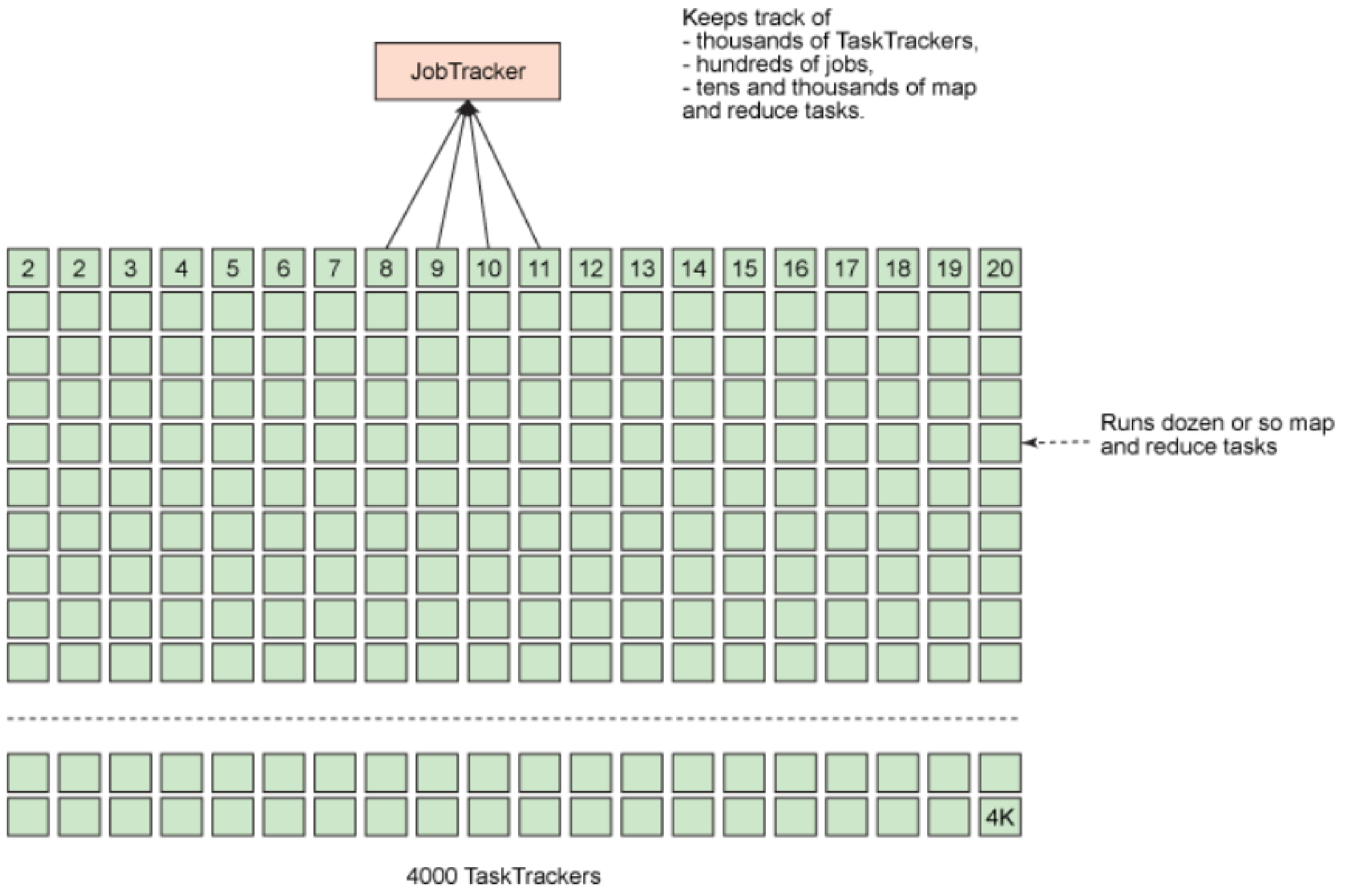
MapReduce (Analytics)

- Programming model for processing sets of data
- Mapping inputs to outputs and reducing the output of multiple Mappers to one (or a few) answer(s)

