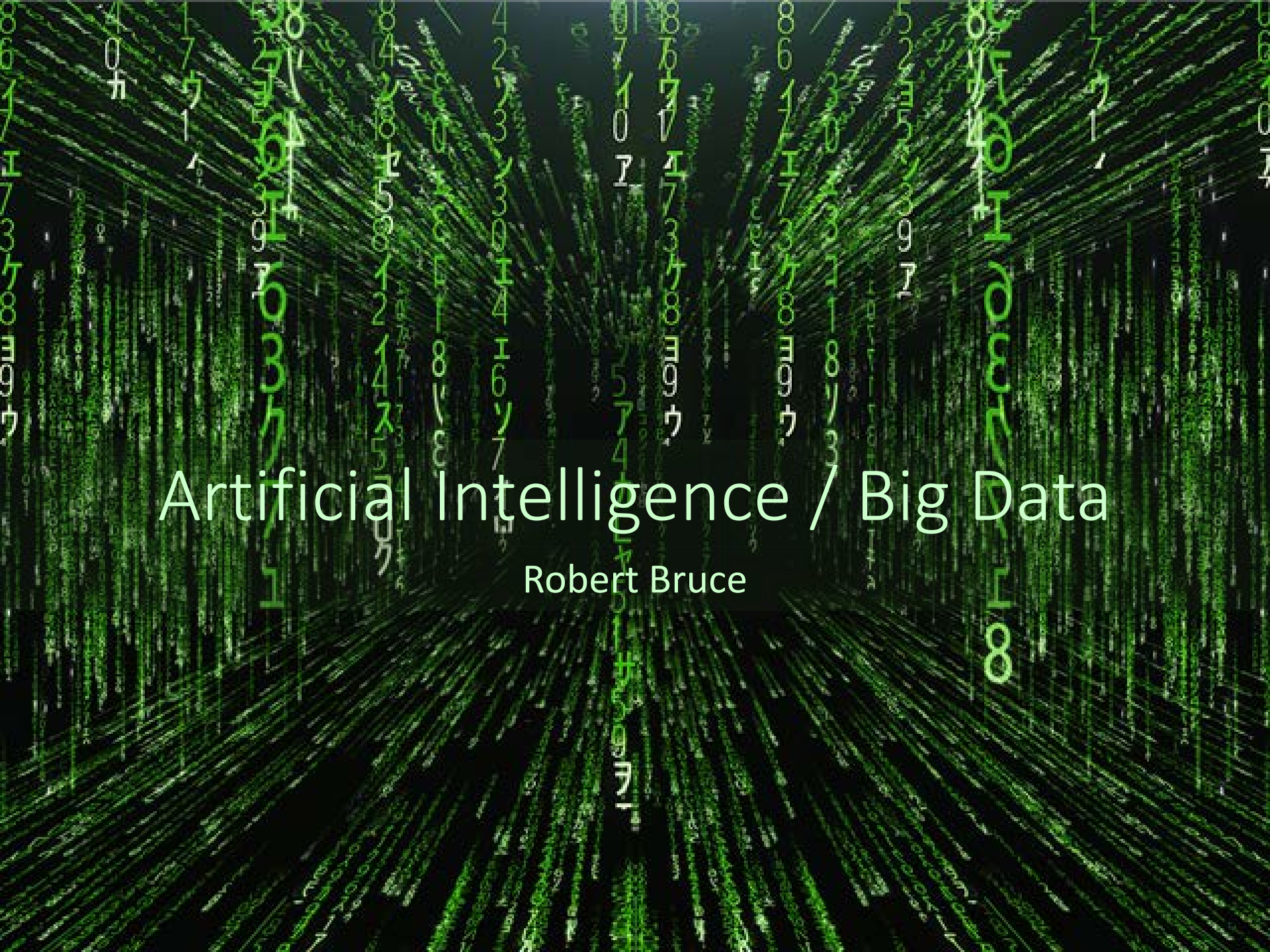


Artificial Intelligence / Big Data

Robert Bruce

Milestones in Database Management Systems

- Hierarchical data model developed by Vern Watts at IBM in 1966[1].
- Relational data model proposed by E. F. Codd in 1970[2].
- Non-relational database model proposed by M. Stronebraker in 1986[3].

