Advanced Data Management for Data Analysis

Stefan Manegold

Data Management @ LIACS

Group leader Database Architectures
Centrum Wiskunde & Informatica (CWI)
Amsterdam

s.manegold@liacs.leidenuniv.nl
http://www.cwi.nl/~manegold/

ADM

Stefan Manegold

Group leader Database Architectures
Centrum Wiskunde & Informatica (CWI)
Amsterdam
http://homepages.cwi.nl/~manegold/

http://www.monetdb.org/

https://duckdb.org/

Prof. Data Management (0.2 fte)
LIACS & LCDS
Faculty of Science, Leiden University

ADM: Logistics

Period: September 07 2022 - November 30 2022 (Wednesdays)
(13 lecture slots in total; 12 actual lectures, one slot skipped)

Time: **11:00 – 12:45**

Place: room **204** in the **"Huygens"** building
(Niels Bohrweg 2, 2333 CA Leiden)

Grading:

4 Assignments, each 25%; 2 individually, 2 in groups of 3-5 students

ADM: Expected Background

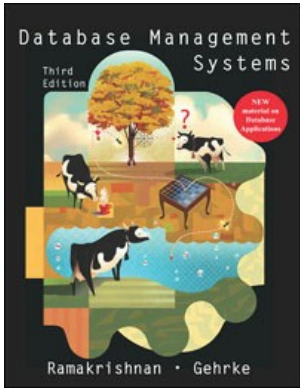
- **standard (relational) database concepts, techniques, algorithms, including**
  - Entity-Relationship model
  - **Relational Data model**
  - **Relational algebra**
  - **SQL query language**
  - transaction management
  - **storage formats and access structures (indexes)**
  - **Query optimization and evaluation**

- **programming experience in**
  - **system-oriented programming languages like C or C++**
  - **Scripting languages like Python**
  - Statistical languages like R

ADM: Expected Background

- **Database systems (e.g.):**

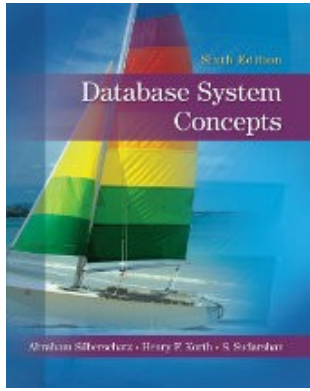
  - *Ramakrishnan*, *Gehrke*: **Database Management Systems (3rd International Edition)**, McGraw-Hill, 2003 (ISBN 0-07-246563-8)



  - *A. Silberschatz*, *H. F. Korth*, *S. Sudarshan:* **Database System Concepts (7th Edition),** McGraw-Hill, 2010 (ISBN 0-07-352332-1)

    book: https://www.db-book.com/

    slides: https://www.db-book.com/slides-dir/index.html



  - *Andy Pavlo:* **Introduction to Database Systems** course @ CMU incl. slides and videos on YouTube: https://15445.courses.cs.cmu.edu/fall2019/

ADM: Background Poll

- **Who is familiar with**
  - Entity-Relationship model
  - **Relational Data model**
  - **Relational algebra**
  - **SQL query language**
  - transaction management
  - **storage formats and access structures (indexes)**
  - **Query optimization and evaluation**

ADM: Background Poll

- **Who has programming experience in**
  - **system-oriented programming languages like C or C++**
  - **Scripting languages like Python**
  - Statistical languages like R
  - Other?

ADM: Background Poll

- Who has experiences using database management systems?
- Which ones?

- Who has experiences with managing / analyzing data sets?
  - Using which tools / platforms?
  - How large is the largest dataset you used?
  - What kind of data / format / complexity?

ADM: Agenda (planned)

- 07.09.2022: Lecture 1: **Introduction**
- 14.09.2022: Lecture 2: **SQL Recap**

  (plus Assignment 1 [in groups; 3 weeks]: TPC-H benchmark)
- 21.09.2022: Lecture 3: **Column-Oriented Database Systems (1/6) - Motivation & Basic Concepts**
- 28.09.2022: Lecture 4: **Column-Oriented Database Systems (2a/6) - Selected Execution Techniques (1/2)**
- 05.10.2022: Lecture 5: **Column-Oriented Database Systems (2b/6) - Selected Execution Techniques (2/2)**

  (plus Assignment 3 [in groups; 3 weeks]: Compression techniques)
- 12.10.2022: Lecture 6: **Column-Oriented Database Systems (3/6) - Cache Conscious Joins**
- 19.10.2022: Lecture 7: **Column-Oriented Database Systems (4/6) - "Vectorized Execution"**
- ~~26.10.2022~~: ***No lecture!***
- 28.09.2022: Lecture 8: **DuckDB: An embedded database for data science (1/2) (guest lecture & *hands-on*)**

  (plus Assignment 2 [individual; 2 weeks]: Analysing NYC Cab dataset with DuckDB)
- 05.10.2022: Lecture 9: **DuckDB: An embedded database for data science (2/2) (guest lecture & *hands-on*)**
- 16.11.2022: Lecture 10: **Branch Misprediction & Predication**

  (plus Assignment 4 [individual; 2 weeks]: Predication)
- 23.11.2022: Lecture 11: **Column-Oriented Database Systems (5/6) - Adaptive Indexing**
- 30.11.2022: Lecture 12: **Column-Oriented Database Systems (6/6) - Progressive Indexing**

Bring your own laptop!

ADM: Literature (1/6)

- **Column-Oriented Database Systems (1/6) - Motivation & Basic Concepts**
  - “An overview of cantor: a new system for data analysis“. Ilkka Karasalo, Per Svensson. SSDBM 1983.
  - “A decomposition storage model“. George P. Copeland, Setrag Khoshafian. SIGMOD Conference, 1985.
  - “Cache Conscious Algorithms for Relational Query Processing“. Ambuj Shatdal, Chander Kant, Jeffrey F. Naughton. VLDB 1994.
  - “MIL Primitives for Querying a Fragmented World“. Peter A. Boncz, Martin L. Kersten. VLDB J. 8(2): 101-119, 1999.
  - “Database Architecture Optimized for the New Bottleneck: Memory Access“. Peter A. Boncz, Stefan Manegold, Martin L. Kersten. VLDB 1999.
  - “DBMSs On A Modern Processor: Where Does Time Go?“. Anastassia Ailamaki, David J. DeWitt, Mark D. Hill, David A. Wood. VLDB 1999.
  - “Weaving Relations for Cache Performance ("PAX")“. Anastassia Ailamaki, David J. DeWitt, Mark D. Hill, Marios Skounakis. VLDB 2001.
  - “A Case for Fractured Mirrors“. Ravishankar Ramamurthy, David J. DeWitt, Qi Su. VLDB 2002.
  - “Data Morphing: An Adaptive, Cache-Conscious Storage Technique“. Richard A. Hankins, Jignesh M. Patel. VLDB 2003.
  - “Clotho: Decoupling Memory Page Layout from Storage Organization“. Minglong Shao, Jiri Schindler, Steven W. Schlosser, Anastassia Ailamaki, Gregory R. Ganger. VLDB 2004.
  - “MonetDB-X100 - A DBMS In The CPU Cache“. Marcin Zukowski, Peter A. Boncz, Niels Nes, Sándor Héman. IEEE Data Eng. Bull. 28(2): 17-22, 2005.
  - “"One size fits all": an idea whose time has come and gone“. Michael Stonebraker, Ugur Çetintemel. ICDE 2005.
  - “Performance Tradeoffs in Read-Optimized Databases“. Stavros Harizopoulos, Velen Liang, Daniel J. Abadi, Samuel Madden. VLDB 2006.
  - …

ADM: Literature (2/6)

- **Column-Oriented Database Systems (1/6) - Motivation & Basic Concepts** *(cont.)*
  - …
  - “One Size Fits All? - Part 2: Benchmarking Results“. Michael Stonebraker, Chuck Bear, Ugur Çetintemel, Mitch Cherniack, Tingjian Ge, Nabil Hachem, Stavros Harizopoulos, John Lifter, Jennie Rogers, Stanley B. Zdonik. CIDR 2007.
  - “C-Store: A Column-oriented DBMS“. Michael Stonebraker, Daniel J. Abadi, Adam Batkin, Xuedong Chen, Mitch Cherniack, Miguel Ferreira, Edmond Lau, Amerson Lin, Samuel Madden, Elizabeth J. O'Neil, Patrick E. O'Neil, Alex Rasin, Nga Tran, Stanley B. Zdonik. VLDB 2005.
  - “Breaking the memory wall in MonetDB“. Peter A. Boncz, Martin L. Kersten, Stefan Manegold. Commun. ACM 51(12): 77-85, 2008.
  - “Column-Stores vs Row-Stores: How Different are They Really?“. Daniel J. Abadi, Samuel Madden, Nabil Hachem. SIGMOD Conference 2008.
  - “DSM vs. NSM: CPU performance tradeoffs in block-oriented query processing“. Marcin Zukowski, Niels Nes, Peter A. Boncz. DaMoN 2008.
  - “Fast Scans and Joins Using Flash Drives“. Mehul A. Shah, Stavros Harizopoulos, Janet L. Wiener, Goetz Graefe. DaMoN 2008.
  - “Read-Optimized Databases, In-Depth“. Allison L. Holloway, David J. DeWitt. Proc. VLDB Endow. 1(1): 502-513, 2008.
  - “Teaching an Old Elephant New Tricks“. Nicolas Bruno. CIDR 2009.
  - “Query Processing Techniques for Solid State Drives“. Dimitris Tsirogiannis, Stavros Harizopoulos, Mehul A. Shah, Janet L. Wiener, Goetz Graefe. SIGMOD Conference 2009.
  - “MonetDB: Two Decades of Research in Column-oriented Database Architectures“. Stratos Idreos, Fabian Groffen, Niels Nes, Stefan Manegold, K. Sjoerd Mullender, Martin L. Kersten. IEEE Data Eng. Bull. 35(1): 40-45, 2012.
  - “The Vertica Analytic Database: C-Store 7 Years Later“. Andrew Lamb, Matt Fuller, Ramakrishna Varadarajan, Nga Tran, Ben Vandiver, Lyric Doshi, Chuck Bear. Proc. VLDB Endow. 5(12): 1790-1801, 2012.

ADM: Literature (3/6)

- **Column-Oriented Database Systems (2/6) - Selected Execution Techniques**
  - Compression
    - "Compressing Relations and Indexes". Goldstein, Ramakrishnan, Shaft. ICDE'98.
    - "Query optimization in compressed database systems". Chen, Gehrke, Korn. SIGMOD'01.
    - "Super-Scalar RAM-CPU Cache Compression". Zukowski, Heman, Nes, Boncz. ICDE'06.
    - "Integrating Compression and Execution in Column-Oriented Database Systems". Abadi, Madden, Ferreira. SIGMOD'06.
    - "Improved Word-Aligned Binary Compression for Text Indexing". Ahn, Moffat. TKDE'06.
  - Tuple Materialization
    - "Materialization Strategies in a Column-Oriented DBMS". Abadi, Myers, DeWitt, Madden. ICDE'07.
    - "Column-Stores vs Row-Stores: How Different are They Really?". Abadi, Madden, Hachem. SIGMOD'08.
    - "Query Processing Techniques for Solid State Drives". Tsirogiannis, Harizopoulos Shah, Wiener, Graefe. SIGMOD'09.
    - "Self-organizing tuple reconstruction in column-stores". Idreos, Manegold, Kersten. SIGMOD'09.
  - Join
    - "Fast Joins using Join Indices". Li and Ross. VLDBJ 8:1-24, 1999.

