Cloud Computing & Big Data

PARALLEL & SCALABLE MACHINE LEARNING & DEEP LEARNING

Prof. Dr. – Ing. Morris Riedel
Adjunct Associated Professor
School of Engineering and Natural Sciences, University of Iceland
Research Group Leader, Juelich Supercomputing Centre, Germany

LECTURE 1

Cloud Computing & Big Data

September 6th, 2018
Room Stapi 108

UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING,
MECHANICAL ENGINEERING AND COMPUTER SCIENCE

JÜLICH
Forschungszentrum

HELMHOLTZ
RESEARCH FOR GRAND CHALLENGES

DEEP
Projects

Outline of the Course

1. Cloud Computing & Big Data
2. Machine Learning Models in Clouds
3. Apache Spark for Cloud Applications
4. Virtualization & Data Center Design
5. Map-Reduce Computing Paradigm
6. Deep Learning driven by Big Data
7. Deep Learning Applications in Clouds
8. Infrastructure-As-A-Service (IAAS)
9. Platform-As-A-Service (PAAS)
10. Software-As-A-Service (SAAS)
11. Data Analytics & Cloud Data Mining
12. Docker & Container Management
13. OpenStack Cloud Operating System
14. Online Social Networking & Graphs
15. Data Streaming Tools & Applications
16. Epilogue

+ additional practical lectures for our hands-on exercises in context

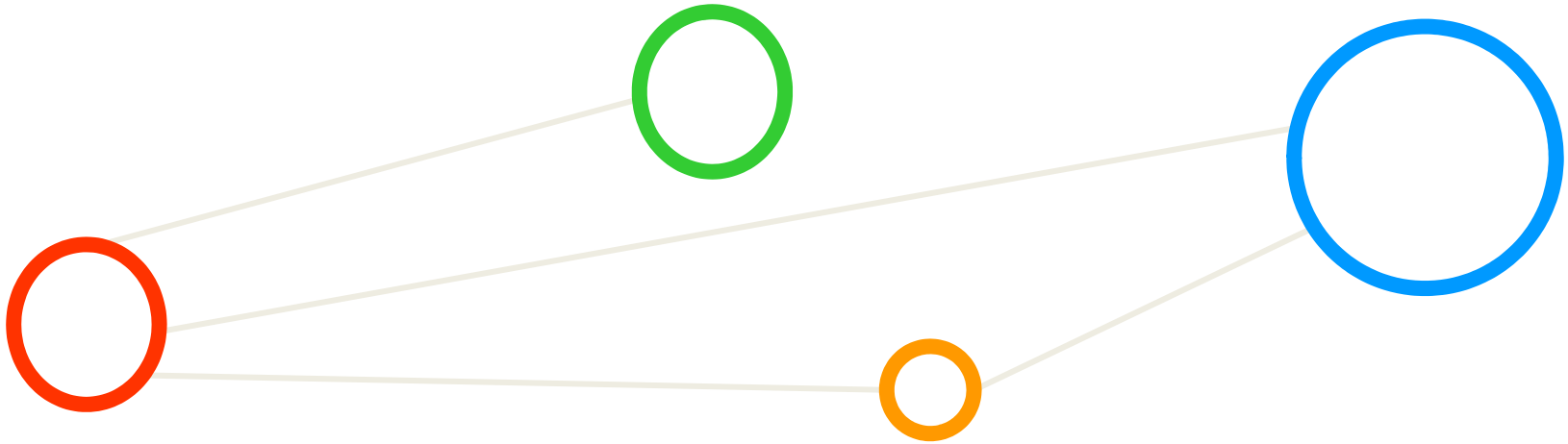
- Practical Topics
- Theoretical / Conceptual Topics

Outline

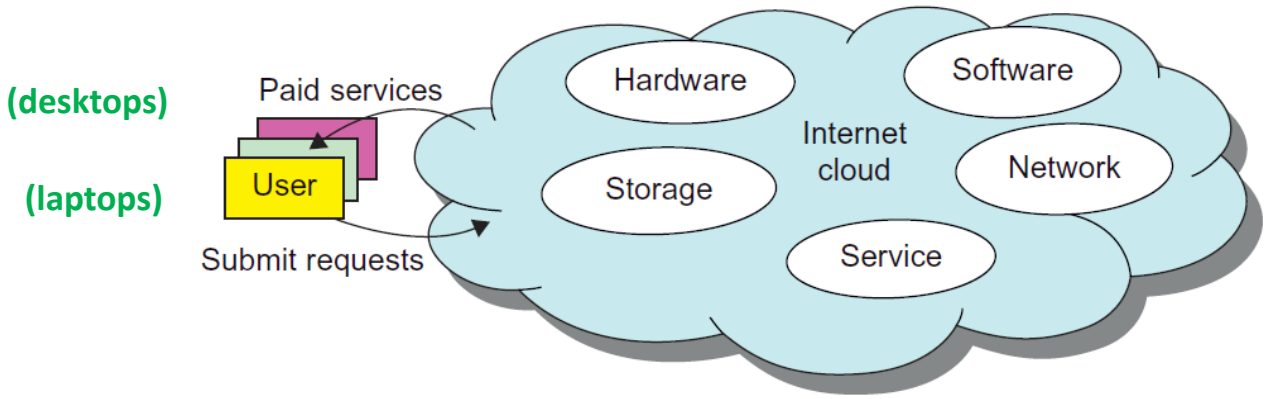
- Foundations of Cloud Computing
  - Parallel & Distributed Computing Evolution
  - Internet Cloud Systems Examples
  - Technology Advances & Parallel Computing
  - SIMD Python & Vectorization Example
  - Multi-core CPUs & Many-core GPUs
- Scalability driven by Big Data
  - What is Big Data & Challenges
  - Evolutions in Memory & Disk Storage
  - Examples of Cloud Storages & Scalability
  - Wide Area Networks & Programming Models
  - Big Data Analytics & Machine/Deep Learning

