

# Large-scale Data Systems

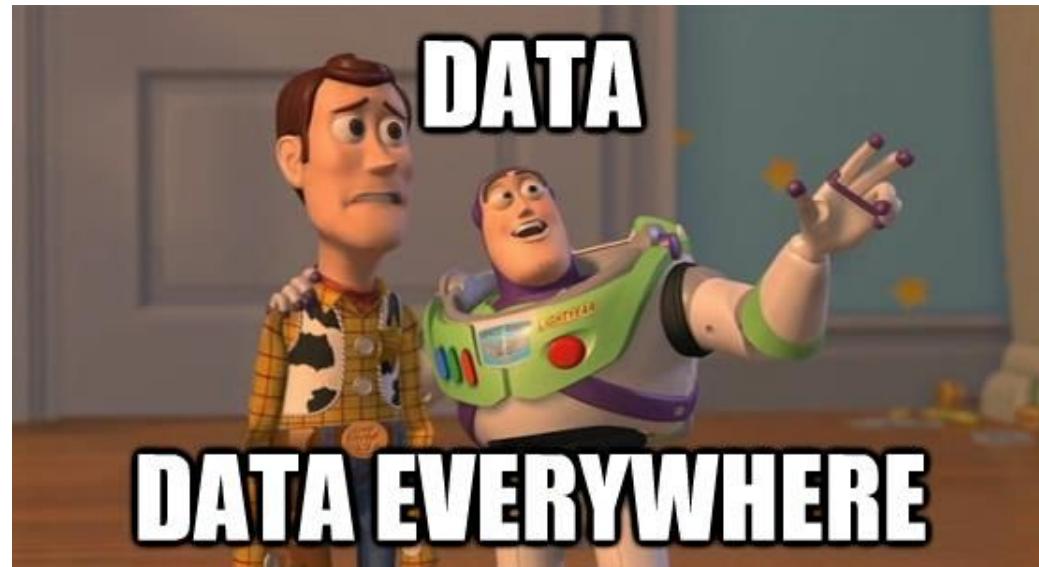
Lecture 1: Introduction

Prof. Gilles Louppe  
[g.louppe@uliege.be](mailto:g.louppe@uliege.be)



# **The data science era**

Big data? Data science?