ADM: Literature (4/6)

- **Column-Oriented Database Systems (3/6) - Cache Conscious Joins**
  - "Cache Conscious Algorithms for Relational Query Processing". Shatdal, Kant, Naughton. VLDB'94.
  - "Fast Joins using Join Indices". Li and Ross. VLDBJ 8:1-24, 1999.
  - "Optimizing main-memory join on modern hardware". Boncz, Manegold, Kersten, TKDE 14(4): 709-730, 2002.
    - "Database Architecture Optimized for the New Bottleneck: Memory Access". Boncz, Manegold, Kersten. VLDB'99.
    - "What Happens During a Join? Dissecting CPU and Memory Optimization Effects". Manegold, Boncz, Kersten. VLDB'00.
    - "Optimizing database architecture for the new bottleneck: memory access". Manegold, Boncz, Kersten. VLDB J. 9(3): 231-246, 2000.
    - "Generic Database Cost Models for Hierarchical Memory Systems". Manegold, Boncz, Kersten. VLDB'02.
  - "Cache-Conscious Radix-Decluster Projections". Manegold, Boncz, Nes. VLDB'04.
- **Column-Oriented Database Systems (4/6) - "Vectorized Execution"**
  - "MonetDB/X100: Hyper-Pipelining Query Execution". Boncz, Zukowski, Nes. CIDR'05.
  - "Buffering Database Operations for Enhanced Instruction Cache Performance". Zhou and Ross. SIGMOD'04.
  - "Block oriented processing of relational database operations in modern computer architectures". Padmanabhan, Malkemus, Agarwal. ICDE'01.
  - "Balancing Vectorized Query Execution with Bandwidth Optimized Storage". Zukowski. PhD Thesis. CWI 2008

ADM: Literature (5/6)

- **DuckDB: An embedded database for data science**
  - “DuckDB: an Embeddable Analytical Database”. Mark Raasveldt & Hannes Mühleisen. SIGMOD’19. Demo.
  - “Data Management for Data Science - Towards Embedded Analytics”. Mark Raasveldt & Hannes Mühleisen. CIDR’20.
  - “Integrating Analytics with Relational Databases”. Mark Raasveldt. PhD Thesis, Leiden University & CWI, 2020.
  - https://duckdb.org

ADM: Literature (6/6)

- **Branch Misprediction & Predication**
  - “Conjunctive Selection Conditions in Main Memory”. Kenneth A. Ross. PODS 2002.
  - “Selection conditions in main memory”. Kenneth A. Ross. ACM Trans. Database Syst. 29: 132-161, 2004.
- **Column-Oriented Database Systems (5/6) - Adaptive Indexing**
  - “Cracking the Database Store”. Martin L. Kersten, Stefan Manegold. CIDR 2005.
  - “Database Cracking”. Stratos Idreos, Martin L. Kersten, Stefan Manegold. CIDR 2007.
  - “Self-selecting, self-tuning, incrementally optimized indexes”. Goetz Graefe, Harumi A. Kuno. EDBT 2010.
  - “Merging What's Cracked, Cracking What's Merged: Adaptive Indexing in Main-Memory Column-Stores”. Stratos Idreos, Stefan Manegold, Harumi A. Kuno, Goetz Graefe. Proc. VLDB Endow. 4(9): 585-597, 2011.
  - “Stochastic Database Cracking: Towards Robust Adaptive Indexing in Main-Memory Column-Stores”. Felix Halim, Stratos Idreos, Panagiotis Karras, Roland H. C. Yap. Proc. VLDB Endow. 5(6): 502-513, 2012.
- **Column-Oriented Database Systems (6/6) - Progressive Indexing**
  - “Progressive Indices: Indexing Without Prejudice”. Pedro Holanda. PhD@VLDB, 2018.
  - “Progressive Indexes: Indexing for Interactive Data Analysis”. Pedro Holanda, Stefan Manegold, Hannes Mühleisen, Mark Raasveldt. Proc. VLDB Endow. 12(13): 2366-2378, 2019.
  - “Progressive Mergesort: Merging Batches of Appends into Progressive Indexes”. Pedro Holanda, Stefan Manegold. EDBT 2021.