Putting it all Together: MapReduce and HDFS

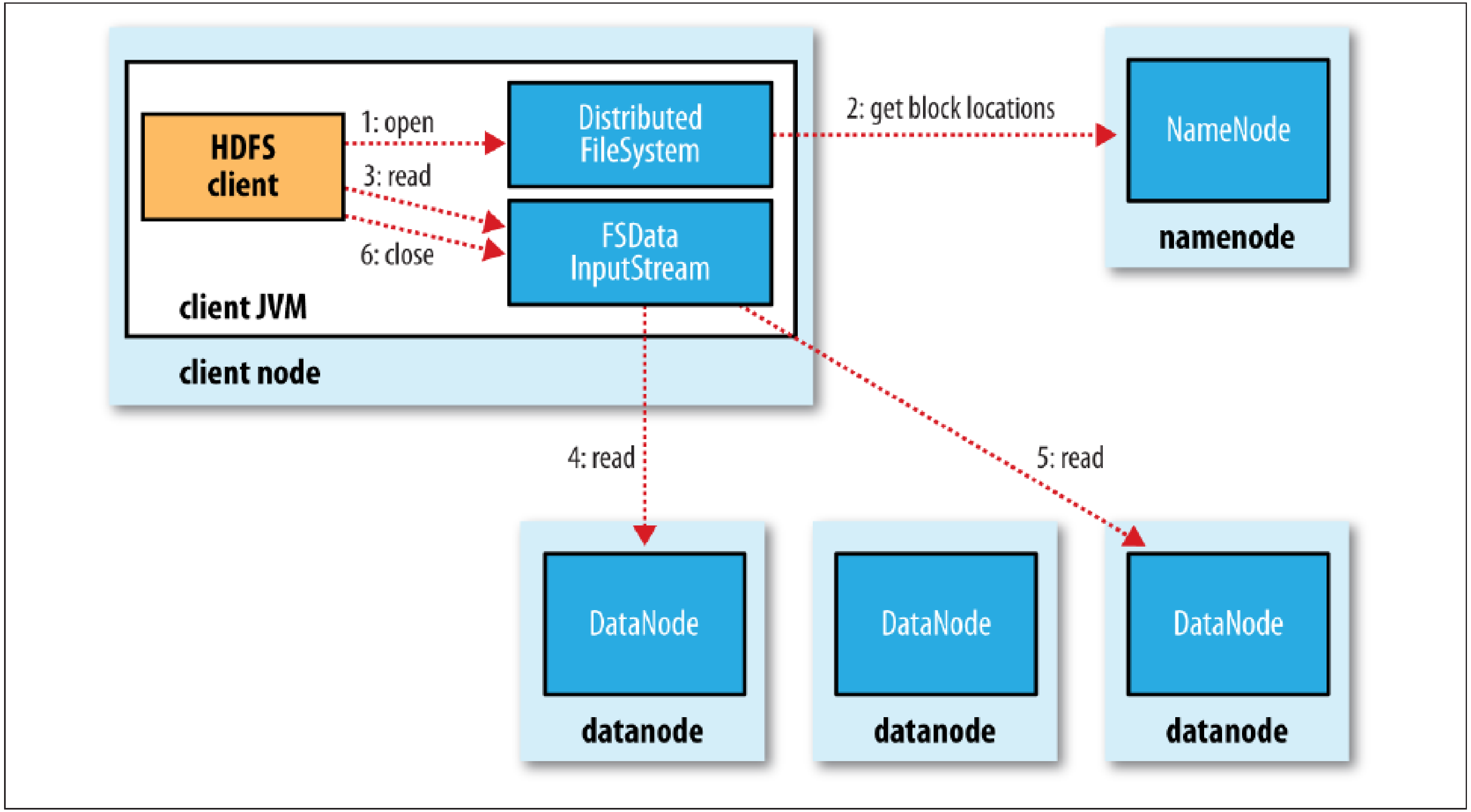


Hadoop

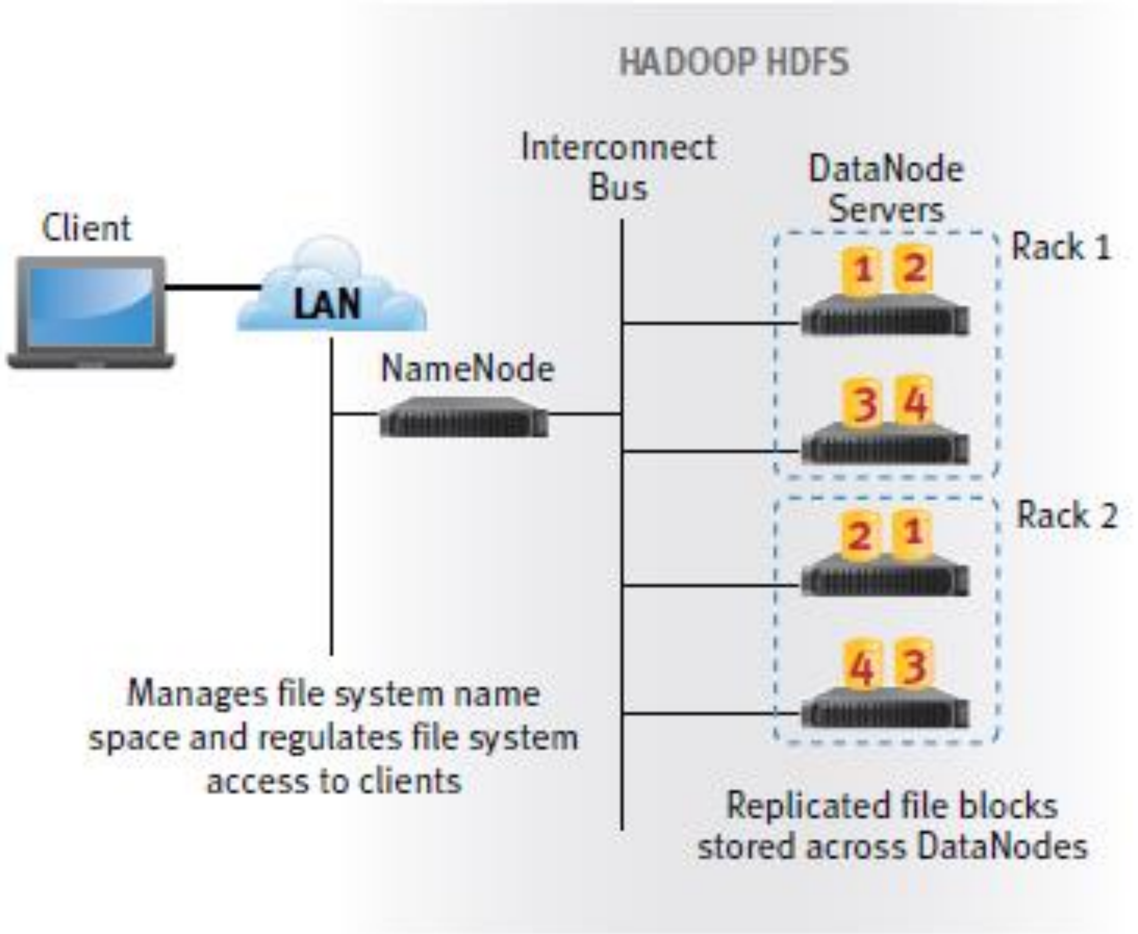


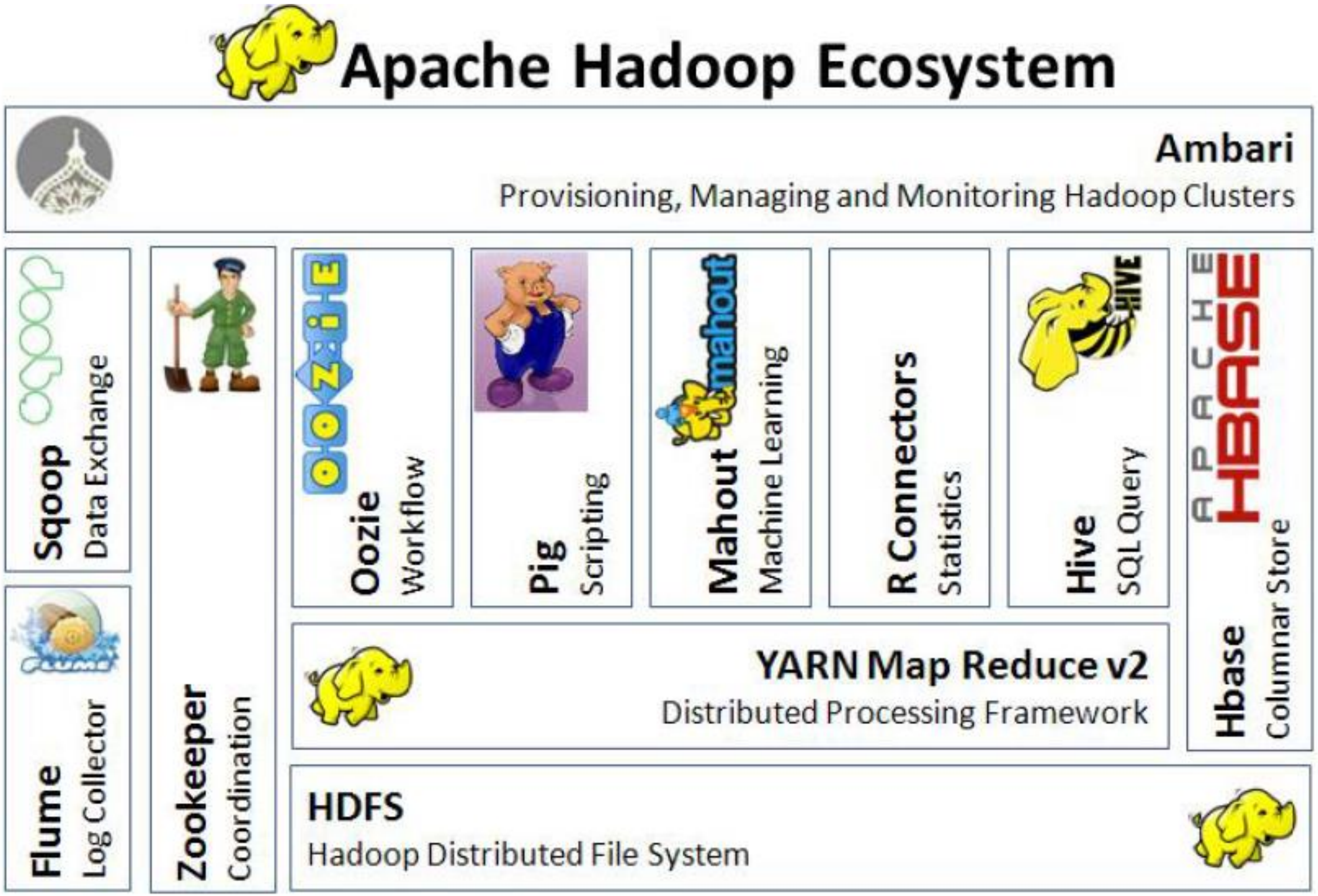
Keeps track of
- thousands of TaskTrackers,
- hundreds of jobs,
- tens and thousands of map
and reduce tasks.
JobTracker
2
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
4K
Runs dozen or so map
and reduce tasks
4000 TaskTrackers