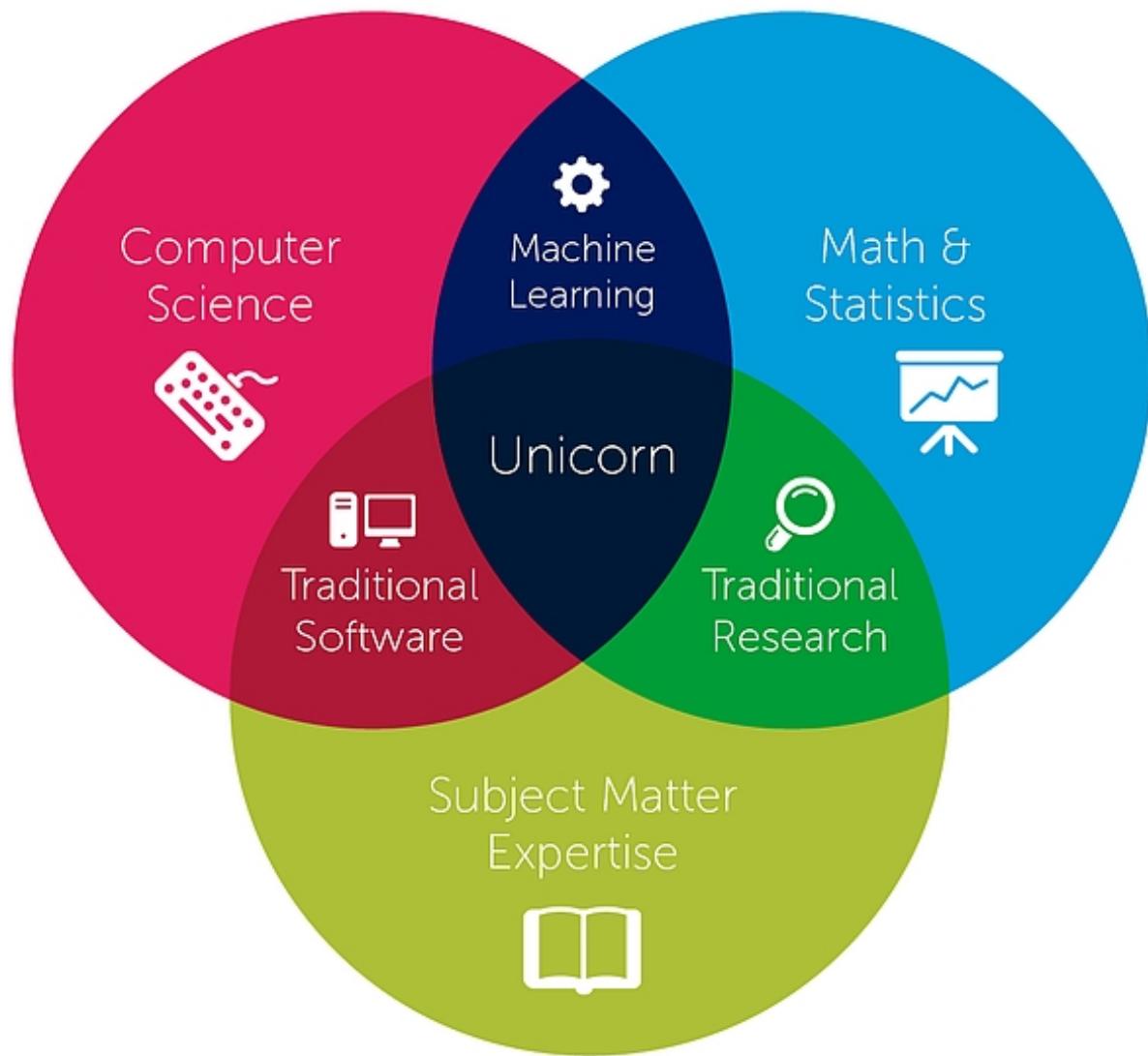
hype **vs.** business **vs.** science

*"A data scientist is someone who knows more statistics than a computer scientist and more computer science than a statistician."*