Sources:

1. https://www-03.ibm.com/ibm/history/ibm100/us/en/icons/ibmims/
2. Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM 13*(6), 377-387.
3. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.58.5370&rep=rep1&type=pdf

Hierarchical Database Model

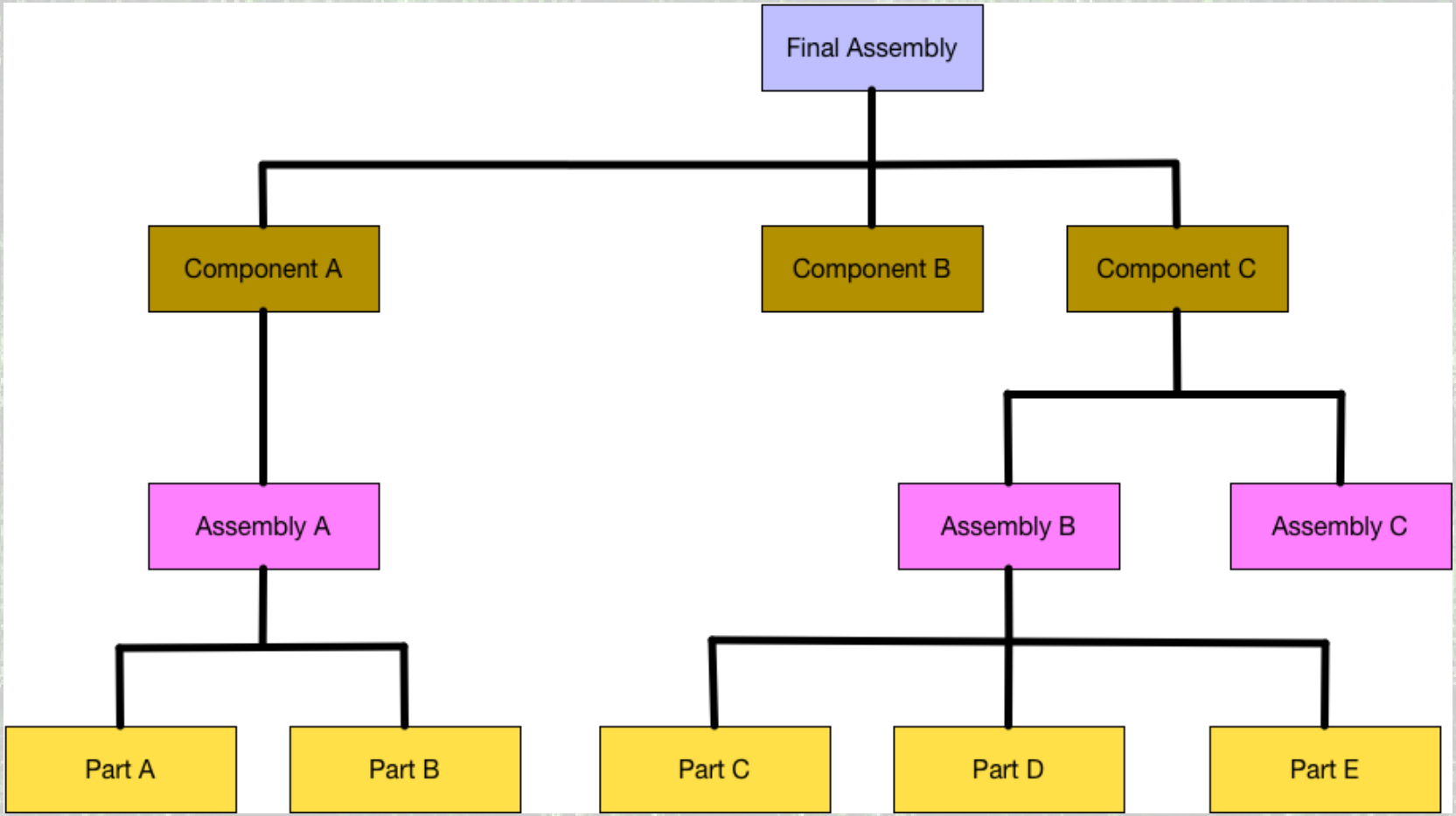
- The hierarchical data model is a tree structure with the following attributes (Rob & Coronel, 2004):