- **Promises from previous lecture(s):**
- ***Lecture 0 – Prologue:*** **Lecture 1 provides technology foundations for clouds and offers insights into terminologies**
- **Lecture 0 – Prologue: Lecture 1 outlines further relationships of intertwined topics big data / clouds / machine learning**

Introduction to Cloud Computing



What is Cloud Computing from 10.000 ft?



[1] Distributed & Cloud Computing Book

(service-oriented platform with virtualized resources)

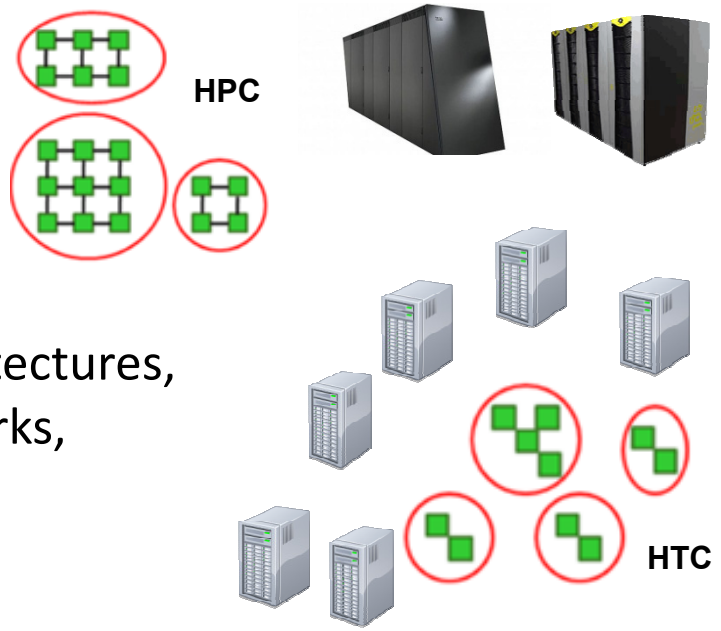
- Data Centres
  - Provide virtualized resources to form an Internet cloud
  - Provisioned with hardware, software, storage, network, and services
  - End user pay to run their applications or services after submitting requests

- **Cloud computing moves desktop computing and laptop computing via the Internet to a service-oriented platform using remote large server clusters and massive storages to data centres**
- **Virtualization has enabled the cost-effectiveness and simplicity of cloud computing solutions**

➢ **Lecture 4 provides more details about the underlying virtualization technology and its cloud impact**

