GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN

Julian Kunkel

# High-Performance Data Analytics (HPDA)

2021-10-25

# Learning Outcomes

After the session, a participant should be able to:

- Name typical applications for high-performance data analytics
- Distinguish HPDA from D/P/S computing and how these topics blend
- Describe use-cases and challenges in the domain of D/P/S computing
- Describe how the scientific method relies on D/P/S computing
- Name big data challenges and the typical workflow
- Recite system characteristics for distributed/parallel/computational science
- Sketch generic D/P system architectures

# Outline

# High-Performance Data Analytics (HPDA)

## Definition

*High performance data analytics is the* ***process*** *of* ***quickly examining extremely large data sets*** *to find insights. This is done by using the* ***parallel processing*** *of high performance computing to run powerful analytic software.*

Source: https://www.omnisci.com/technical-glossary/high-performance-data-analytics

## Components to undestand

- Understanding analysis processes
- Managing large scale data sets
- Applying parallel processing
- Characterizing performance factors of high-performance compute and storage

# Distributed Computing

Field in computer science that studies **distributed systems**[1]

## Definition

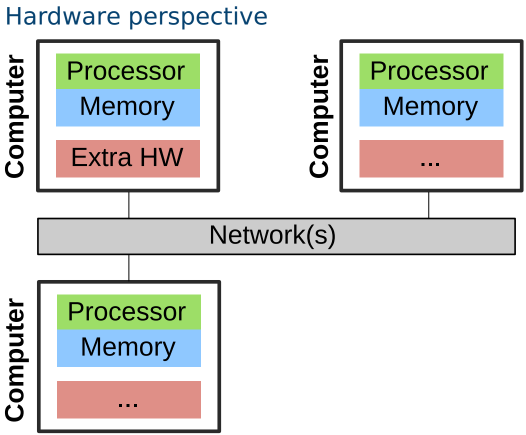
- System which components[2] are located on different networked computers
- Components communicate and coordinate actions by passing messages
- Components interact to achieve a common goal
- *In the wider sense*: autonomous processes coordinated by passing messages

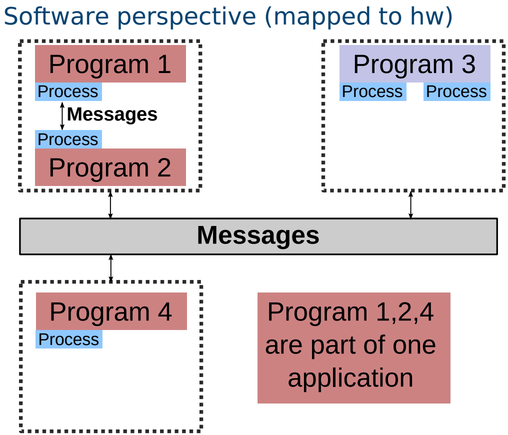
## Characteristics

- Distributed memory: components have their own (private) memory
- Concurrency of components: different components compute at the same time
- Lack of a global clock: clocks may diverge
- Independent failure of components, e.g., due to power outage

# Example Distributed System and Distributed Program

- A **distributed program** (DP) runs on a distributed system
  - Processes are instances of one program running on one computer
- A **distributed applications/algorithm** may involve various DPs/different vendors



# Example Distributed Applications and Algorithms

## Applications

- The Internet and telecommunication networks
- Cloud computing
- Wireless sensor networks
- The Internet of Things (IoT) – "everything is connected to the Internet"

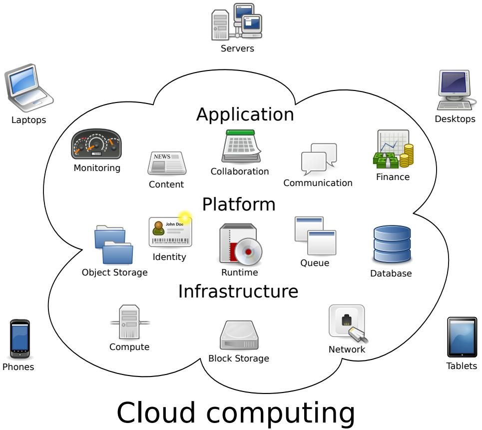
## Algorithms (selection from real world examples)

- Consensus: reliable agreement on a decision (malicious participants?)
- Leader election
- Reliable broadcast (of a message)
- Replication

