NoSQL Database Systems

Prof. Dr. Bernhard Seeger
Database Systems Working Group
Department of Mathematics and Computer Science, Philipps University, 35032 Marburg
https://uni-marburg.de/dIgQLp

Organization of the module

- *Lecture*
  - Lecture: 2SWS
    - Thu 4:15 p.m. – 5:55 p.m. in HS A (@Lahnberge))

- *Exercises*
  - Tutorial (2 SWS)
    - Fri (every two weeks) from 10:00 a.m. to noon in HS IV

- *Office hours*
  - Just send an email and make an appointment!

Ilias learning platform

- *All participants in the module must register for the lecture on the Ilias learning platform.*

  https://uni-marburg.de/dIgQLp

- *Services via Ilias*
  – Lecture notes, exercises, data sets, tutorials, forum, etc.

Excercises

- *Teaching Assistant*
  - Manuel Oed
- *One exercise sheet for two weeks (6 sheets in total)*
  - Optional Homework
- *Group work in the exercises*
  - 3 students per group
- *Start of the exercise from the 2nd week of the semester*

Requirements for the Examination

- *There are two programming assignments*
  - First assignment
    - Implementation of a versioning approach in NoSQL systems
  - Second assignment
    - Problem solving using a NoSQL system
- *There are 6 excercise sheets*
  - Excercise sheets offer typical exam tasks.
  - One task has to be submitted and will be corrected.
- *Requirements*
    - 50% of the assignments successfully completed.
    - 25% of every assignment sucessfully completed
    - 40% of the points from the excercise sheets

Exams

- *Examination*
  - 1. Written exam, February 19, 2:00–5:00 p. m.
  - 2. Written exam, Mar 20, 1:00 - 4:00 p.m.

Bibliography

- *Dan Sullivan: **“NoSQL for Mere Mort**als,” Pearson Education (US), 2015*
- *Tom White: **Hadoop: The Definitive Guide**, 4th Edition, O'Reilly Media, 2015*
- *Lars George: **"HBase: The Definitive Guide,"** O'Reilly Media, 2011*
- *Pramod J. Sadalage, Martin Fowler: **"NoSQL Distilled,"** Addison-Wesley, 2013*
- ***...***

Overview of topics

- *Motivation*
  - Deficits of relational database systems
- *Fundamentals of NoSQL database systems*
  - Distribution
  - Consistency
- *Key-value stores*
  - Redis
  - Hash tables / Consistent hashing
  - LSM trees
- *Document stores*
- *Column stores*
- *Graph databases*
- *Time-series databases*

Motivation

- *Advantages of relational database systems*
  - Transaction concept with ACID property
    - Consistency and fault tolerance
  - Simple data model with SQL as standard query language
    - Tables + relational algebra
  - Integration of data from different users in one database
    - Easy data exchange and collaboration
  - Declarative formulation of queries + optimization

- *Proven technology with many applications in the field of classic enterprise applications*
  - Integration of data from different applications in a central database system.

Motivation

- *Disadvantages of relational database systems*
  - Impedance mismatch
    - Data model in the application (in main memory) and the relational model in the persistent database
      - ORM tools such as JPA help, but do not eliminate the underlying problem.
  - Functionality
    - Oversupply
      - The system must be fully understood.
      - However, only a small part of it is used.
    - Lack of analytical functions
  - **Rudimentary distribution options** in a cluster
    - Typical model for distribution: Shared disk
    - Desired model: Shared Nothing

Architectures for distribution

- *Shared Everything*
  - Parallel Architecture
- *Shared Disk*
- *Shared Nothing*