Josh Blumenstock

*"Data scientist = statistician + programmer + coach + storyteller + artist"*

Shlomo Aragmon





*Nate Silver*

# FiveThirtyEight Forecast

Updated 12:27 AM ET on Oct. 1



Barack Obama

320.1

+10.7 since Sept. 23

President  
Now-cast

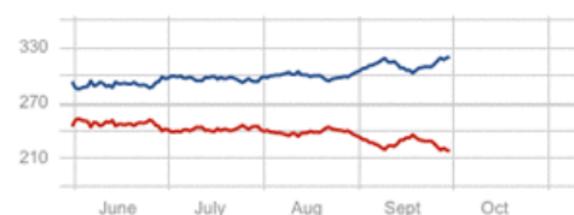
Senate  
Nov. 6 Forecast

Mitt Romney

217.9

-10.7 since Sept. 23

Electoral  
vote



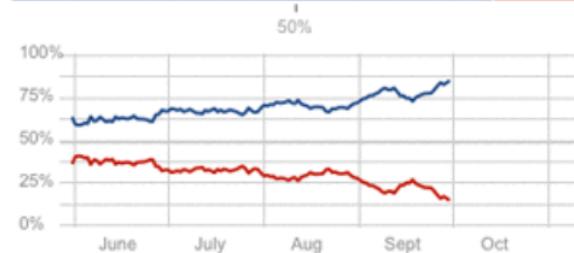
85.1%

+7.5 since Sept. 23

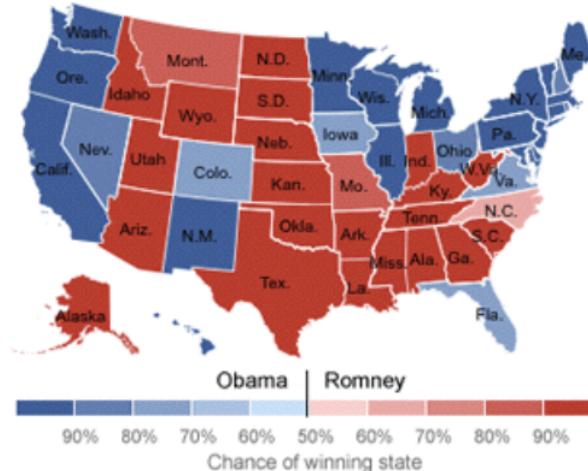
Chance of  
Winning

14.9%

-7.5 since Sept. 23

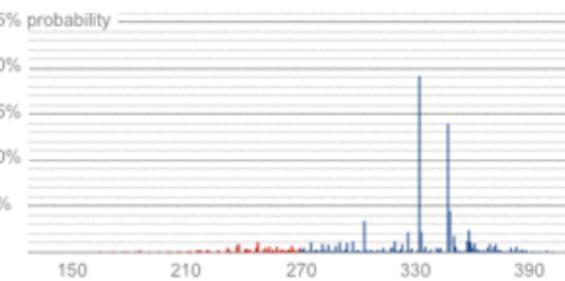


## State-by-State Probabilities

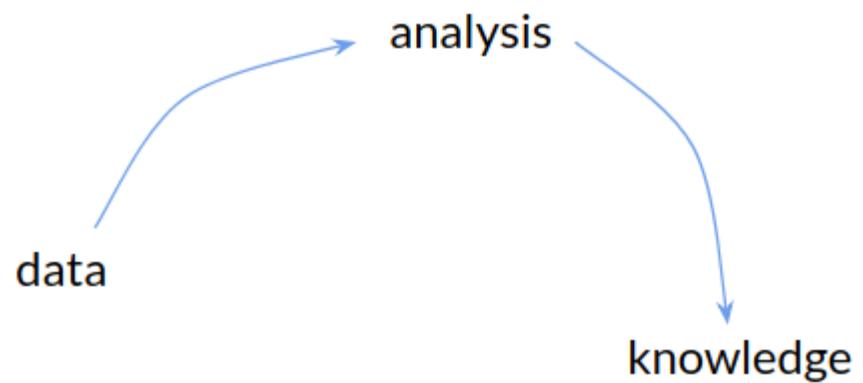


## Electoral Vote Distribution

The probability that President Obama receives a given number of Electoral College votes.



*"Nate Silver won the election" - Harvard Business Review*





*Haven't we be doing data analysis forever?*



*"Every two days now we create as much information as we did from the dawn of civilization up until 2003, according to Schmidt. That's something like five exabytes of data, he says.*

*Let me repeat that: we create as much information in two days now as we did from the dawn of man through 2003.*

Eric Schmidt, 2010.

# 1 Zettabyte (ZB) = 1 Trillion Gigabytes (GB)

We face an overwhelming amount of data in every industry

**>2.5 PB**

of customer data  
stored by Walmart  
**every hour.**

**292 exabytes**

of mobile traffic by  
2019, up from 30  
**exabytes** in 2014.

