# Introduction

## Lecture BigData Analytics

Julian M. Kunkel

julian.kunkel@googlemail.com

University of Hamburg / German Climate Computing Center (DKRZ)

16-10-2015

# Outline

# About DKRZ

## German Climate Computing Center (DKRZ)

**Partner for Climate Research**

Maximum Compute Performance.

Sophisticated Data Management.

Competent Service.

# Scientific Computing

- Research Group of Prof. Ludwig at the University of Hamburg
- Embedded into DKRZ

## Research

- Analysis of parallel I/O
- I/O & energy tracing tools
- Middleware optimization
- Alternative I/O interfaces
- Data reduction techniques
- Cost & energy efficiency

# Lecture

## Concept of the lecture

- The lecture is focussing on applying technology and some theory
- Theory
  - Data models and processing concepts
  - Algorithms and data structures
  - System architectures
  - Statistics and machine learning
- Applying technology
  - Learning about various state-of-the art technology
  - Hands-on for understanding the key concepts
  - Languages: Java, Python, R
- The domain of big data is overwhelming, especially in terms of technology
- It is a crash course for several topics such as statistics and databases

$\Rightarrow$ it is not the goal to learn and understand every aspect in this lecture

# Lecture (2)

## Slides

- Many openly accessable sources have been used
- Citation to them by a number
- The reference slide provides the link to the source
- For figures, a reference is indicated by *Source: [Author]*[1] *[title] [ref]*
- In the title, an [ref] means that this reference has been used for the slide, some text may be taken literally

## Excercise

- Weekly delivery, processing time about 8 hours / per week estimated
- Teamwork of 2 or 3 people (groups are mandatory!)
- Supported by: Hans Ole Hatzel

[1] If available

# Idea of BigData

## Methods of obtaining knowledge (Erkenntnissprozess)

Theory (model), hypothesis, experiment, analysis (repeat)

- Explorative: start theory with observations of phenomena
- Constructivism: starts with axioms and reason implications

## The Fourth Paradigm

- (Big) Data + Analytics $\Rightarrow$Insight (prediction of the future)
  - For industry: insight = business advantage and money...
- Analytics: follow an explorative approach and study the data
  - To infer knowledge, use statistics / machine learning
- Construct a theory (model) and validate it with the data

# Example Models

Similarity is a (very) simplistic model and predictor for the world

- Humans use this approach in their cognitive process
- Uses the advantage of BigData

## Weather prediction

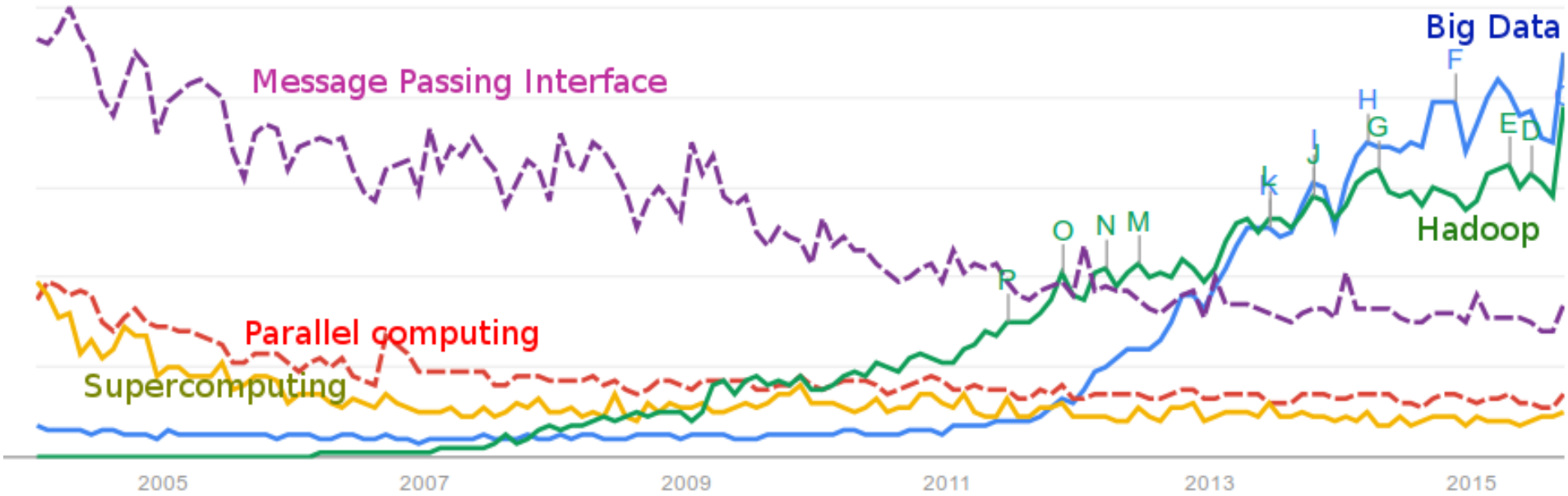
- You may develop and rely on complex models of physics
- Or use a simple model for a particular day; e.g. expect it to be similar to the weather of the day over the last X years
- Used by humans: rule of thumb for farmers

## Preferences of Humans

- Identify a set of people which liked items you like
- Predict you like also the items those people like (items you haven't rated so far)