# Cloud Computing

## Definition

- On-demand availability of computer system resources (data storage and computing)
  - Without direct active management by the user
- Typically relates to distributed resources
  - provided by data centers
  - to many users
  - over the Internet
- Fog/Edge Computing: brings cloud closer to user




## Examples

- Applications: Dropbox, Google Mail, Office 365
- Infrastructure: Amazon, Google, Microsoft, Oracle

# Some Facts: Cloud Computing and Data Centers

- Server workload (VMs or hardware): 350 Million, about 10 instances per server
- Data Center storage capacity: 1,750 Exabyte ($10^{18}$), 720 Exabyte actually stored
  - 180 Exabyte from Big Data
- Global data center IP traffic: 14 Zettabyte ($10^{21}$), 440 Terabyte/s
  - 15% volume communicated to the user: 20 KB/s per human
- Power consumption: US data centers alone 40% UK or 3% of global energy[3]
  - 416 Terawatt = energy bill: 50 Billion £ (12 cents/kWh)
  - Estimate for 2025: 20% worldwide for all DCs?

# Challenges using Distributed Systems

- Programming: concurrency introduces new types of programming mistakes
  - It is difficult to think about all cases of concurrency
  - Must coordinate between programs
  - No global view and debugging
- Resource sharing: system shares resources between all users
- Scalability: system must be able to grow with the requirements
  - numbers of users/data volume/compute demand
  - retain performance level (response time)
  - requires to add hardware, though
- Fault handling: detect, mask, and recover from failures
  - Failures are innevitable and the normal mode of operation
- Heterogenity: system consists of different hardware/software
- Transparency: Users do not care about how/where code/data is
- Security: Availability of services, confidentiality of data

# Outline

# Definition: Parallel Computing

Many calculations **or** the execution of processes are carried out simultaneously[4]

## Characteristics

- Goal is to improve performance for an application
  - Either allowing to solve problems within a deadline or increased accuracy
- Application/System must coordinate the otherwise independent parallel processing
  - There are various programming models for parallel applications
- Different architectures to speed up computation: **may use** distributed systems

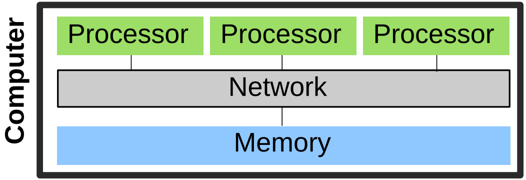
## Levels of parallelism (from hardware perspective)

- Bit-level: process multiple bits concurrently (e.g., in an ALU)
- Instruction-level: process multiple instructions concurrently on a CPU
- Data: run the same computation on **different data**
- Task: run **different** computations concurrently

# Parallel Architectures

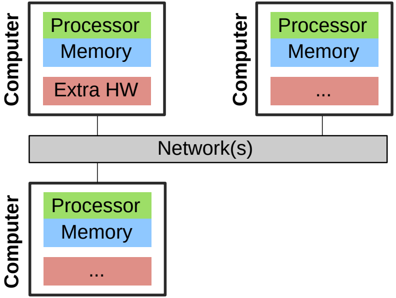
In practice, systems are a mix of two paradigms:

## Shared memory




- Processors can access a joint memory
  - Enables communication/coordination
- Cannot be scaled up to any size
- Very expensive to build one big system

## Distributed memory systems (again!)

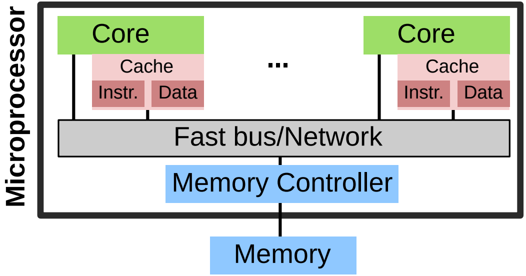



- Processor can only see own memory
- Performance of the network is key

# Parallel Programs

- A **parallel program** runs on parallel hardware
  *In the strict sense*: A parallel application coordinates concurrent processing

## Schema of a multicore processor




## Processor provides all levels of parallelism

- Multiple ALU/other units
- Pipelining of processing stages
- SIMD: Single Instruction - Multiple Data
  - Same operation on multiple data
  - Instruction set: SSE, AVX
- Multiple cores
  - Each with own instruction pointer