  a. Each parent node can have zero or more children nodes.

  b. Each child node belongs to only one parent node.

1. Rob, P. & Coronel, C. (2004). Database systems: Design, implementation, & management (6th ed.). Boston, MA: Thomson Learning.

Hierarchical Database Model



1. Rob, P. & Coronel, C. (2004). Database systems: Design, implementation, & management (6th ed.). Boston, MA: Thomson Learning.

Relational Database Model

- The relational database model uses mathematical set theory to retrieve data results based on entities and relations (Codd, 1970).
- Users enter Structured Query Language (SQL) commands in a Relational Database Management System (RDBMS) to retrieve data from the database (Codd, 1970).
- Example RDBMS include PostgreSQL, MariaDB, MySQL, Mini SQL, Microsoft Access, and Oracle (Codd, 1970).
- A relational database management system can be very effective for highly structured data; however one must consider:
    a. Can the data be accessed quickly and efficiently **at scale** for very large datasets?

1. Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM 13*(6), 377-387.

Non-Relational Database Model

- The non-relational database model harkens back to databases that pre-dated the relational database model (Tiwari, 2011, p. 4).
- Non-relational databases grew in popularity with the introduction of the World Wide Web and the need to organize, traverse, or search massive datasets of highly unorganized data (e.g. Google search) (Tiwari, 2011, p. 4).

Tiwari, S. (2011). *Professional NoSQL*. Indianapolis, IN: Wiley.

What is Big Data?

The National Institute of Standards and Technology (2015) defines Big Data as "extensive datasets – primarily in the characteristics of volume, variety, velocity, and/or variability – that requires a scalable architecture for efficient storage, manipulation, and analysis" (p. 5).

- **Volume**: refers to "the size of the dataset" (NIST, 2015, p. 4).
- **Variety**: refers to "data from multiple repositories, domains, or types" (NIST, 2015, p. 4).
- **Velocity**: refers to the rate that data is "created, stored, analyzed and visualized" (NIST, 2015, p. 15).
- **Variability**: refers to "any change in data over time including the flow rate, the format, or the composition" (NIST, 2015, p. 15).

National Institute of Standards and Technology (NIST). (2015). *NIST Big Data Interoperability Framework: Volume 1, Definitions* (NIST Special Publication 1500-1). Washington, DC: U.S. Department of Commerce.

An alternative definition for Big Data

Boyd and Crawford (2012) define Big Data as a confluence of “technology”, “analysis”, and “mythology” (p. 663).

- **“Technology**: maximizing computation power and algorithmic accuracy to gather, analyze, link, and compare large data sets” (Boyd & Crawford, 2012, p. 663).
- **“Analysis**: drawing on large data sets to identify patterns in order to make economic, social, technical, and legal claim” (Boyd & Crawford, 2012, p. 663).
- **“Mythology**: the widespread belief that large data sets offer a higher form of intelligence and knowledge that can generate insights that were previously impossible, with the aura of truth, objectivity, and accuracy” (Boyd & Crawford, 2012, p. 663).

