Department of Computer Science

University of Reading

CS3DP: Introduction to Distributed and Parallel Computing

Julian Kunkel and Christopher Maynard

2020-08-03

Learning Outcomes

After the session, a participant should be able to:

- Recite system characteristics for distributed/parallel/computational science
- Describe use-cases and challenges in the domain of D/P/S computing
- Sketch generic D/P system architectures
- Describe how the scientific method relies on D/P/S computing
- Name big data challenges and the typical workflow

Outline

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

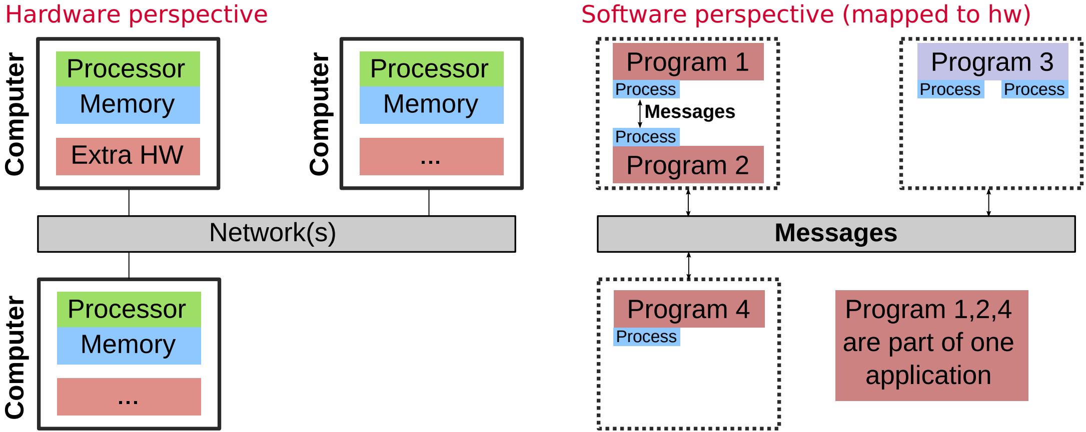
- Distributed memory: components have their own (private) memory
- Concurrency of components: different components compute at the same time
- Lack of a global clock: clocks may diverge
- Independent failure of components, e.g., due to power outage

[1] https://en.wikipedia.org/wiki/Distributed_computing
[2] In this context, means a component from a software architecture.

Example Distributed System and Distributed Program

University of Reading

- 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)

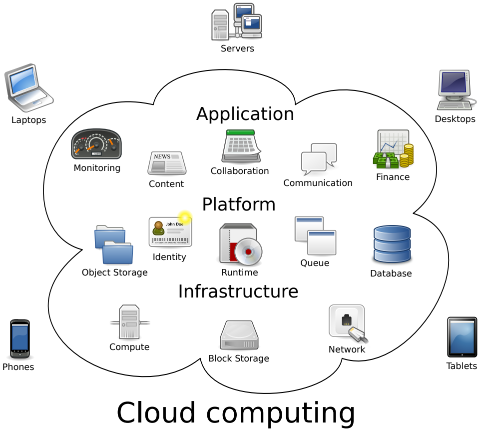
- Consensus: reliable agreement on a decision (malicious participants?)
- Leader election
- Reliable broadcast (of a message)
- Replication

Cloud Computing

University of Reading

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

Image source: Frank, B. Wilson - CloudNINE, https://en.wikipedia.org/wiki/Cloud_computing

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?

[3] For 2017: https://www.forbes.com/sites/forbestechcouncil/2017/12/15/why-energy-is-a-big-and-rapidly-growing-problem-for-data-centers/
Estimate for 2019: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.pdf

Challenges

- 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

University of Reading

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)

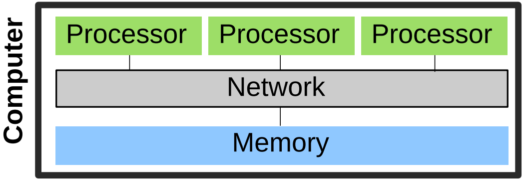
- Bit-level: process multiple bits concurrently (e.g., in an ALU)
- Instruction-level: process multiple instructions concurrently on a CPU
- Data: run the same computation on **different data**
- Task: run **different** computations concurrently

[4] https://en.wikipedia.org/wiki/Parallel_computing

Parallel Architectures

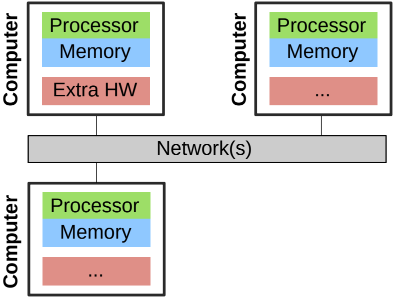
In practice, systems are a mix of two paradigms:

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!)