CWI

ADM

Universiteit Leiden

Introduction & Motivation

CWI

History of Databases

Universiteit Leiden

https://youtu.be/KG-mqHoXOXY

CWI

Universiteit Leiden

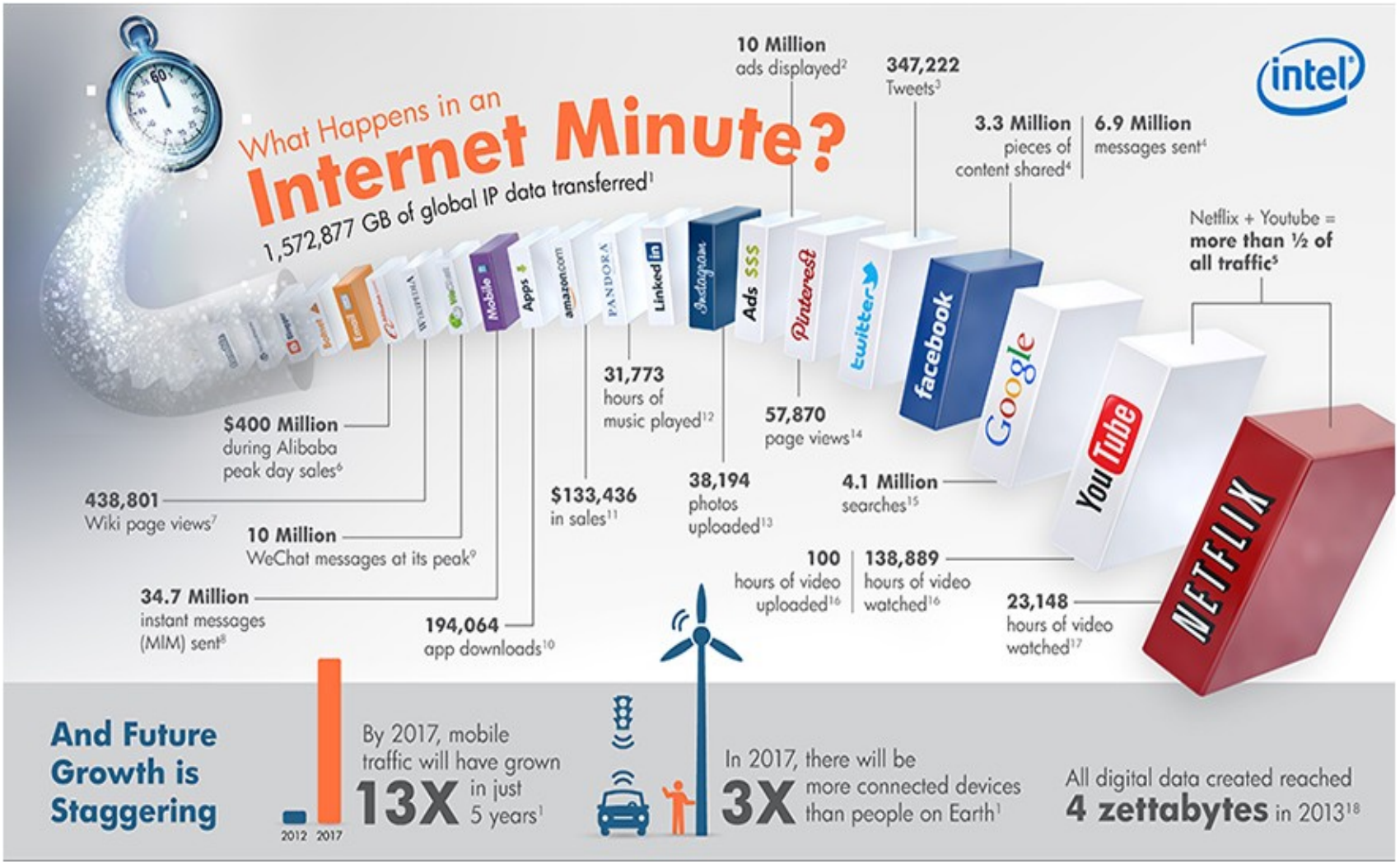
Data

CWI

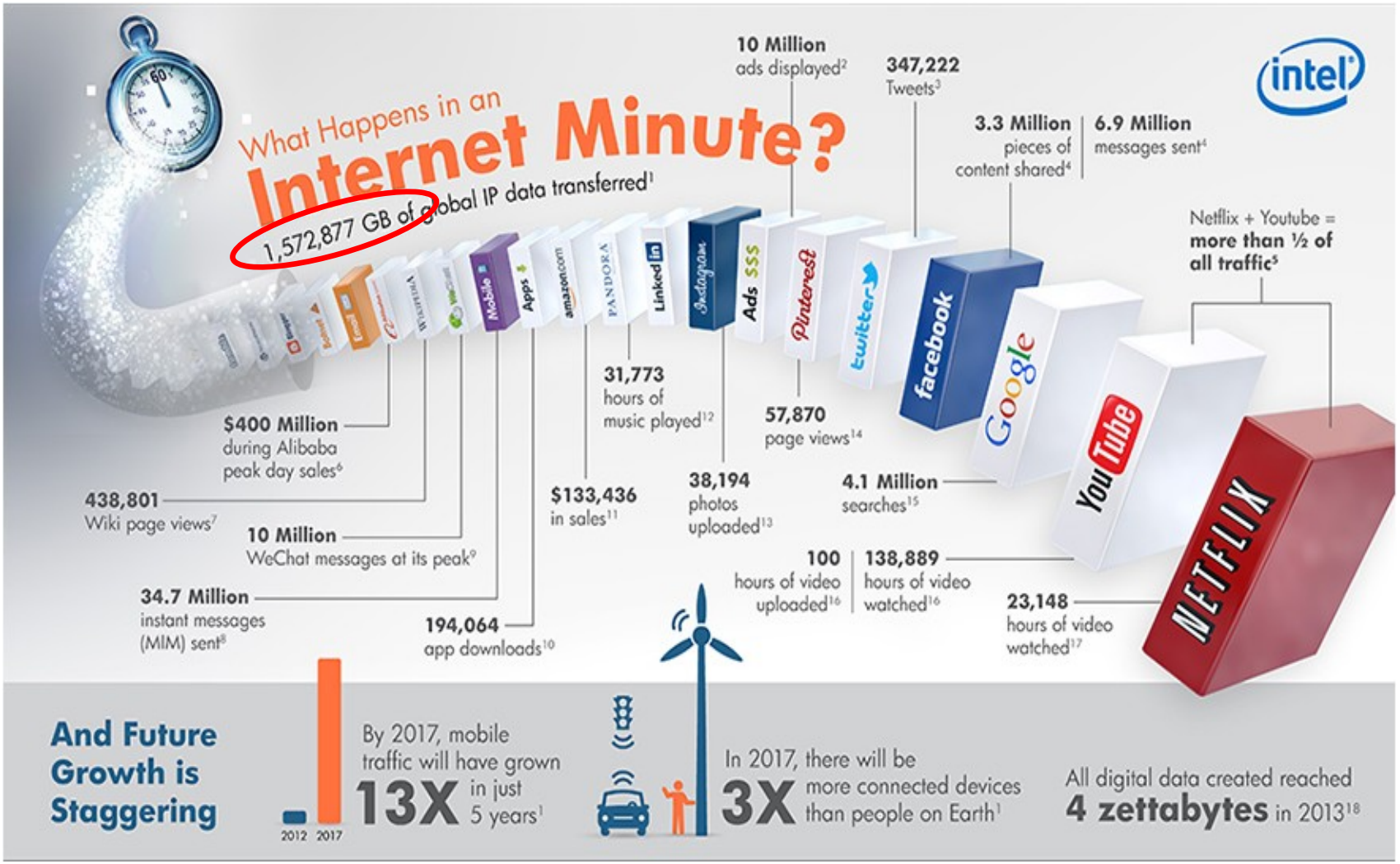
Universiteit Leiden

Data

The age of Big Data



The age of Big Data



The age of Big Data

1500TB/min =
1000 full drives
per minute

= a stack of
20meter high

The age of Big Data

1500TB/min =
1000 full drives
per minute

= a stack of
20meter high

The age of Big Data

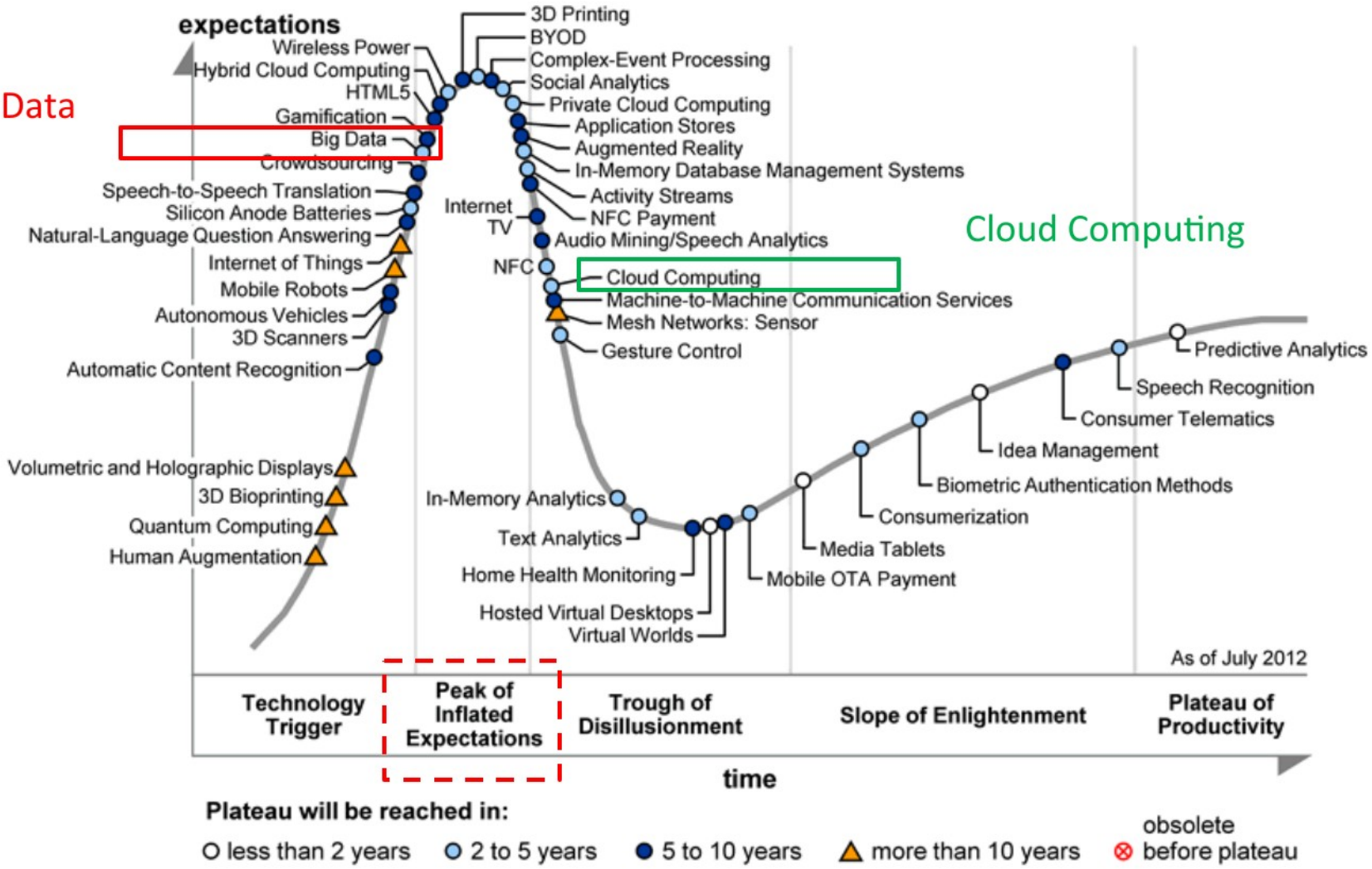
1500TB/min =
1000 full drives
per minute

= a stack of
20meter high

4000 million
TeraBytes
=
3 billion
full disk drives

BigData is the new hype

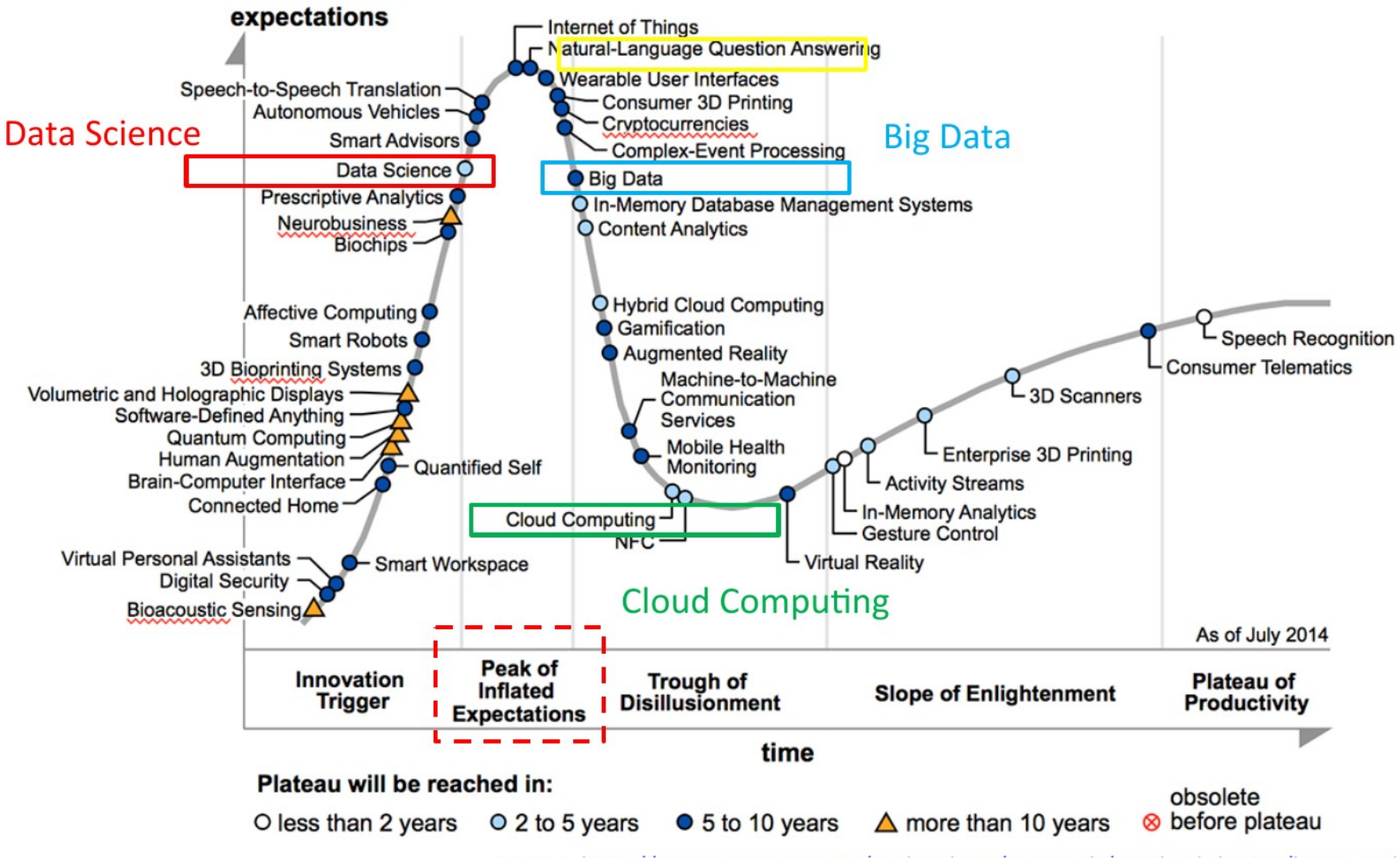
Big Data



Cloud Computing

Source http://www.gartner.com/technology/research/methodologies/hype-cycle.jsp

Gartner Technology Hypercycle (2014)



