Course Information and Introduction to Big Data

COMP9313: Big Data Management

Thanks to Dr. Xin Cao for sharing useful materials for the preparation of this course

Course Information

- **Lectures / Tutorials (weeks 1- 10)**
  - Wed 13:00 to 15:00 (Ritchie Theatre, K-G19-LG02)
  - Thu 16:00 to 18:00 (Sir John Clancy Auditorium, K-C24-G17)
- **Consultation with Lecturer (weeks 1-10)**
  - After Thu lectures (max. 1 hour per week overall)
- **Labs (weeks 4-9)**
- **Tutors (may change):**
  - Maisie Badami
  - Alireza Tabebordbar
  - More to be confirmed...

Course Aims

This course aims to introduce students to the concepts behind Big Data, the core technologies used in managing large-scale data sets, and a range of technologies for developing solutions to large-scale data analytics problems.

This course is intended for students who want to understand modern large-scale data analytics systems. It covers a wide range of topics and technologies, and will prepare students to be able to build such systems as well as use them efficiently and effectively to address challenges in big data management.

Not possible to cover all aspects of big data management!

Lectures / Tutorials

- Lectures will focus on presenting concepts and technologies on big data management
- Tutorials will focus on discussions of lecture materials and concrete scenarios where big data management can play an important role
- Schedule and length of lectures may vary based on the progress of the course

Resources

- Hadoop: The Definitive Guide . Tom White. 4th Edition - O'Reilly Media
- Learning Spark. Matei Zaharia, Holden Karau, Andy Konwinski, Patrick Wendell. O'Reilly Media
- Apache MapReduce Tutorial
- Apache Spark Quick Start
- Elasticsearch Reference
- Datasets: Assignments will make use of a number of datasets from different areas (e.g., cyber security)

Course Prerequisite

- Have experiences and good knowledge of algorithm design (equivalent to COMP9024)
- Have a solid background in database systems (equivalent to COMP9311)
- Have solid programming skills in Java
- Be familiar with working on a Unix-style operating system
- Be familiar with working with web services (e.g., RESTful services)

Learning Outcomes

On successfully completing this course, students should be able to:

- Describe the important characteristics of Big Data
- Understand key concerns in the management of Big Data
- Develop an appropriate storage structure for a Big Data repository
- Utilise the Map/Reduce paradigm and the Spark platform to manipulate Big Data
- Use a high-level query language to manipulate Big Data

Assessment

- **Quizzes (15%):** This component will help review the concepts introduced in lectures/tutorials
- **Assignments (35%):** This component assesses the student's ability to apply big data technologies to solve problems
- **Written Final Exam (50%):** This component is going to assess the various facts-and-knowledge level learning outcomes

Quizzes, Assignments and Final Exam

Quizzes:

- We will have quizzes after each lecture to help review concepts introduced in lectures

Assignments (plan):

- 1 assignment on Hadoop MapReduce
- 1 assignment on Spark
- 1 assignment on Elasticsearch

Final Exam:

- Final written exam (lab-based)

Tentative Course Schedule

Week	Lecture	Lab
1	Course Info & Intro to Big Data	
2	Big Data Processes and Management	
3	Hadoop	
4	Spark	**Lab 1:** Hadoop
5	Data Curation	**Lab 2:** Hadoop
6	NoSQL Technologies	**Lab 3:** Spark
7	Cybersecurity Information Indexing	**Lab 4:** Spark
8	Cognitive Services and Big Data	**Lab 5:** Elasticsearch
9	Finding Similar Items	**Lab 6:** Elasticsearch
10	Course wrap-up & Review	

General recommendations

- Read to the notice board in **WebCMS** (and check your emails)
- Participate in discussions in message boards
- Tutors can help with questions on assignments and practical/implementation aspects

Introduction to Big Data

Big Data: What is it?

- **Wikipedia:**

  "Big data is a field that treats ways to analyze, systematically extract information from, or otherwise deal with data sets that are too large or complex to be dealt with by traditional data-processing application software"

- **Oxford English dictionary:**

  "Extremely large data sets that may be analysed computationally to reveal patterns, trends, and associations, especially relating to human behaviour and interactions"

- **Apache Hadoop[1]:**

  "Datasets which could not be captured, managed, and processed by general computers within an acceptable scope"

[1]Chen et al. 2014. Big data: A survey. Mobile networks and applications, 19(2), pp.171-209.

Examples of Big Data

source

source

e-commerce

source

Open Data

source

source

Social Media

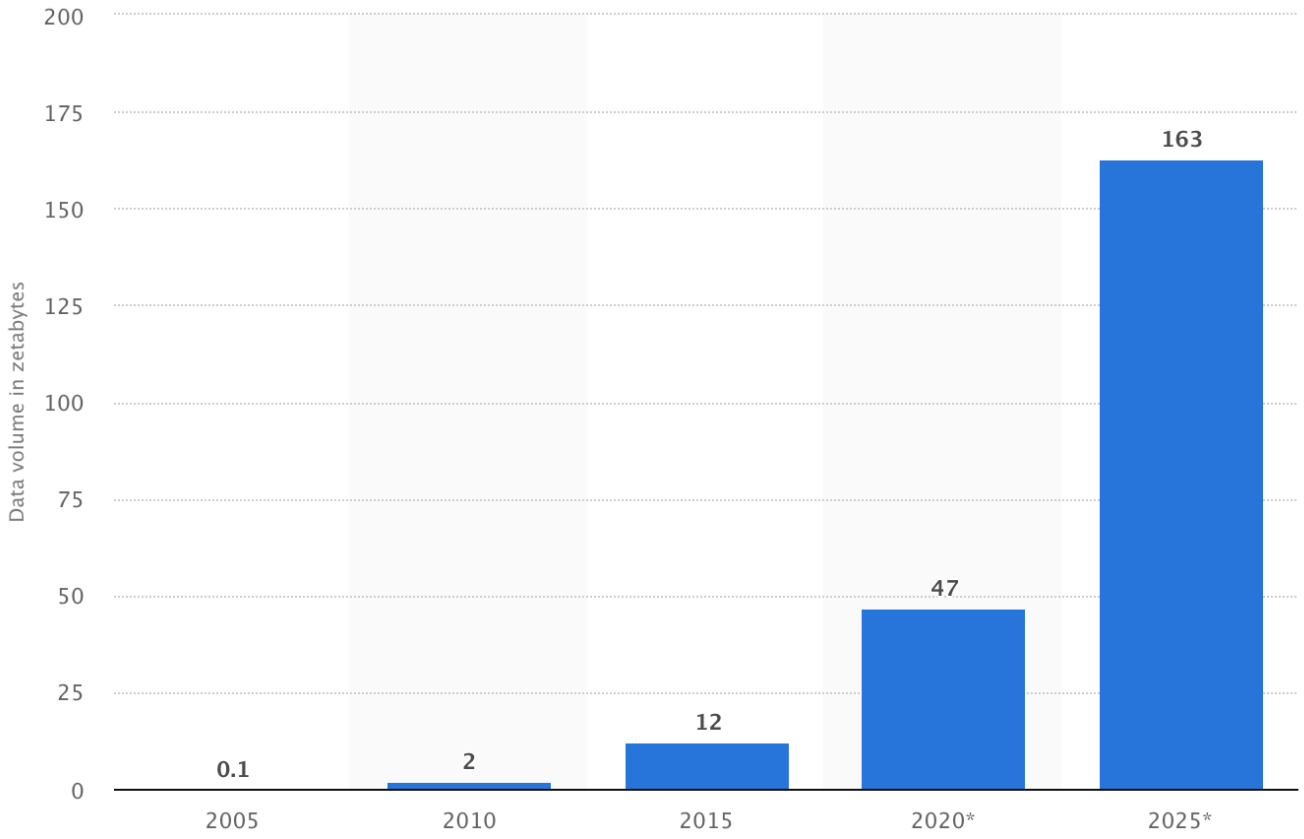
Healthcare

source

Internet of Things

Financial Sector

Volume of data/information created worldwide from 2005 to 2025



1 zettabyte = 1 million petabytes = 1 billion terabytes

https://www.statista.com/statistics/871513/worldwide-data-created/

More stats on Big Data...

- **Internet:**
  - 3.7+ billion humans using the Internet
  - Every second, Google processes 40K searches
  - 5 billion searches a day, worldwide

- **Social Media**

  Every minute:
  - Snapchat: 527K photos are shared
  - LinkedIn: 120 professionals joining the network
  - Twitter: 456K tweets sent
  - Instagram: 46K photos

https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#5d64ff7560ba

More stats on Big Data...