**1 TB**

of data produced  
by a cancer patient  
**every day.**

2010

2018

2025

Today

44 Zettabytes

Gap

Sensors & Devices

Images/Multimedia

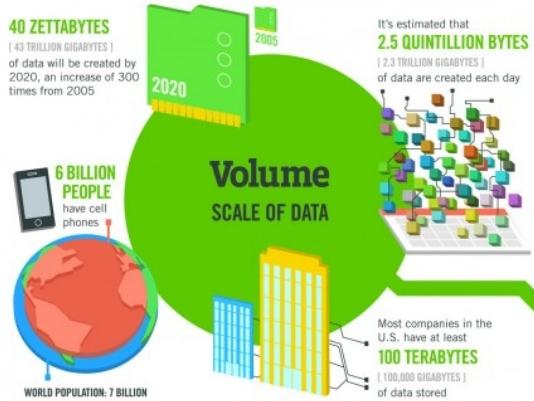
Text

Enterprise Data

Traditional

Source

© 2018 Dvmobile Inc. All Rights Reserved



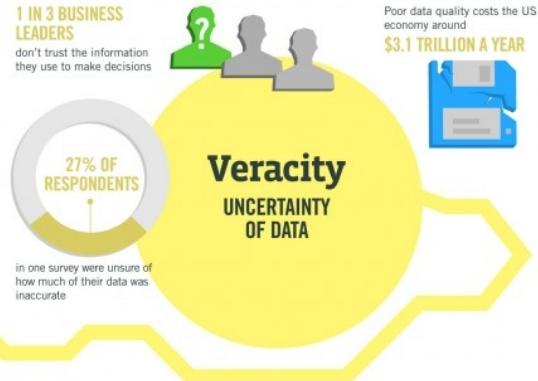
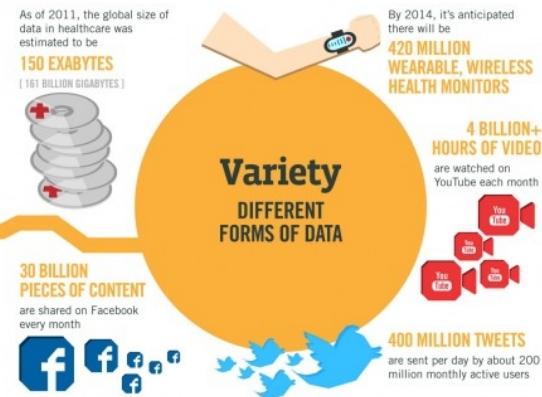
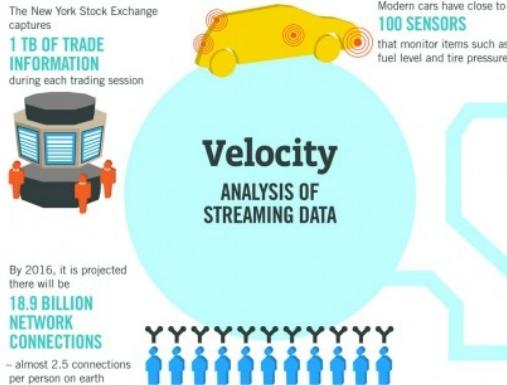
# The FOUR V's of Big Data

From traffic patterns and music downloads to web history and medical records, data is recorded, stored, and analyzed to enable the technology and services that the world relies on every day. But what exactly is big data, and how can these massive amounts of data be used?

As a leader in the sector, IBM data scientists break big data into four dimensions: **Volume, Velocity, Variety, and Veracity**.

Depending on the industry and organization, big data encompasses information from multiple internal and external sources such as transactions, social media, enterprise content, sensors and mobile devices. Companies can leverage data to adapt their products and services to better meet customer needs, optimize operations and infrastructure, and find new sources of revenue.

By 2015  
**4.4 MILLION IT JOBS**  
will be created globally to support big data, with 1.9 million in the United States



Sources: McKinsey Global Institute, Twitter, Cisco, Gartner, EMC, SAS, IBM, MPTEC, QAS

IBM

Actually, **none of that is really new**... What is new is:

- our ability to **store machine generated data**, at unprecedented scale and rate.
- the broad understanding that **we cannot just manually get value out of data**.