Platform for
relational database systems

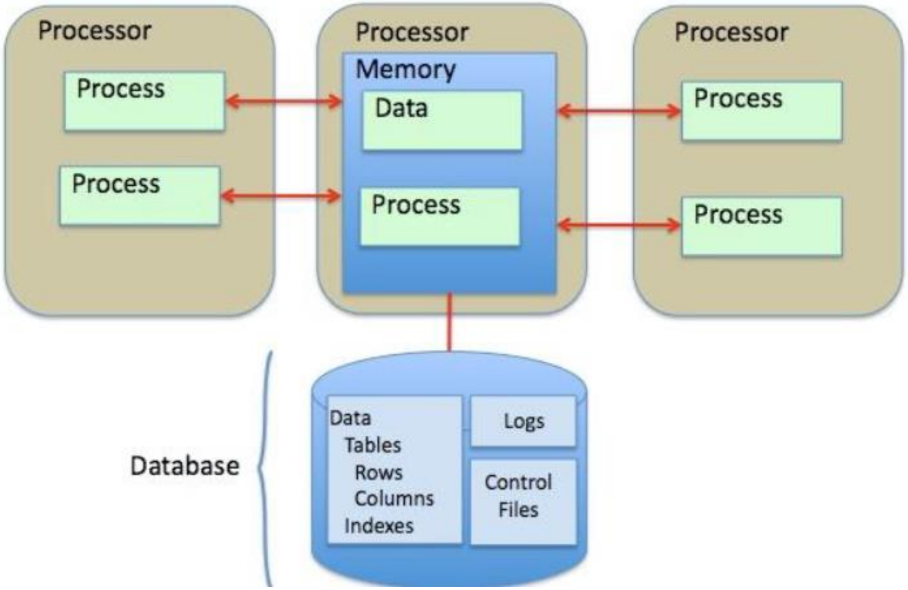
- Performance improvement through **scale-up**

Platform for NoSQL systems

- Performance improvement through **scale-out**

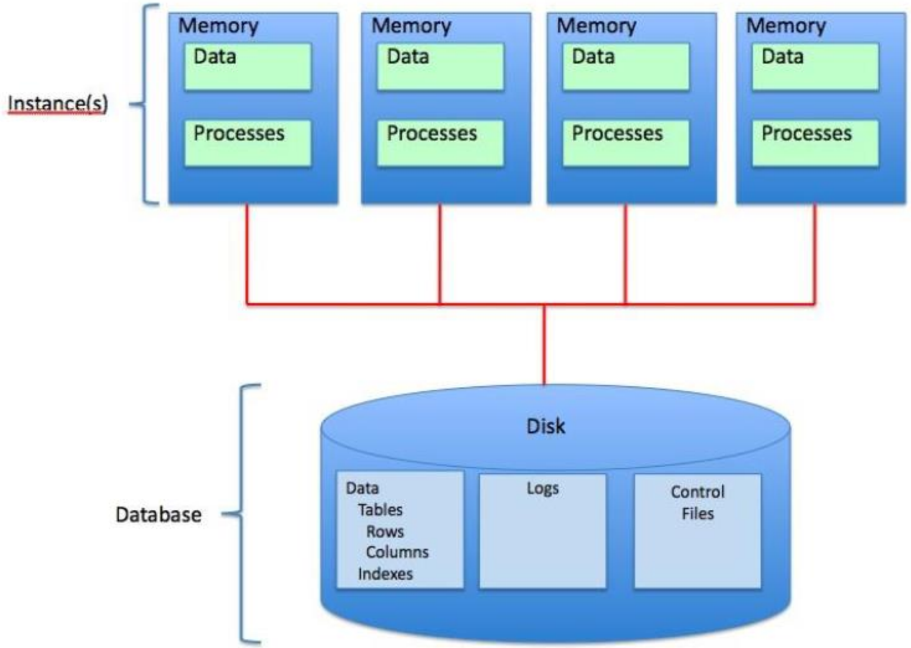
Shared Everything

- *All computing units use a shared memory and a shared multiprocessor.*
  - No database system customization required.
  - Inexpensive hardware with up to 64 cores (128 threads)
  - Limited scalability because the hardware determines the maximum degree of parallelism.