# High-Performance Computing

## Definitions

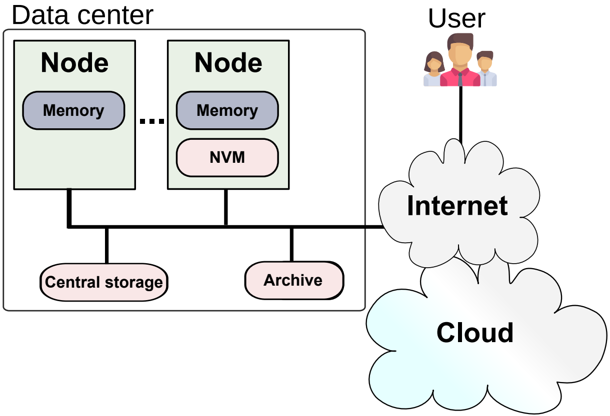
- HPC: Field providing massive compute resources for a computational task
  - Task needs too much memory or time for a normal computer
  - $\Rightarrow$ Enabler of complex challenging simulations, e.g., weather, astronomy
- Supercomputer: aggregates power of many compute devices
  - Nowadays: 100-1,000s of servers that are clustered together

## Example: Summit (Oak Ridge National Laboratories)

- Compute: 4,608 nodes; 2.4 Million cores
  - Peak 200 Petaflop/s ($10^{15}$)
  - 2x IBM POWER9 22C 3.07GHz; 6x NVIDIA Volta V100 GPU
- 10 Petabyte memory (DRAM + HBM + GPU)
- Network: 100G Infiniband; 12.5 GB/s per node; 115 TB/s bisection bandwidth
- Storage: 32 PB capacity; 1 TB/s throughput

The Top500 is a list of the most performant supercomputers

# Supercomputers & Data Centers




Credits: STFC

JASMIN Cluster at RAL / STFC
Used for data analysis of the Centre for Environmental Data Analysis (CEDA)

# HPC in Göttingen

## GWDG: unversity data center and providing innovative technology solutions

- HPC sytems for local scientists, German wide and for DLR
- Integrates research for HPC systems and services

# Challenges

- Programming: imports errors from distributed computed
  - Low-level APIs and code-optimization to achieve performance
  - Performance-optimized code is difficult to maintain
  - Expensive and challenging to debug 1’000 concurrently running processes
  - Utilizing all compute resources efficiently (load balancing)
  - Grand challenges are difficult to test, as nobody knows the true answer
- Scalability: stricter than distributed systems
  - Strong-scaling: same problem, more parallelism shall improve performance
  - Weak-scaling: data scales with processors, retain time-to-solution
- Environment: bleeding edge and varying hardware/software systems
  - Obscure special-purpose hardware (FPGA/ASIC Application-Specific Integrated Circuit)
  - Limited knowledge to administrate, use, and to compare performance

# Outline

# Computational Science

## Definitions

- Multidiciplinary field using advanced computing capabilities to understand and solve complex problems
  - Typically using mathematical models and computer simulation
  - Problems are motivated by industrial or societal challenges
- May utilize single computer, distributed systems, or supercomputers

## Examples utilizing distributed computing

- Finding the higgs boson (CERN)
- Bioinformatics applications, e.g., gene sequencing

## Examples utilizing high-performance computing

- Computing the weather forecast for tomorrow / next week
- Simulating a tokamak fusion reactor