HDFS


HDFS client
1: open
3: read
6: close
Distributed FileSystem
FSData InputStream
client JVM
client node
2: get block locations
NameNode
namenode
4: read
5: read
DataNode
datanode
DataNode
datanode
DataNode
datanode

HDFS


HADOOP HDFS
Client
LAN
Interconnect Bus
DataNode Servers
Rack 1
1 2
3 4
Rack 2
2 1
4 3
NameNode
Manages file system name space and regulates file system access to clients
Replicated file blocks stored across DataNodes



http://thebigdatablog.weebly.com/blog/the-hadoop-ecosystem-overview

Which Interface Should You Choose?

Pig

- Replacement for MapReduce Java coding
- When need exists to customize part of the processing phases (UDF)

Hive

- Use when SQL skills are available
- Customize part of the processing via UDFs

HBase

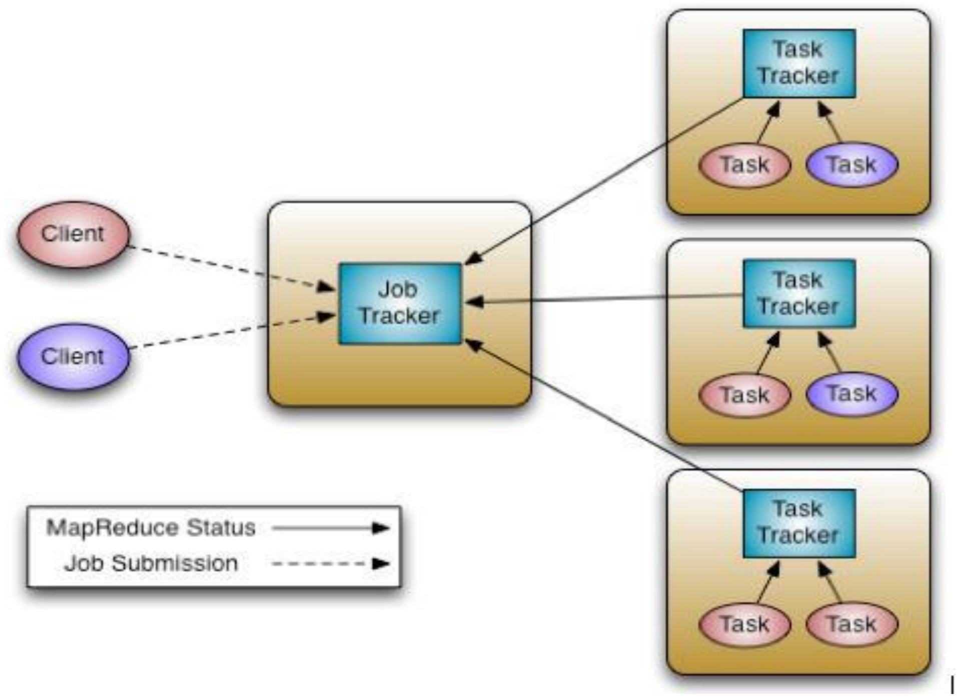
- Use when random queries and partial processing is required, or when specific file layouts are needed

Hadoop v2

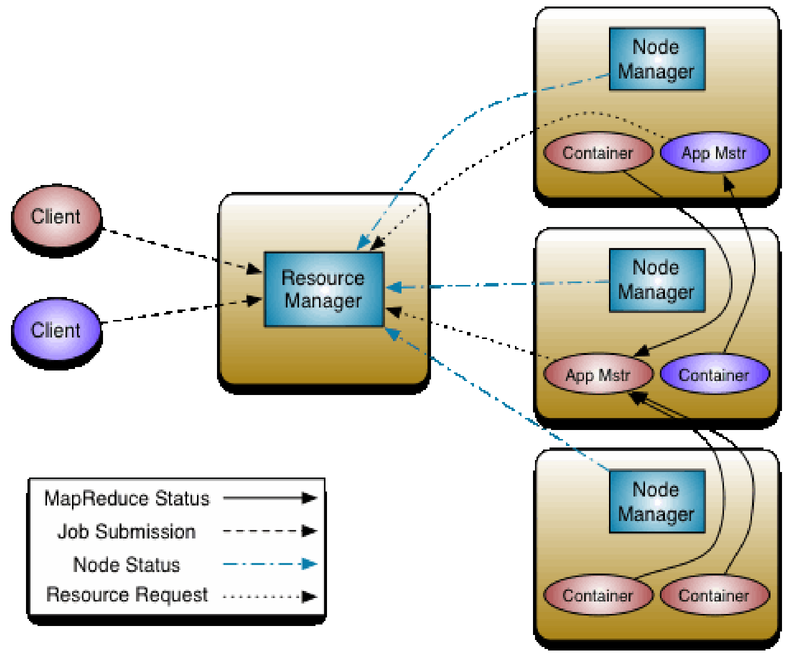
- Main changes:
- MapReduce NextGen aka YARN aka MRv2
  - divides the two major functions of the JobTracker: resource management and job life-cycle management into separate components.
  - Now, MapReduce is just one of the applications running on top of YARN.
  - YARN permits alternative programming models.
- HDFS Federation
  - Scale the name service horizontally.