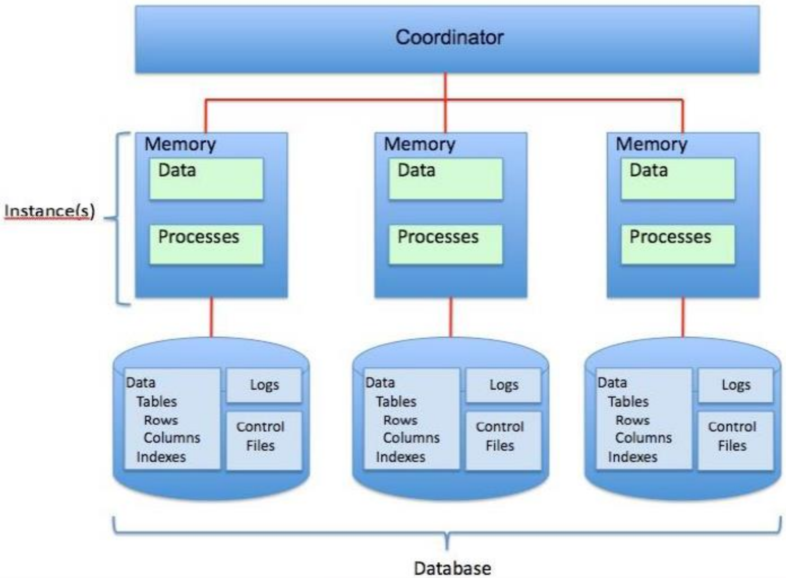
Shared disk

– Each computing unit has its own multiprocessor with local memory, but they all use a shared database
  - Use of shared external memory and a database
  - The disadvantage is the high overhead involved in writing for synchronization of the local memories.



Shared Nothing

- *Each computing unit has its own processor with local main memory and local database.*
  - Division of the database into several small databases.
  - Any **scale-out** possible by adding new computer nodes.
  - Disadvantage: potentially high communication overhead.



Motivation

- *Further disadvantages of relational database systems*
  - Rudimentary distribution options in a cluster
  - Licensing model not designed for large clusters
    - Linearly increasing costs in the number of nodes/CPUs
  - **New applications** in the **big data** environment are not really well supported.
    - High insertion rates → very large databases
    - Heterogeneous data (without fixed schema)
    - Fast response time with low latency
    - High demands on analytical functionality

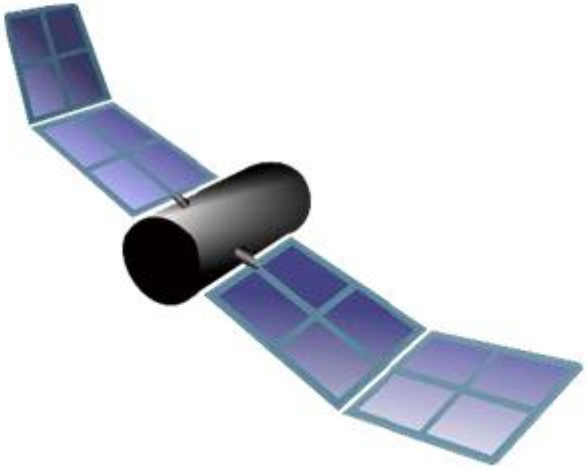
One size does not fit all

- *Data in a company requires multiple types of DBMS to meet the needs of all applications.*
  - Relational DBMS will play an important role.
    - Integration of important company data
  - New NoSQL database systems will be added as needed for specific applications.

- *A zoo of data systems is already a reality in many companies.*

Data Producers

Example: Remote sensing



- *Next generation of satellites*
  - More channels
  - Better spatial resolution
  - Better temporal resolution
  - ➔ Increase in data volume by a factor of 100-1000

- *Use of drones and wingcopters*
  - "Drones are a cheaper way for NASA to pursue some aspects of its mission, compared to rocket launches."