*The*  
**F O U R T H  
P A R A D I G M**

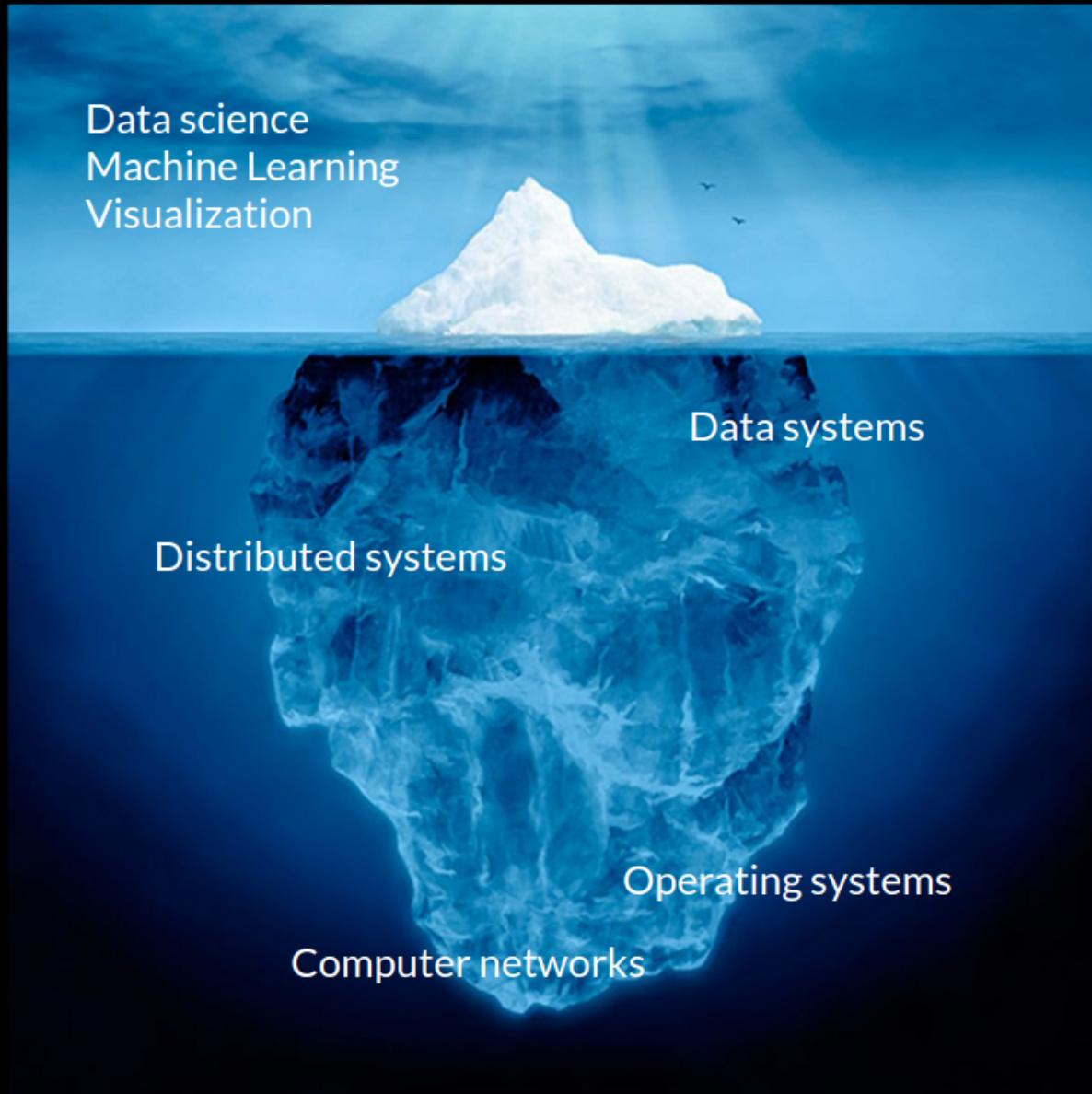
DATA-INTENSIVE SCIENTIFIC DISCOVERY

EDITED BY TONY HEY, STEWART TANSLEY, AND KRISTIN TOLLE

*"Increasingly, scientific breakthroughs will be powered by advanced computing capabilities that help researchers manipulate and explore massive datasets.*

*The speed at which any given scientific discipline advances will depend on how well its researchers collaborate with one another, and with technologists, in areas of eScience such as databases, workflow management, visualization, and cloud computing technologies."*

# Data systems



# Operating systems

Can you name examples of **operating systems**?