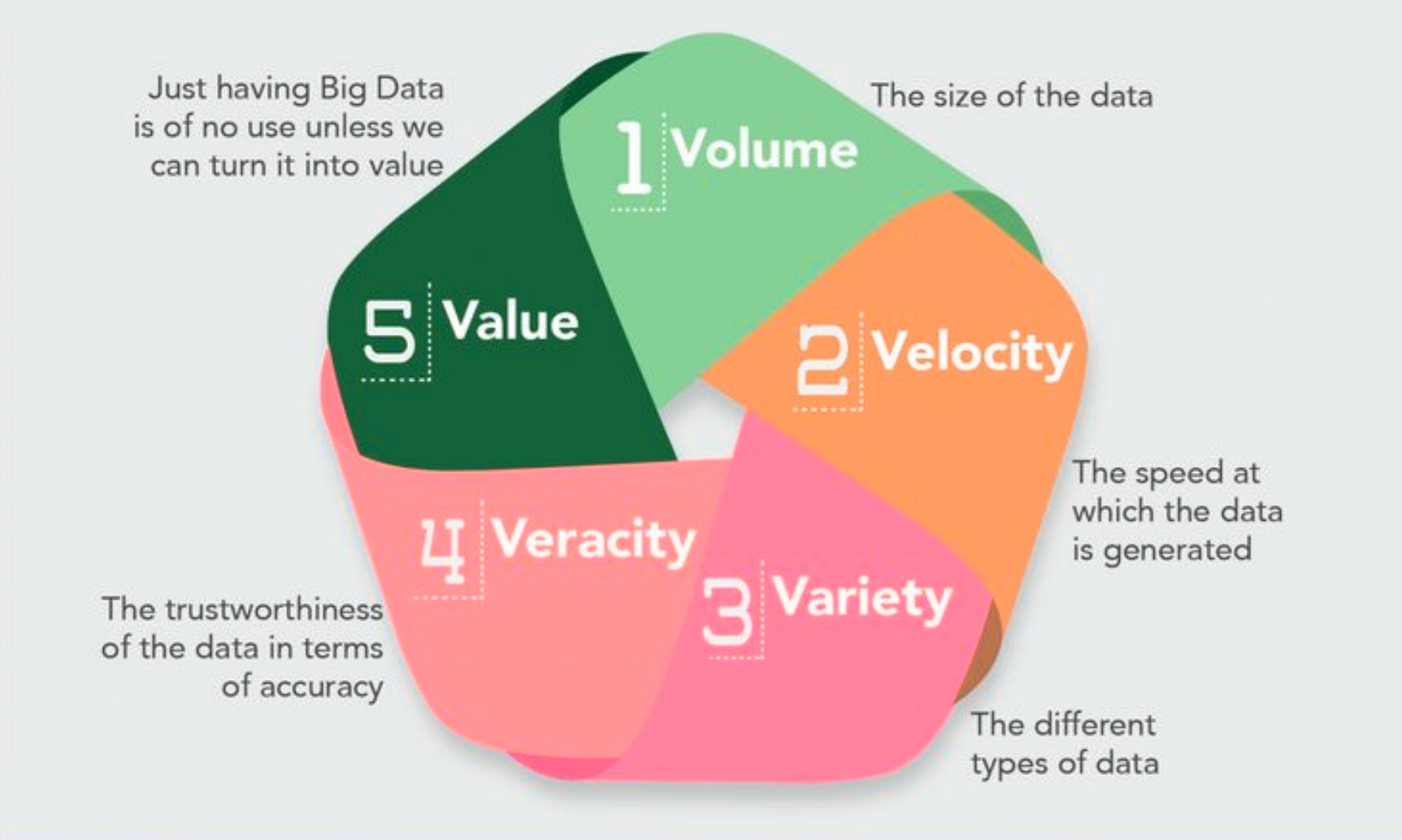
# Relevance of Big Data

- Big Data Analytics is emerging
- Relevance increases compared to supercomputing




Google Search Trends, relative searches

# BigData Challenges & Characteristics




Source: MarianVesper [4]

# Volume: The size of the Data

## What is Big Data

Terrabytes to 10s of petabytes

## What is not Big Data

A few gigabytes

## Examples

- Wikipedia corpus with history ca. 10 TByte
- Wikimedia commons ca. 23 TByte
- Google search index ca. 46 Gigawebpages[2]
- YouTube per year 76 PByte (2012[3])

[2] http://www.worldwidewebsize.com/

[3] https://sumanrs.wordpress.com/2012/04/14/youtube-yearly-costs-for-storagenetworking-estimate/

# Velocity: Data Volume per Time

## What is Big Data

30 KiB to 30 GiB per second
(902 GiB/year to 902 PiB/year)

## What is not Big Data

A never changing data set

## Examples

- LHC (Cern) with all experiments about 25 GB/s [4]
- Square Kilometre Array 700 TB/s (in 2018) [5]
- 50k Google searches per s [6]
- Facebook 30 Billion content pieces shared per month [7]

[4] http://home.web.cern.ch/about/computing/processing-what-record
[5] http://venturebeat.com/2014/10/05/how-big-data-is-fueling-a-new-age-in-space-exploration/
[6] http://www.internetlivestats.com/google-search-statistics/
[7] https://blog.kissmetrics.com/facebook-statistics/

# Data Sources

## Enterprise data

- Serves business objectives, well defined
- Customer information
- Transactions, e.g. Purchases

## Experimental/Observational data (EOD)

- Created by machines from sensors/devices
- Trading systems, satellites
- Microscopes, video streams, Smart meters

## Social media

- Created by humans
- Messages, posts, blogs, Wikis

# Variety: Types of Data

- Structured data
  - Like tables with fixed attributes
  - Traditionally handled by relational databases
- Unstructured data
  - Usually generated by humans
  - E.g. natural language, voice, Wikipedia, Twitter posts
  - Must be processed into (semi-structured) data to gain value
- Semi-structured data
  - Has some structure in tags but it changes with documents
  - E.g. HTML, XML, JSON files, server logs

## What is Big Data

- Use data from multiple sources and in multiple forms
- Involve unstructured and semi-structured data

# Veracity: Trustworthiness of Data

## What is Big Data

- Data involves some uncertainty and ambiguities
- Mistakes can be introduced by humans and machines
  - People sharing accounts
  - Like sth. today, dislike it tomorrorw
  - Wrong system timestamps

## Data Quality is vital!

Analytics and conclusions rely on good data quality

- Garbage data + perfect model => garbage results
- Perfect data + garbage model => garbage results

GIGO paradigm: *Garbage In – Garbage Out*

# Value of Data

## What is Big Data

- Raw data of Big Data is of low value
  - For example, single observations
- Analytics and theory about the data increases the value