# Scientific Method

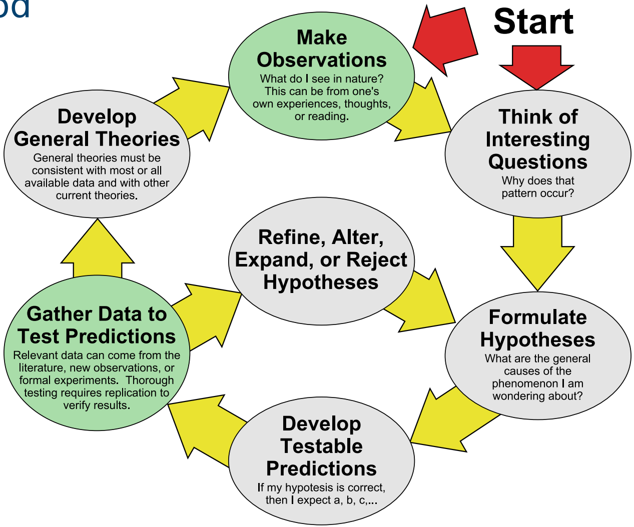
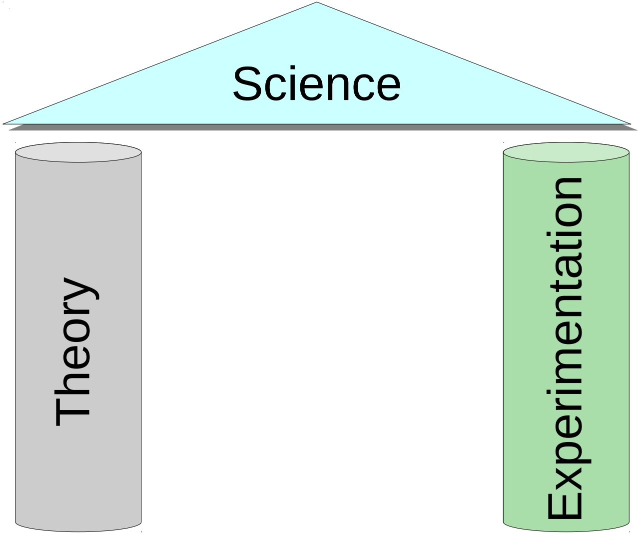
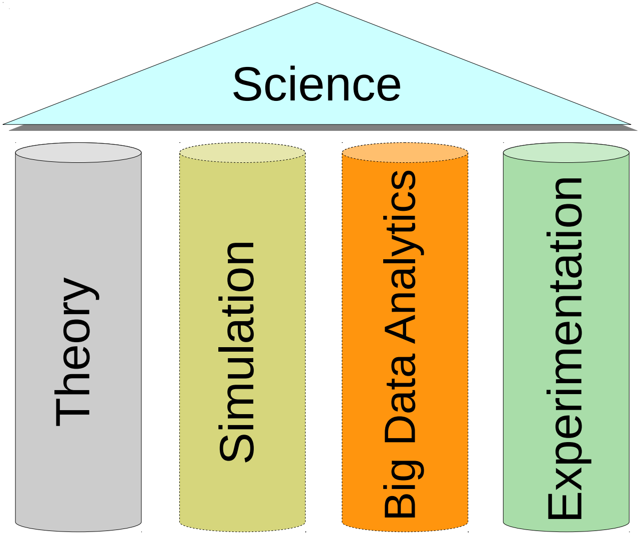



Figure: Based on “The Scientific Method as an Ongoing Process”, ArchonMagnus https://en.wikipedia.org/wiki/Scientific_method

# Pillars of the Scientific Method



# Pillars of Science: **Modern Perspective**



# Relation of the Scientific Method to D/P/S Computing

## Simulation models real systems to gain new insight

- Instrument to make observations, e.g., high-resolution and fast timescale
- Typically used to validate/refine theories, identify new phenomen
- Classical computational science: hard facts (based on models)
- The frontier of science needs massive computing resources on supercomputers
- Data-intensive sciences like climate imposes challenges to data handling, too

## Big Data Analytics extracts insight from data

- Provides a data pool to identify/mine new insight and to validate theories
- In business often approximate insight is enough (a small advantage)
- Distributed and parallel systems are needed to manage and analyze the data
- Gained knowledge is often made available as part of the cloud (for money)...

# Big Data Analytics

## Definition

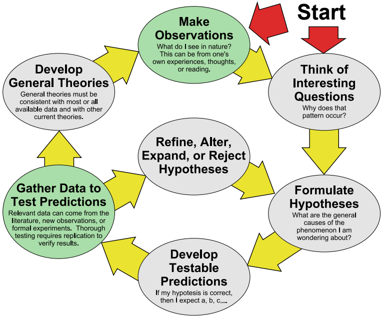
- Extracting insight from data to support decisions
  - Vast amounts of data are available
  - Many different/heterogene data sources that can be correlated
  - Raw data is of low value (fine grained)