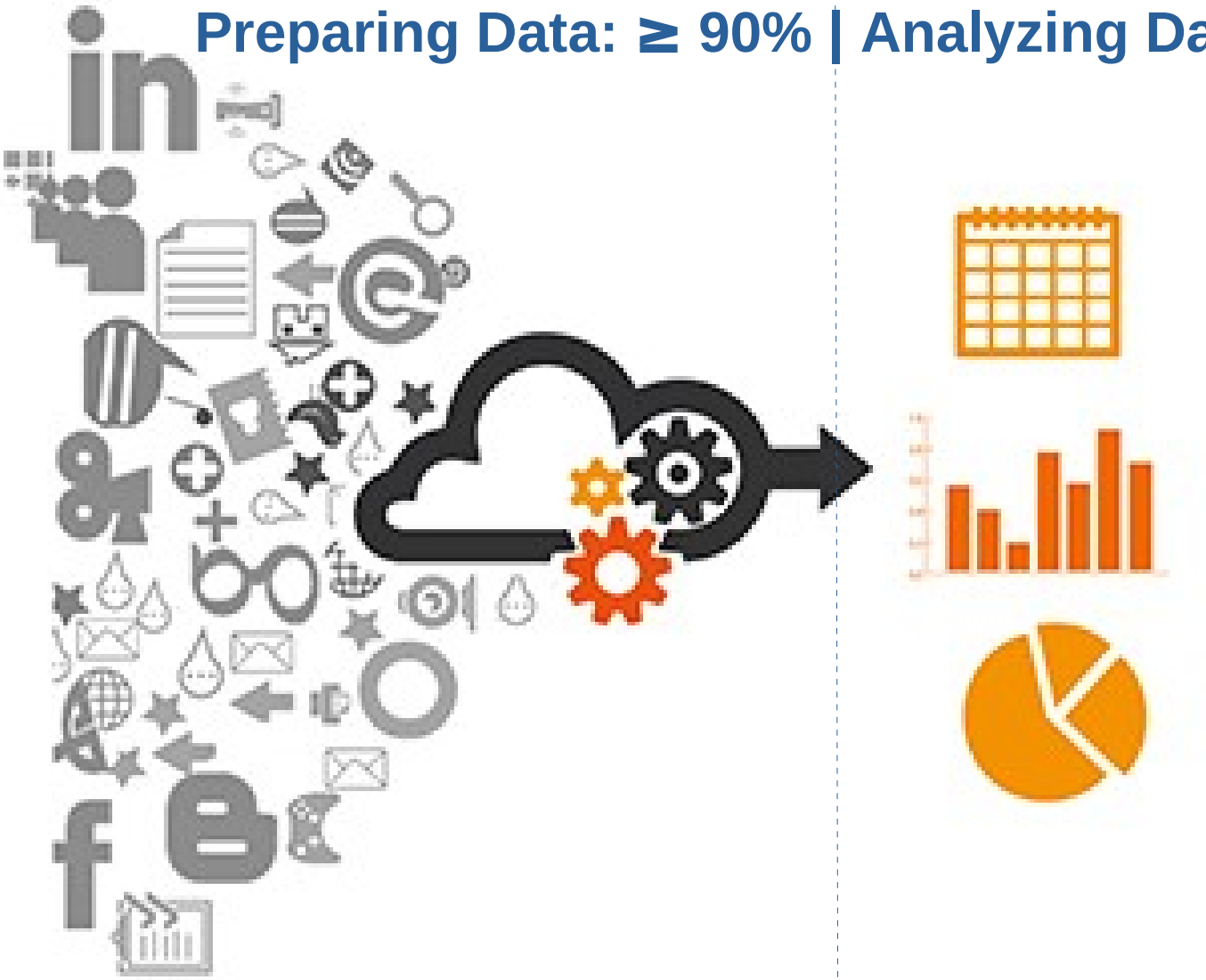
CWI

Universiteit Leiden

Big Data / Data Science == AI/ML ?

Big Data / Data Science == AI/ML ?

Preparing Data: ≥ 90% | Analyzing Data: ≤ 10%



Big Data / Data Science == AI/ML ?

Preparing Data: ≥ 90% | Analyzing Data: ≤ 10%

See also (e.g.)

Michael Stonebraker: "Top 10 Big Data Blunders"

Keynote Address, ODSC East 2019

https://www.youtube.com/watch?v=DX77pAHlVHY

CWI

Conference on Neural Information Processing Systems (NIPS/NeurIPS), 2015, pp. 2503-2511

Universiteit Leiden

Hidden Technical Debt in Machine Learning Systems

D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips
{dsculley,gholt,dgg,edavydov,toddphillips}@google.com
Google, Inc.

Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo, Dan Dennison
{ebner,vchaudhary,mwyoung,jfcrespo,dennison}@google.com
Google, Inc.

Abstract

Machine learning offers a fantastically powerful toolkit for building useful complex prediction systems quickly. This paper argues it is dangerous to think of these quick wins as coming for free. Using the software engineering framework of *technical debt*, we find it is common to incur massive ongoing maintenance costs in real-world ML systems. We explore several ML-specific risk factors to account for in system design. These include boundary erosion, entanglement, hidden feedback loops, undeclared consumers, data dependencies, configuration issues, changes in the external world, and a variety of system-level anti-patterns.

Hidden Technical Debt in Machine Learning Systems

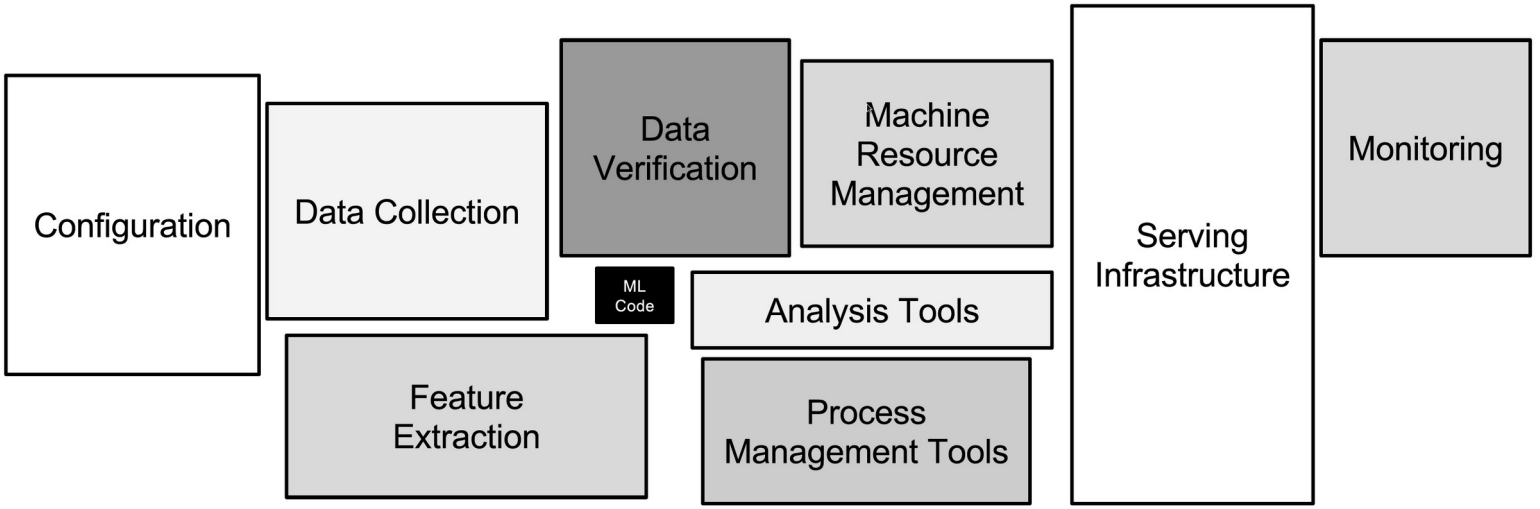
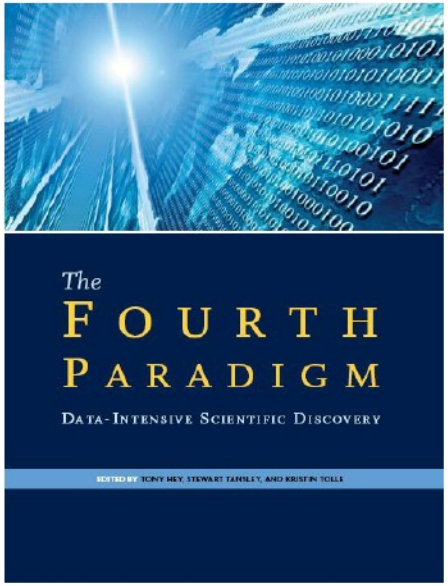


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

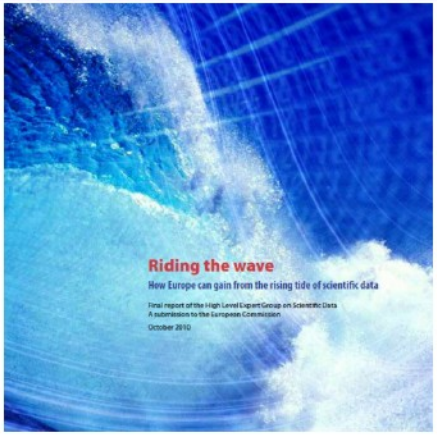
Visionaries and Drivers: Seminal works and High level reports



The Fourth Paradigm: Data-Intensive Scientific Discovery.

By Jim Gray, Microsoft, 2009. Edited by Tony Hey, et al.

http://research.microsoft.com/en-us/collaboration/fourthparadigm/



Riding the wave: How Europe can gain from the rising tide of scientific data.

Final report of the High Level Expert Group on Scientific Data. October 2010.

http://cordis.europa.eu/fp7/ict/e-infrastructure/docs/hlg-sdi-report.pdf

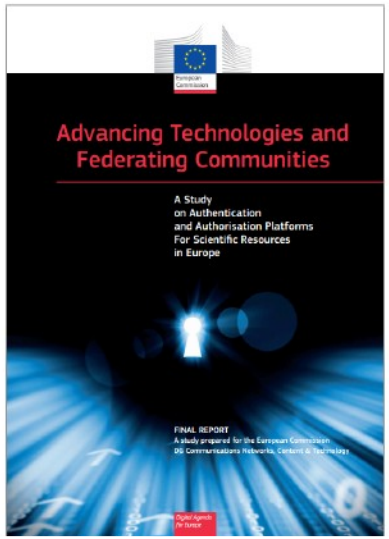
RDA RESEARCH DATA ALLIANCE

Research Data Sharing without barriers

https://www.rd-alliance.org/

NIST Big Data Working Group (NBD-WG)

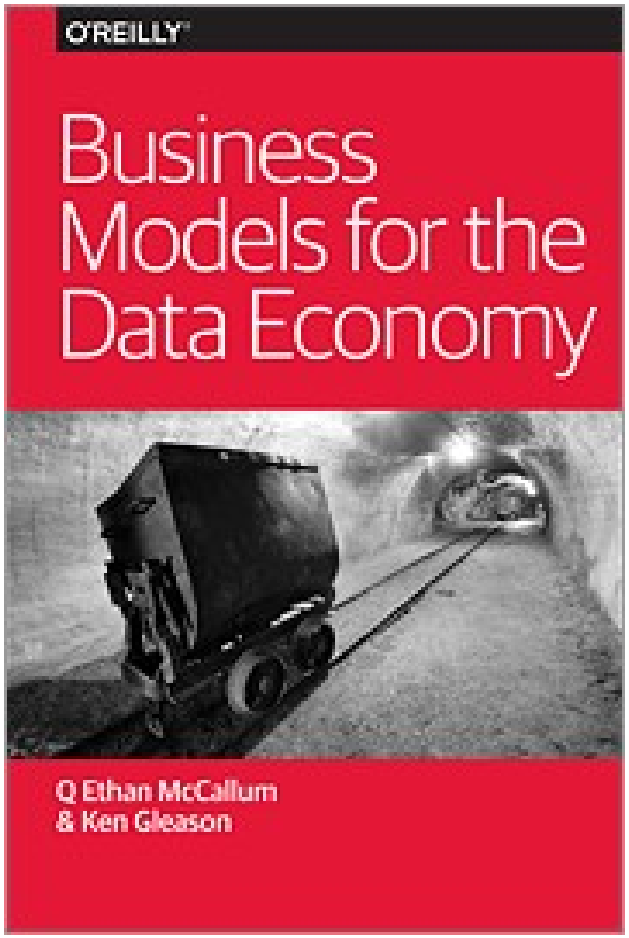
https://www.rd-alliance.org/



AAA Study: Study on AAA Platforms For Scientific data/ information Resources in Europe,

TERENA, UvA, LIBER, UinvDeb.