Hadoop

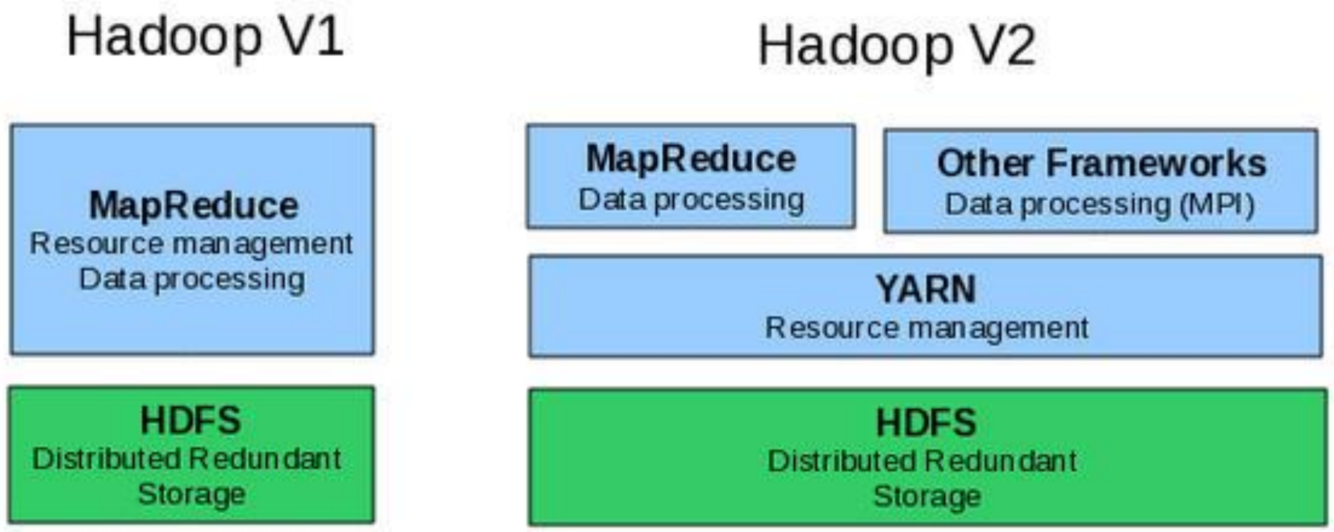
Hadoop 1.0



Hadoop 2.0



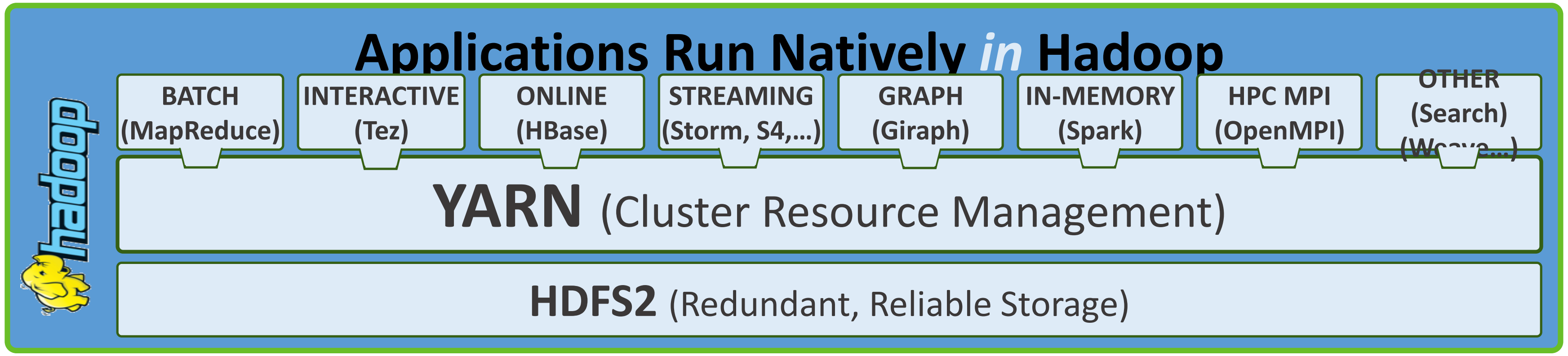
Hadoop MRv1 *versus* MRv2

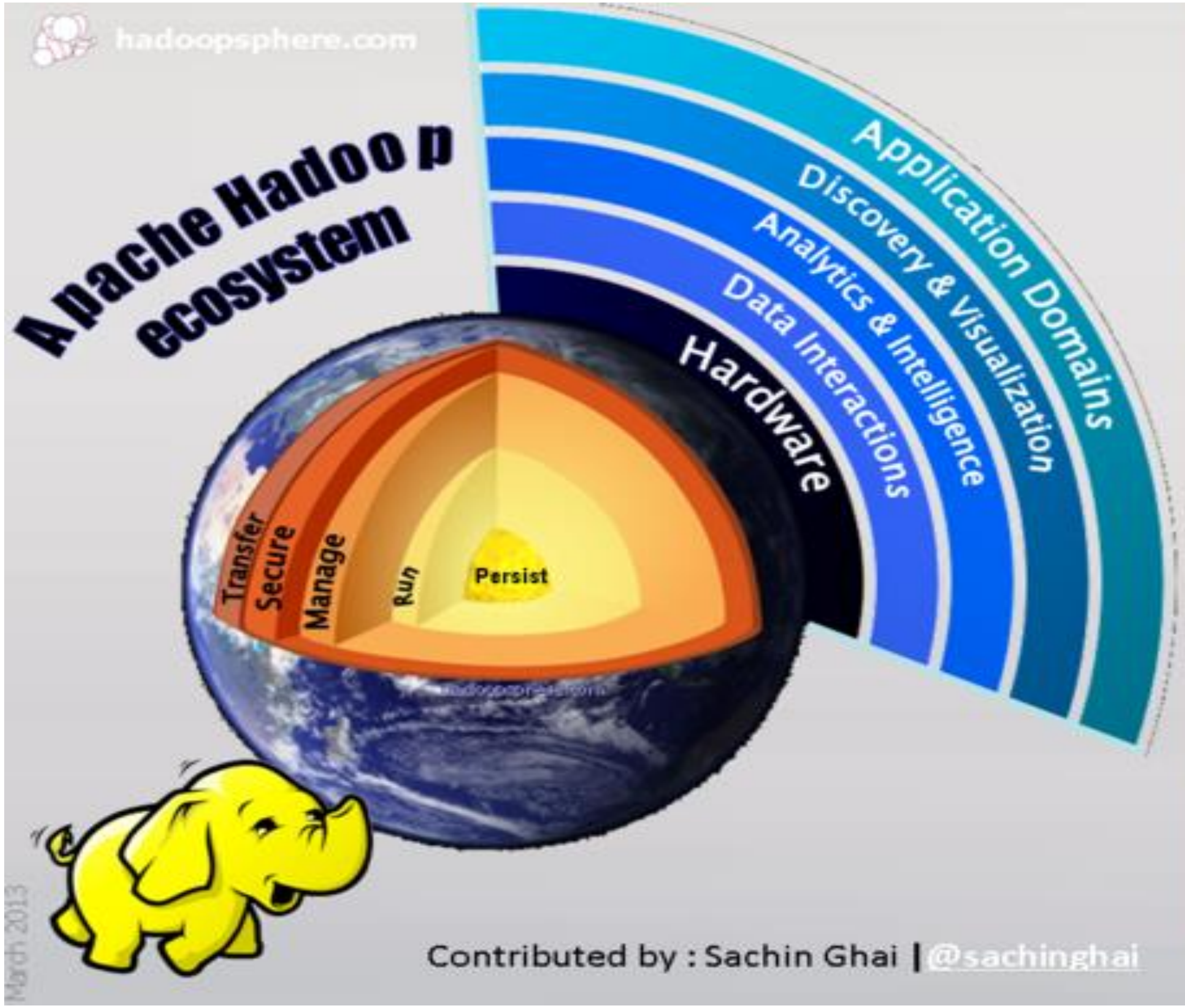


Hadoop v2 divides the two major functions of the JobTracker: resource management and data processing into separate components.