Boyd, D. & Crawford, K. (2012). Critical questions for Big Data. *Information, Communication & Society 15*(5), 662-679. DOI:10.1080/1369118X.2012.678878

Why process Big Data?

Processing Big Data can potentially answer previously unattainable questions such as:

- "How can a potential pandemic reliably be detected early enough to intervene?" (NIST, 2015, p. 1)
- "Can new materials with advanced properties be predicted before these materials have ever been synthesized?" (NIST, 2015, p. 1)
- "How can the current advantage of the attacker over the defender in guarding against cybersecurity threats be reversed?" (NIST, 2015, p. 1)

National Institute of Standards and Technology (NIST). (2015). *NIST Big Data Interoperability Framework: Volume 1, Definitions* (NIST Special Publication 1500-1). Washington, DC: U.S. Department of Commerce.

Data Mining and Big Data: the relationship

- Data mining provides a means for processing big data regardless of the data volume, variety, velocity, or variability.
- Data mining attempts to finds patterns and relationships in data.
- Data mining uses machine learning – a form of artificial intelligence – to find these patterns and relationships.
- On a deeper level, data mining is built upon statistics to infer patterns and relationships.

Data Mining: top ten algorithms

According to Wu, et al. (2008) the top ten most influential algorithms used in data mining are:

1. C4.5
2. K-means
3. Support Vector Machines (SVM)
4. Apriori
5. Expectation Maximization (EM)
6. PageRank
7. Adaboost
8. *k*NN: k-nearest neighbor classification
9. Naïve Bayes
10. Classification and Regresion Trees (CART)

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H.,...Steinberg, D. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems 14*(1), 1-37. DOI:10.1007/s10115-007-0114-2

Distributed Computing and MapReduce

Google's problem:

It takes a lot of time to traverse and archive contents on the World Wide Web.

Google's solution:

Implement a programming model called MapReduce on a networked cluster of Linux-based personal computers. Each cluster is comprised of:

- "100s/1000s of 2-CPU x86 machines, 2-4 GB of memory"†
- "Limited bisection bandwidth"†
- "Storage is on local IDE disks"†
- "GFS: distributed file system manages data"†
- "Job scheduling system: jobs made up of tasks, scheduler assigns tasks to machines"†

† Source: https://research.google.com/archive/mapreduce-osdi04-slides/index-auto-0006.html

What is MapReduce?

"MapReduce is a programming model and an associated implementation for processing and generating large data sets."†

"Users specify a map function that processes a key/value pair to generate a set of intermediate key/value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key."†

† Source: https://static.googleusercontent.com/media/research.google.com/en//archive/mapreduce-osdi04.pdf