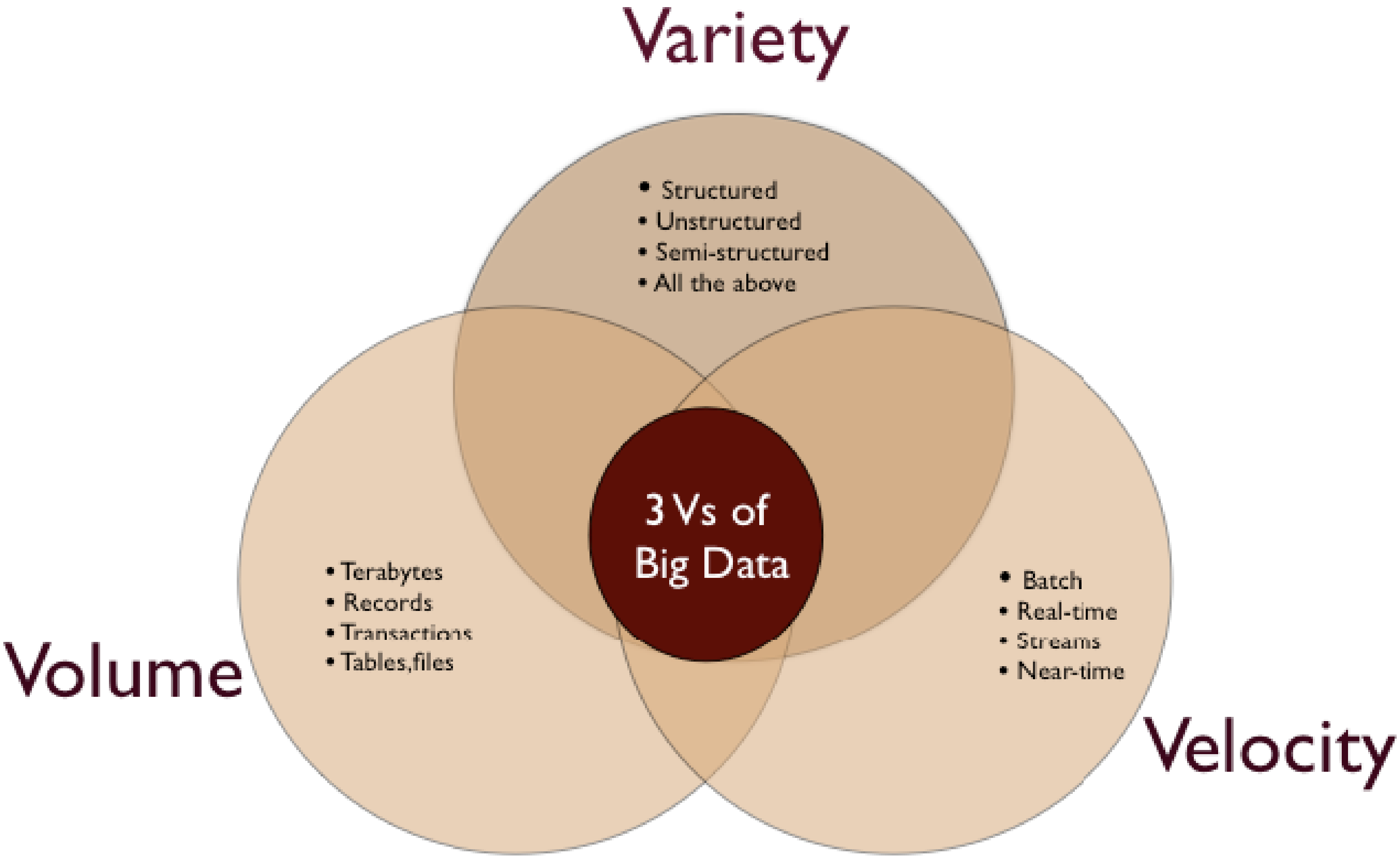
Benefit from heterogeneous data

- *New applications collect data of very different types.*
  - Structured data
  - Log data and time series
  - Text documents
  - Images, videos
  - Sensor data
  - Data from social media

Data analysis

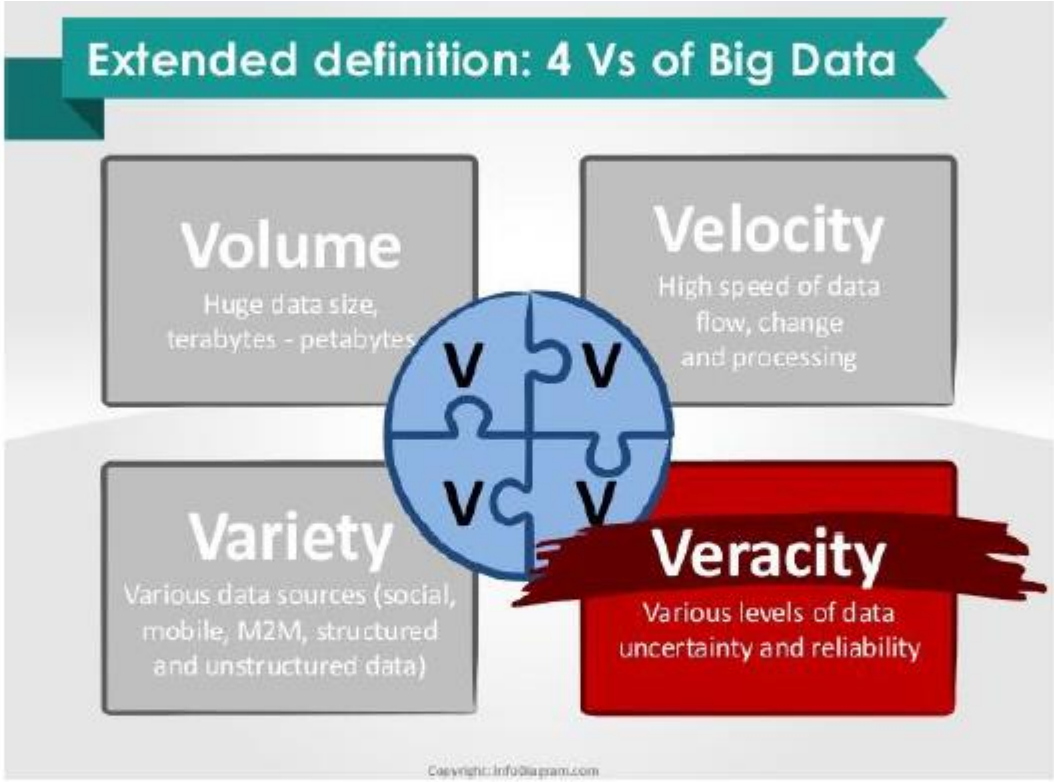
- Statistics, cubes, reports
- Recommender Systems
- Classification & Clustering
- Knowledge Extraction
- LLMs