- **Communication:**

  Every minute:

  - 16 million text messages
  - 156 millions e-mails sent
  - 15K GIFs sent via Facebook Messenger
  - 154K calls on Skype

- **Services:**

  Every minute:

  - 18M forecast requests (Weather Channel)
  - $51K peer-to-peer transactions (Venmo)
  - 13 new songs added to Spotify
  - 45K rides in Uber
  - 600 new page edits in Wikipedia

https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#5d64ff7560ba

What is big data used for?

Manufacturing

source

- Equipment failure prediction
- Maximization of parts and equipment uptime
- Remaining optimal lifetime
- Overall production optimization

https://www.oracle.com/big-data/guide/big-data-use-cases.html

What is big data used for?

Retail

source

- Product and services prediction
- Demand forecasting
- In-store shopping experience
- Customer experience
- Pricing analytics and optimization

https://www.oracle.com/big-data/guide/big-data-use-cases.html

What is big data used for?

Healthcare

- Genomic research
- Patience experience and outcomes
- Claims and Fraud
- Healthcare billing analytics

https://www.oracle.com/big-data/guide/big-data-use-cases.html

What is big data used for?

Oil and Gas

source

- Predictive equipment maintenance
- Oil exploration and discovery
- Oil production optimization

https://www.oracle.com/big-data/guide/big-data-use-cases.html

What is big data used for?

Telecommunications

- Optimize network capacity
- Telecom customer churn
- New product offerings

source

https://www.oracle.com/big-data/guide/big-data-use-cases.html

What is big data used for?

Financial Services

source

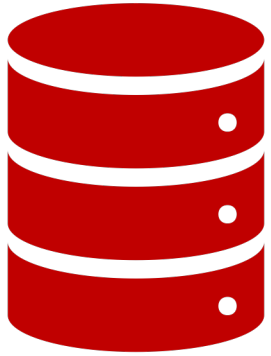
- Fraud and Compliance
- Anti-money laundering
- Financial regulatory and compliance analytics

https://www.oracle.com/big-data/guide/big-data-use-cases.html

Characterizing Big Data

The 3 Vs of Big Data

Volume



- Big data -> Big opportunities
- Challenges for traditional IT infrastructures
- Large amounts of data, low capacity to process it
- Distributed technologies: Filesystems and computation (e.g., Hadoop)

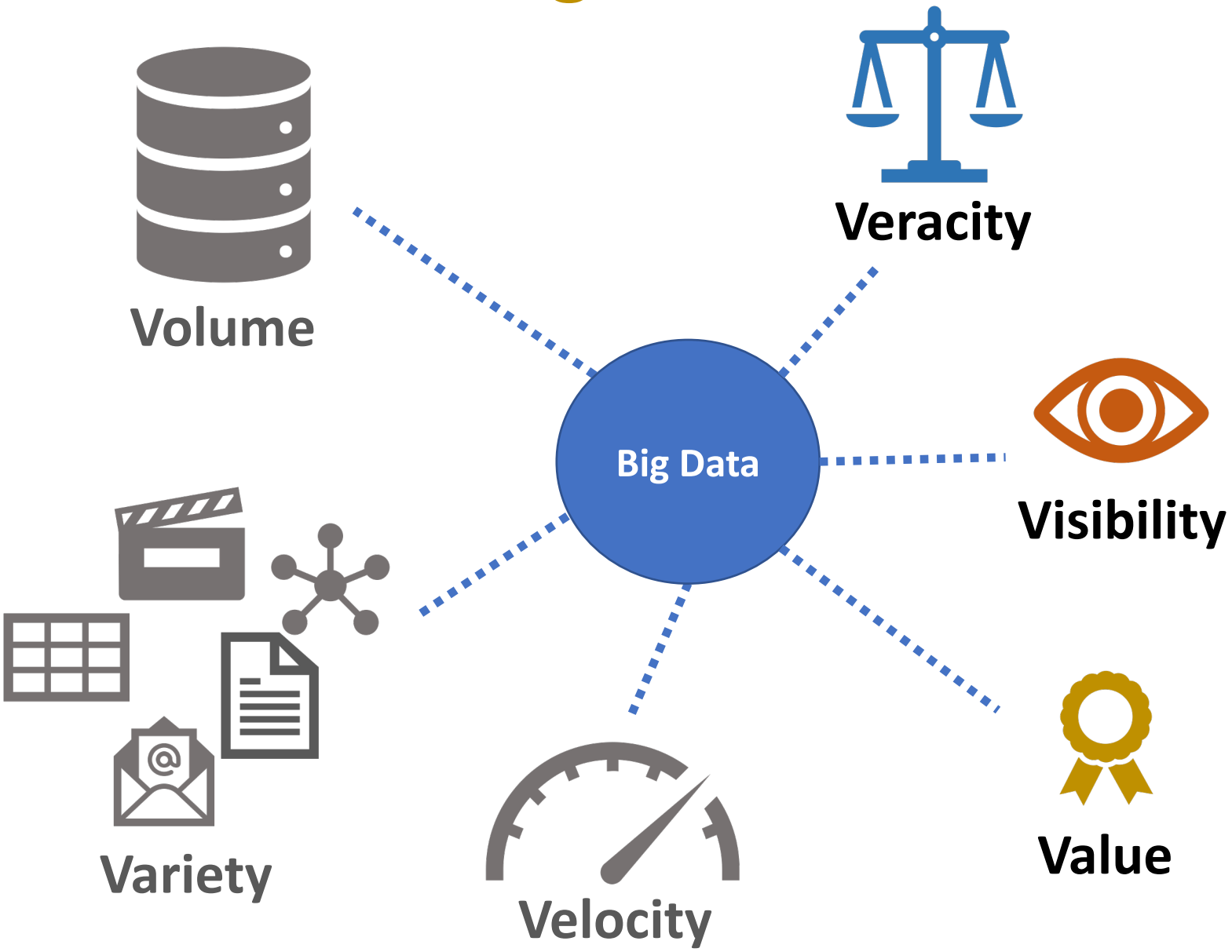
Variety

- Diverse sources of data (not just relational)
- Text, image, videos, sensor data, etc.
- Big data integration and curation challenges
- Flexibility in data representation

Velocity

- Internet / mobile: Faster creation and consumption of data
- How to keep up with the pace of it?
- Streaming processing -> (pre) analyze data it arrives

Other Vs for Big Data



Veracity

- Data -> **quantity + quality (trust)**
  - System capacity to handle sheer amounts of data?
  - Manage queries with resources we have?
- Can we trust the answers to our queries?
  - Unhandled, low quality data -> bad decisions
  - Loss of reputation, revenue, customers
- Quality is as important as quantity

Visibility

- Visibility -> The state of being able to see or be seen - is implied
- Not enough for information to “exists”. It should be visible to the right person
- Information silos -> Important roadblocks in the extraction of value from data