## Analytics

- Analyzing data $\Rightarrow$ Insight == value
  - For academia: knowledge
  - For industry: business advantage and money
- Levels of insight – primary abstraction levels of analytics
  - **Exploration**: study data and identify properties of (subsets) of data
  - **Induction/Inference**: infer properties of the full population
- Big data tools allow to construct a theory/model and validate it with data
  - **Statistics** and **machine learning** provide **algorithms and models**
  - Visual methods support data exploration and analysis

# Group Work

- What question(s) you'd like to solve using the scientific method?
- Define the question, hypotheses, how could this be tested? What data is needed?
- Time: 5 min
- Organization: breakout groups - please use your mic or chat



# Example Predictive 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 typical 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 but haven't rated

# Relevance of Big Data and Parallel Computing

- Big Data Analytics is emerging, relevance increases compared to supercomputing
- Nowadays all processors provide parallelism, thus, experts are needed


Figure: Google Search Trends, relative searches

# Outline

# BigData Challenges & Characteristics

Dealing with large data is challenging in Big Data Analytics but also in Computational Science

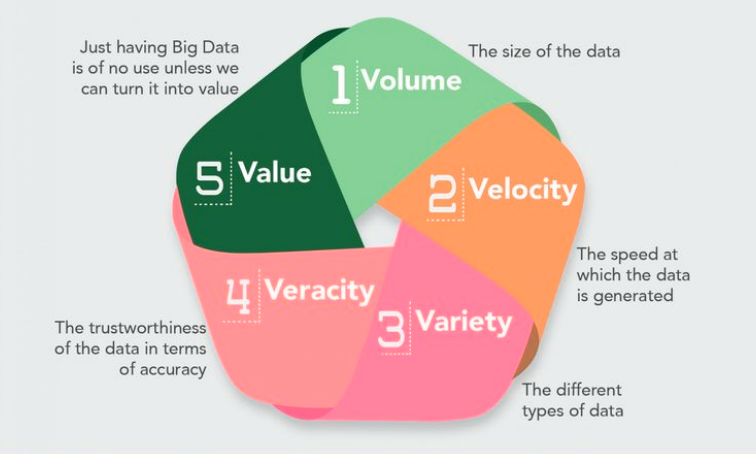



Figure: Source: MarianVesper (Forrester Big Data Webinar. Holger Kisker, Martha Bennet. Big Data: Gold Rush Or Illusion?)

# 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[5]
- YouTube per year 76 PByte (2012[6])

# 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 [7]
- Square Kilometre Array 700 TB/s (in 2018) [8]
- 50k Google searches per s [9]
- Facebook 30 Billion content pieces shared per month [10]

# 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
  - Examples: 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
  - Examples: 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
- Examples
  - 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 of the weather, a bill
- The output of a large scale climate simulation that cost 10k to run
  - It still needs to be analyzed to come to conclusions!

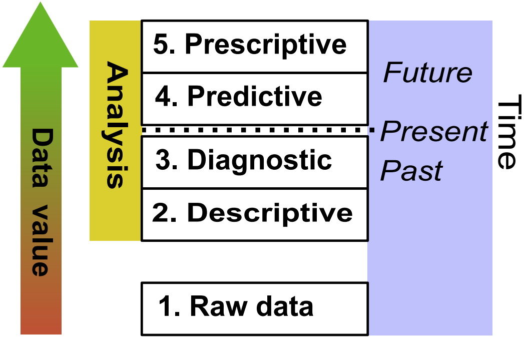
## Analytics and theory about the data increases the value

- Analytics transform big data into smart (valuable) data!

# Abstraction Levels of Analytics and the Value of Data

5. Prescriptive analytics
   - "What should we do and why?"
4. Predictive analytics
   - "What will happen?"
3. Diagnostic analytics
   - "What went wrong?"
   - "Why did this happen"
2. Descriptive analytics[a]
   - "What happened?"
1. Raw (observed) data