The 3 Big Data Challenges



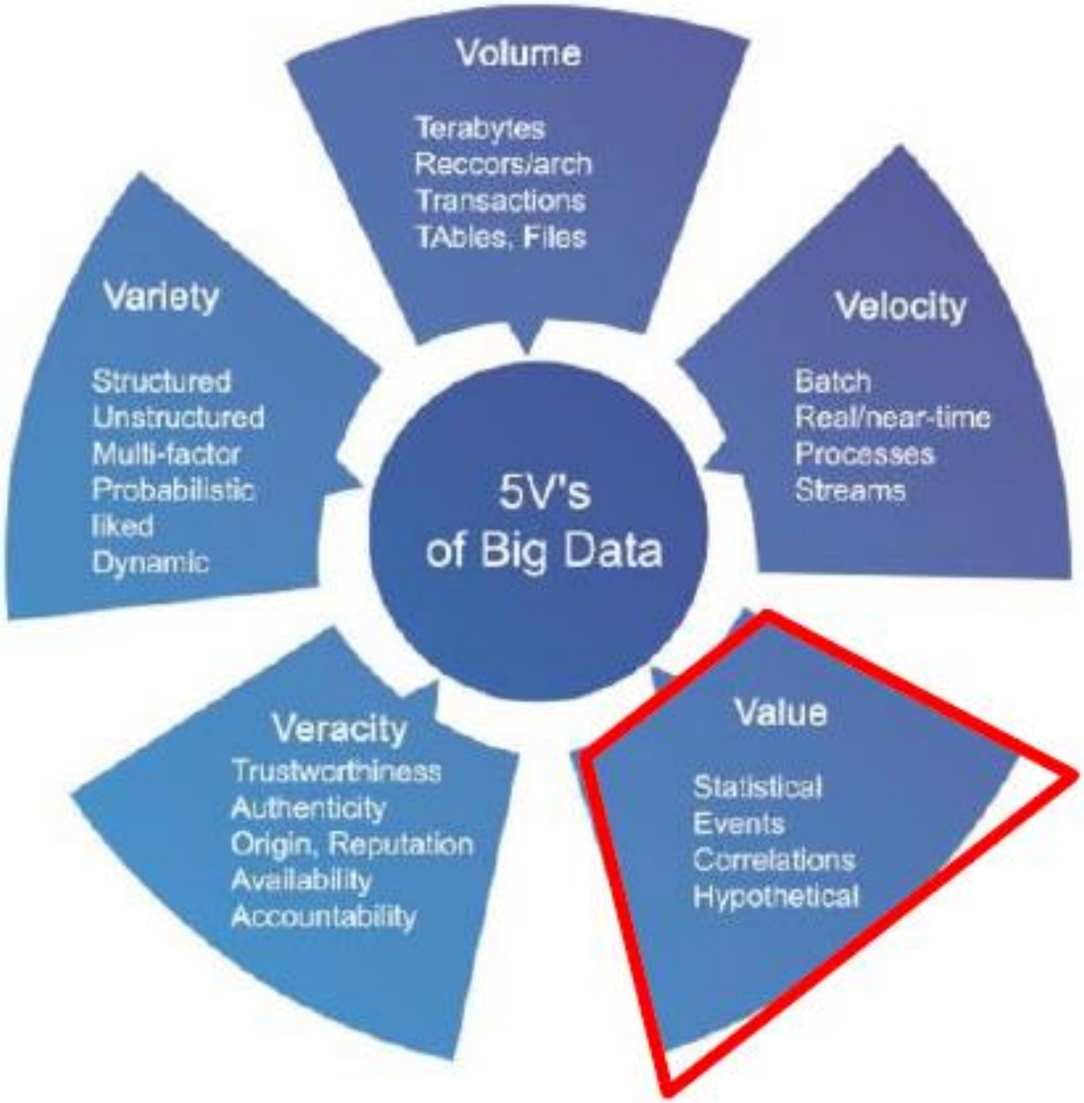
Quelle: http://kavyamuthanna.wordpress.com/category/big-data/