YARN

- Multitenancy: multiple access engine (batch, interactive and real-time)
- Cluster utilization: dynamic allocation of cluster resource
- Resource management: scale scheduling as number of nodes grows
- Compatibility: retrocompatible applications

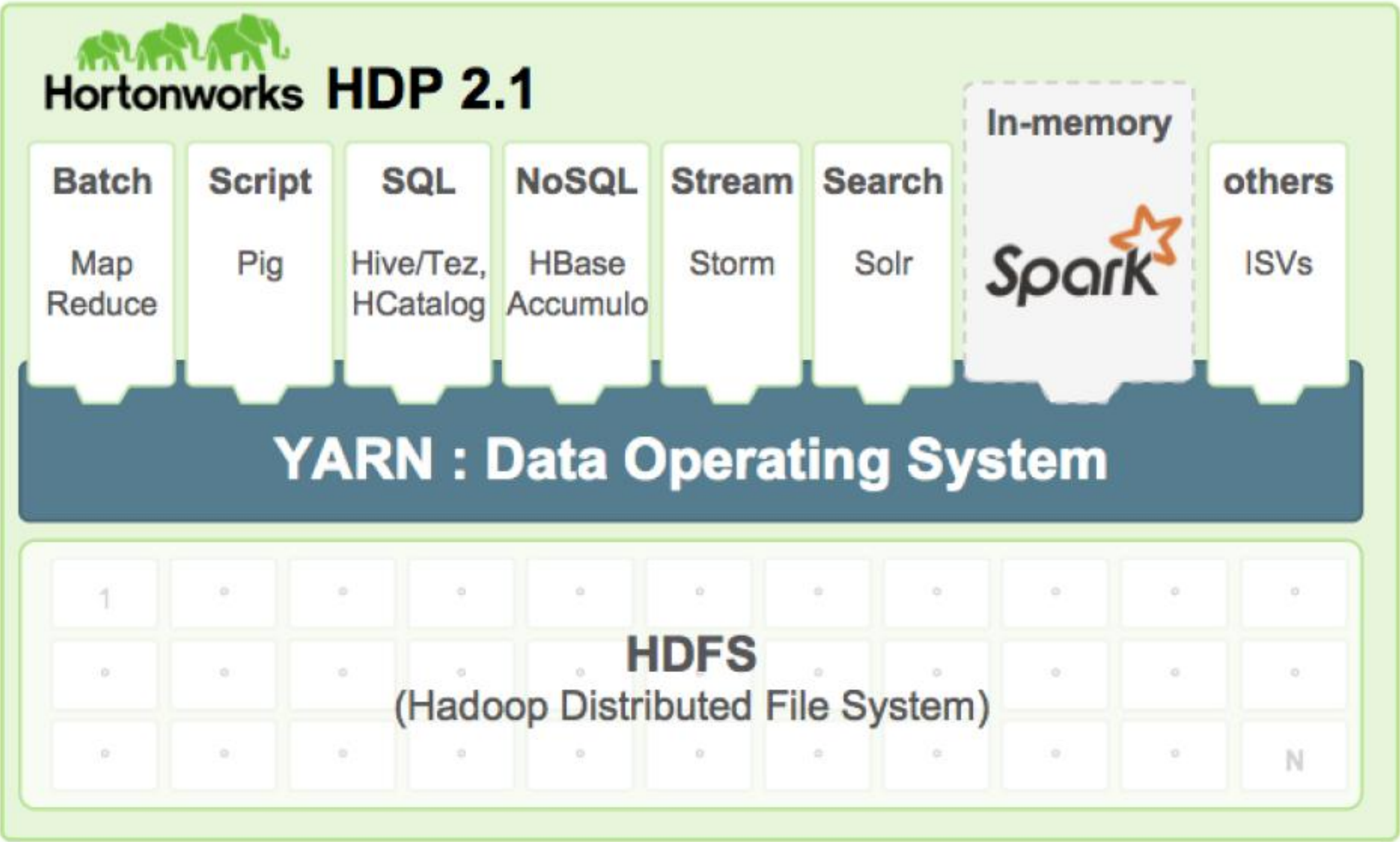




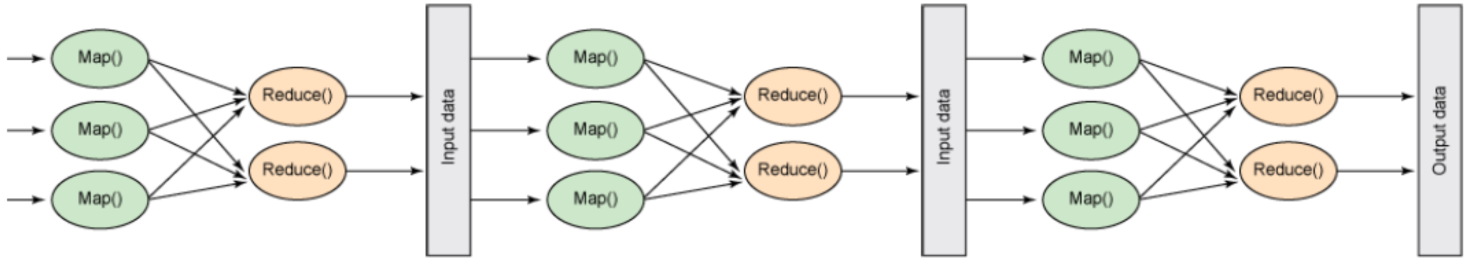
'Atmospheric' Layers	Application Domains	Distribution, Financial, Government, Heavy Industry, Internet, Oil & Energy, Research, Telecom
	Discovery & Visualization	Lucene, Blur, Giraph
	Analytics & Intelligence	Mahout, Drill
	Data Interactions	Pig, Hive, HCatalog, Tez, Gora
	Hardware (& Appliances)	Commodity H/w
'Core' Layers	Distribution	Apache
	Secure	Knox
	Manage	Oozie, Zookeeper, Crunch, MRUnit, HDT, Ambari, Vaidya, BigTop, Whirr
	Run	MapReduce, YARN, Hama
	Persist	HDFS, HBase, Cassandra, Accumulo, Avro, Trevni, Thrift
	Transfer	Flume, Sqoop, Chukwa, Kafka

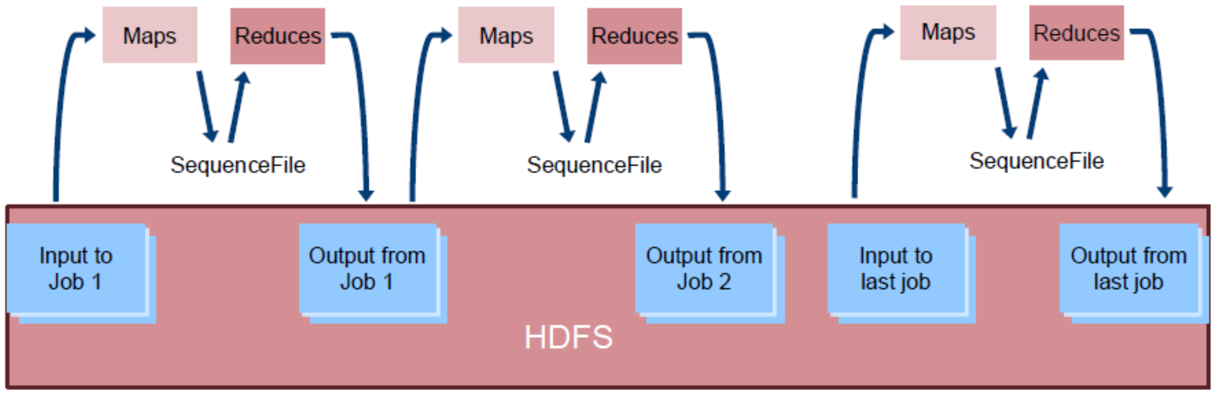
Who uses Hadoop?