# Operating systems

Can you name examples of **operating systems**?

- Android
- Chrome OS
- FreeBSD
- iOS
- macOS
- OS/2
- RISC OS
- Solaris
- Windows
- ...

## **Definition**

The low-level software which handles the interface to peripheral hardware, schedules tasks, allocates storage, and presents a default interface to the user when no application program is running.

# Distributed systems

Can you name examples of **distributed systems**?

# Distributed systems

Can you name examples of **distributed systems**?

- A client/server system
- The web
- Wireless networks
- Telephone networks
- DNS
- Massively multiplayer online games
- Distributed databases
- BitTorrent (peer-to-peer overlays)
- A cloud, e.g. Amazon EC2/S3, Microsoft Azure
- A data center, e.g. a Google data center, AWS
- The bitcoin network

## Definition

A distributed system is a collection of entities with a common goal, each of which is **autonomous, programmable, asynchronous** and **failure-prone**, and which communicate through an **unreliable** communication medium.

- **Entity**: a process on a device.
- **Communication medium**: Wired or wireless network.

A distributed system appears to its users as a **single coherent** system.

# Internet



*What are the entities? What is the communication medium?*

## Data center



*What are the entities? What is the communication medium?*

# Why study distributed systems?

- Distributed systems are **everywhere**:
  - Internet
  - WWW
  - Mobile devices
  - Internet of Things
- **Technical** importance:
  - Improve **scalability**
    - Adding computational resources to a system is an easy way to scale its performance to many users.
  - Improve **reliability**
    - We want high availability and durability of the system.

- Distributed systems are **difficult** to build.
  - **Scale:** hundreds or thousands of machines.
    - Google: 4k-machine MapReduce cluster
    - Facebook: 60k machines providing the service
  - **Fault tolerance:** machines and networks do fail!
    - 50 machine failures out of 20k machine cluster per day (reported by Yahoo!)
    - 1 disk failure out of 16k disks every 6 hours (reported by Google)
  - **Concurrency:**
    - Nodes execute in parallel
    - Messages travel asynchronously
  - **Consistency:**
    - Distributed systems need to ensure user guarantees about the data they store.
    - E.g., all read operations return the same value, no matter where it is stored.
- But only a few **core problems** reoccur.

# Teaser: Two Generals' Problem

Two generals need to coordinate an attack.

- They must **agree** on time to attack.
- They will win only if they attack **simultaneously**.
- They communicate through **messengers**.
- Messengers may be **killed** on their way.



Let's try to solve the problem for generals  $g_1$  and  $g_2$ .

- $g_1$  sends time of attack to  $g_2$ .
- Problem: how to ensure  $g_2$  received the message?
- Solution: let  $g_2$  acknowledge receipt of message.
- Problem: how to ensure  $g_1$  received the acknowledgment?
- Solution: let  $g_1$  acknowledge receipt of acknowledgment.
- ...

This problem is **impossible** to solve!

(Unless we make additional assumptions)

- Applicability to distributed systems:
  - Two nodes need to **agree** on a **value**.
  - They communicate by **messages** using an **unreliable channel**.
- **Agreement** is one of the core problems of distributed systems.

# Data systems

Can you name examples of **data systems**?

# Data systems

Can you name examples of **data systems**?

- A database
- A file system
- A ledger
- Search engines
- Data flow frameworks
- Social networks
- ...

## **Definition**

In this course, data systems will broadly refer to any kind of computer systems, distributed or not, that can be used to store, retrieve, organize or process data.

Our main focus will be on data systems for data science purposes.