... or 4Vs



Extended definition: 4 Vs of Big Data
Volume
Huge data size, terabytes - petabytes
Velocity
High speed of data flow, change and processing
Variety
Various data sources (social, mobile, M2M, structured and unstructured data)
Veracity
Various levels of data uncertainty and reliability
V
V
V
V

... or 5Vs


Volume
Terabytes
Reccors/arch
Transactions
TAbles, Files
Variety
Structured
Unstructured
Multi-factor
Probabilistic
liked
Dynamic
Velocity
Batch
Real/near-time
Processes
Streams
5V's
of Big Data
Veracity
Trustworthiness
Authenticity
Origin, Reputation
Availability
Accountability
Value
Statistical
Events
Correlations
Hypothetical

... or 10Vs

- ***#1: Volume***
- ***#2: Velocity***
  - speed at which data is being generated or refreshed.
    - Facebook claims 600 terabytes of incoming data per day.
  - Google alone processes on average more than "40,000 search queries every second.
- ***#3: Variety***
  - Semi-structured and unstructured data such as audio, image, video files, social media updates, and other text formats
  - log files, click data, machine and sensor data, etc.

... or 10Vs

- ***#4: Variability***
  - inconsistencies in the data (anomalies and outliers).
  - multitude of data dimensions resulting from multiple disparate data types and sources.
  - Variable speed at which big data is loaded into your database.
- ***#5: Veracity***
  - Veracity refers to the provenance and reliability of the data source, its context, and how meaningful it is to the analysis based on it.
- ***#6: Validity***
  - Similar to veracity, validity refers to how accurate and correct the data is for its intended use.
  - good data governance practices to ensure consistent data quality, common definitions, and metadata.

… or 10Vs

- ***#7: Vulnerability***
  – Big data brings new security concerns.
  – Information on many others can be found at Information is Beautiful.
- ***#8: Volatility***
  – How old does your data need to be before it is considered irrelevant, historic, or no longer useful?
- ***#9: Visualization***
  – Current big data visualization tools face technical challenges due to limitations of in-memory technology and poor scalability, functionality, and response time.
- ***#10: Value***
  – The other characteristics of big data are meaningless if you don't derive business value from the data

NoSQL DBMS

- *The term NoSQL stands for Not only SQL and refers to non-relational systems with a specific range of functions.*
  - It originates from a hashtag in a Twitter message.
- *There are now a large number of these systems.*

Development Effort

- *Relational systems*
  - Extremely complex software systems with a wide range of applications
- *NoSQL systems*
  - Simpler systems with a limited range of functions for a specific range of applications
    - ➔Low development costs
    - ➔Implementation with ready-made building blocks

- *The effort required to develop a new NoSQL system is lower than that required to expand a relational system.*

  ➔ Low time-to-market

Features of NoSQL DBMS

- No use of SQL as a query language.
  - Instead, simpler query language
- Distribution in a shared-nothing cluster
  - Use of inexpensive consumer computers as nodes, standard network, SSDs, etc.
- Open source
  - At least initially, the systems were developed as open source.
- No fixed schema
  - Schema is defined in the application and not in the DBMS
    - No impedance mismatch
  - Dynamic adaptation of the schema
- Weaker or different forms of consistency

Classification of NoSQL DBMS

- *Based on the underlying data model*
  - **Key-value**
  - Wide-Column
  - Document
  - Search engine
  - Graph
  - **TimeSeries**