http://wiki.apache.org/hadoop/PoweredBy

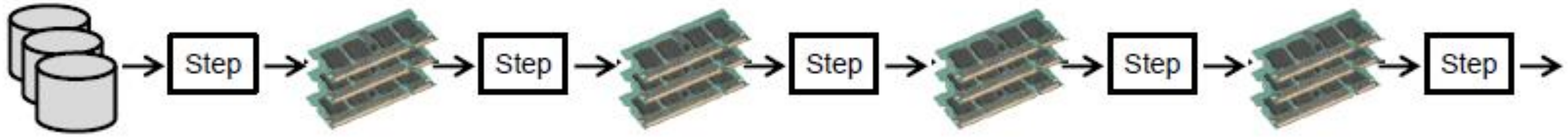


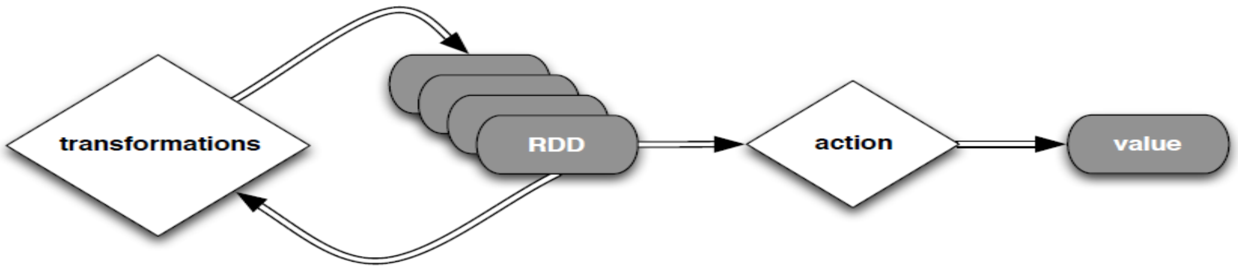
http://hortonworks.com/blog/apache-spark-yarn-ready-hortonworks-data-platform/



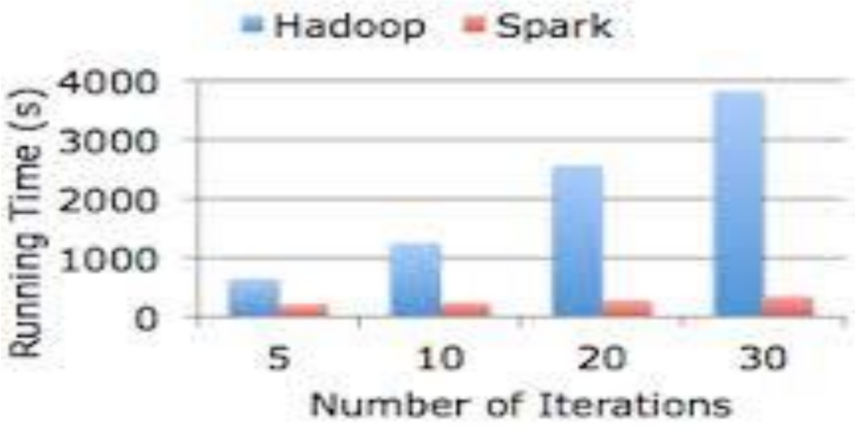


Carol McDonald: An Overview of Apache Spark


Step
Step
Step
Step
Step


transformations
RDD
action
value

Apache Spark



- In contrast to Hadoop's two-stage disk-based MapReduce paradigm, Spark's in-memory primitives provide performance up to 100 times faster for certain applications. By allowing user programs to load data into a cluster's memory and query it repeatedly, Spark is well suited to machine learning algorithms.
- https://**spark.apache**.org/

Spark Architecture

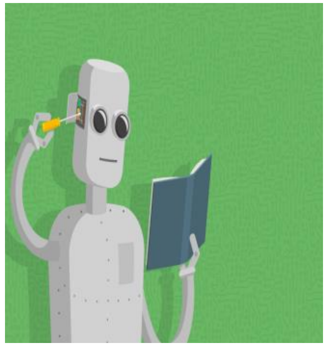
- Spark Core and Resilient Distributed Datasets (RDDs): provides distributed task dispatching, scheduling, and basic I/O functionalities.
- Spark SQ - provides support for structured and semi-structured data.
- Spark Streaming - leverages Spark Core's fast scheduling capability to perform streaming analytics.
- MLlib Machine Learning Library - distributed machine learning framework on top of Spark,
- GraphX - a distributed graph processing framework on top of Spark

Spark Users

sharethrough
NETFLIX
DATASTAX
stratio
CISCO
SIGMOID ANALYTICS
guavus
Atigeo
PEARSON
VELOCIDATA
OOYALA
AsiaInfo
viadeo
kelkoo
FAiMDATA

Machine Learning

What is Machine Learning (ML)?



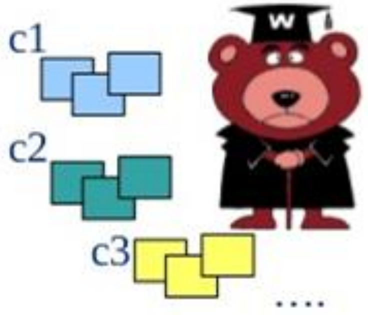
"A computer program is said to learn from experience *E* with respect to some task *T* and some performance measure *P*,
if its performance on *T*, as measured by *P*, improves with experience *E*."

Tom Mitchell (1997), Carnegie Mellon University

Machine Learning Algorithms

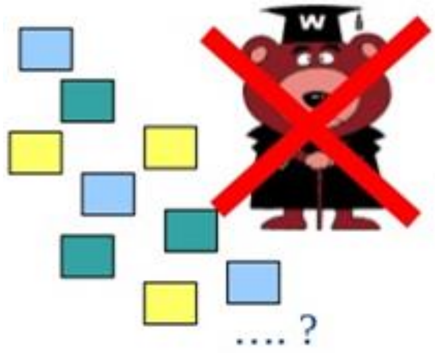
Supervised Learning

- induces a prediction function using a set of examples, called a *training set.*
    - data is **labelled**
    - to predict the correct label associated with any new observation



Unsupervised Learning

- consider **unlabeled** training examples and try to uncover regularities in the data



Machine Learning Algorithms

	Supervised	Unsupervised
Categorical	(Classification) *k* Nearest Neighbors Support Vector Machine Neural Networks	Hidden Markov Model
Continuous	Decision Trees Random Forests Regression	(Clustering and Dimensionality Reduction) Principal Component Analysis Singular Value Decomposition Gaussian Mixture Model

Machine Learning Algorithms

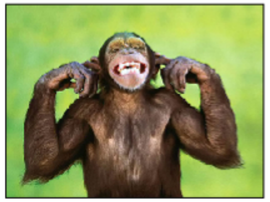
	Supervised	Unsupervised
Categorical	(Classification) *k* Nearest Neighbors Support Vector Machine Neural Networks	Hidden Markov Model
Continuous	Decision Trees Random Forests Regression	(Clustering and Dimensionality Reduction) Principal Component Analysis Singular Value Decomposition Gaussian Mixture Model

How do we classify (images)?

Training Images

→ Images Features → Classifier Training → Trained Classifier

Trained Labels ↓ Classifier Training



Test Image

→ Images Features → Trained Classifier → Prediction Monkey

k Nearest Neighbors (*k*NN)

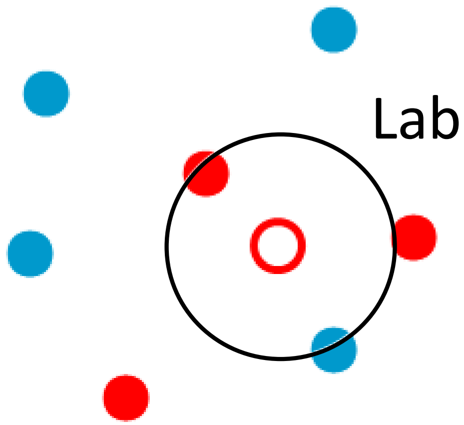
[

- One of the simplest of all machine learning classifiers. **No training** phase.