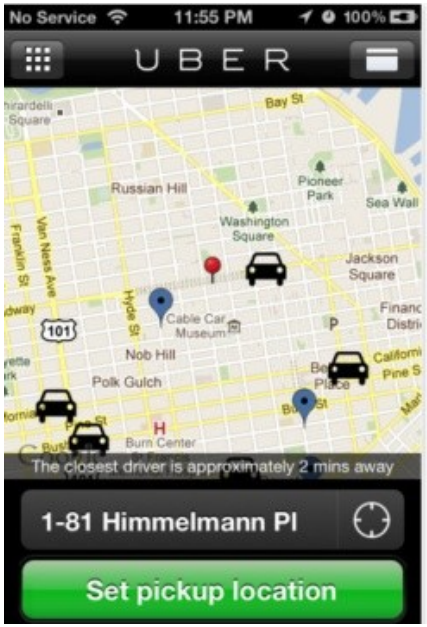
The Data Economy

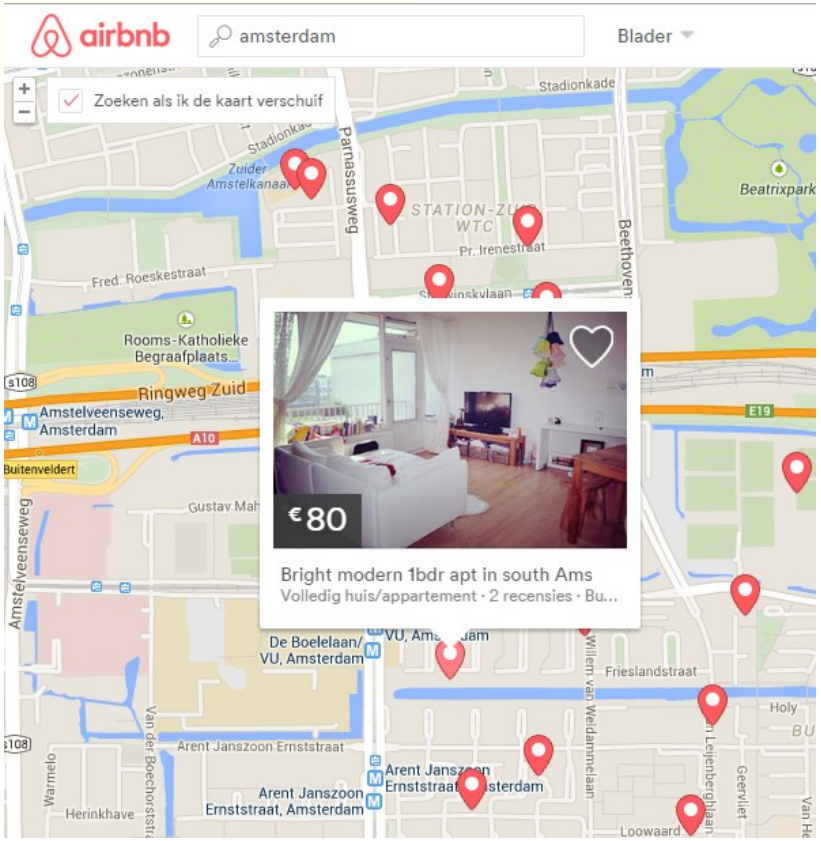


Disruptions

UBER

EVERYONE'S PRIVATE DRIVER™

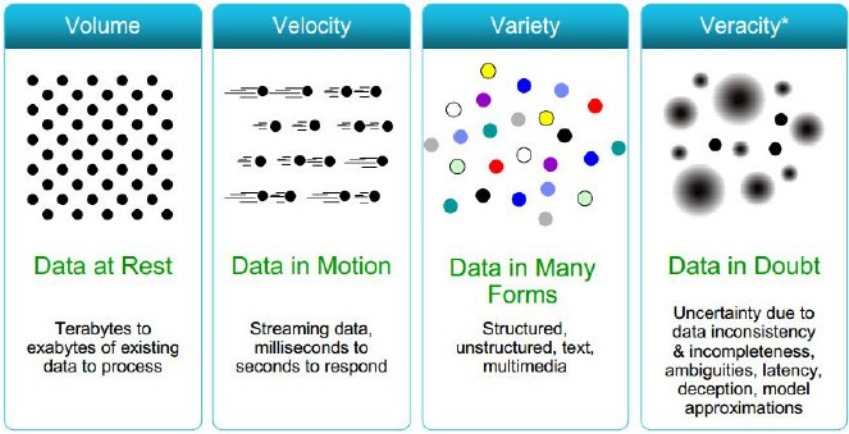




Where Big Data Comes From?

- Big Data is not **Specific application type**, but rather a **trend** –or even a collection of Trends- napping multiple application types

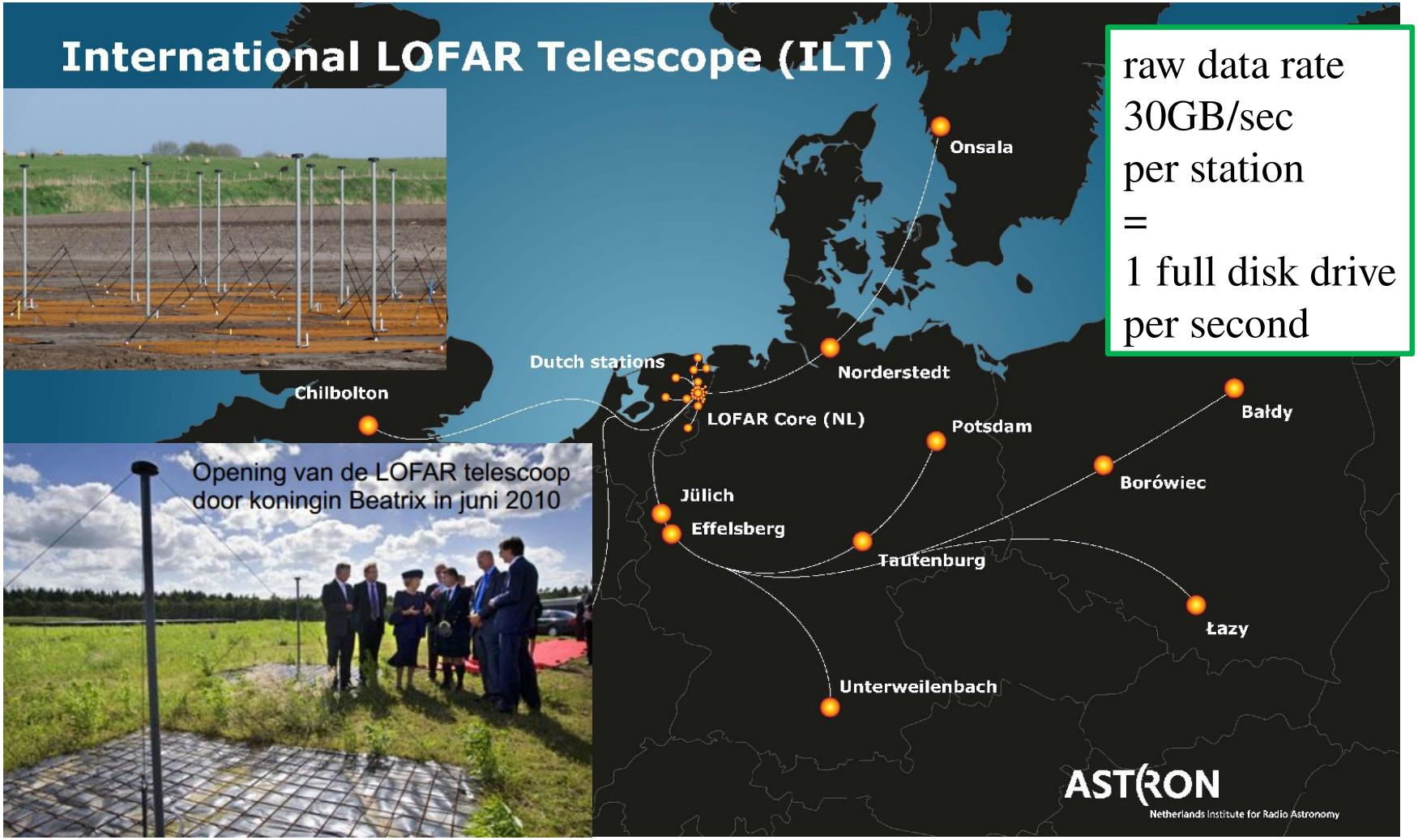
- Data growing in multiple ways
  - More data (volume of data)
  - More Type of data (variety of data)
  - Faster Ingest of data (velocity of data)
  - More Accessibility of data (internet, instruments , …)
  - Data Growth and availability exceeds organization ability to make intelligent decision based on it



CWI

Universiteit Leiden

Data Driven Science



Selected Big Data Challenges

GIS (LIDAR):

Massive point clouds: 640 Billion (x,y,z) points / 15 TB

=> spatial joins between point cloud and polygons

Logistics:

> 5 trillion (10^12) GPS points (grows with >60k points/sec)

Seismology:

~ 4 M files, ~ 500 GB (10x compressed)

=> Transparent data ingestion: *Data Vault*

Remote sensing:

~2 PB satellite image data

=> Array data processing: *SciQL*

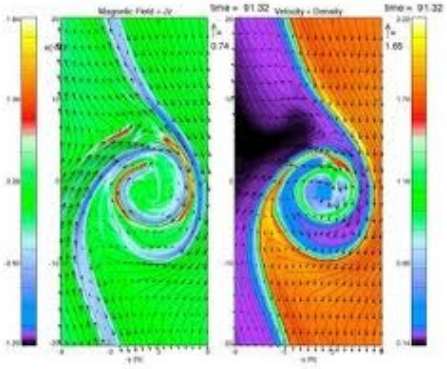
Astronomy:

Raw data: *25 TB / hour*; derived data: *100 TB / year*

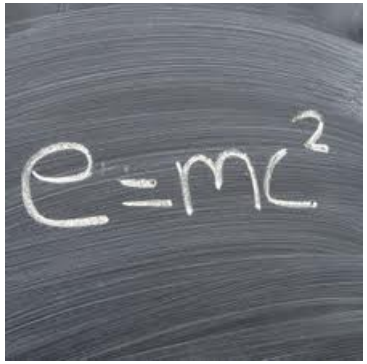
=> Transient detection inside DBMS

Paradigm Shift in Scientific Research

data exploration
(eScience)

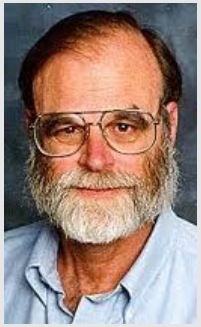


computational
3rd

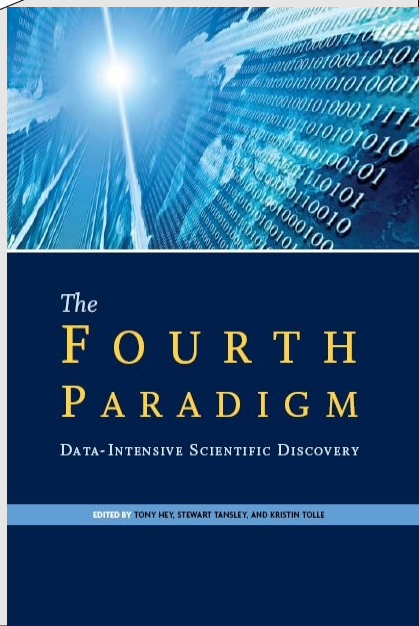


theoretical
2nd

empirical
1st



Jim Gray
(1944 - 2007)



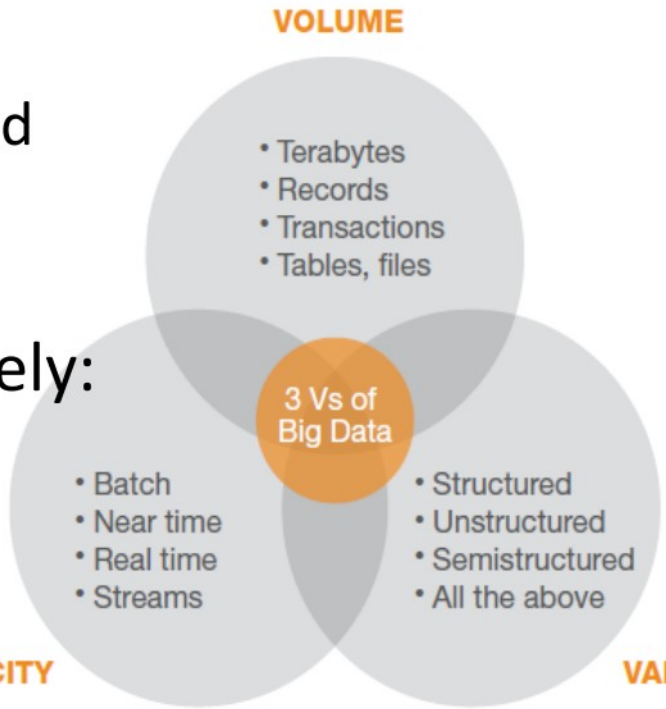
Big Data

- Big Data is a relative term
  - If things are breaking, you have Big Data
  - Big Data is not always Petabytes in size
  - Big Data for Informatics is not the same as for Google
- Big Data is often hard to understand
  - A model explaining it might be as complicated as the data itself
  - This has implications for Science
- The game may be the same, but the rules are completely different
  - What used to work needs to be reinvented in a different context

How We Define Big Data

- **Big** in Big Data refers to:
  - **Big size** is the primary definition.
  - **Big complexity** rather than big volume. it can be small and not all large datasets are big data
  - size matters... but so does **accessibility**, **interoperability** and **reusability**.

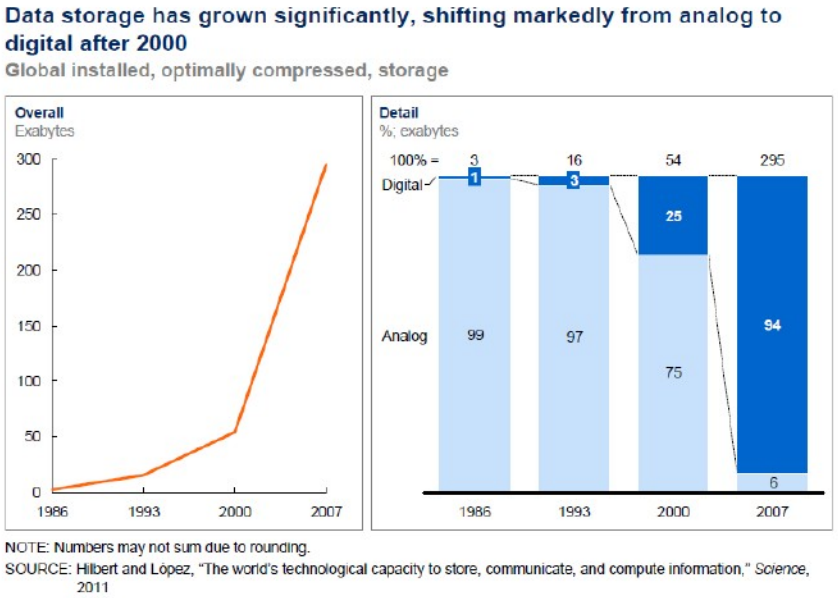
- define Big Data using 3 Vs; namely:
  - volume, variety, velocity

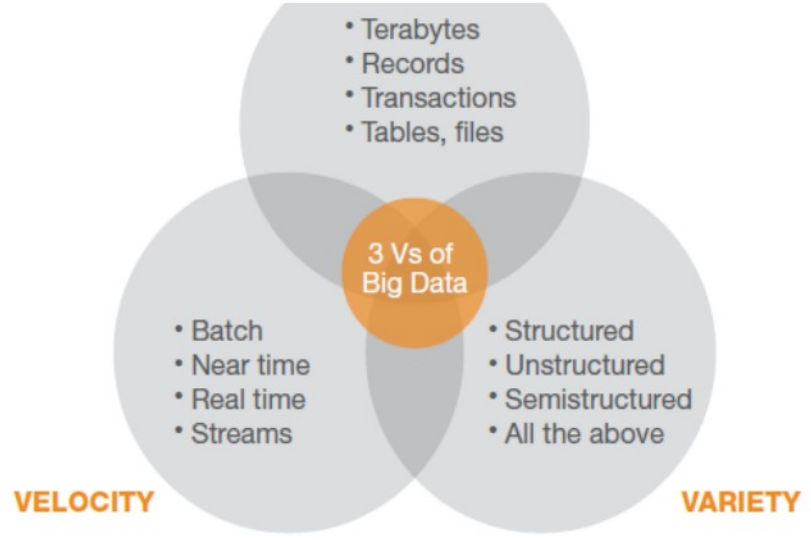


volume, variety, and velocity

- Aggregation that used to be measured in petabytes (**PB**) is now referenced by a term: **zettabytes** (**ZB**).
  - A **zettabyte** is a **trillion gigabytes** (GB)
  - or a **billion terabytes**

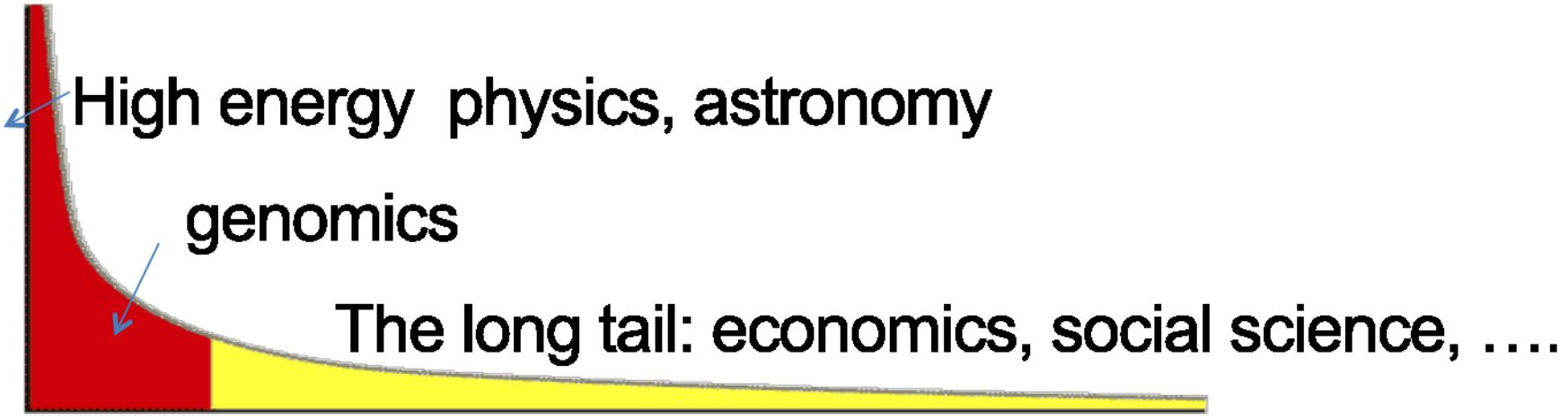
- in 2010, we crossed the **1ZB** marker, and at the end of 2011 that number was estimated to be **1.8ZB**





volume, variety, and velocity
How much data?