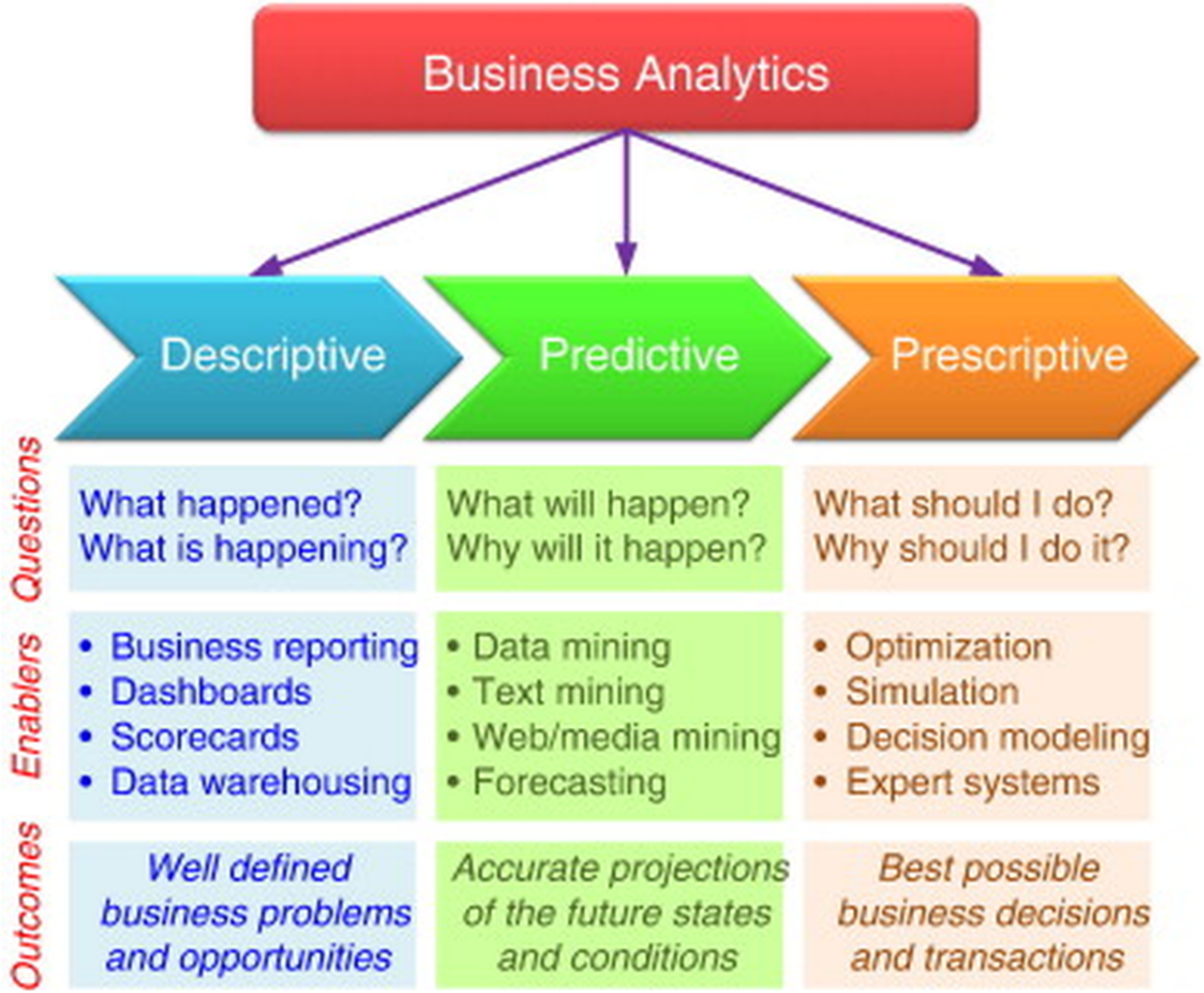
Analytics transform big data into smart data!

# Types of Data Analytics and Value of Data

1. Descriptive analytics (Beschreiben)
   - “What happened?”
2. Diagnostic analytics
   - “Why did this happen, what went wrong?”
3. Predictive analytics (Vorhersagen)
   - “What will happen?”
4. Prescriptive analytics (Empfehlen)
   - “What should we do and why?”

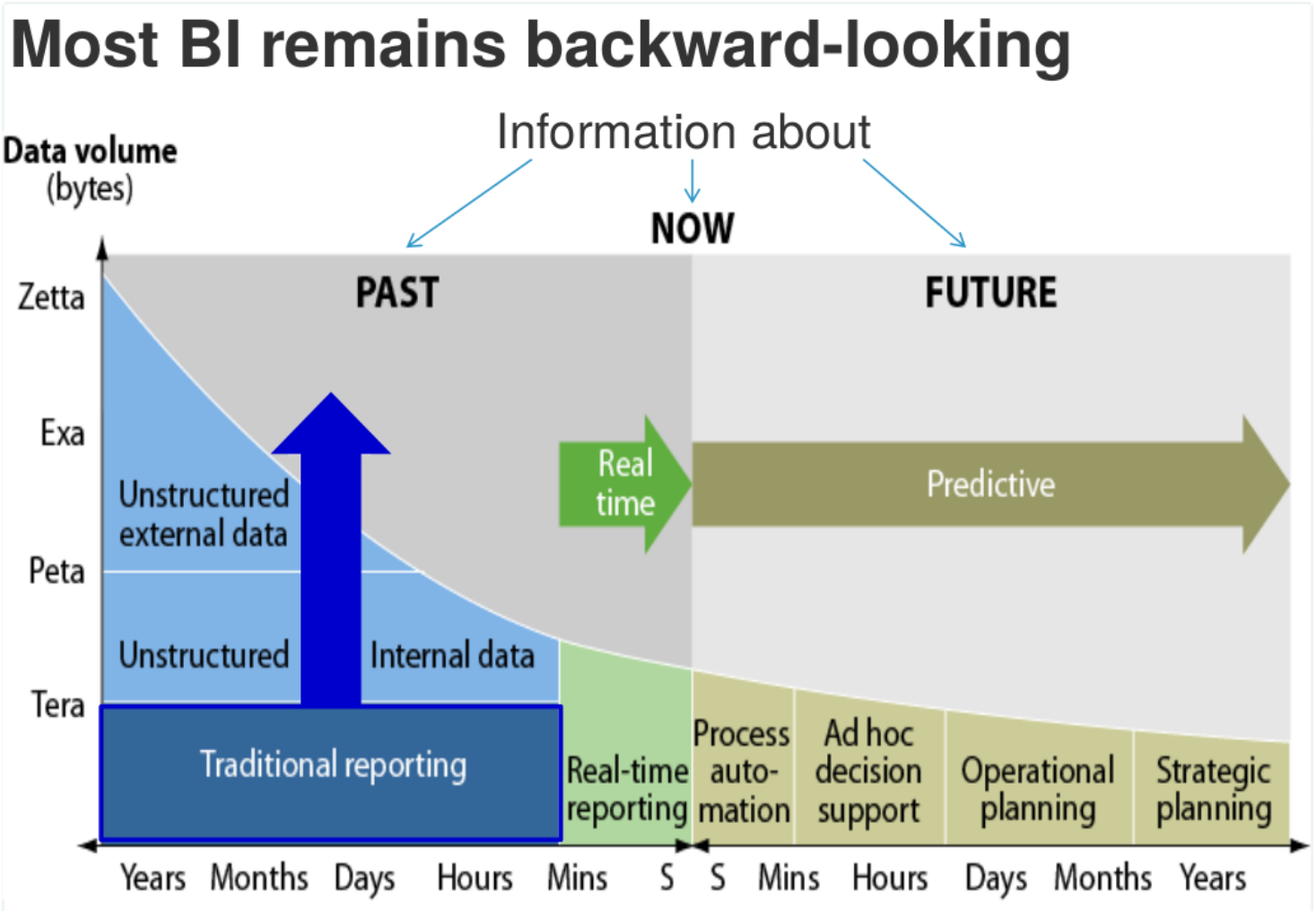
The level of insight and value of data increases from step 1 to 4

