

Big Data

Isabella Aspöding, Alexander Pils

Paris Lodron Universität Salzburg

22. November, 2019

Inhalt

- 1 Allgemein
 - Definition
 - Unterschied
 - Datenherkunft
 - Wachstum von Daten
 - Anwendung
- 2 Analysemethoden
- 3 Entwicklungen
 - NoSQL
 - JSON
 - Map Reduce
 - Hadoop
 - R
 - Spark

Allgemein

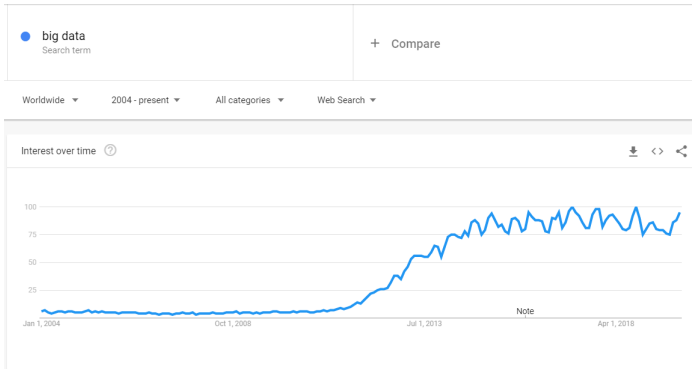


Figure: <https://trends.google.com/trends/>

Definition

- Volume (Datenvolumen)
- Velocity (Geschwindigkeit der Datenverarbeitung und Veränderungsdynamik)
- Variety (Vielfalt der Datenstrukturen und -klassen)
- Veracity (Echtheit der Daten)
- Value (unternehmerischer Mehrwert)
- Validity (Datenqualität)

Unterschied

Traditionelle Analytik

- Schrittweise Analyse von kleinen Datenmengen
- Daten werden angesammelt, bearbeitet, gespeichert und erst dann analysiert

Big Data Analytik

- Bearbeitung der ganzen Datenmenge
- Daten werden unverändert bearbeitet
- Analyse und Bearbeitung werden je nach Eingang durchgeführt

Datenherkunft

- ① Aufzeichnungen verschiedenster Überwachungssysteme.
- ② die Nutzung von Kunden- oder Bank- bzw. Bezahlkarten
- ③ die Nutzung eines Smartphones
- ④ Social-Media
- ⑤ Kraftfahrzeuge
- ⑥ vernetzte Technik in Häusern
- ⑦ von Behörden und Unternehmen erhobene und gesammelte Daten.

Wachstum von Daten

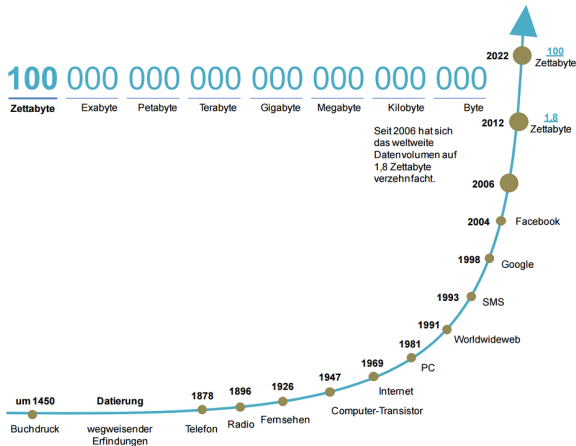


Figure: <https://bigdatablog.actgruppe.de/2017/04/06/die-geschaefftliche->

Anwendung

- ① Wahlen
- ② Social Scoring
- ③ Bildungswesen
- ④ Wirtschaft
- ⑤ Staat
- ⑥ Gesundheit
- ⑦ Umwelt

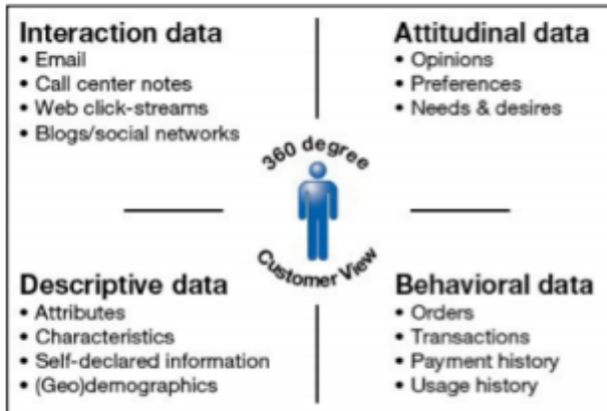
Analysemethoden

- Repräsentative Stichprobe
- Kopplungsanalyse
- Predictive Analytics

Entwicklungen

- NOSQL (Not Only SQL)
- JSON
- Map Reduce
- Hadoop
- Spark
- R

Kopplungsanalyse



NoSQL

- Objektdatenbanken
- Grid- und Cloud-Datenbanken
- XML-Datenbanken
- Andere nicht-relationale Datenbanken

NoSQL

Kriterien

- Nichtrelationales Datenmodell
- Schemafrei (oder nur schwache Restriktionen)
- Bieten einfache API
- Verteilte Architektur, optimiert für einfache Replikation
- Skalierung
- Kein ACID-Konsistenzmodell
- Open Source