[a] Decriptive and diagnostic analysis are like forensics




## Relation to Computational Science

- These analysis steps are still done just by running computational experiments

# Analytics Abstraction Level

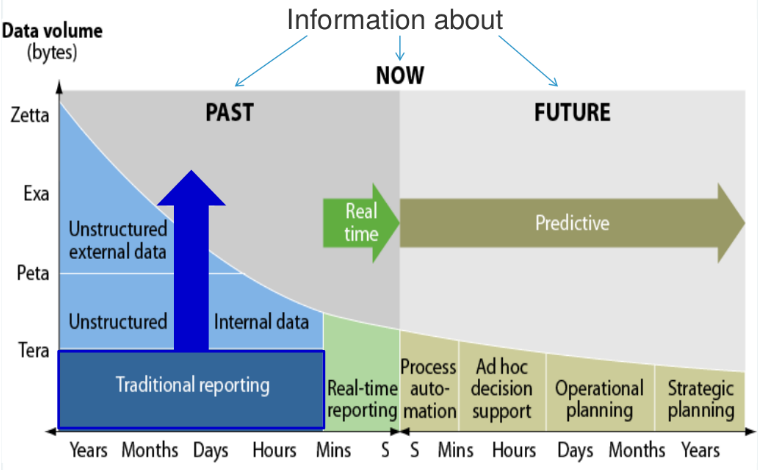
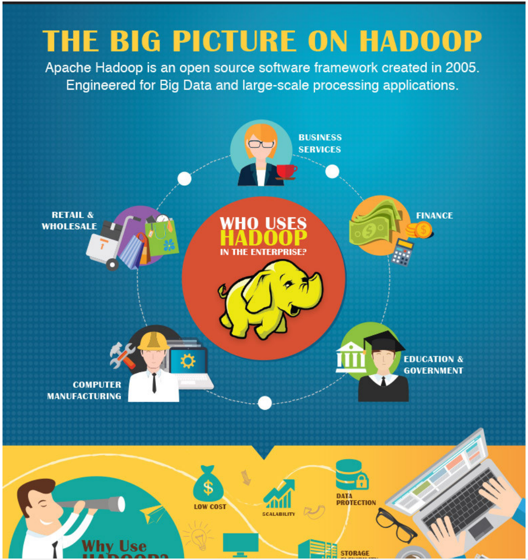
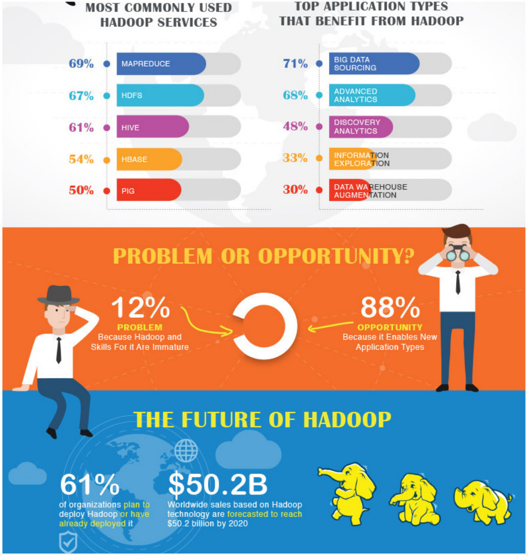



Figure: Source: Forrester report. Understanding The Business Intelligence Growth Opportunity. 20-08-2011

# Outline

# Advertisement for a Big Data Platform



# 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
- Monetization: Extract money from gamers [27]

## Science

- Epidemiology research: Google searches indicate Flu spread
- Personalized Healthcare: Recommend good treatment
- Physics: Finding the Higgs-Boson, analyze telescope data

# 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 number of 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”

# Learning Behavior

## Games

- DeepMind playing atari games [29]
- AlphaGo wins vs. humans in playing Go [26]
- AI beating world's best gamer in Dota 2 [28]

## Motion

- Learning hand motion by human training [30]
- Robots learning to pick up items [31]

# Systems: Fault Prediction and Anomaly Detection

## Smart buildings [24]

- Predicting faults of heating and ventilation of an hospital
- Predicted 76 of 124 real faults and 41 of 44 exceptional temperatures
- May consider weather to control systems automatically

## Google DeepMind AI [25]

- Controlling 120 variables in the data center (fans, ...)
- Saves 15% energy of the overall bill

# Automatize Classification

## Analysis of multimedia

- Voice, face, biometric recognition
- Speech recognition
- Counting (animal) species on pictures / videos
- Finding patterns on satellite images (e.g., damn, thunderstorms)
- Anomalies in behavior (depressed people)
- Anomalies in structures (operational condition)