# The Value of Data (alternative view)




Source: Dursun Delen, Haluk Demirkan [9]

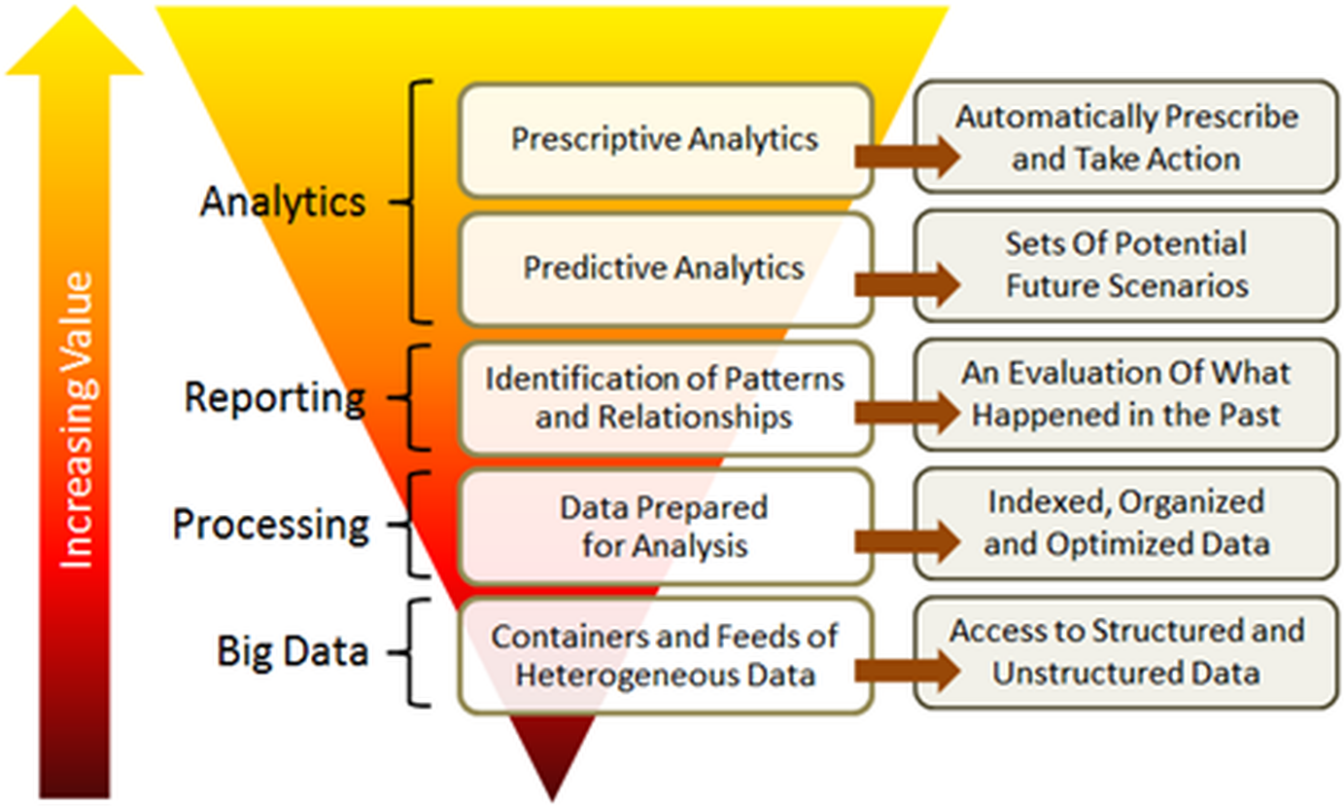
# The Value of Data (alternative view 2)




Source: Forrester report. Understanding The Business Intelligence Growth Opportunity. 20-08-2011