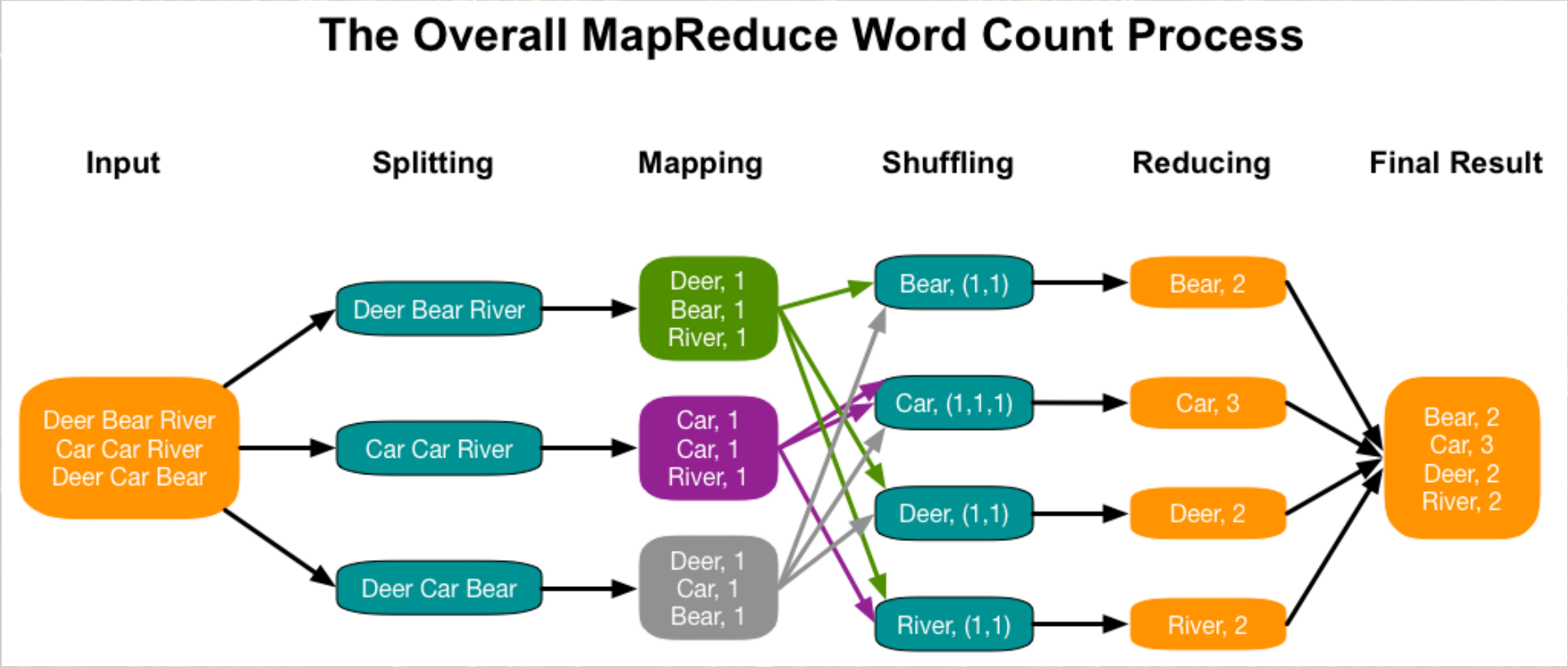
MapReduce features

MapReduce features:

- "Automatic parallelization and distribution"†
- "Fault-tolerance"†
- "I/O scheduling"†
- "Status and monitoring"†

† From: https://research.google.com/archive/mapreduce-osdi04-slides/index-auto-0002.html

MapReduce Example 1



Derived from source: https://www.edureka.co/blog/mapreduce-tutorial/#what_is_mapreduce

MapReduce Example 2

In the example below, MapReduce aggregated the number of students in each department.

Student	Department	Count	(Key, Value), Pair
Student 1	D1	1	(D1, 1)
Student 2	D1	1	(D1, 1)
Student 3	D1	1	(D1, 1)
Student 4	D2	1	(D2, 1)
Student 5	D2	1	(D2, 1)
Student 6	D3	1	(D3, 1)
Student 7	D3	1	(D3, 1)

Department	Total students
D1	3
D2	2
D3	2

Source: https://www.edureka.co/blog/hadoop-ecosystem

Introduction to Hadoop

- Hadoop is a distributed framework for parallel processing of big data.†
- Hadoop has two components:
  - A storage component: the Hadoop Filesystem (HDFS).†
  - A processing component: YARN.†

† Source: https://www.edureka.co/blog/what-is-hadoop/

Hadoop File System (HDFS)

Hadoop File System (HDFS):

- A distributed file structure comprised of many clusters spanning a massive number of machines.
- Each cluster is comprised of a *Namenode* and one or more *Datanodes* in a master/slave hierarchical tree structure.
- HFDS is hardware fault-tolerant through replication.
- HDFS is designed for enormous datasets.