NoSQL

Size vs. Complexity

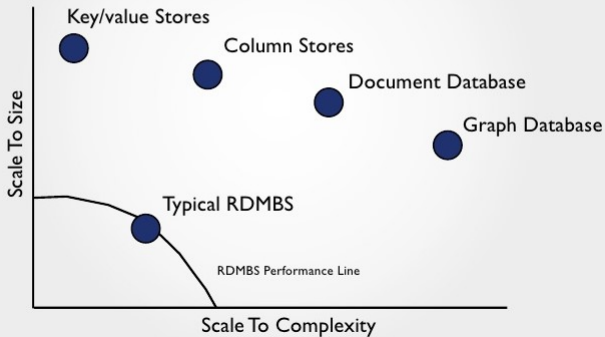


Figure: <https://www.slideshare.net/timjuravich/mysql-nosql-from-a-php-perspective>

JavaScript Object Notation

JSON ist ein kompaktes Datenformat in einer einfach lesbaren Textform zum Zweck des Datenaustauschs zwischen Anwendungen.

```
{  
  "Herausgeber": "Xema",  
  "Nummer": "1234-5678-9012-3456",  
  "Deckung": 2e+6,  
  "Waehrung": "EURO",  
  "Inhaber": {  
    "Name": "Mustermann",  
    "Vorname": "Max",  
    "maennlich": true,  
    "Hobbys": ["Reiten", "Golfen", "Lesen"],  
    "Alter": 42,  
    "Kinder": [],  
    "Partner": null  
  }  
}
```

Map Reduce

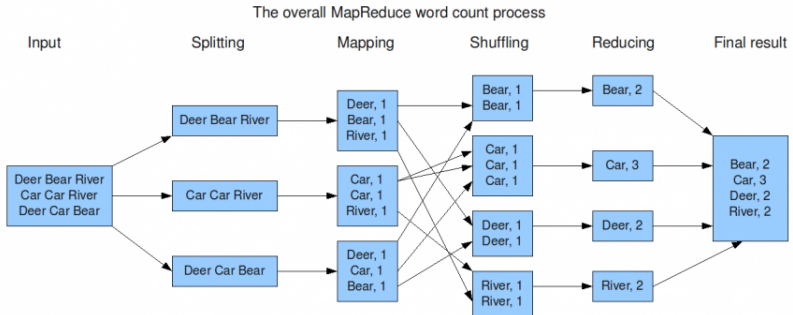


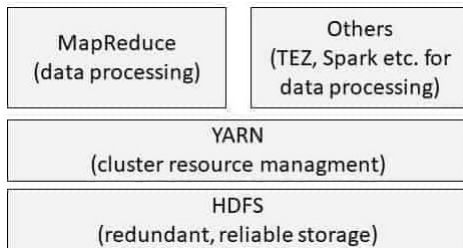
Figure: <http://beyondthegreek.com/2016/07/13/mapper-reducer/>

Map Reduce

Wortzahl

```
1 function map(String name, String documentPart):  
2     for each word w in documentPart:  
3         emit (w, 1)  
4  
5 function reduce(String word, List<Int> partialCounts):  
6     sum = 0  
7     for each pc in partialCounts:  
8         sum += pc  
9     emit (word, sum)
```

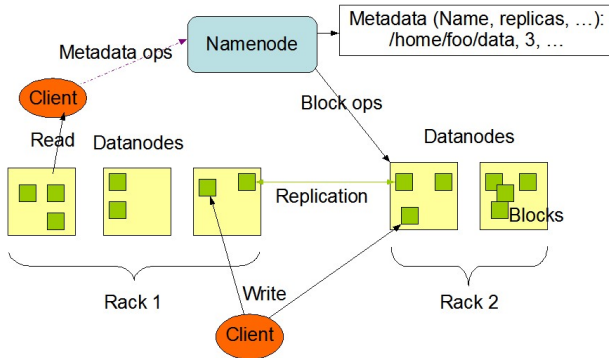
Hadoop



- HDFS (Hadoop Distributed File System)
- YARN (Yet Another Resource Negotiator)
- Map Reduce

HDFS

HDFS Architecture



R

Paradigmen:

- funktional
- dynamisch
- objektorientiert

R Beispiel Code

```
1 | Gewicht <- c(60, 72, 57, 90, 95, 72)
2 | Groesse <- c(1.75, 1.80, 1.65, 1.90, 1.74, 1.91)
3 | BMI <- Gewicht / Groesse^2
4 | sum(Gewicht)
5 | length(Gewicht)
6 | sum(Gewicht) / length(Gewicht)
7 | table(Gewicht)
```

Spark

- einheitliches In- Memory System
- zur Verarbeitung von enormen Datenmengen geeignet
- Framework für Clustercomputing
- Open Source
- Konzept der Resilient Distributed Datasets (RDD)

Spark

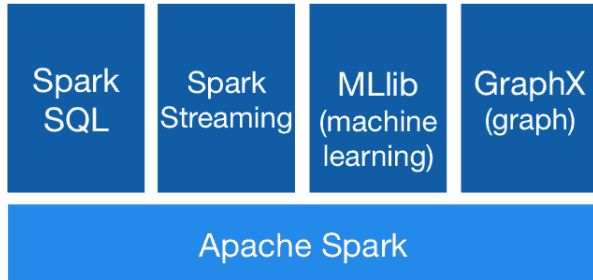


Figure: <http://archive.ibmssystemsmag.com/mainframe/business-strategy/bi-and-analytics/dynamic-spark/>

Zusammenfassung

Schöne neue Welt?



Danke für Ihre Aufmerksamkeit!