Key-value stores

- Data model: (key) -> value
- Simple interface: CRUD (Create, Read, Update, Delete)

users:42:name → Kemper

users:42:friends → [2, 45, 23, 99]

users:42:settings → busy → "true", position → C4

- Examples (https://db-engines.com/)

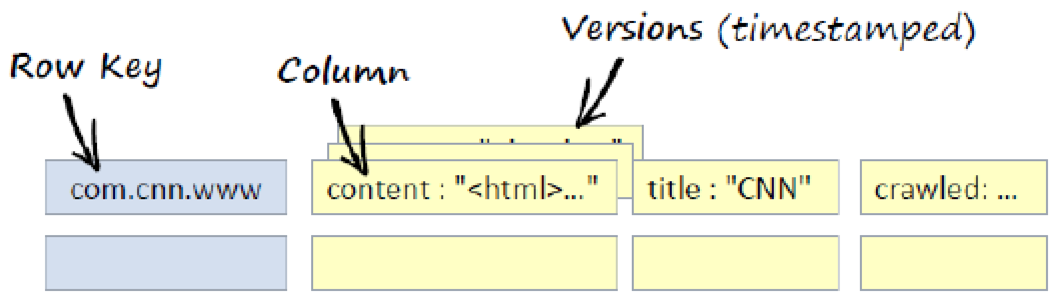
☐ sekundäre Datenbankmodelle berücksichtigen

69 Systeme im Ranking, April 2023

Rang Apr 2023	Rang Mär 2023	Rang Apr 2022	DBMS	Datenbankmodell	Punkte Apr 2023	Punkte Mär 2023	Punkte Apr 2022
1.	1.	1.	Redis	Key-value, Multi-Model	173,55	+1,10	-4,05
2.	2.	2.	Amazon DynamoDB	Multi-Model	77,45	-3,32	-5,46
3.	3.	3.	Microsoft Azure Cosmos DB	Multi-Model	35,08	-1,03	-5,26
4.	4.	4.	Memcached	Key-value	21,99	-0,62	-3,19

Wide-column DBMS

- *Data model:* (row key, column, timestamp) → value
- Interfaces:
  CRUD (Create, Read, Update, Delete), Map-Reduce



- Examples (https://db-engines.com/)

☐ sekundäre Datenbankmodelle berücksichtigen — 13 Systeme im Ranking, April 2023

Rang Apr 2023	Rang Mär 2023	Rang Apr 2022	DBMS	Datenbankmodell	Punkte Apr 2023	Punkte Mär 2023	Punkte Apr 2022
1.	1.	1.	Cassandra	Wide column	111,81	-1,98	-10,19
2.	2.	2.	HBase	Wide column	37,79	+0,17	-6,54
3.	3.	3.	Microsoft Azure Cosmos DB	Multi-Model	35,08	-1,03	-5,26
4.	4.	4.	Datastax Enterprise	Wide column, Multi-Model	6,79	-0,55	-3,09

Document Stores

- *Data model: (collection, key) → document*
- *Interface: CRUD,* Map-Reduce, declarative query language

users-42 → {Last name: Kemper,
 Lectures: [DBS1, IDBS, MM-DBS]
 Address: {City: "Garching," Street: "Boltzmannstraße"}
}

… → …

- Examples (https://db-engines.com/)

☐ sekundäre Datenbankmodelle berücksichtigen — 57 Systeme im Ranking, April 2023

Rang			DBMS	Datenbankmodell	Punkte		
Apr 2023	Mär 2023	Apr 2022			Apr 2023	Mär 2023	Apr 2022
1.	1.	1.	MongoDB	Document, Multi-Model	441,90	-16,89	-41,48
2.	2.	2.	Amazon DynamoDB	Multi-Model	77,45	-3,32	-5,46
3.	3.		Databricks	Multi-Model	60,97	+0,11	
4.	4.	↓ 3.	Microsoft Azure Cosmos DB	Multi-Model	35,08	-1,03	-5,26

Search engines

- *Data model: (collection, attribute, value) → document_list*
- *Interface: CRUD, similarity queries*

users-last name-Kemper → 42, 77, 99, 234

… → …

- Examples (https://db-engines.com/)