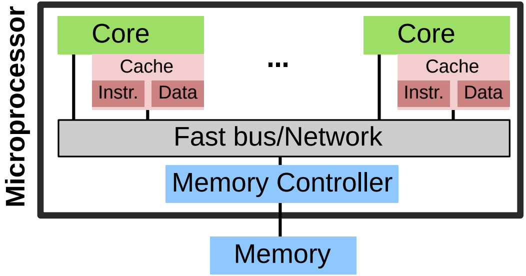


- Processor can only see own memory
- Performance of the network is key

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

https://en.wikipedia.org/wiki/Microarchitecture

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
  - ⇒ 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

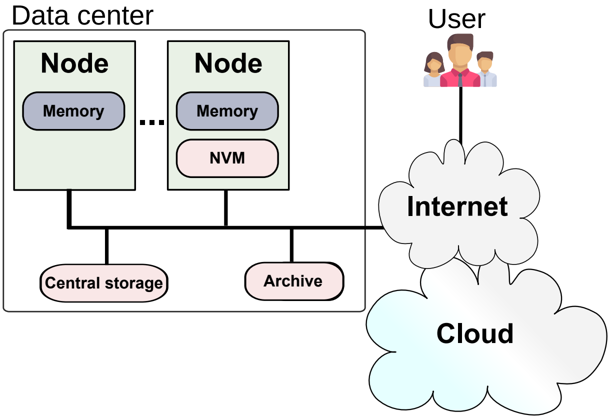
Example: Summit (Oak Ridge National Laboratories)

- 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

University of Reading



Credits: STFC

JASMIN Cluster at RAL / STFC

Used for data analysis of the Centre for Environmental Data Analysis (CEDA)

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

University of Reading

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

---

https://en.wikipedia.org/wiki/Computational_science

Scientific Method

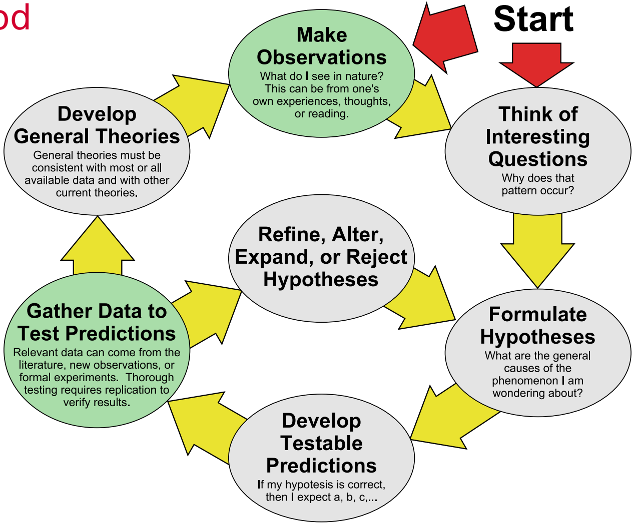
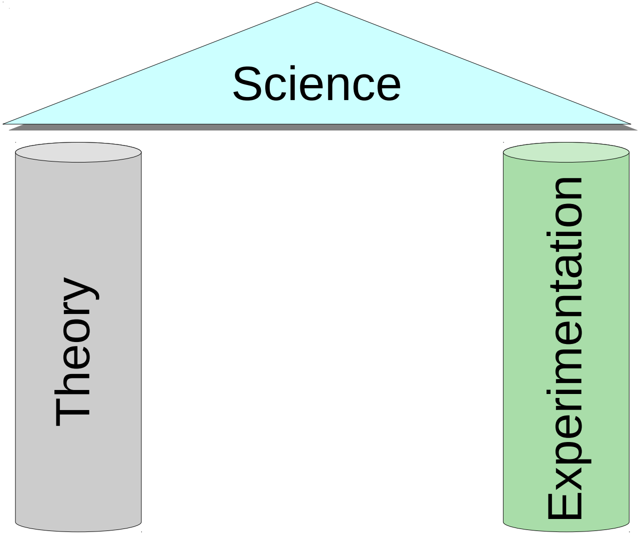
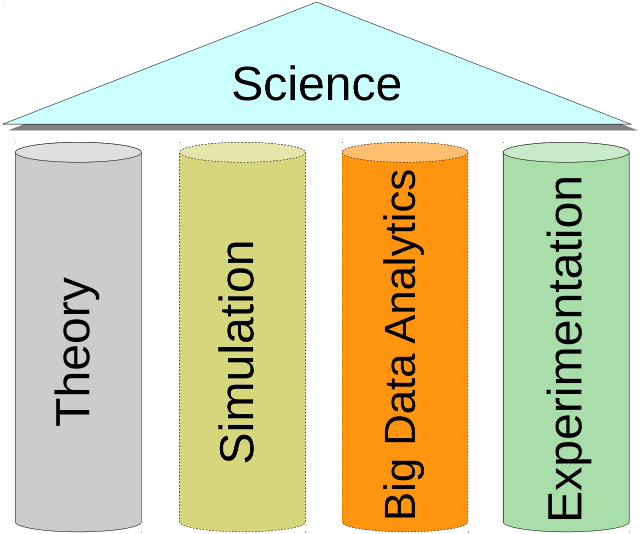