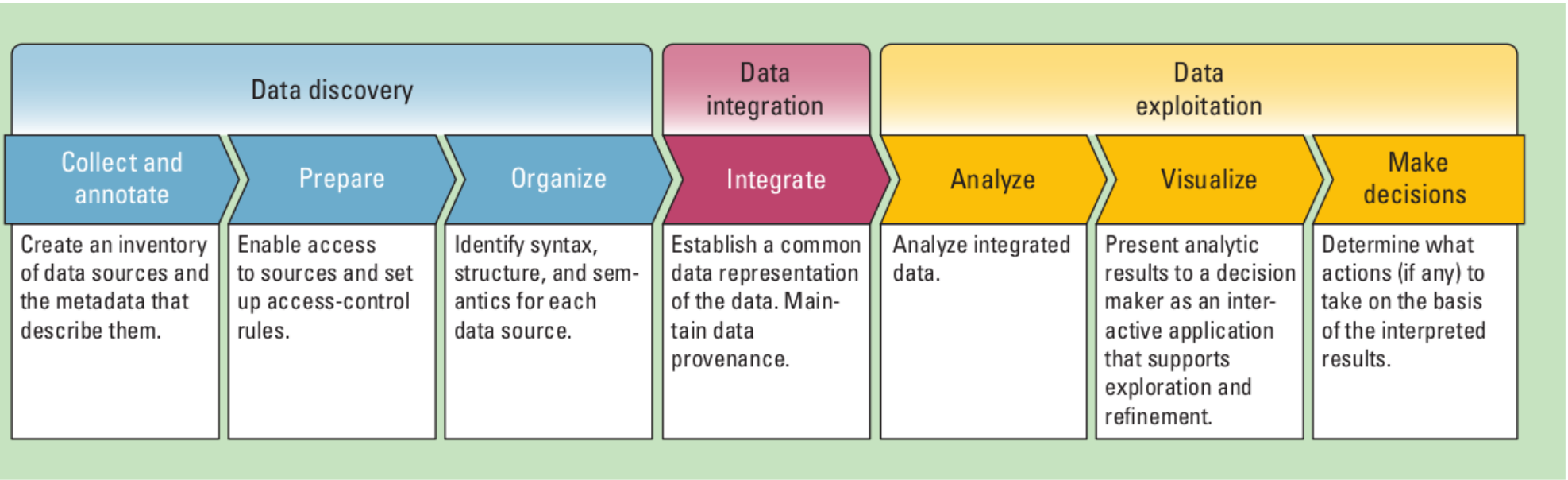
# Big Data Analytics Value Chain

There are many visualizations of the processing and value chain [8]




Source: Andrew Stein [8]

# Big Data Analytics Value Chain (2)




Source: Miller and Mork [7]

# Roles in the Big Data Business

## Data scientist

*Data science is a systematic method dedicated to knowledge discovery via data analysis* [1]

- In business, optimize organizational processes for efficiency
- In science, analyze experimental/observational data to derive results

## Data engineer

*Data engineering is the domain that develops and provides systems for managing and analyzing big data*

- Build modular and scalable data platforms for data scientists
- Deploy big data solutions

# Typical Skills

## Data scientist

- Statistics + (Mathematics)
- Computer science
  - Programming e.g.: Java, Python, R, (SAS, ...)
  - Machine learning
- Some domain knowledge for the problem to solve

## Data engineer

- Computer science
  - Databases
  - Software engineering
  - Massively parallel processing
  - Real-time processing
- Languages: C++, Java, Python
- Understand performance factors and limitations of systems

# Data Science vs. Business Intelligence (BI)

## Characteristics of BI

- Provides pre-created dashboards for management
  - Repeated visualization of well known analysis steps
- Deals with structured data
- Typically data is generated within the organization
- Central data storage (vs. multiple data silos)
- Handeled well by specialized database techniques

## Typical types of insight

- Customer service data: “what business causes the largest customer wait times”
- Sales and marketing data: “which marketing is most effective”
- Operational data: “efficiency of the help desk”
- Employee performance data: “who is most/least productive”

# Privacy

Be aware of privacy issues if you deal with personal/private information.
German privacy laws are more strict than those of other countries

## Ziel des Datenschutzes

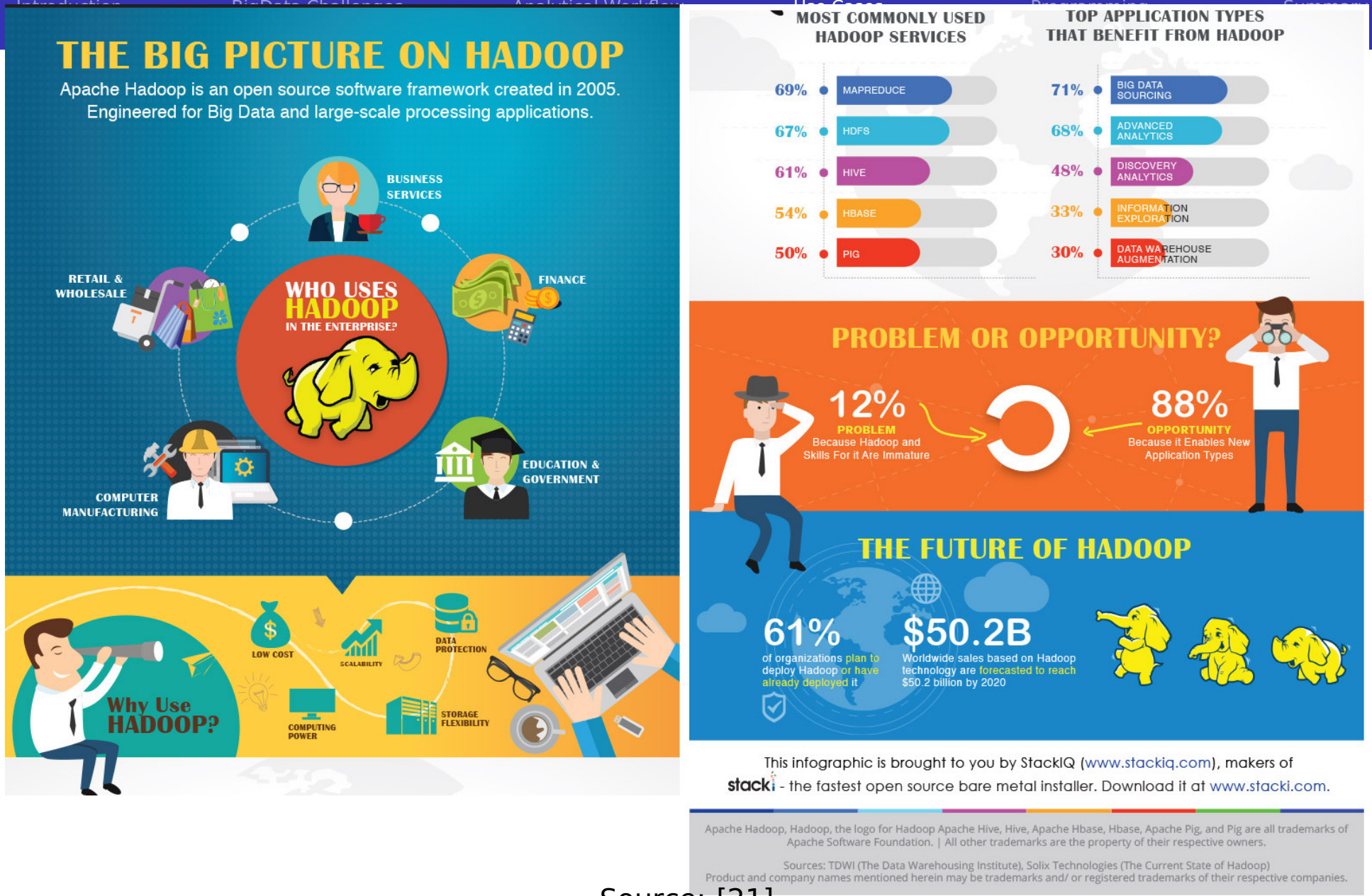
Recht auf informationelle Selbstbestimmung