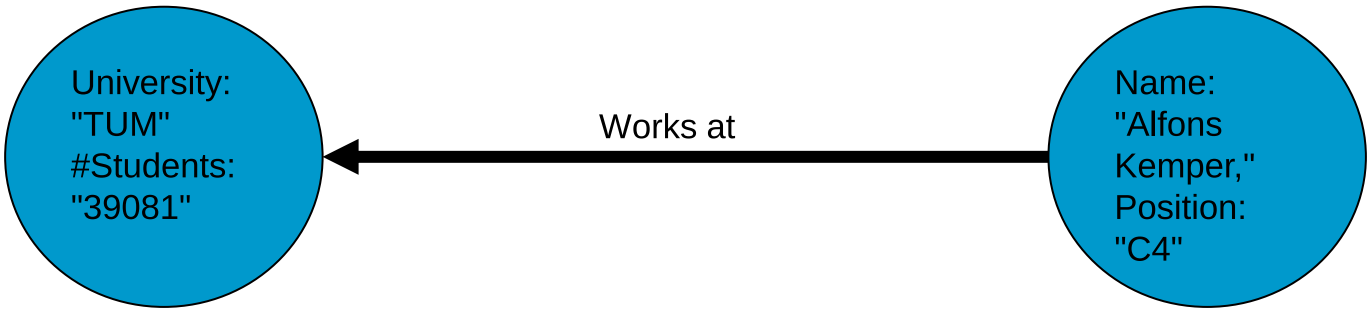
☐ sekundäre Datenbankmodelle berücksichtigen — 28 Systeme im Ranking, April 2023

Rang Apr 2023	Rang Mär 2023	Rang Apr 2022	DBMS	Datenbankmodell	Punkte Apr 2023	Punkte Mär 2023	Punkte Apr 2022
1.	1.	1.	Elasticsearch	Suchmaschine, Multi-Model	141,08	+2,01	-19,76
2.	2.	2.	Splunk	Suchmaschine	85,44	-2,54	-9,81
3.	3.	3.	Solr	Suchmaschine, Multi-Model	48,22	+0,94	-9,52
4.	4.	↑ 10.	OpenSearch	Suchmaschine	11,09	-0,03	+5,85

Graph DBMS

Current research area

- Data model: G = (V, E): Graph property model
- Interface: query languages, graph traversals



- Examples (https://db-engines.com/)

☐ sekundäre Datenbankmodelle berücksichtigen

39 Systeme im Ranking, April 2023

Rang			DBMS	Datenbankmodell	Punkte		
Apr 2023	Mär 2023	Apr 2022			Apr 2023	Mär 2023	Apr 2022
1.	1.	1.	Neo4j	Graph	51,60	-1,91	-7,92
2.	2.	2.	Microsoft Azure Cosmos DB	Multi-Model	35,08	-1,03	-5,26
3.	3.	3.	Virtuoso	Multi-Model	6,24	-0,16	+0,57

Time-Series DBMS

Current research area

- Data model: (Key, [(t_1 , $value_1$), (t_2 , $value_2$), …, (t_n , $value_n$)]
- Interface: Dedicated query language

Kemper place of residence → [(1980, Dortmund), (1982, LA), (1985, Karlsruhe), (1989, Passau), (2003, Garching)]

… → ...

- Examples (https://db-engines.com/)

☐ sekundäre Datenbankmodelle berücksichtigen

43 Systeme im Ranking, April 2023

Rang			DBMS	Datenbankmodell	Punkte		
Apr 2023	Mär 2023	Apr 2022			Apr 2023	Mär 2023	Apr 2022
1.	1.	1.	InfluxDB	Time Series, Multi-Model	28,59	-0,56	-1,43
2.	2.	2.	Kdb	Time Series, Multi-Model	8,47	+0,43	-0,31
3.	3.	3.	Prometheus	Time Series	6,99	-0,34	+0,68
4.	4.	4.	Graphite	Time Series	6,30	-0,28	+0,94

Next steps

- *Fundamentals*
  - Architecture, distribution, consistency, request processing
- *Key-value*
  - Redis, LSM index structure, consistent hashing
  - Versioning
- *Document*
  - MongoDB, JSON
- *Wide-Column*
  - Cassandra, compression
- *Search Engine*
  - Elastic Search, Ranking
- *Graph*
  - Neo4J, query language with pattern matching
- *Time Series*
  - InfluxDB, SQL extensions