BIG DATA & AI LANDSCAPE 2018



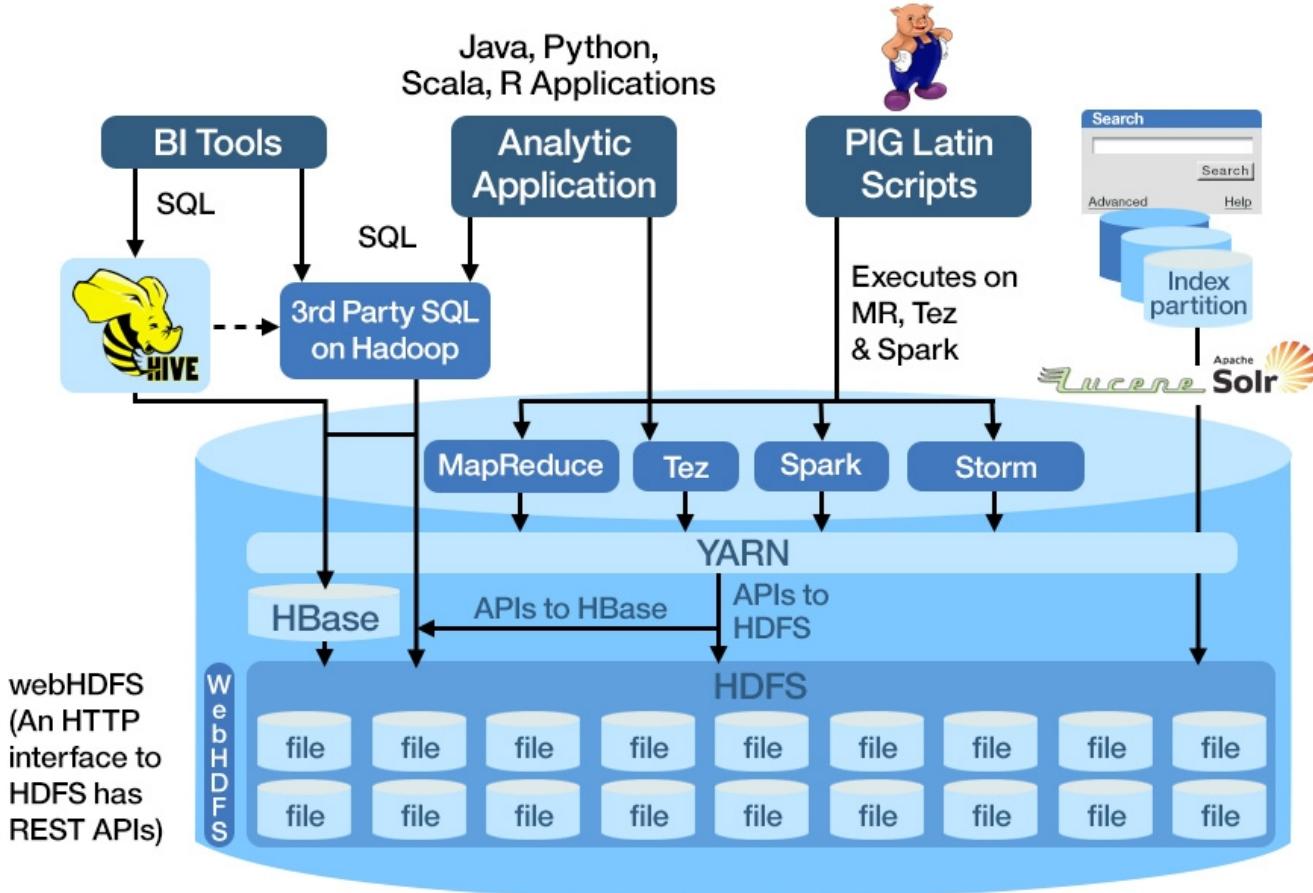
Final 2018 version, updated 07/15/2018

© Matt Turck (@mattturck), Demi Obavomi (@demi\_obavomi), & FirstMark (@firstmarkcap)