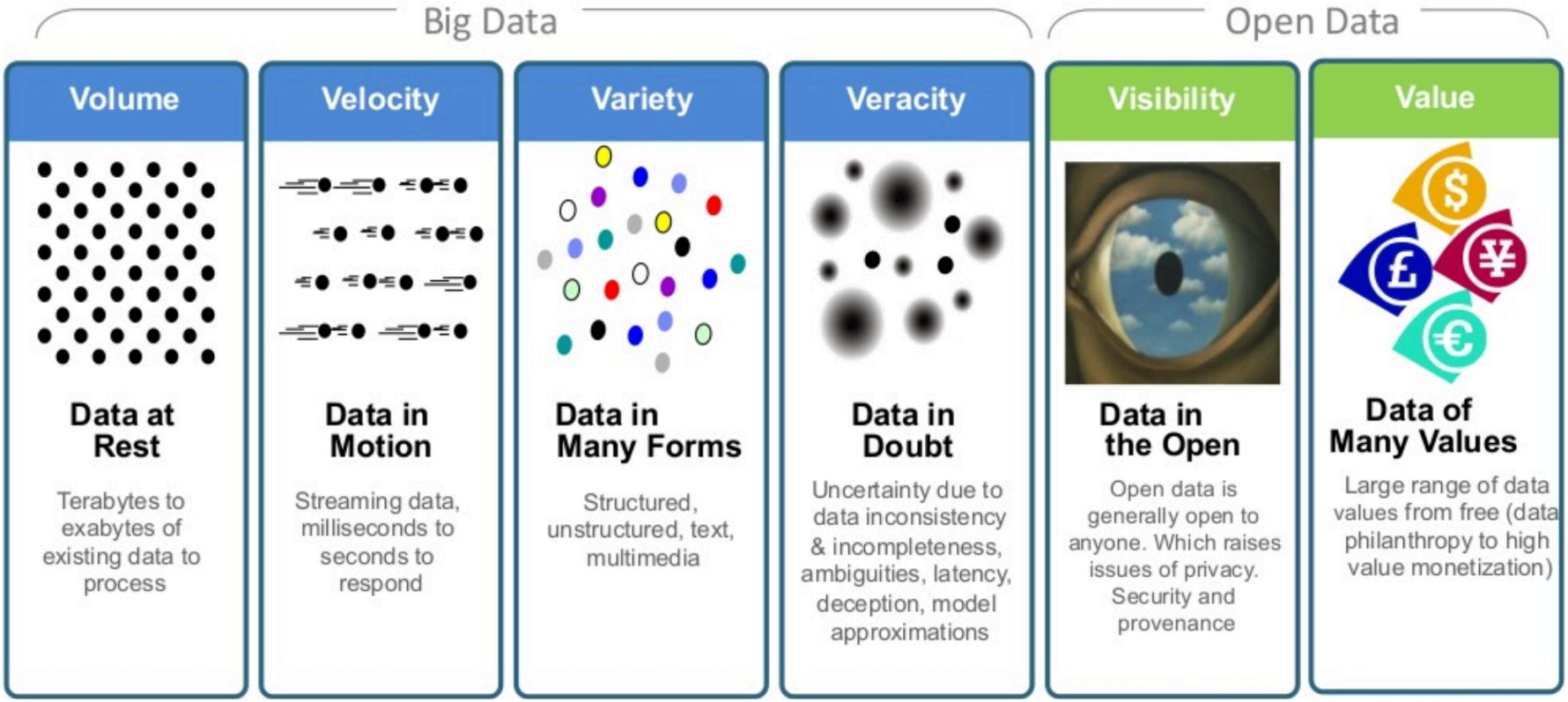
Value

- Holy grail of big data -> main goal
- Data in and of itself has no real value
- Does it justify the effort / investment?

Interest over time on “Big Data”

https://trends.google.com/trends/explore?date=all&q=big%20data

Summary of Big Data's 6V



Transforming Energy and Utilities through Big Data & Analytics
By Anders Quitzau, IBM, 2014

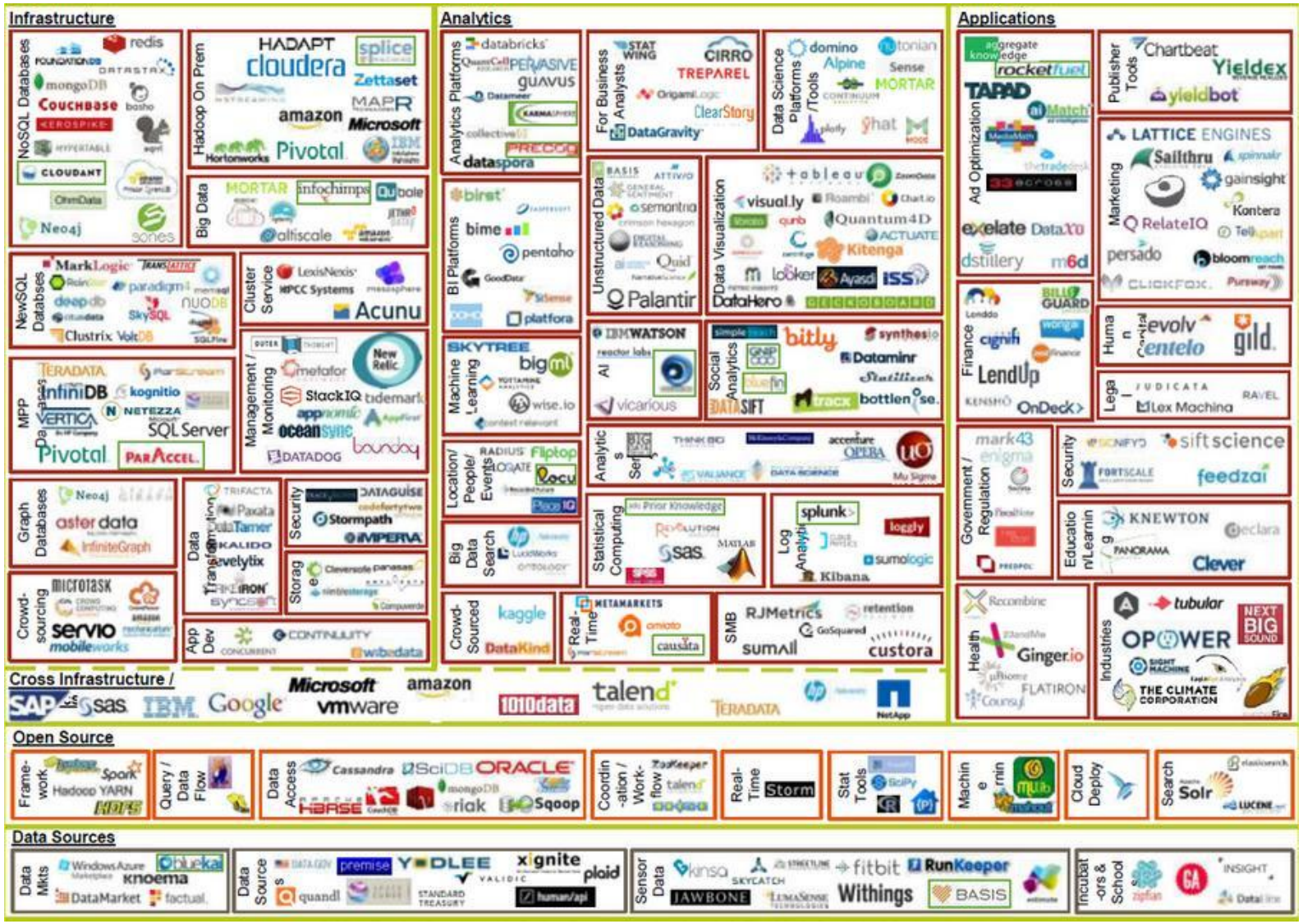
Expansion of the original Vs

- Vagueness
- Validity
- **Value**
- Variability
- **Variety**
- **Velocity**
- Venue
- **Veracity**
- Viability
- Vincularity
- Viscosity
- **Visibility**
- Visible
- Visualization
- Vitality
- Vocabulary
- Volatility
- **Volume**

Big Data Technologies: Why Learn Them?

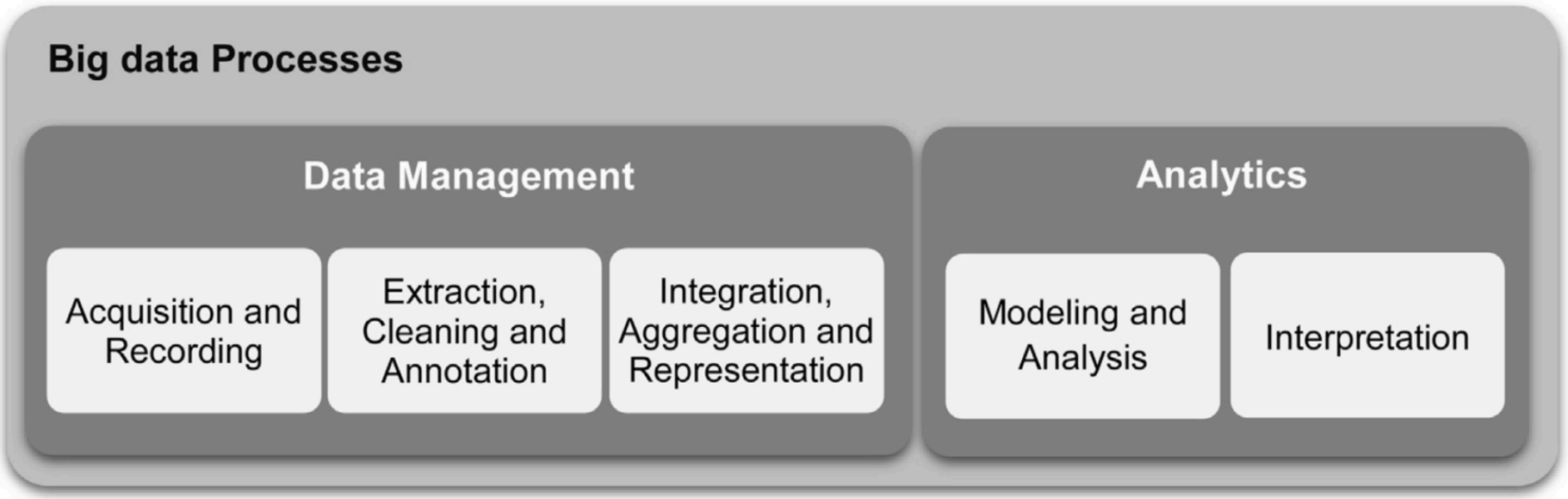
- One of the hottest topics in both research and industry!
- High demand for big data experts
- A promising future career
- Research and development of big data systems:
  - Distributed systems (e.g., Hadoop), visualization tools, data warehouse, OLAP, data integration, data quality control, …
  - Big data applications: social marketing, healthcare, …
  - Data analysis: to get values out of big data discovering and applying patterns, predictive analysis, business intelligence, privacy and security, …