# Outline

# Learning Objectives of the Lecture

- Assign big data challenges to a given use-case
- Outline use-case examples for high-performance data analytics
- Estimate performance and runtime for a given workload and system
- Create a suitable hardware configuration to execute a given workload within a deadline
- Construct suitable data models for a given use-case and discuss their pro/cons
- Discuss the rationales behind the design decisions for the tools
- Describe the concept of visual analytics and its potential in scientific workflows
- Compare the features and architectures of NoSQL solutions to the abstract concept of a parallel file system
- Appraise the requirements for designing system architectures for systems storing and processing data
- Apply distributed algorithms and data structures to a given problem instance and illustrate their processing steps in pseudocode
- Explain the importance of hardware characteristics when executing a given workload

# Organization of the Module: Components

- Lecture (2h / week)
  - Delivers concepts and gives an overview
  - 1 invited talk (and this overview presentation)
- Practical for discussion of the exercise (2h / week)
  - Follows the schedule of the lecture, **optional**
  - Part 1: Students present their solution/questions to exercise tasks
  - Part 2: We discuss the new excercise such that everyone understands the questions
- Exercise (prescribed 4h / week)
  - Self-study to practice lecture content (feel free to team up!)
  - Each task comes with an estimated time for you to spend on it
  - Contains introductory and harder tasks
  - Recommend to store your work in a Git Repository – a portfolio of the course
- Group work: Some time of practical may be used for group work

# Role of Exercises and Group Work

## Assessment

- Module: Assessment is 100% exam, however,
- Exercises and group work is formative assessment that prepares for the exam
- Feedback of the lecturer during practicals for your exercises
- Some quizes are provided during lecture/exercises and for your self-study

## Group work

- Discuss/Critice exercises of peers (groups of 2-4)
- Brainstorm/Design/Solve small tasks (groups of 2-4)
- The outcome should be stored in the Git portfolio

# Proposed Learning Strategy/How to Achieve Good Marks

- Understand learning outcomes (provided in each slide deck)
- Participate in exercises
  - To understand the topic, types of questions, and how to solve issues
  - To get feedback from the lecturer (e.g., if you present) and from peers
- Schedule time for the exercises, best to team up in learning groups
  - Try to do the 3h/week!
  - Always do the easy tasks, if you are busy you may miss some harder tasks
  - Partial solutions are better than no attempt
- (Do further reading of topics you are interested in)
- Team up again to prepare for the exam
- Ask questions to colleagues and to us
- We will support your learning journey but **YOU** are responsible for it

# Communication

- Webpage: https://hps.vi4io.org/teaching/autumn_term_2021/hpda
- Webpage provides
  - Slides for lectures/practical
  - Exercise sheets
  - Reading lists for topics
- StudIP for communication
  - We use it for announcements
  - Please use it for any purpose around the topic!
  - To solve exercises, to share an interesting link, to ask a question
  - To find peers to work with

# Summary

- HPDA: process of quickly examing large data sets
- Simulation and Big data analytics is a pillar of science
  - Supports building of hypothesis and experimentation
- Challenges: 5 Vs – Volume, velocity, variety, veracity, value

## Characteristics and Differences of DC/PC

| | Distibuted computing | Parallel computing |
|---|---|---|
| Motivation | Decentrality/low costs | Performance/feasability |
| Enables | business/cloud/big data analytics | interactivity/computational science |
| Communication | message passing | may use shared resources |
| Fault-tolerance | tolerate errors | needs reliable hardware |
| Application | Weakly-coupled<br>Multiple programs/vendors | Tightly-coupled<br>Single application/vendor |

# 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

22 https://en.wikipedia.org/wiki/Scientific_method

23 https://en.wikipedia.org/wiki/Exploratory_data_analysis

24 https://www.newscientist.com/article/2118499-smart-buildings-predict-when-critical-systems-are-about-to-fail/

25 https://www.theverge.com/2016/7/21/12246258/google-deepmind-ai-data-center-cooling

26 https://deepmind.com/research/alphago/

27 https://www.ibm.com/developerworks/library/ba-big-data-gaming/index.html

28 http://money.cnn.com/2017/08/12/technology/future/elon-musk-ai-dota-2/index.html

29 http://www.wired.co.uk/article/google-deepmind-atari

30 https://arxiv.org/abs/1603.06348

31 https://spectrum.ieee.org/automaton/robotics/artificial-intelligence/google-large-scale-robotic-grasping-project