- Google processes **20 PB a day (2008)**
- Wayback Machine has 3 PB **+ 100 TB/month** (3/2009)
- Facebook has 2.5 PB of user data **+ 15 TB/day** (4/2009)
- eBay has 6.5 PB of user data **+ 50 TB/day** (5/2009)
- CERN's Large Hydron Collider (LHC) generates **15 PB a year**

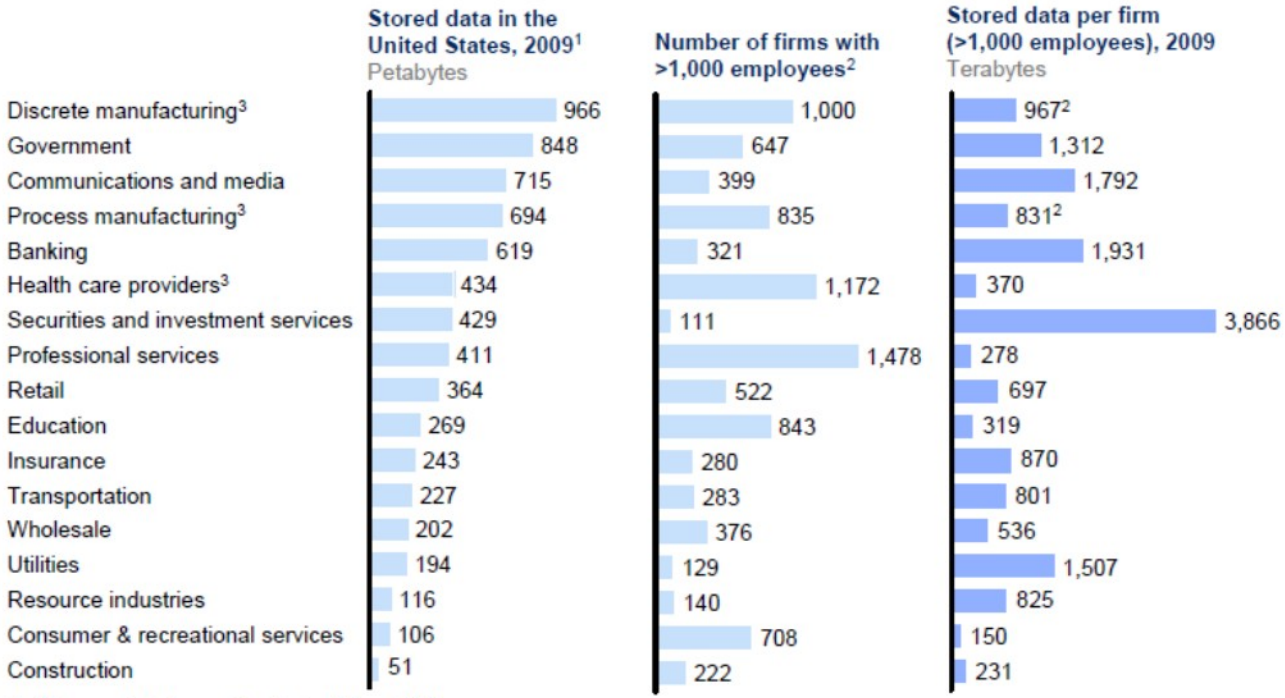


http://econsultancy.com/nl/blog/10891-big-data-shifting-the-bell-curve-to-the-long-tail

volume, variety, and velocity

How much data?

Companies in all sectors have at least 100 terabytes of stored data in the United States; many have more than 1 petabyte



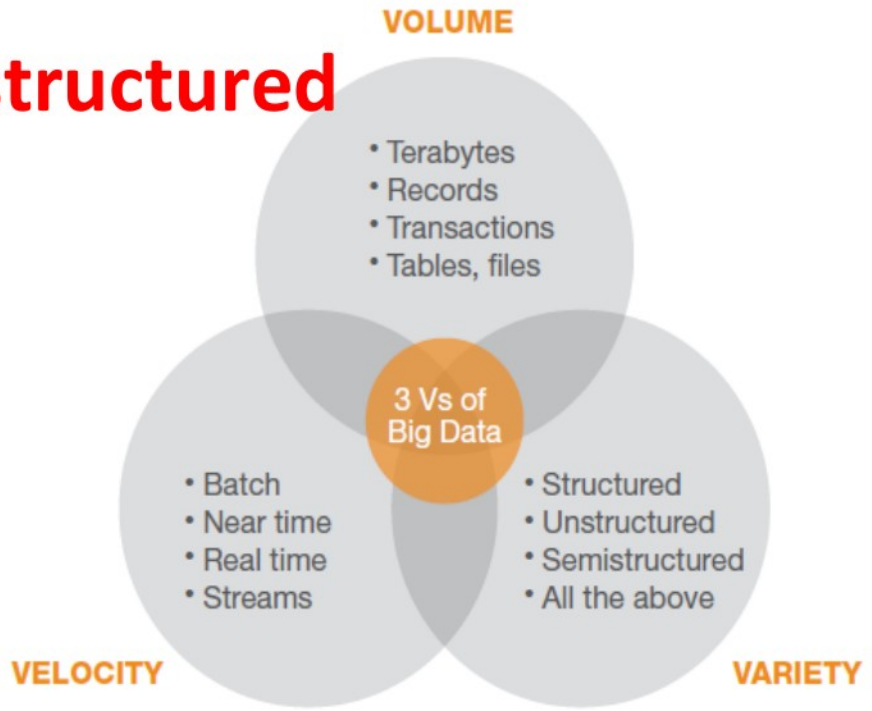
1 Storage data by sector derived from IDC.
2 Firm data split into sectors, when needed, using employment
3 The particularly large number of firms in manufacturing and health care provider sectors make the available storage per company much smaller.

SOURCE: IDC; US Bureau of Labor Statistics; McKinsey Global Institute analysis

volume, variety, and velocity

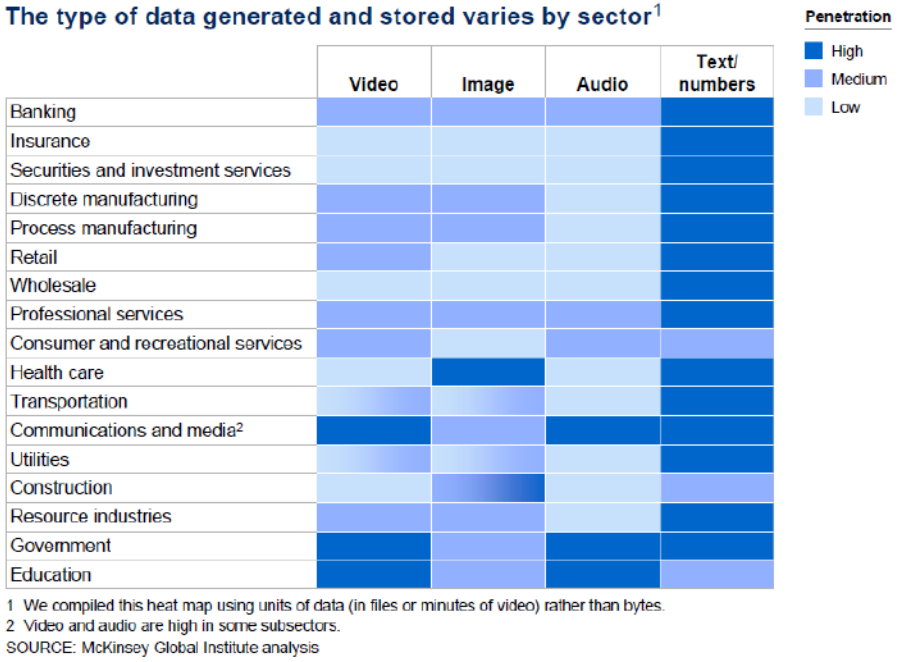
- The variety characteristic of Big Data is really about trying to **capture all** of the data that pertains to our **decision-making** process.

- Making sense out of **unstructured** data, such as **opinion**, or analysing images.



volume, variety, and velocity Type of Data

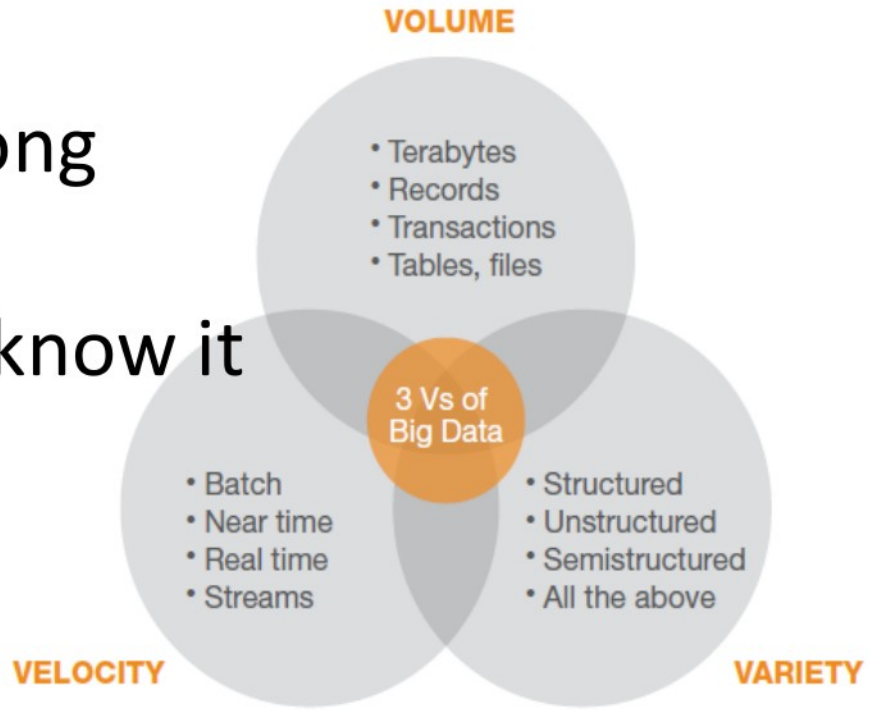
- Relational Data
  - (Tables/Transaction/Legacy Data)
- Text Data (Web)
- Semi-structured Data (XML)
- Graph Data
  - Social Network,
  - Semantic Web (RDF), …
- Streaming Data
  - You can only scan the data once



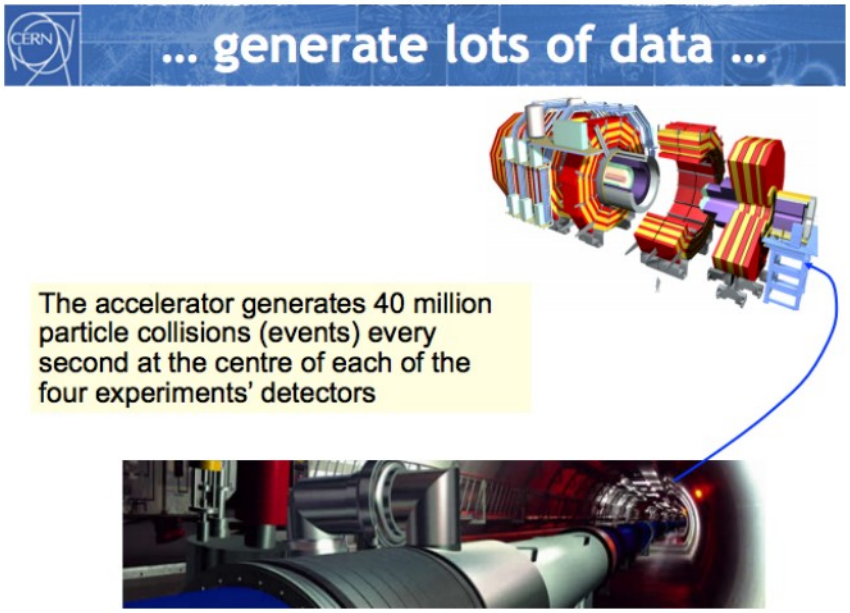
volume, variety, and velocity

- velocity is the **rate** at which data is generated and is **processed** or **well understood**

- In other terms “How long does it take you to do something about it or know it has even arrived?”



volume, variety, and velocity



Today, it is possible using real-time analytics to optimize Like buttons across both website and on Facebook.

FaceBook use anonymised data to show you the number of times people saw Like buttons, clicked Like buttons, saw Like stories on Facebook, and clicked Like stories to visit a given website.