Examples of Big Data Technologies



source

Big Data Processes



https://www.sciencedirect.com/science/article/pii/S0268401214001066

Big Data Analytical Techniques

Text Analytics

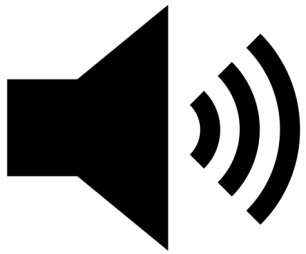
- Extracting information from textual data
- Statistical analysis, computational linguistics, ML
- Information extraction
- Text summarization
- Question answering
- Sentiment analysis

https://www.sciencedirect.com/science/article/pii/S0268401214001066

Big Data Analytical Techniques

Audio Analytics

- Analyzing/Extracting information from audio data
- Live call analysis
- Cross/up-selling recommendations
- Real-time feedbacks to agents
- Interactive voice response

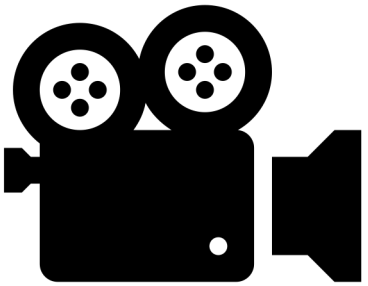


https://www.sciencedirect.com/science/article/pii/S0268401214001066

Big Data Analytical Techniques

Video Analytics

- Analyzing/Extracting information from video streams
- Automated security and surveillance systems
- Detecting breaches (e.g. restricted zones)
- Object identification (e.g. object removal)
- Suspicious activity recognition
- Camera tampering detection

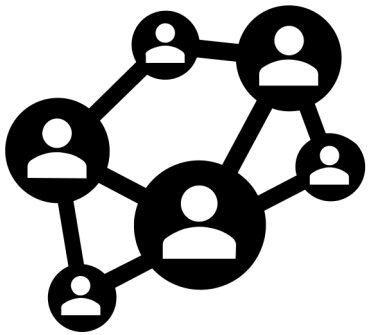


https://www.sciencedirect.com/science/article/pii/S0268401214001066

Big Data Analytical Techniques

Social Media Analytics

- Analyzing structured/unstructured data from social media
- Content- and structured-based analytics
- Community detection
- Social influence analysis
- Link prediction



https://www.sciencedirect.com/science/article/pii/S0268401214001066

Big Data Analytical Techniques

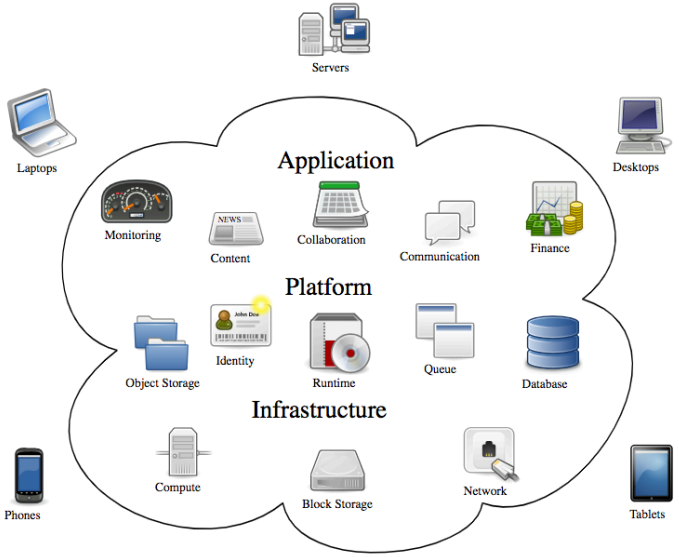
Predictive Analytics

- Heterogeneity (e.g., subpopulation / outliers)
- Noise accumulation (e.g. noise in estimations)
- Spurious correlation (e.g. correlation vs. size of dataset)

https://www.sciencedirect.com/science/article/pii/S0268401214001066

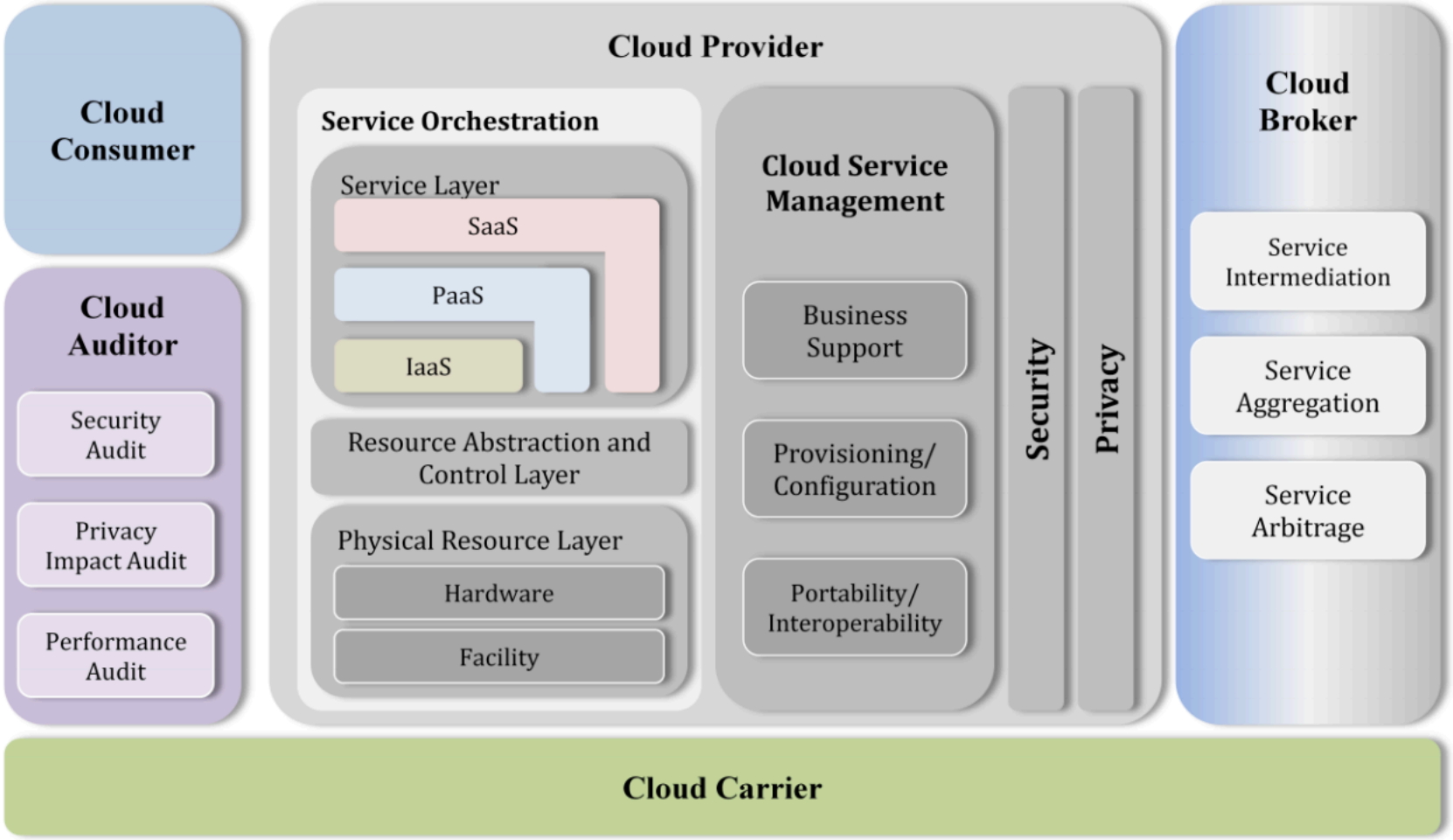
Cloud Computing

- Computing resources made available on-demand, over the Internet
- Resources: data storage and computing power
- Types of ”Clouds”:
  - Enterprise clouds
  - Public clouds
  - Hybrid-clouds