mattturck.com/bigdata2018

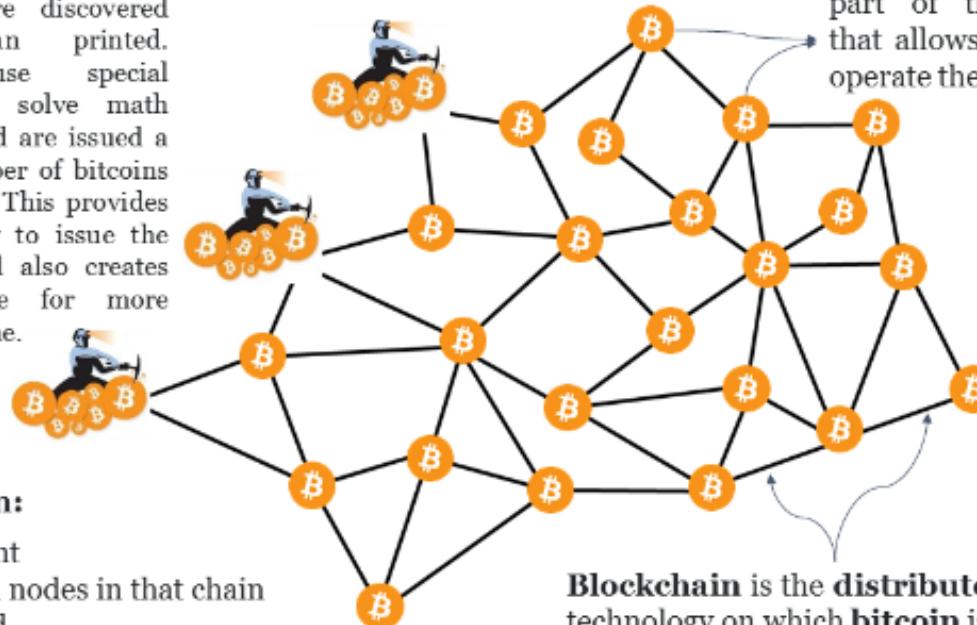
FIRSTMARK  
EARLY STAGE VENTURE CAPITAL

# The Hadoop ecosystem



## A distributed ledger

**Bitcoins** are discovered rather than printed. **Miners** use special software to solve math problems and are issued a certain number of bitcoins in exchange. This provides a smart way to issue the currency and also creates an incentive for more people to mine.



# Outline

# Fundamentals of distributed systems

Understand the **foundational principles** required for the **design, implementation** and **maintenance** of distributed systems.

- Communications
- Failures
- Consistency
- Concurrency
- Consensus

## Communications

- How do you talk to another machine?
  - Reliable networking.
- How do you talk to multiple machines at once, with ordering guarantees?
  - Multicast, Gossiping.

## Failures and consistency

- How do you know if a machine has failed?
  - Failure detection.
- How do you program your system to operate continually even under failure?
  - Gossiping, replication.
- What if some machines do not cooperate?
  - Byzantine fault tolerance.

## Concurrency

- How do you control access to shared resources?
  - Distributed mutual exclusion, distributed transactions, etc.

## Consensus

- How do multiple machines reach an agreement?
  - Time and synchronization, global states, leader election, Paxos, proof of work, blockchain.
- **Bad news:** it is impossible!
  - The impossibility of consensus for asynchronous systems.

# Case studies

From these building blocks, understand how to build and architecture data systems for large volumes or data or for data science purposes.

## Distributed storage

- How do you locate where things are and access them?
  - Distributed file systems
  - Key-value stores
- How do you record and share sensitive data?
  - Proof of work, blockchain

## Distributed computing for data science

- What are the distributed computing systems for data science?
  - Map Reduce (Hadoop)
  - Computational graph systems (Spark, Dask)
- Is distributed computing always necessary?



# References

- Hey, Tony, Stewart Tansley, and Kristin M. Tolle. The fourth paradigm: data-intensive scientific discovery. Vol. 1. Redmond, WA: Microsoft research, 2009.
- Kersten, Martin L., et al. "The researcher's guide to the data deluge: Querying a scientific database in just a few seconds." PVLDB Challenges and Visions 3.3 (2011).
- Silver, Nate. The signal and the noise: the art and science of prediction. Penguin UK, 2012.