CWI

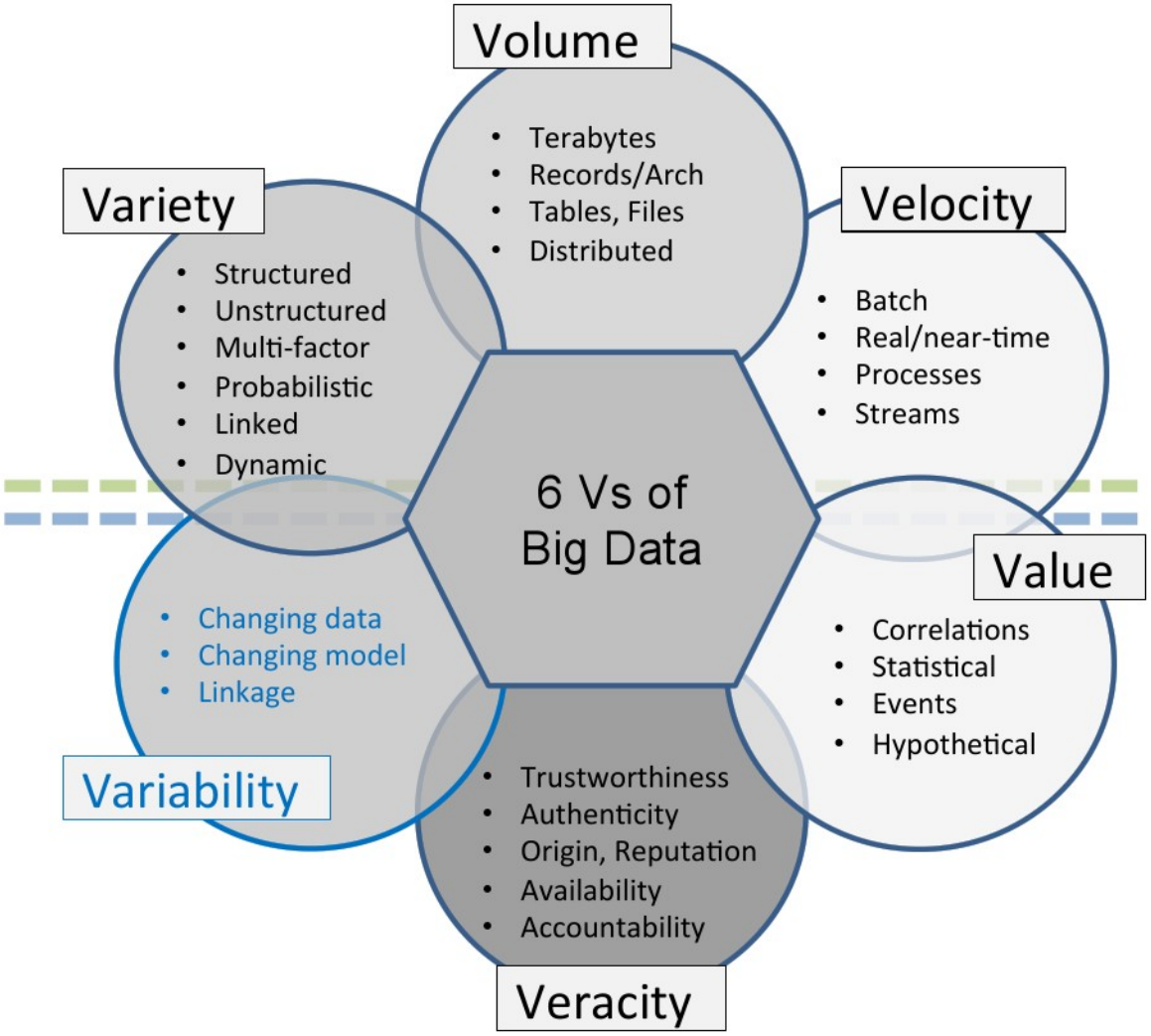
Universiteit Leiden

volume, variety, velocity, and veracity

- Veracity refers to the **quality** or trustworthiness of the data.

- A common complication is that the data is saturated with both **useful signals** and **lots of noise** (data that can't be trusted)

Improved: 5+1 V's of Big Data



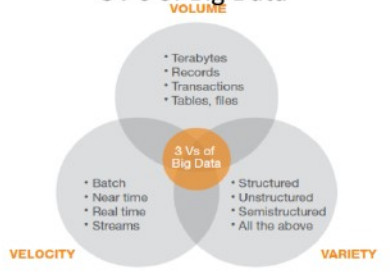
Generic Big Data Properties

- Volume
- Variety
- Velocity

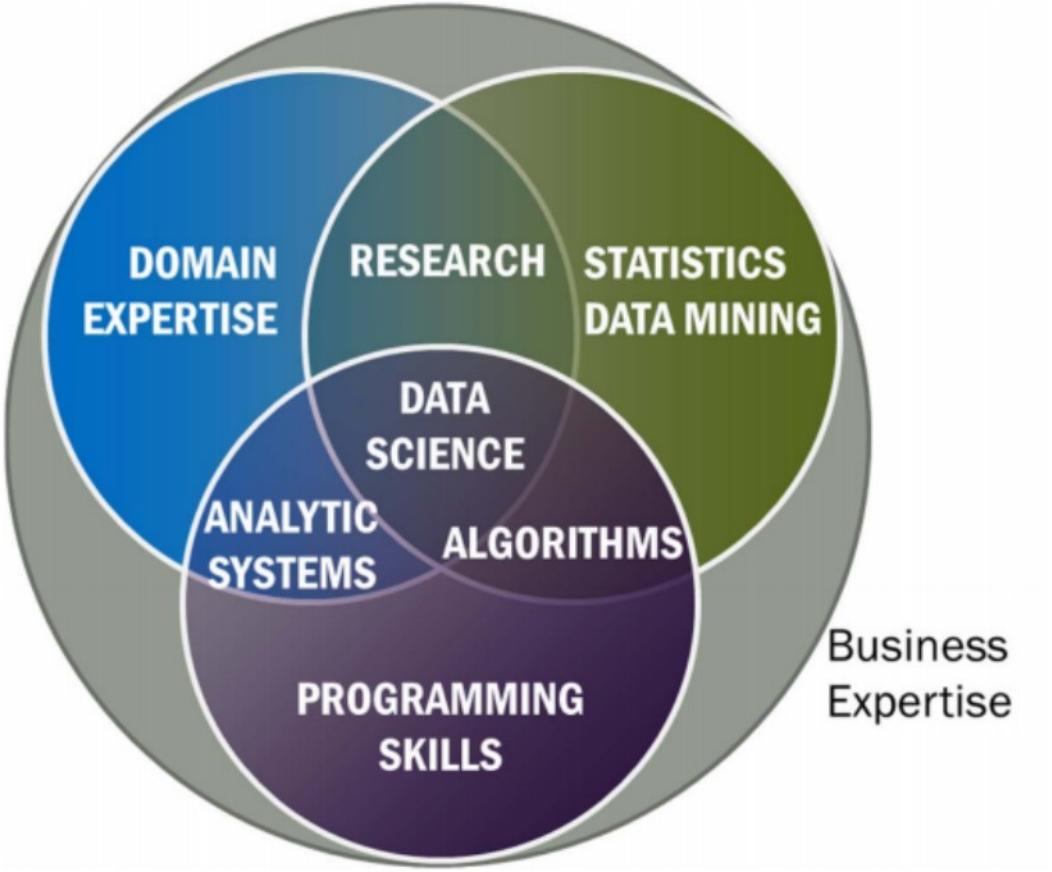
Acquired Properties (after entering system)

- Value
- Veracity
- Variability

Commonly accepted 3V's of Big Data



Skills required for Data Analytics



Nancy Grady, PhD, SAIC Co-Chair Definitions and Taxonomy Subgroup NIST Big Data Working

ADM: Agenda (planned)

- 07.09.2022: Lecture 1: **Introduction**
- 14.09.2022: Lecture 2: **SQL Recap**

  (plus Assignment 1 [in groups; 3 weeks]: TPC-H benchmark)
- 21.09.2022: Lecture 3: **Column-Oriented Database Systems (1/6) - Motivation & Basic Concepts**
- 28.09.2022: Lecture 4: **Column-Oriented Database Systems (2a/6) - Selected Execution Techniques (1/2)**
- 05.10.2022: Lecture 5: **Column-Oriented Database Systems (2b/6) - Selected Execution Techniques (2/2)**

  (plus Assignment 3 [in groups; 3 weeks]: Compression techniques)
- 12.10.2022: Lecture 6: **Column-Oriented Database Systems (3/6) - Cache Conscious Joins**
- 19.10.2022: Lecture 7: **Column-Oriented Database Systems (4/6) - "Vectorized Execution"**
- ~~26.10.2022~~: ***No lecture!***
- 28.09.2022: Lecture 8: **DuckDB: An embedded database for data science (1/2) (guest lecture & hands-on)**

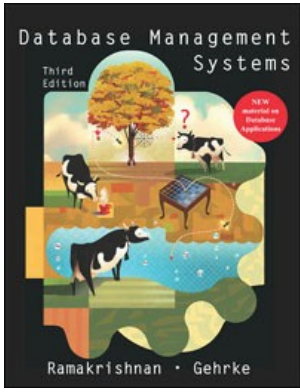
  (plus Assignment 2 [individual; 2 weeks]: Analysing NYC Cab dataset with DuckDB)
- 05.10.2022: Lecture 9: **DuckDB: An embedded database for data science (2/2) (guest lecture & hands-on)**
- 16.11.2022: Lecture 10: **Branch Misprediction & Predication**

  (plus Assignment 4 [individual; 2 weeks]: Predication)
- 23.11.2022: Lecture 11: **Column-Oriented Database Systems (5/6) - Adaptive Indexing**
- 30.11.2022: Lecture 12: **Column-Oriented Database Systems (6/6) - Progressive Indexing**

ADM: Expected Background

- **Database systems (e.g.):**

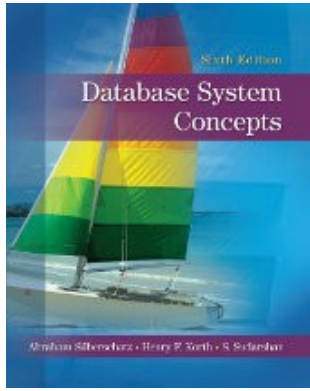
  - *Ramakrishnan*, *Gehrke*: **Database Management Systems (3rd International Edition)**, McGraw-Hill, 2003 (ISBN 0-07-246563-8)



  - *A. Silberschatz*, *H. F. Korth*, *S. Sudarshan:* **Database System Concepts (7th Edition),** McGraw-Hill, 2010 (ISBN 0-07-352332-1)

    book: https://www.db-book.com/

    slides: https://www.db-book.com/slides-dir/index.html



  - *Andy Pavlo:* **Introduction to Database Systems** course @ CMU incl. slides and videos on YouTube: https://15445.courses.cs.cmu.edu/fall2019/