source

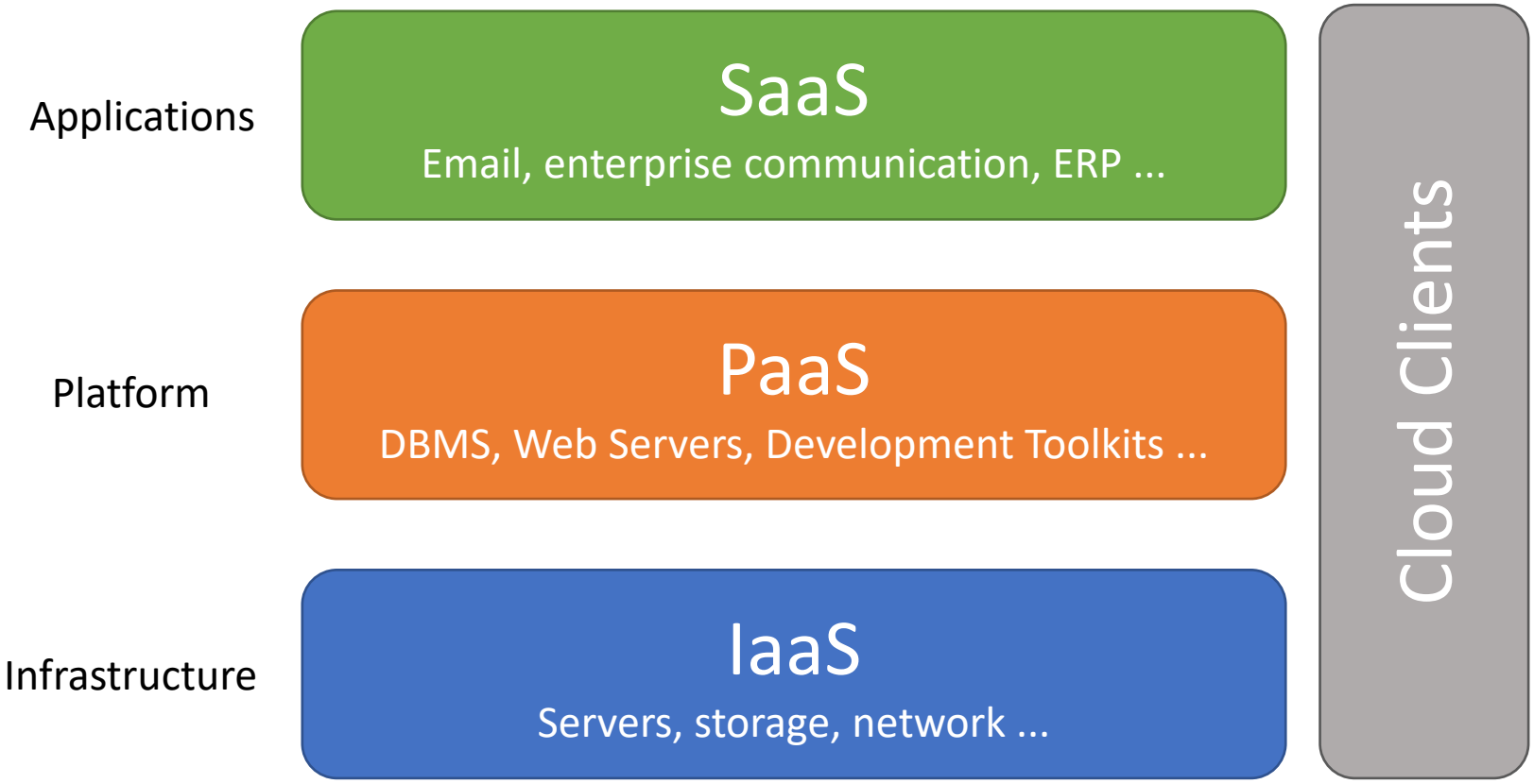
Architecture: Cloud Computing



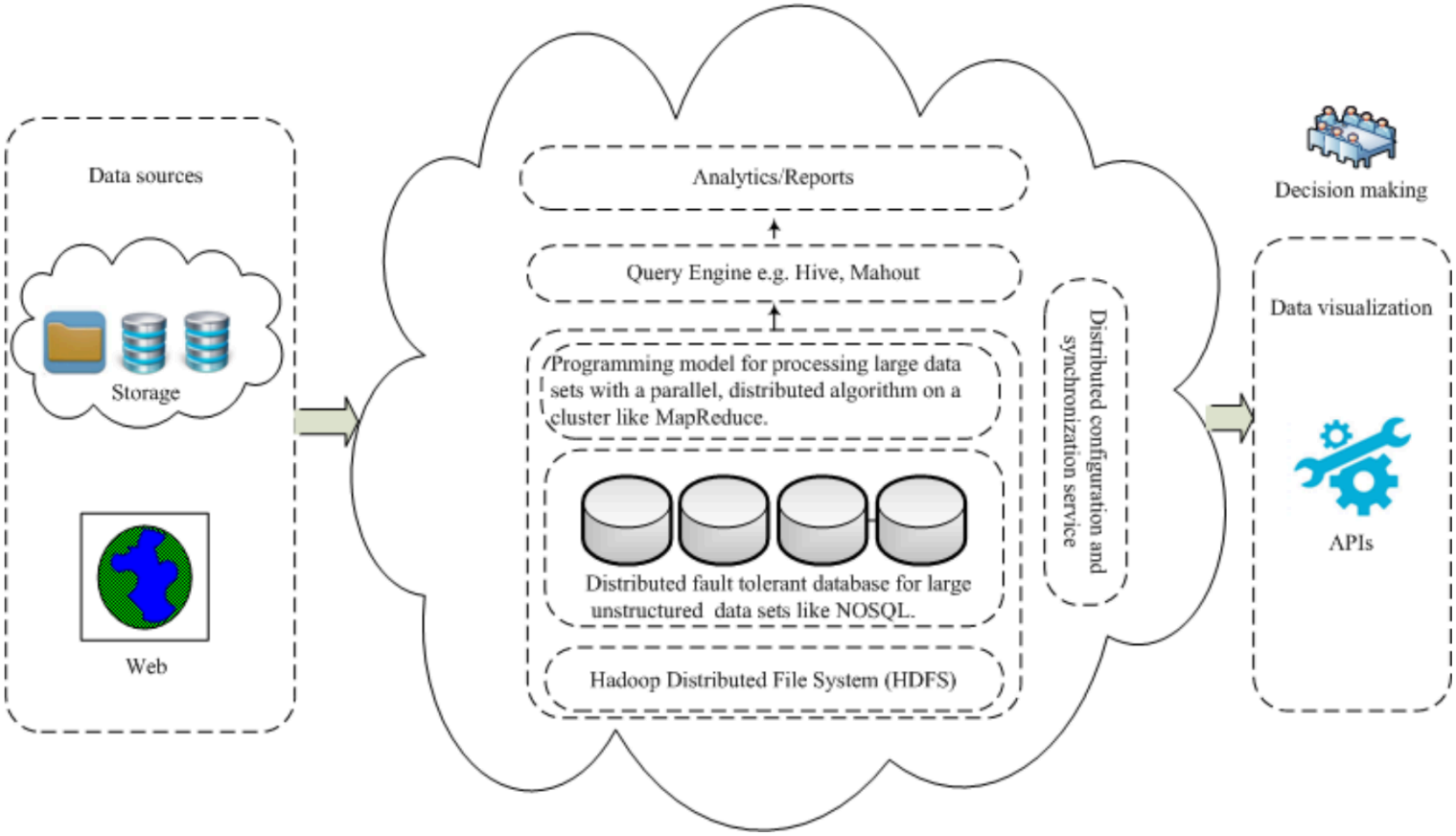
NIST cloud computing reference architecture

https://ws680.nist.gov/publication/get_pdf.cfm?pub_id=909505

Cloud Computing Services



Cloud Computing for Big Data



https://www.sciencedirect.com/science/article/abs/pii/S0306437914001288

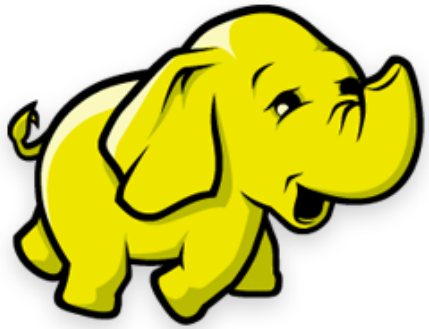
Comparison of Big Data Cloud Platforms

	Google	Microsoft	Amazon	Cloudera
Big data storage	Google cloud services	Azure	S3	
MapReduce	AppEngine	Hadoop on Azure	Elastic MapReduce (Hadoop)	MapReduce YARN
Big data analytics	BigQuery	Hadoop on Azure	Elastic MapReduce (Hadoop)	Elastic MapReduce (Hadoop
Relational database	Cloud SQL	SQL Azure	MySQL or Oracle	MySQL, Oracle, PostgreSQL
NoSQL database	AppEngine Datastore	Table storage	DynamoDB	Apache Accumulo
Streaming processing	Search API	Streaminsight	Nothing prepackaged	Apache Spark