- Schutz des Einzelnen vor beeinträchtigung des Persönlichkeitsrechts durch den Umgang mit seinen personenbezogenen[8] Daten
- Besonderer Schutz für Daten über Gesundheit, ethnische Herkunft, religiöse, gewerkschaftschliche oder sexuelle Orientierung

[8] §3 BDSG, Einzelangaben über persönliche oder sachliche Verhältnisse einer bestimmten oder bestimmbaren natürlichen Person

# Wichtige Grundsätze des Gesetzes [10]

- Verbotsprinzip mit Erlaubsnisvorbehalt
  - Erhebung, Verarbeitung, Nutzung und Weitergabe von personenbezogenen Daten sind verboten
  - Nutzung nur mit Rechtsgrundlage oder mit Zustimmung der Person
- Unternehmen mit 10 Personen benötigen Datenschutzbeauftragten
- Verfahren zur automatischen Verarbeitung sind vom Datenschutzbeauftragten zu prüfen und anzeigepflichtig
- Sitz der verantwortlichen Stelle maßgeblich
  - Bei einer Niederlassung in D gilt BDSG
- Prinzipien: Datenvermeidung, -sparsamkeit
- Schutz vor Zugriffen, Änderungen und Weitergabe
- Betroffene haben Recht auf Auskunft, Löschung oder Sperrung
- Anonymisierung/Pseudonymisierung: Ist die Zuordnung zu Einzelpersonen (nahezu) ausgeschlossen, so können Daten verabeitet werden


THE BIG PICTURE ON HADOOP
Apache Hadoop is an open source software framework created in 2005.
Engineered for Big Data and large-scale processing applications.
BUSINESS SERVICES
FINANCE
EDUCATION & GOVERNMENT
COMPUTER MANUFACTURING
RETAIL & WHOLESALE
WHO USES HADOOP IN THE ENTERPRISE?
Why Use HADOOP?
LOW COST
SCALABILITY
DATA PROTECTION
COMPUTING POWER
STORAGE FLEXIBILITY
MOST COMMONLY USED HADOOP SERVICES
69% MAPREDUCE
67% HDFS
61% HIVE
54% HBASE
50% PIG
TOP APPLICATION TYPES THAT BENEFIT FROM HADOOP
71% BIG DATA SOURCING
68% ADVANCED ANALYTICS
48% DISCOVERY ANALYTICS
33% INFORMATION EXPLORATION
30% DATA WAREHOUSE AUGMENTATION
PROBLEM OR OPPORTUNITY?
12% PROBLEM
Because Hadoop and Skills For it Are Immature
88% OPPORTUNITY
Because it Enables New Application Types
THE FUTURE OF HADOOP
61%
of organizations plan to deploy Hadoop or have already deployed it
$50.2B
Worldwide sales based on Hadoop technology are forecasted to reach $50.2 billion by 2020
This infographic is brought to you by StackIQ (www.stackiq.com), makers of
stacki - the fastest open source bare metal installer. Download it at www.stacki.com.
Apache Hadoop, Hadoop, the logo for Hadoop Apache Hive, Hive, Apache Hbase, Hbase, Apache Pig, and Pig are all trademarks of Apache Software Foundation. | All other trademarks are the property of their respective owners.
Sources: TDWI (The Data Warehousing Institute), Solix Technologies (The Current State of Hadoop)
Product and company names mentioned herein may be trademarks and/ or registered trademarks of their respective companies.

Source: [21]

# Use Cases for BigData Analytics

## Increase efficiency of processes and systems

- Advertisement: Optimize for target audience
- Product: Acceptance (like/dislike) of buyer, dynamic pricing
- Decrease financial risks: fraud detection, account takeover
- Insurance policies: Modeling of catastrophes
- Recommendation engine: Stimulate purchase/consume
- Systems: Fault prediction and anomaly detection
- Supply chain management

## Science

- Epidemiology research: Google searches indicate Flu spread
- Personalized Healthcare: Recommend good treatment
- Physics: Finding the Higgs-Boson, analyze telescope data
- Enabler for social sciences: Analyze people's mood

# Big Data in Industry

| INDUSTRY | USE CASE | DATA TYPE | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Sensor | Server Logs | Text | Social | Geographic | Machine | Clickstream | Structured | Unstructured |
| **Financial Services** | New Account Risk Screens | | ✔ | ✔ | | | | | | |
| | Trading Risk | | ✔ | | | | | | | |
| | Insurance Underwriting | ✔ | | ✔ | | ✔ | | | | |
| **Telecom** | Call Detail Records (CDR) | | | | | ✔ | ✔ | | | |
| | Infrastructure Investment | | ✔ | | | | ✔ | | | |
| | Real-time Bandwidth Allocation | | ✔ | ✔ | ✔ | | | | | |
| **Retail** | 360° View of the Customer | | | ✔ | | | | ✔ | | |
| | Localized, Personalized Promotions | | | | | ✔ | | | | |
| | Website Optimization | | | | | | | ✔ | | |
| **Manufacturing** | Supply Chain and Logistics | ✔ | | | | | | | | |
| | Assembly Line Quality Assurance | ✔ | | | | | | | | |
| | Crowd-sourced Quality Assurance | | | | ✔ | | | | | |
| **Healthcare** | Use Genomic Data in Medial Trials | ✔ | | | | | | | ✔ | |
| | Monitor Patient Vitals in Real-Time | | | | | | | | | |
| **Pharmaceuticals** | Recruit and Retain Patients for Drug Trials | | | | ✔ | | | ✔ | | |
| | Improve Prescription Adherence | | | | ✔ | ✔ | | | | ✔ |
| **Oil & Gas** | Unify Exploration & Production Data | ✔ | | | | ✔ | | | | ✔ |
| | Monitor Rig Safety in Real-Time | ✔ | | | | | | | | ✔ |
| **Government** | ETL Offloaded Response to Federal Budgetary Pressures | | | | | | | | ✔ | |
| | Sentiment Analysis for Government Programs | | | | ✔ | | | | | |