Figure: Based on "The Scientific Method as an Ongoing Process", ArchonMagnus
https://en.wikipedia.org/wiki/Scientific_method

Pillars of the Scientific Method



Pillars of Science: **Modern Perspective**



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

University of Reading

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

University of Reading

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

Relevance of Big Data and Parallel Computing

University of Reading

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

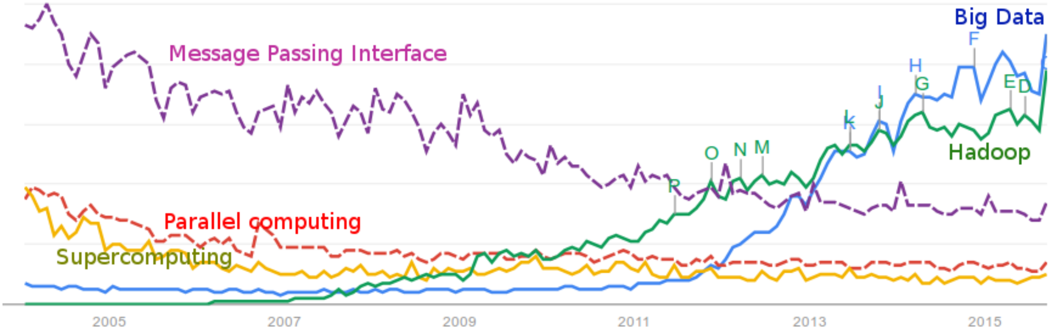


Figure: Google Search Trends, relative searches

Outline

BigData Challenges & Characteristics

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

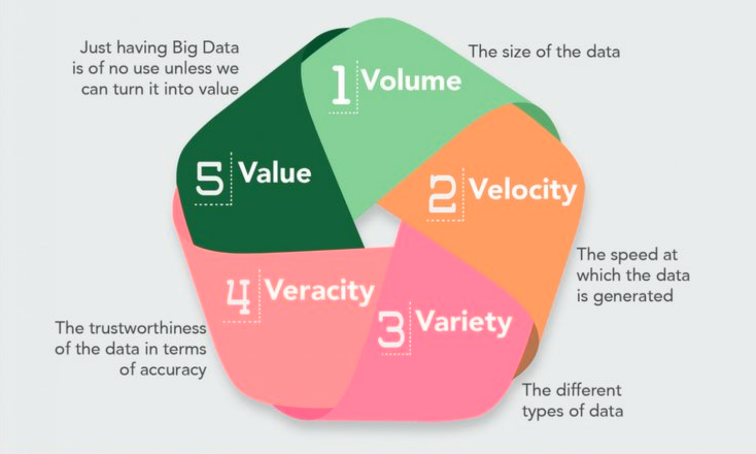


Figure: Source: MarianVesper (Forrester Big Data Webinar. Holger Kisker, Martha Bennet. Big Data: Gold Rush Or Illusion?)

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])

---

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

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

Velocity: Data Volume per Time

University of Reading

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]

---

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

University of Reading

What is Big Data

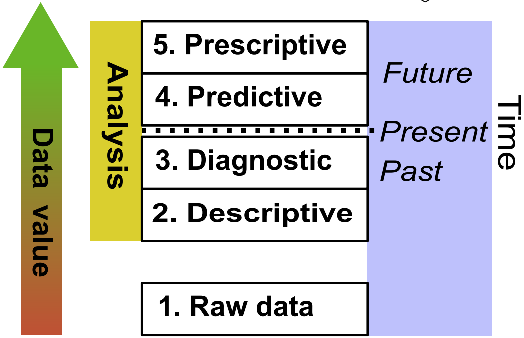
- Raw data of Big Data is of low value
  - For example, single observations of the weather, a bill
- The output of a large scale climate simulation that cost 10k to run
  - It still needs to be analyzed to come to conclusions!