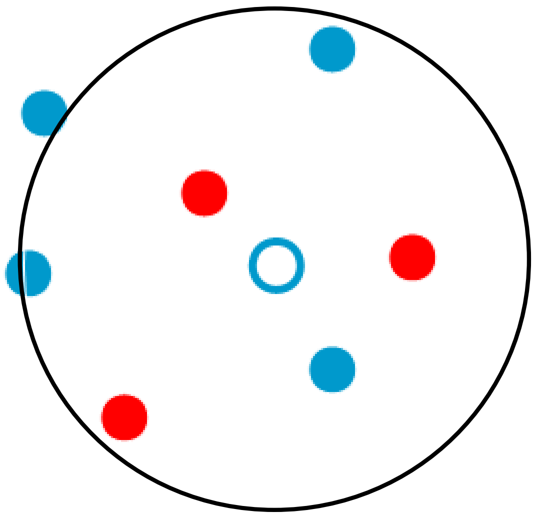
- The examples are classified based on the class of their *k* nearest neighbors in the descriptor space.

[Duda et al., 2001] Duda, R. O., Hart, P. E., and Stork, D. G. Pattern Classification. John Wiley and Sons, 2 edition, 2001.

k Nearest Neighbors (*k*NN)



Label it **red**, when k = 3.



Label it **blue**, when k = 7.

Support Vector Machines (SVM)

Given training instances (x,y), learn a model f such that $f(x) = y$. Use f to predict y for new x.

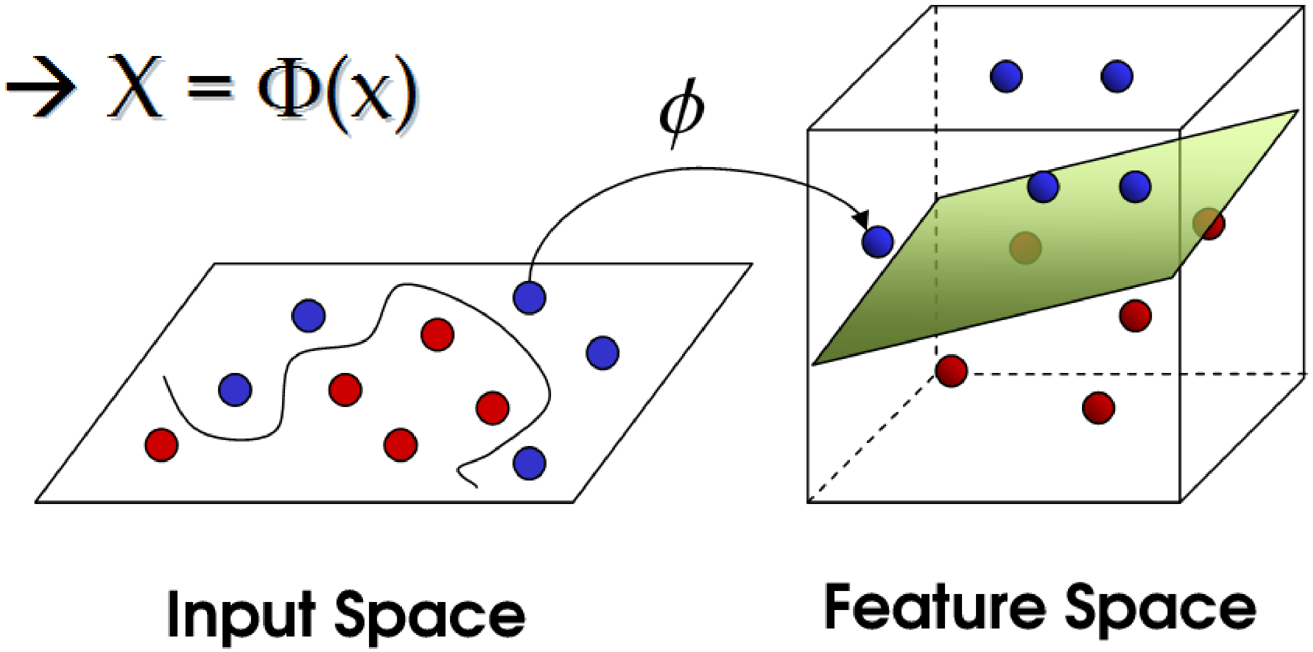
- Good generalization (in theory & in practice!)
- Work well with few training instances
- Find globally best model
- Efficient algorithms

[Vapnik, 1998] Vapnik, V. N. Statistical Learning Theory. John Wiley & Sons, 1998.

Support Vector Machines (SVM)

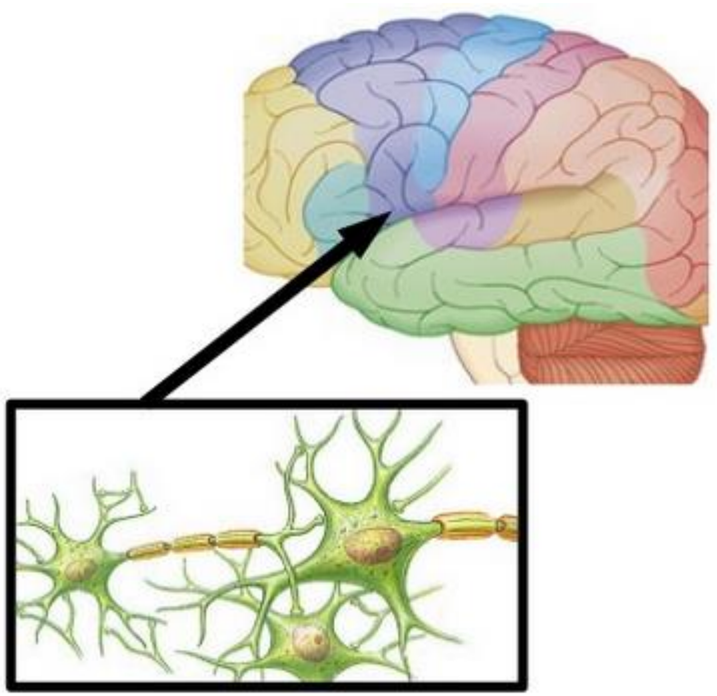
What if data is not linearly separable?
Mapping into a new feature space