Excerpt from:
https://www.sciencedirect.com/science/article/abs/pii/S0306437914001288

Big Data Technologies Overview

Hadoop



source

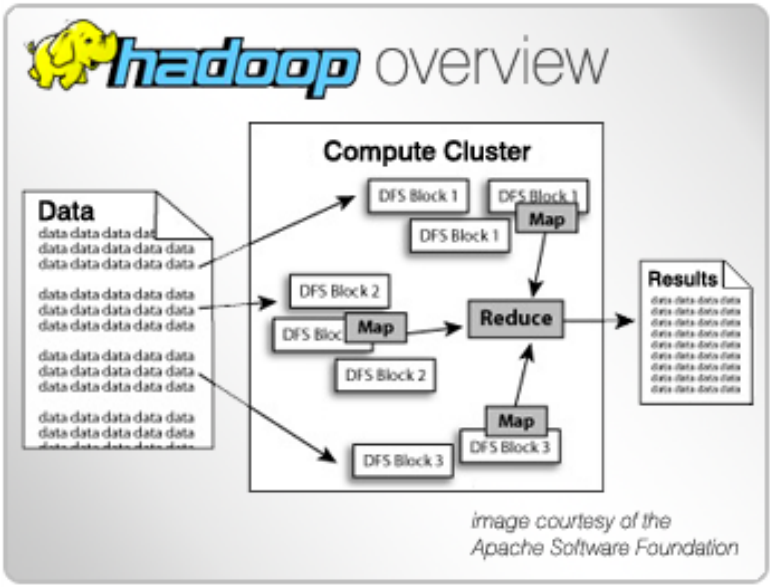
- Open-source data storage and processing platform
- Before the advent of Hadoop, storage and processing of big data was a big challenge
- Massively scalable, automatically parallelizable
  - Based on work from Google
    - Google: GFS + MapReduce + BigTable (Not open)
    - Hadoop: HDFS + Hadoop MapReduce + Hbase (opensource)
- Named by Doug Cutting in 2006 (worked at Yahoo! at that time), after his son's toy elephant

What's offered by Hadoop?

- Redundant, fault-tolerant data storage
- Parallel computation framework
- Job coordination
- Programmers do not need to worry about:
  - Where are files located?
  - How to handle failures and data loss?
  - How to distribute computation?
  - How to program for scaling?

Why use Hadoop?

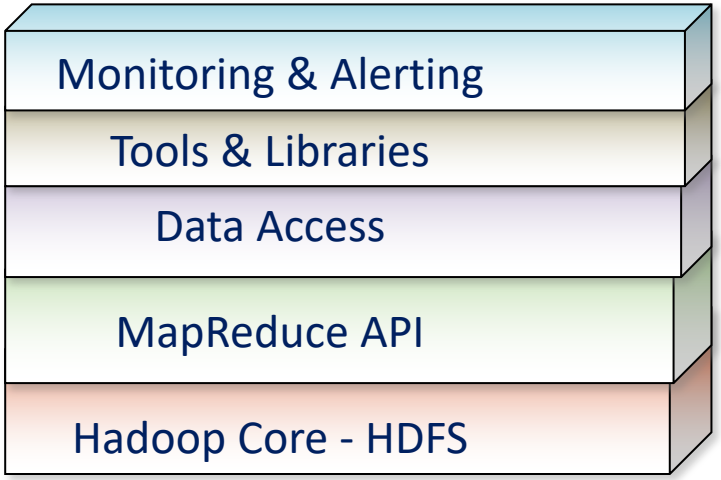
- Cheaper
  - Scales to Petabytes or more easily
- Faster
  - Parallel data processing
- Better
  - Suited for particular types of big data problems



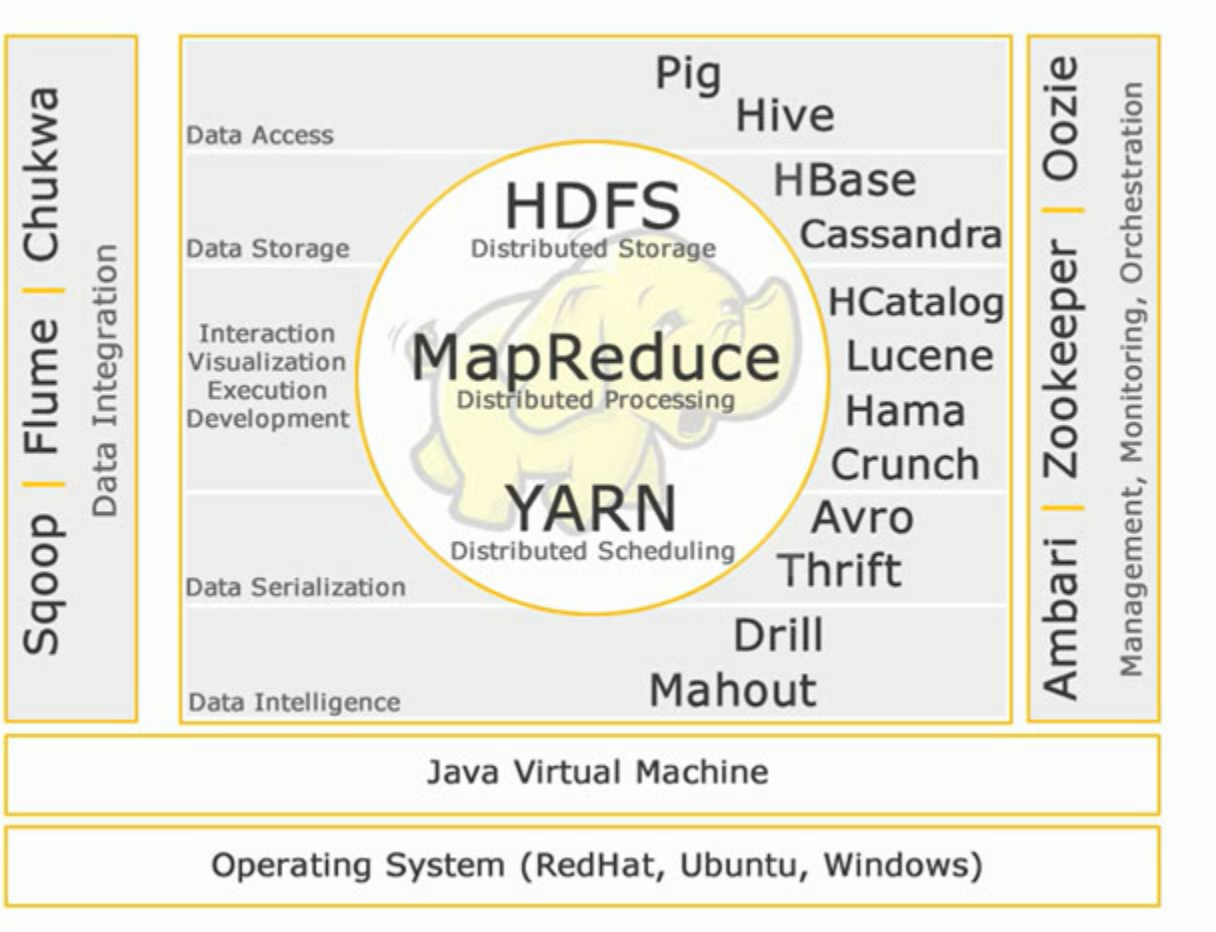
source

Hadoop is a set of Apache Frameworks...

- Data storage (HDFS)
  - Runs on commodity hardware
  - Horizontally scalable
- Processing (MapReduce)
  - Parallelized (scalable) processing
  - Fault Tolerant
- Other Tools / Frameworks
  - Data Access
    - Hbase (column store), Hive (Data warehouseing), Pig (high-level language on top of Hadoop), Mahout (library for ML / Data Analytics)
  - Tools
    - Hue (SQL Cloud editor), Sqoop (data transfer Hadoop/Structured stores)