Source: https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

Hadoop File System (HDFS)

Namenode:

- Contains HDFS metadata such as “permissions, modification and access times, namespace and disk space quotas” (Shvachko, Kuang, Radia, Chansle, 2010, p. 1).
- “maintains the namespace tree and the mapping of file blocks to DataNodes” (Shvachko, Kuang, Radia, Chansle, 2010, p. 1).

Datanode:

- Contains chunks of data typically 128 MB in size (user can also define size of each chunk) (Shvachko, Kuang, Radia, Chansle, 2010, p. 2).
- Data chunks typically span multiple datanodes (Shvachko, Kuang, Radia, Chansle, 2010, p. 2).

Source: Shvachko, K., Kuang, H., Radia, S., & Chansle, R. (2010). The Hadoop Distributed File System. *Proceedings of the 26th Symposium on Mass Storage Systems and Technologies (MSST)*, (pp. 1-10). doi:10.1109/MSST.2010.5496972

Hadoop: Yarn

Yarn (Yet Another Resource Negotiator) is “the brain of your Hadoop Ecosystem”[†]. It is comprised of two separate processes:

- **Resource manager**

  “arbitrates resources among all the applications in the system”[‡]

- **Node manager**

  “responsible for containers, monitoring their resource usage (cpu, memory, disk, network) and reporting the same to the ResourceManager/Scheduler”[‡]

[†] Source: https://www.edureka.co/blog/hadoop-ecosystem
[‡] Source: https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html

Apache Spark

What is Apache Spark?

- "A framework for real time data analytics in a distributed computing environment"†
- "executes in-memory computations to increase speed of data processing over Map-Reduce"†
- "100x faster than Hadoop for large scale data processing by exploiting in-memory computations and other optimizations"†
- Apache Spark homepage at: https://spark.apache.org/

† Source: https://www.edureka.co/blog/hadoop-ecosystem

Apache Pig

What is Apache Pig?

- Translates a high level SQL-like programming language called “Pig Latin” into a series of MapReduced tasks.†
- “a platform for building data flow for ETL (Extract, Transform and Load), processing and analyzing huge data sets”†
- Apache Pig homepage at: https://pig.apache.org/

† Source: https://www.edureka.co/blog/hadoop-ecosystem

Apache Hive

What is Apache Hive?

- "facilitates reading, writing, and managing large datasets residing in distributed storage using SQL."†
- Apache Hive homepage at: https://hive.apache.org/

† Source: https://hive.apache.org/

Apache Mahout

What is Apache Mahout?

- Provides machine learning algorithms to perform clustering, recommender systems, and regression on a distributed storage platform.[†]
- Apache Mahout homepage at: https://mahout.apache.org/

[†] Source: https://mahout.apache.org/docs/latest/

Apache Hbase

What is Apache Hbase?

- "open-source, distributed, versioned, non-relational database modeled after Google's Bigtable: A Distributed Storage System for Structured Data"†
- Works in conjunction with Hadoop and HDFS.†
- Apache Hbase homepage at: https://hbase.apache.org/

† Source: https://hbase.apache.org/

Apache Drill

What is Apache Drill?

- Enables programmers to query and join data from multiple non-relational database management systems together into one dataset.
- Apache Drill homepage at: https://drill.apache.org/

† Source: https://drill.apache.org/

Apache ZooKeeper

What is Apache ZooKeeper?

- "a distributed, open-source coordination service for distributed applications"†
- Apache ZooKeeper homepage at: https://zookeeper.apache.org/

† Source: https://zookeeper.apache.org/doc/current/zookeeperOver.html

Apache Oozie

What is Apache Oozie?

- “workflow scheduler system to manage Apache Hadoop jobs”†
- “jobs triggered by time (frequency) and data availability”†
- Apache Oozie homepage at: https://oozie.apache.org/

† Source: https://oozie.apache.org/

Apache Flume

What is Apache Flume?

- “a distributed, reliable, and available system for efficiently collecting, aggregating and moving large amounts of log data from many different sources to a centralized data store.”†
- Apache Flume homepage at: https://flume.apache.org/

† Source: https://flume.apache.org/

Apache Sqoop

What is Apache Sqoop?