$$\Phi : x \rightarrow X = \Phi(x)$$



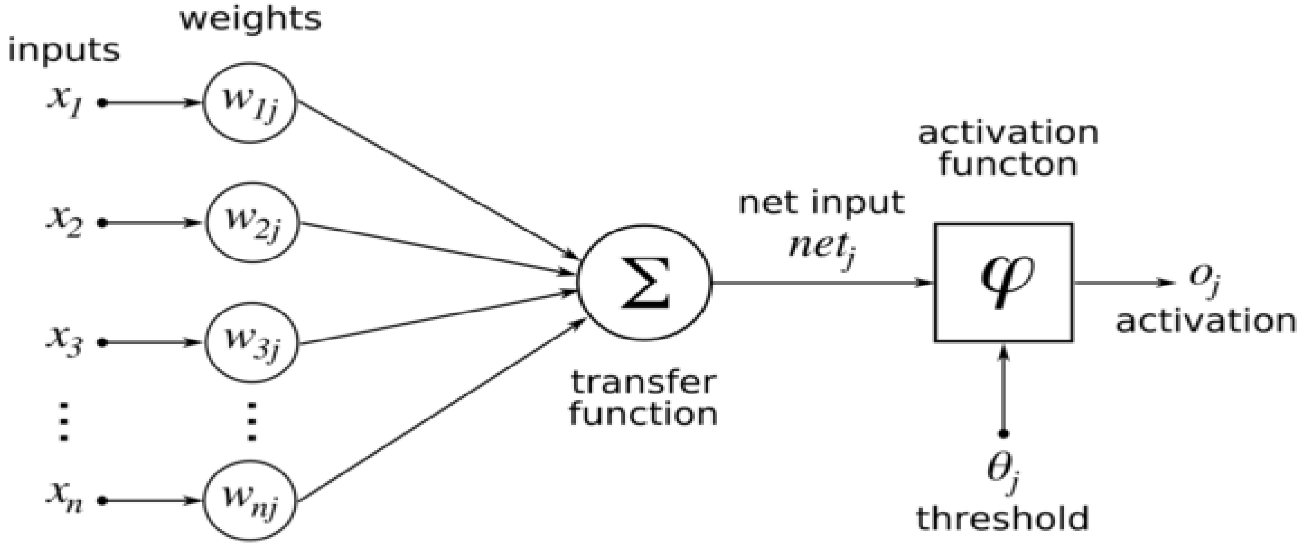
Artificial Neural Networks

Human brain as a collection of biological **neurons and synapses**



[Hubel and Wiesel] Hubel, D. and Wiesel, T. Receptive e
cortex. The Journal of Physiology, 148(3):574–591, 1
[Rosenblatt, 1959] Rosenblatt, F.. The perceptron: a pr
and organization in the brain. Psychological Review,

Artificial Neural Networks



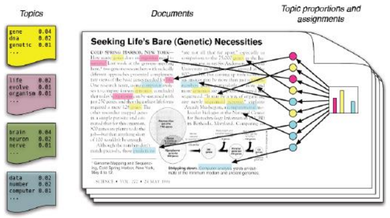
An *n*-dimensional input vector *x* is mapped to output variable *o* by means of the scalar product and a nonlinear function mapping *f*.

Deep Neural Networks

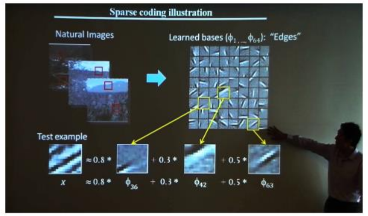
- Same networks as before, just BIGGER.
- Combination of three factors:
  - ★ Big data!
  - ★ Better algorithms!
  - ★ Parallel computing (GPU)!

[Hinton et al., 2006] Hinton, G., Osindero, S. and Teh, Y-W.. A Fast Learning Algorithm for Deep Belief Nets. Neural Computing, 18(7):1527–1554, 2006.

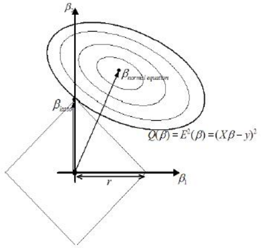
But new tasks have emerged; demand today's ML algos



Topic models
make sense of documents

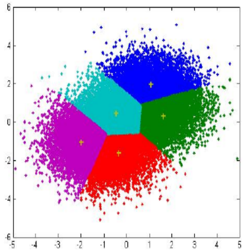


Deep learning
make sense of images, audio

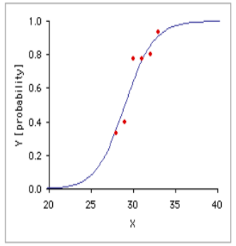


Lasso regression
find significant genes, predict stock market

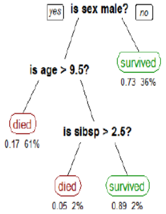
Classic ML algorithms used for decades



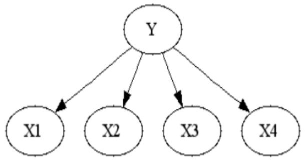
K-means



Logistic regression

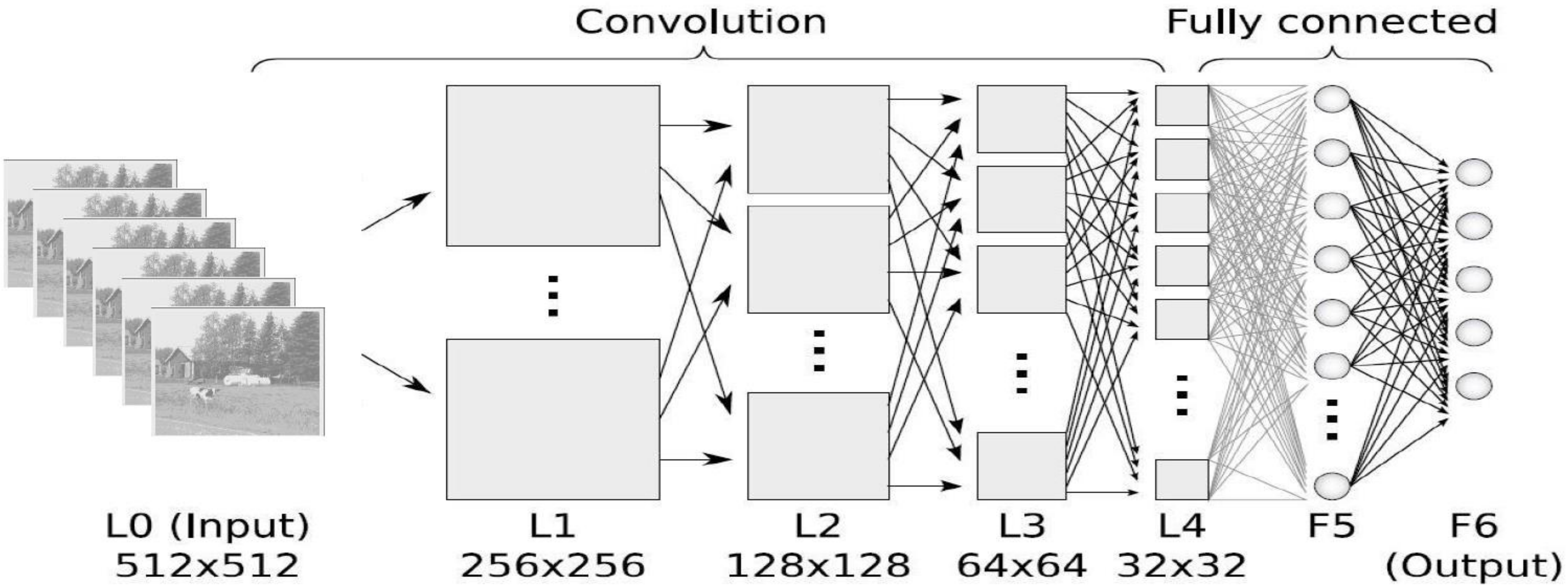


Decision trees



Naive Bayes

Qirong Ho and Eric P. Xing,Big ML Software for Modern ML Algorithms, , IEEE BigData 2014



Source: Universitv of Bonn