Source: [20]

# Example Use Case: Deutschland Card [2]

## Goals

- Customer bonus card which tracks purchases
- Increase scalability and flexibility
- Previous solution based on OLAP

## Big Data Characteristics

- Volume: O(10) TB
- Variety: mostly structured data, schemes are extended steadily
- Velocity: data growth rate O(100) GB / month

## Results

- Much better scalability of the solution
- From dashboards to ad-hoc analysis within minutes

# Example Use Case: DM [2]

## Goals

- Predict required employees per day and store
- Prevent staff changes on short-notice

## Big Data Characteristics

- Input data: Opening hours, incoming goods, empl. preferences, holidays, weather ...
- Model: NeuroBayes (Bayes + neuronal networks)
- Predictions: Sales, employee planning
- 450.000 predictions per week

## Results

- Daily updated sales per store
- Reliable predictions for staff planning
- Customer and employee satisfaction

# Example Use Case: OTTO [2]

## Goals

Optimize inventory and prevent out-of-stock situations

## Big Data Characteristics

- Input data: product characteristics, advertisement
- Volume/Velocity: 135 GB/week, 300 million records
- Model: NeuroBayes (Bayes + neuronal networks)
- 1 billion predictions per year

## Results

- Better prognostics of product sales (up to 40%)
- Real time data analytics

# Example Use Case: Smarter Cities (by KTH) [2]

## Goals

- Improve traffic management in Stockholm
- Prediction of alternative routes

## Big Data Characteristics

- Input data: Traffic videos/sensors, weather, GPS
- Volume/Velocity: 250k GPS-data/s + other data sources

## Results

- 20% less traffic
- 50% reduction in travel time
- 20% less emissions

# Example Facebook Studies

## Insight from [11] by exploring posts

- Young narcissists tweet more likely.
  Middle-aged narcissists update their status
- US students post more problematic information than German students
- US Government checks tweets/facebook messages for several reasons
- Human communication graph has an average diameter of 4.74

## Manipulation of news feeds [13]

- News feeds have been changed to analysis people's behavior in subsequent posts
- Paper: "Experimental evidence of massive-scale emotional contagion through social networks"

# From Big Data to the Data Lake [20]

- With cheap storage costs, people promote the concept of the data lake
- Combines data from many sources and of any type
- Allows for conducting future analysis and not miss any opportunity

## Attributes of the data lake

- Collect everything: all data, both raw sources over extended periods of time as well as any processed data
  - Decide during analysis which data is important, e.g. no “schema“ until read
- Dive in anywhere: enable users across multiple business units to refine, explore and enrich data on their terms
- Flexible access: enable multiple data access patterns across a shared infrastructure: batch, interactive, online, search, and others

# Programming BigData Analytics

## High-level concepts

- SQL and derivatives
- Domain-specific languages (Cypher, PigLatin)

## Programming languages

- Java interfaces are widely available but low-level
- Python and R have connectors to popular BigData solutions

In the exercises, we'll learn and use basics of those languages/interfaces

# Introduction to Java

- Developed by Sun Microsystems in 1995
- Object oriented programming language
- OpenJDK implementation is open source
- Source code ⇒byte code ⇒just-in-time compiler
  - Byte code is portable & platform independent
  - Virtual machine abstracts from systems
- Strong and static type system
- Popular language for Enterprise & Big Data applications
  - Most popular programming language (Pos. 1 on the TIOBE index)
- Development tools: Eclipse

## Specialties

- Good runtime and compile time error reporting
- Generic data types (vs. templates of C++)
- Introspection via. Reflection

# Example Java Program

```
1  import java.util.Scanner;
2  import java.io.FileReader;
3  import java.io.FileNotFoundException;
4  // compile with javac program.java
5  // run with java program
6  public class program{
7     // the main method is part of a class
8     public static void main(String [ ] args) throws FileNotFoundException{
9        try{
10          // read from file "program.java" and create simple tokens
11          Scanner data = new Scanner(new FileReader("program.java"));
12          while(data.hasNext()){
13             System.out.println(data.next());
14          }
15       }catch(Exception e){
16          // handle error here, we'll just rethrow the error
17          throw(e);
18       }
19    }
20 }
```

# Example Java Classes

```
1  // Run: javac classes1.java and java Rabbit
2  // An abstract class is not completely implemented
3  abstract class Animal{
4     // instance member
5     private float weight;
6     // not-implemented instance function
7     public abstract String name();
8     // constructor
9     public Animal(float weight){ this.weight = weight; }
10    public String toString(){ return "I'm a " + name() + " with " +
11       weight + " kg"; }
12 }
13
14 class Rabbit extends Animal{
15    // invoke the constructor of the parent
16    public String name(){ return "Rabbit"; }
17    public Rabbit(){ super(2.5f); }
18
19    // the main method is part of a class
20    public static void main(String [ ] args){
21       Animal a = new Rabbit();
22       System.out.println(a); // I'm a Rabbit with 2.5 kg
23    }
24 }
25
26
```

# Introduction to Python

- Open source
- Position 5 on TIOBE index
- Interpreted language
- Weak type system (errors at runtime)
- Development tools: any editor, interactive shell
- Note: Use and learn python3 explicitly
- Recommended plotting library: matplotlib[9]

## Specialties

- Strong text processing
- Simple to use
- Support for object oriented programming
- Indentation is relevant for code blocks

[9] http://matplotlib.org/gallery.html

# Example Python Program

```
1  #!/bin/env python
2  import re # use the module 're'
3
4  # function reading a file
5  def readFile(filename):
6     with open(filename, 'r') as f:
7        data = f.readlines()
8        f.close()
9        return data
10    return [] # return an empty array/list
11
12 # the main function
13 if __name__ == "__main__":
14    data = readFile('intro.py')
15    # iterate over the array
16    for x in data:
17       # extract imports from a python file using a regex
18       m = re.match("import[ \t]+(?P<WHAT>[^# ]*)", x)
19       if m:
20          print(m.group("WHAT"))
21          # dictionary (key value pair)
22          dic = m.groupdict()
23          dic.update( {"FILE" : 'intro.py'}) # append a new dict. with one key
24          # use format string with dictionary
25          print("Found import '%(WHAT)s' in file %(FILE)s" % dic )
26          # Prints: Found import 're' in file intro.py
```