- “a tool designed for efficiently transferring bulk data between Apache Hadoop and structured datastores such as relational databases”†
- Apache Sqoop homepage at: http://sqoop.apache.org/

† Source: http://sqoop.apache.org/

Apache Lucene

What is Apache Lucene?

- “high performance search server”†
- “provides Java-based indexing and search technology, as well as spellchecking, hit highlighting and advanced analysis/tokenization capabilities”†
- Apache Lucene homepage at: https://lucene.apache.org/

† Source: https://lucene.apache.org/

Apache Ambari

What is Apache Ambari?

- A software tool used for “provisioning, managing, and monitoring Apache Hadoop clusters”[†]
- Apache Ambari homepage at: https://ambari.apache.org/

[†] Source: https://ambari.apache.org/

Big Data and Ethics: Consequences

- Analyzing consumer purchasing habits, Target (a consumer department store) predicted a female shopper was pregnant.

https://www.nytimes.com/2012/02/19/magazine/shopping-habits.html

Impediments to Big Data

Some impediments that limit progress in the implementation of Big Data include:

- "What attributes define Big Data solutions?" (NIST, 2015, p. 1)
- "How is Big Data different from traditional data environments and related applications?" (NIST, 2015, p. 1)
- "What are the essential characteristics of Big Data environments?" (NIST, 2015, p. 1)
- How do these environments integrate with currently deployed architectures?" (NIST, 2015, p. 1)
- What are the central scientific, technological, and standardization challenges that need to be addressed to accelerate the deployment of robust Big Data solutions?" (NIST, 2015, p. 1)

National Institute of Standards and Technology (NIST). (2015). *NIST Big Data Interoperability Framework: Volume 1, Definitions* (NIST Special Publication 1500-1). Washington, DC: U.S. Department of Commerce.

Legal issues with Big Data

Legal issues with Big Data:

- "Who 'owns' a piece of data and what rights come attached with a dataset?" (p. 11)
- "What defines 'fair use' of data?" (p. 11)
- "Who is responsible when an inaccurate piece of data leads to negative consequences?" (pp. 11-12)

Source:
https://www.mckinsey.com/~/media/McKinsey/Business%20Functions/McKinsey%20Digital/Our%20Insights/Big%20data%20The%20next%20frontier%20for%20innovation/MGI_big_data_full_report.ashx

Legal issues with Big Data

Legal issues with Big Data:

- "Who 'owns' a piece of data and what rights come attached with a dataset?" (p. 11)
- "What defines 'fair use' of data?" (p. 11)
- "Who is responsible when an inaccurate piece of data leads to negative consequences?" (pp. 11-12)

Source:
https://www.mckinsey.com/~/media/McKinsey/Business%20Functions/McKinsey%20Digital/Our%20Insights/Big%20data%20The%20next%20frontier%20for%20innovation/MGI_big_data_full_report.ashx

Ethical issues with Big Data

Ethical issues with Big Data:

- Who decides if specific data should be included or excluded in aggregate of a big data collection? (Boyd & Crawford, 2012, p. 672)
- How does one properly determine the context of the raw data collected? (Boyd & Crawford, 2012, p. 672)
- Does publicly available information require informed consent from the author of that information? (Boyd & Crawford, 2012, p. 672)

Source: Boyd, D. & Crawford, K. (2012). Critical questions for Big Data. *Information, Communication & Society 15*(5), 662-679. DOI:10.1080/1369118X.2012.678878

Professional Organization Codes of Ethics relevant to Big Data

- Association for Computing Machinery:

  “Section 1.6: Respect Privacy”

  “Section 1.7: Honor confidentiality”

  https://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct

- Data Science Association

  “Rule 5: Confidential Information”

  http://www.datascienceassn.org/code-of-conduct.html