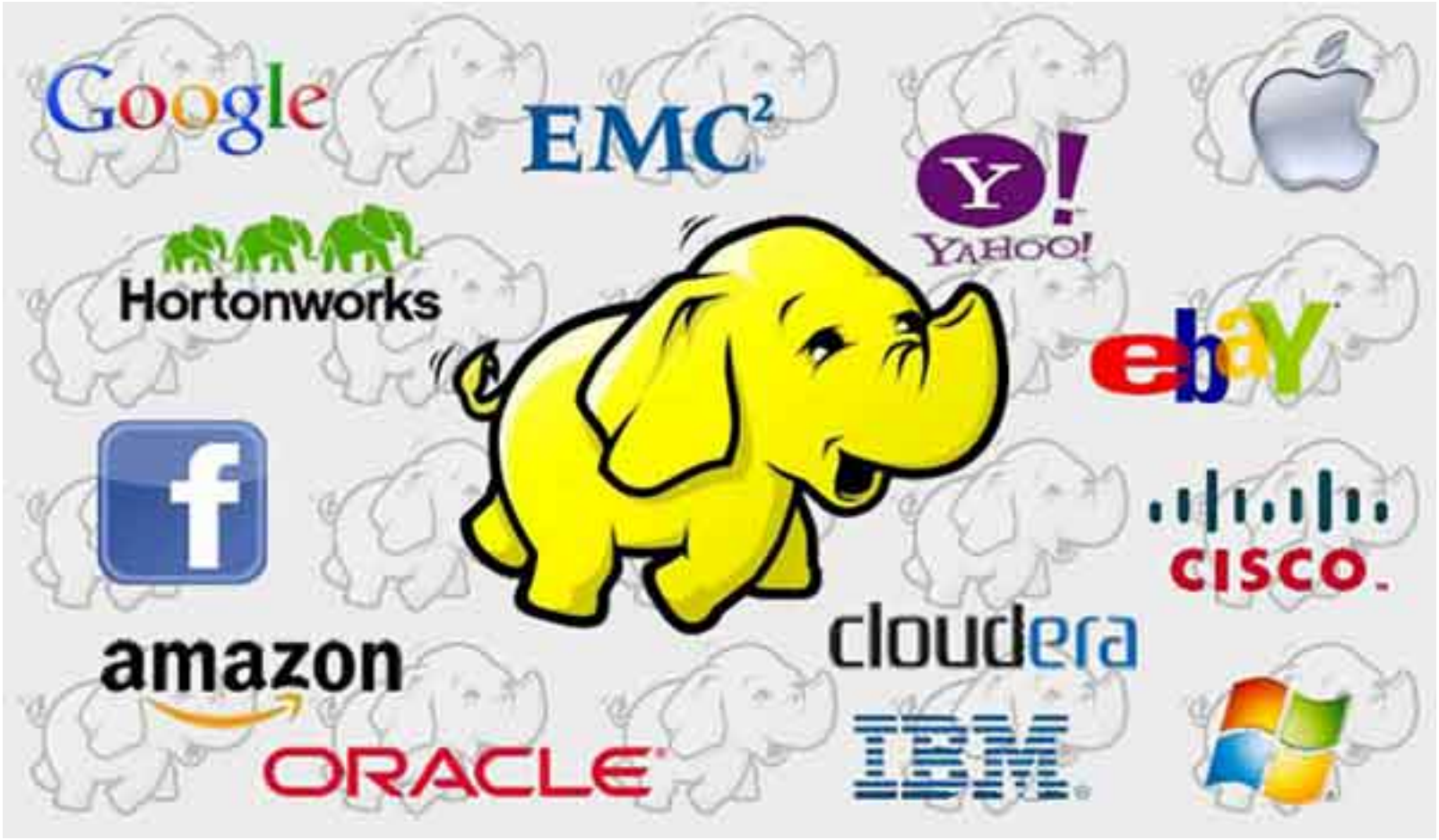


Hadoop ecosystem



source

Companies using Hadoop



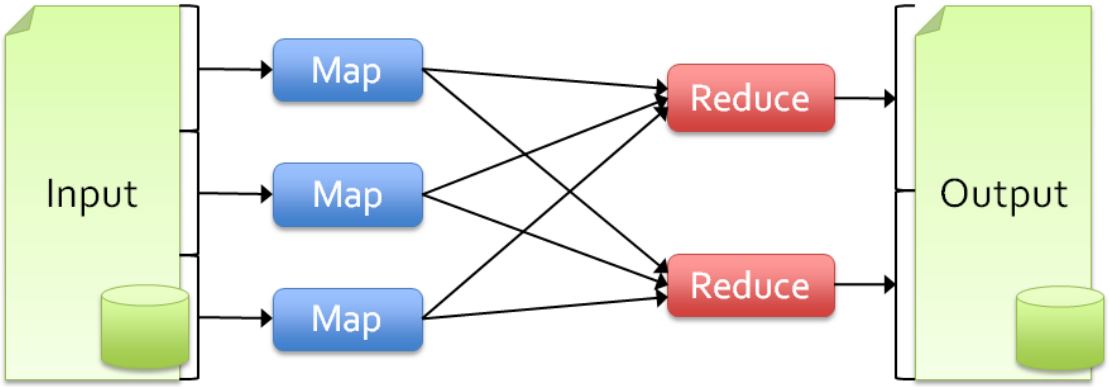
source

Spark

source

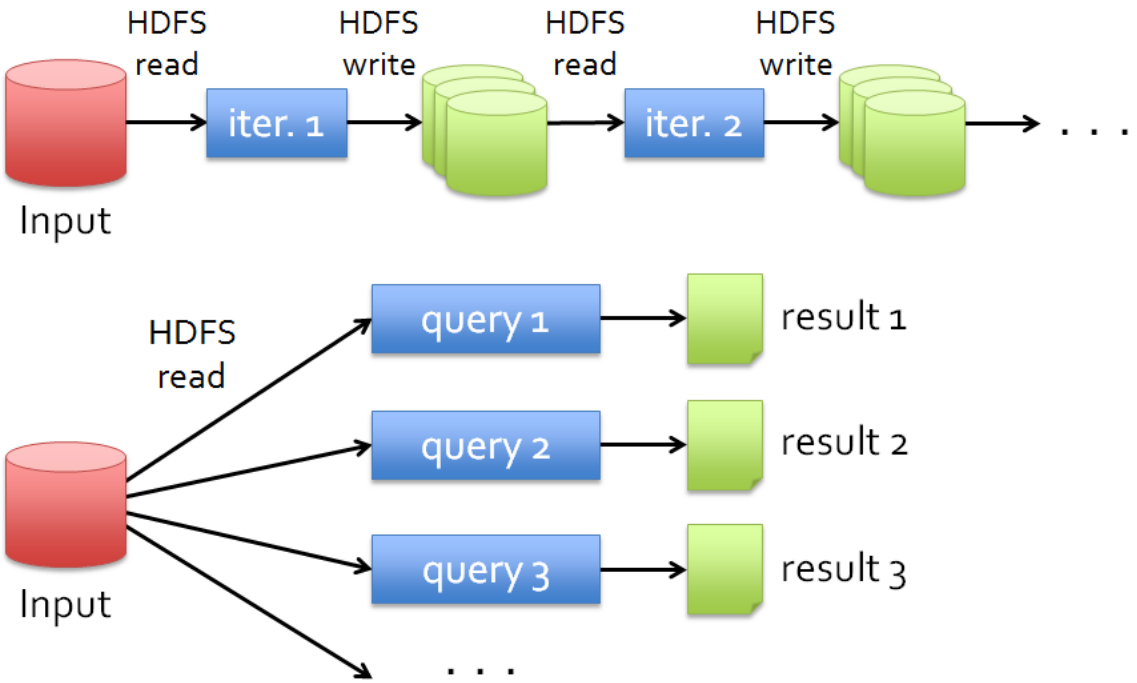
- MapReduce greatly simplified big data analysis on large, unreliable clusters. It is great at one-pass computation
- But as soon as it got popular, users wanted more:
  - More complex, multi-pass analytics (e.g. ML, graph)
  - More interactive ad-hoc queries
  - More real-time stream processing
- All 3 need faster data sharing across parallel jobs
  - One reaction: specialized models for some of these apps, e.g.,
    - Pregel (graph processing)
    - Storm (stream processing)

Limitations of MapReduce



- As a general programming model:
  - It is more suitable for one-pass computation on a large dataset
  - Hard to compose and nest multiple operations
  - No means of expressing iterative operations
- As implemented in Hadoop
  - All datasets are read from disk, then stored back on to disk
  - All data is (usually) triple-replicated for reliability
  - Not easy to write MapReduce programs using Java

Data sharing in MapReduce



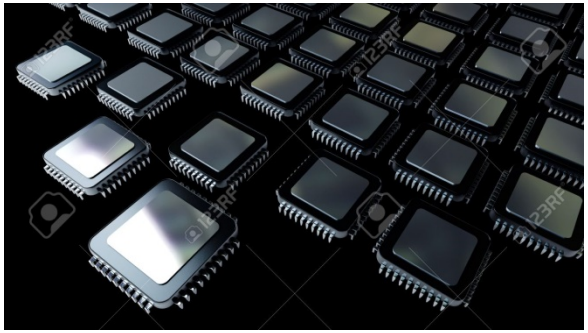
Slow due to replication, serialization, and disk IO

- Complex apps, streaming, and interactive queries all need one thing that MapReduce lacks:

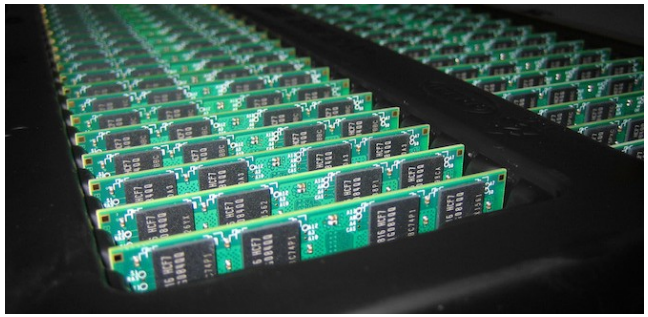
  Efficient primitives for **data sharing**

Hardware for Big Data

Lots of hard drives



Lots of CPUs



And lots of memory!