# Example Python Classes

```python
from abc import abstractmethod

class Animal():
   # constructor, self are instance methods, else class methods
   def __init__(self, weight):
      self.__weight = weight # private variables start with __

   # decorator
   @abstractmethod
   def name(self):
      return self.__class__.__name__ # reflection like

   def __str__(self):
      return "I'm a %s with weight %f" % (self.name(), self.__weight)

class Rabbit(Animal):
   def __init__(self):
      # super() is available with python 3
      super().__init__(2.5)

   def name(self):
      return "Small Rabbit" # override name

if __name__ == "__main__":
   r = Rabbit()
   print(r) # print: I'm a Small Rabbit with weight 2.500000
```

# Introduction to R

- Based on S language for statisticians
- Open source
- Position 19 on TIOBE index
- Interpreter with C modules (packages)
  - Easy installation of packages via CRAN[10]
- Popular language for data analytics
- Development tools: RStudio (or any editor), interactive shell
- Recommended plotting library: ggplot2[11]

## Specialties

- Vector/matrix operations. Note: Loops are slow, so avoid them
- Table data structure (data frames)

[10] Comprehensive R Archive Network

[11] http://docs.ggplot2.org/current/

# Course for Learning R Programming

```r
# Run with "Rscript intro.R" or run "R" and copy&paste into interactive shell
# Installing a new package is as easy as:
install.packages("swirl")
# Note: sometimes packages are not available on all mirrors!
library(swirl) # load the package

help(swirl) # read help about the function swirl

swirl() # start an interactive course to learn R

# a simple for loop
for (x in 1:10){
   if (x < 5){
      print(x)
   }else{
      print(x * 2)
   }
}
```

# Example R Program

```
1  # create an array
2  x = c(1, 2, 10:12)
3
4  # apply an operator on the full vector and output it
5  print( x*2 ) # prints: 2  4 20 22 24
6
7  # slice arrays
8  print ( x[3:5] ) # prints: 10 11 12
9  print( x[c(1,4,8)] ) # prints: 1 11 NA
10
11 r = runif(100, min=0, max=100) # create array with random numbers
12 m = matrix(r, ncol=4, byrow = TRUE) # create a matrix
13
14 # slice matrix rows "m[row(s), column(s)]"
15 print( m[10:12, ] ) # Output:
16 #          [,1]      [,2]       [,3]      [,4]
17 #[1,] 85.46609 60.749703 10.5062183  7.449173
18 #[2,] 79.76042 52.199321 96.9699856 97.877946
19 #[3,] 37.34286  8.266282  0.3398741  1.957607
20
21 # slice rows & columns
22 print ( m[10, c(1,4)]  ) # Output: [1] 85.466085  7.449173
23
24 # subset the table based on a mask
25 set = m[ (m[,1] < 20 & m[,2] > 2) , ]
```

# Accessing CSV Files with R

```
# function to create a table (data frame) and fill it with random data
createTable = function (size){
   tbl = read.table(text="", col.names = c("Type", "Time"))
   tbl[1:size, ] = 0  # initialize size times a full rows
   tbl$Time = runif(size, min=0, max=100) # address by column name
   # create random types, factor() for nominal data and
   # ordered() for ordinal data
   tbl$Type = factor(round(runif(size, min=0.5, max=3.49)),
      levels=1:3, # three categories
      labels=c("unknown", "good", "bad"))
   tbl$Type[size] = "bad" # assign last element to be bad
   return (tbl)
}
# change columnnames
colnames(tbl) = c("Typ", "Duration")

d = createTable(5)
# Assign the column with the name
print( d )
print( summary (d) ) # some statistics about d
# Write CSV incl. header
write.table(d, file = "mydata.csv", sep=",", row.names=FALSE)
# reread table
d =  read.table("mydata.csv", header = TRUE, sep = ",")
```

# Summary

- Big data analytics
  - Explore data and model causalities to gain knowledge & value
- Challenges: 5 Vs – Volume, velocity, variety, veracity, value
- Data sources: Enterprise, humans, Exp./Observational data (EOD)
- Types of data: Structured, unstructured and semi-structured
- Levels of analytics: Descriptive, predictive and prescriptive
- Roles in big data business: Data scientist and engineer
- Data science != business “intelligence”

# Bibliography


1 Book: Lillian Pierson. **Data Science for Dummies**. John Wiley & Sons

2 Report: Jürgen Urbanski et.al. **Big Data im Praxiseinsatz – Szenarien, Beispiele, Effekte**. BITKOM

3 http://winfwiki.wi-fom.de/

4 Forrester Big Data Webinar. Holger Kisker, Martha Bennet. Big Data: Gold Rush Or Illusion?

5 http://blog.eoda.de/2013/10/10/veracity-sinnhaftigkeit-und-vertrauenswuerdigkeit-von-bigdata-als-kernherausforderung-im-informationszeitalter/

6 http://lehrerfortbildung-bw.de/kompetenzen/projektkompetenz/methoden/erkenntnis.htm

7 Gilbert Miller, Peter Mork From Data to Decisions: A Value Chain for Big Data. http://www.fh-schmalkalden.de/Englmeier-p-790/_/ValueChainBigData.pdf

8 Andrew Stein. The Analytics Value Chain. http://steinvox.com/blog/big-data-and-analytics-the-analytics-value-chain/

9 Dursun Delen, Haluk Demirkan,. Decision Support Systems, Data, information and analytics as services.http://j.mp/11bl9b9

10 Wikipedia

11 Kashmir Hill. 46 Things We've Learned From Facebook Studies. Forbe. http://www.forbes.com/sites/kashmirhill/2013/06/21/46-things-weve-learned-from-facebook-studies/

12 Hortonworks http://hortonworks.com/

13 http://www.huffingtonpost.com/2014/12/10/facebook-most-popular-paper_n_6302034.html

20 http://hortonworks.com/blog/enterprise-hadoop-journey-data-lake/

21 http://www.stacki.com/hadoop/?utm_campaign=Stacki+Hadoop+Infographic