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[11]
   - "What happened?"
1. Raw (observed) data



Relation to Computational Science

- These analysis steps are still done just by running computational experiments
- Also the output of the simulation must be analyzed

[11] Decriptive and diagnostic analysis are like forensics

Analytics Abstraction Level

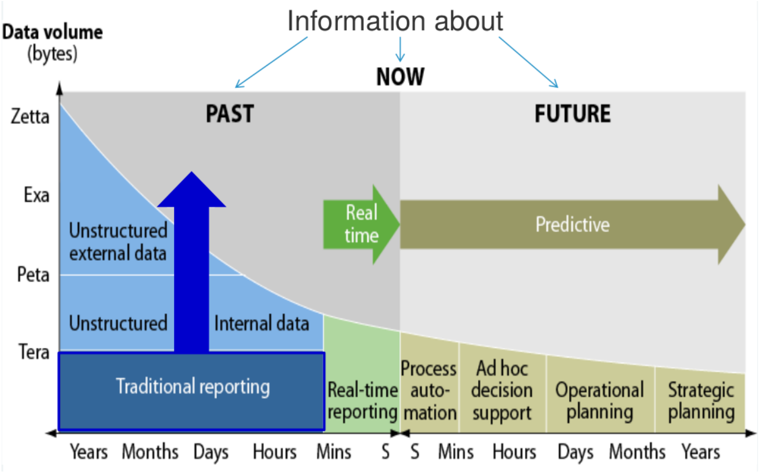


Figure: Source: Forrester report. Understanding The Business Intelligence Growth Opportunity. 20-08-2011

Data Analysis Workflow

University of Reading

The traditional approach proceeds in phases:

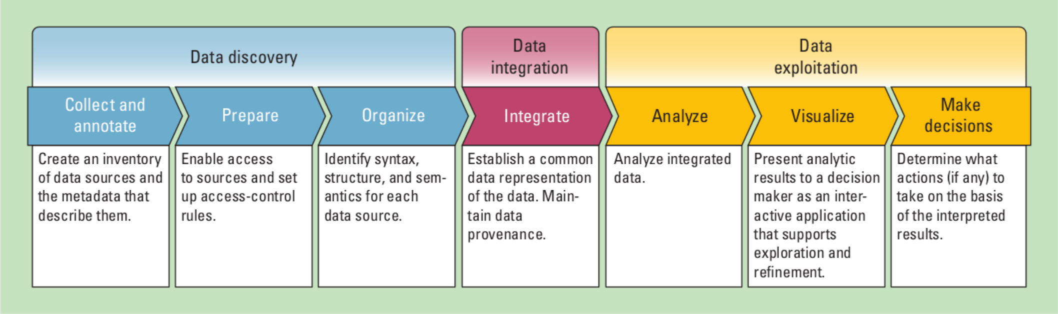


Figure: Source: Gilbert Miller, Peter Mork From Data to Decisions: A Value Chain for Big Data.

- Analysis tools: machine learning, statistics, interactive visualization
- Limitation: Interactivity by browsing through prepared results
- Indirect feedback between visualization and analysis

From Big Data to the Data Lake

University of Reading

- 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 time all data: raw sources and 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: shared infrastructure supports various patterns
  - Batch, interactive, online, search

---

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

Outline

Organization of the Module: Components

University of Reading

- Lecture (1h / week)
  - Delivers concepts and gives an overview
  - 4 lectures about distributed computing
  - 4 lectures about parallel computing
  - 1 invited talk (and this overview presentation)
- Practical (1h / week) – follows the schedule after the tutorial
  - 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 3h / 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
  - Store your work in a Git Repository – the portfolio of the course
- Group work: Some time of practical/tutorial 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/tutorials
- Some quizes are provided during lecture 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 the quizzes
- 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

- Blackboard provides
  - Slides for lectures/practical, exercise sheets
  - Most important announcements
  - Reading lists for topics
- Teams channel for communication/tutorial
  - We use it for any announcement
  - 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

- 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 Multiple programs/vendors	Tightly-coupled Single application/vendor