Goals of Spark

- Keep more data in-memory to improve the performance!
- Extend the MapReduce model to better support two common classes of analytics apps:
  - Iterative algorithms (machine learning, graphs)
  - Interactive data mining
- Enhance programmability:
  - Integrate into Scala programming language
  - Allow interactive use from Scala interpreter

Spark in Summary

- Fast and expressive cluster computing system interoperable with Apache Hadoop
- Improves efficiency through:
  - In-memory computing primitives → Up to 100 × faster (10 × on disk)
  - General computation graphs
- Improves usability through:
  - Rich APIs in Scala, Java, Python
  - Interactive shell → Often 5 × less code

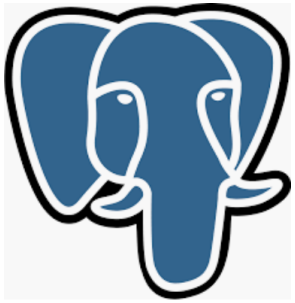
- **Spark is not**
  - a modified version of Hadoop
  - dependent on Hadoop because it has its own cluster management. Spark uses Hadoop for storage purpose only

NoSQL Technologies

What is NoSQL?

- The name stands for Not Only SQL
- Does not use SQL as its primary querying language
- Class of non-relational data storage systems
- The term NoSQL was introduced by Eric Evans (Apache Cassandra committer) when an event was organized to discuss open source distributed databases
- It's not a replacement for a RDBMS but compliments it
- NoSQL offerings typically relax one or more of the ACID properties

Traditional RDBMS



source

- Relational model with schemas
- Powerful, flexible query language (SQL)
- Transactional semantics: ACID
  - **A**tomicity
  - **C**onsistency
  - **I**solation
  - **D**urability
- Rich ecosystem, lots of tool support (MySQL, PostgreSQL, etc.)

NoSQL Key Features

- Non-relational
- Do not require strict schema
- Data can be replicated to multiple nodes (so, identical & fault-tolerant) and can be partitioned:
- Down nodes easily replaced
- No single point of failure
- Horizontal scalable
- Cheap, easy to implement (open-source)
- Massive write performance
- Fast key-value access

Why NoSQL?

- Web apps have different needs (than the apps that RDBMS were designed for)
  - Low and predictable response time (latency)
  - Scalability & elasticity (at low cost!)
  - High availability
  - Flexible schemas / semi-structured data
  - Geographic distribution (multiple datacenters)
- Web apps can (usually) do without
  - Transactions / strong consistency / integrity
  - Complex queries

Who are using NoSQL?

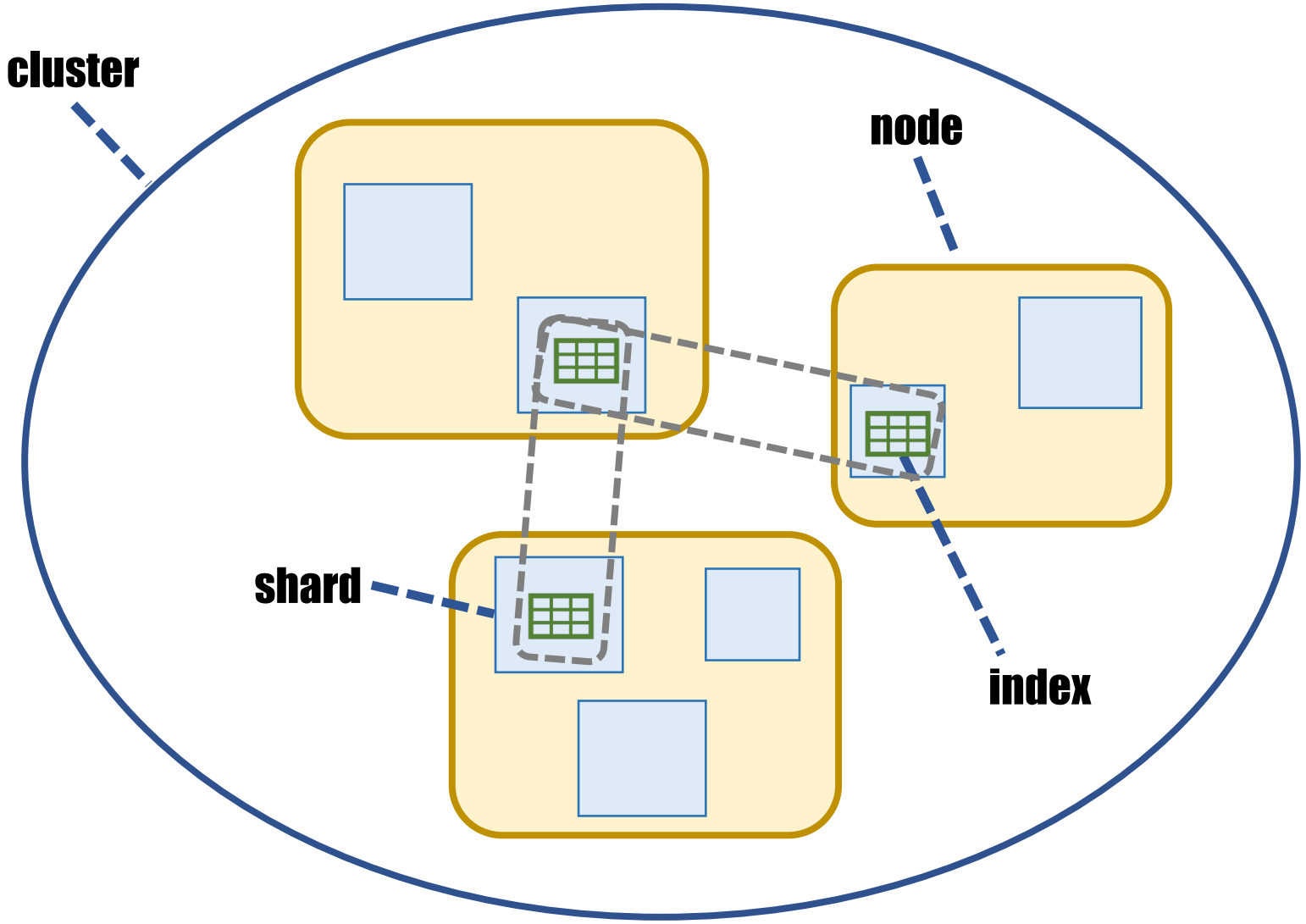
- Google (BigTable)
- LinkedIn (Voldemort)
- Facebook (Cassandra)
- Twitter (HBase, Cassandra)
- Baidu (HyperTable)

Elasticsearch

source

- Open source search engine based on Apache Lucene
- Initial release in 2010
- Provides a distributed, full-text search engine with a REST APIs
- Document oriented (JSON as serialization format for documents)
- Developed in Java (cross platform)
- Focused on scalability – distributed by design

Elasticsearch Ecosystem



Elasticsearch Ecosystem

- Cluster
  - An Elasticsearch cluster is a collection of nodes (servers)
  - Identified by a unique name
  - Data is stored in this collection of nodes
  - Provide indexing and search capabilities across all nodes
- Node
  - A single server in the cluster
  - Identified by a unique name
  - Stores all or parts of the whole dataset
  - Contributes to the indexing and search capabilities of Elasticsearch

Elasticsearch Ecosystem

- Shard
  - Individual instances of Lucene index
  - Elasticsearch leverages Lucene indexing in a distributed system
- Index
  - Distributed across shards
  - Replicas (fault tolerance)

Individual instances of Lucene index Ø Elasticsearch leverages Lucene indexing in a distributed system

Elasticsearch Use Cases

- E-commerce
  - Online web stores
  - Fast search for products
  - Autocomplete suggestions
- Storage, analysis and mining of transaction data
  - Trends
  - Statistics
  - Summarizations
- Analytics/Business intelligence
  - Investigation
  - Analysis
  - Visualization
  - Ad-hoc business questions

Who Uses Elasticsearch?

source

www.elastic.co/use-cases