Evolutions towards Cloud Computing

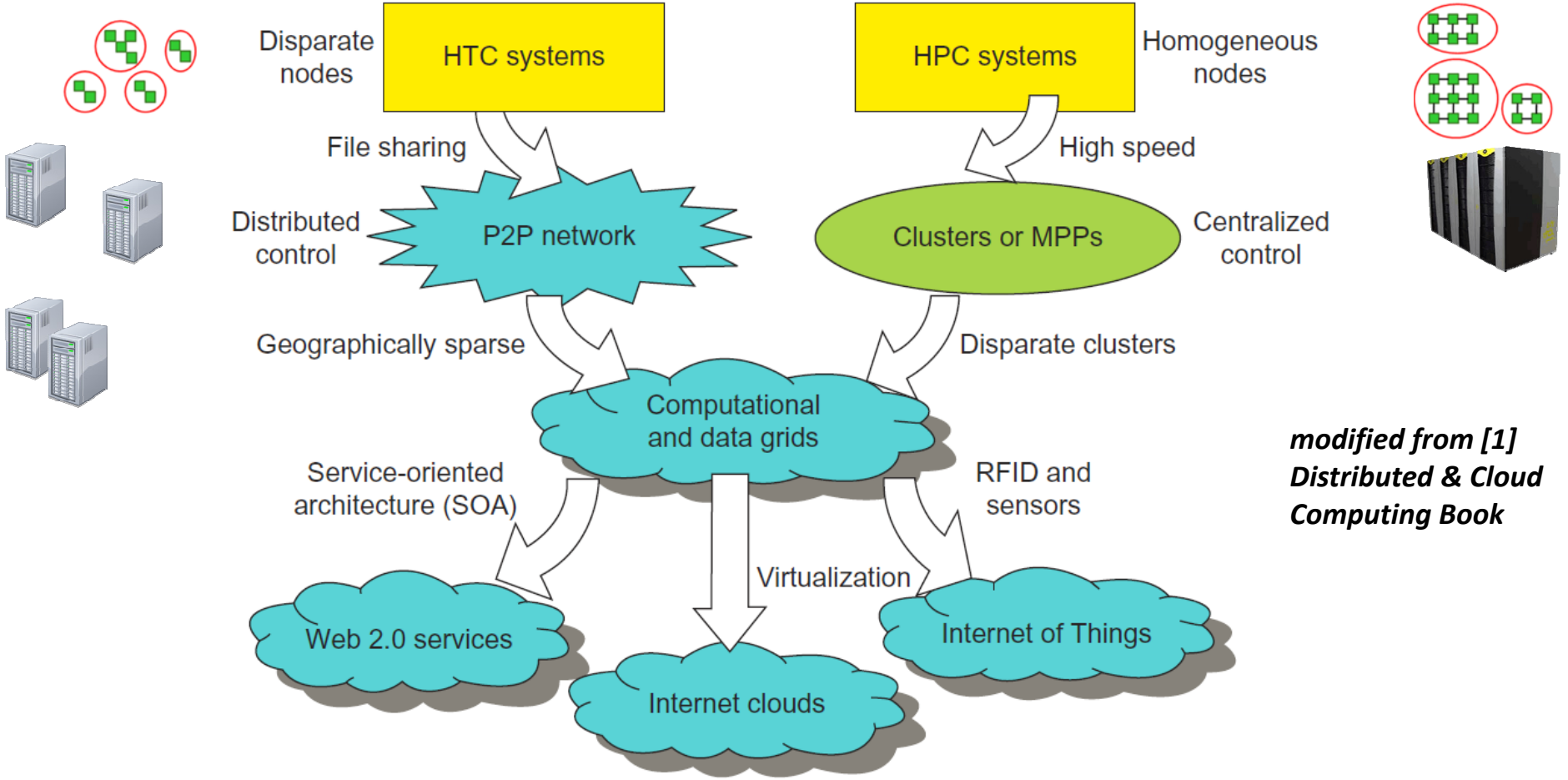
- Many evolutions in parallel and distributed computing
  - Over the past 30 years
  - Driven by applications with variable workloads and large big data sets
- Established computing paradigms
  - High Performance Computing (HPC)
  - High Throughput computing (HTC)
- Increase use of parallel computers
  - Computer clusters, service oriented architectures, computational Grids, peer-to-peer networks, Internet Clouds, Internet of Things, …



[1] Distributed & Cloud Computing Book

- **Instead of using a centralized computer to solve computational problems, a parallel and distributed computing system uses multiple computers to solve large-scale problems over the Internet**

Evolution over time



modified from [1] Distributed & Cloud Computing Book

➢ **Lecture 4 provides more details about the computing paradigms HPC and HTC and their systems**

Internet Cloud Systems – Examples from Every Day Life

- Selected Cloud Systems (aka 'Clouds') known today
  - Google AppEngine **(massive computing & storage & applications)**
  - Amazon Web Service **(massive computing & storage)**
  - Facebook **(online social networking & advertisement)**
  - SalesForce.com **(customer relationship management)**
  - Rackspace **(managed cloud provider & hosting)**
  - IBM Bluemix **(cloud platform)**
  - Enomaly **(elastic computing cloud)**


- Cloud systems play an increasingly important role in upgrading traditional Web services and Internet applications

[1] Distributed & Cloud Computing Book

- **Large-scale Internet applications have significantly enhanced the quality of life in society today**

➢ **Lecture 8 & 9 & 10 offer more insights into concrete cloud systems and their architectures today**

Terminologies: Centralized Computing & Parallel Computing

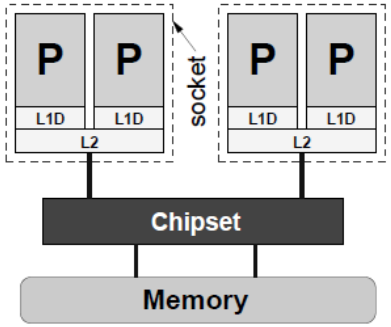
- Centralized Computing
  - All computer resources are centralized in one physical system
  - All resources (e.g. processors, memory, storage, etc.) fully shared and tightly coupled within one integrated operating system
  - Many data centers & supercomputers are centralized systems

- Parallel Computing
  - All processors can be tightly coupled using shared memory
  - All processors can be loosely coupled using distributed memory
  - Interprocessor communication via shared memory or message passing
  - Computer systems capable of parallel computing is a parallel computer
  - Programs running in a parallel computer are called parallel programs
  - Process of writing parallel programs is referred to as parallel programming

➢ **Complementary HPC course offers shared memory & message passing parallel programming**

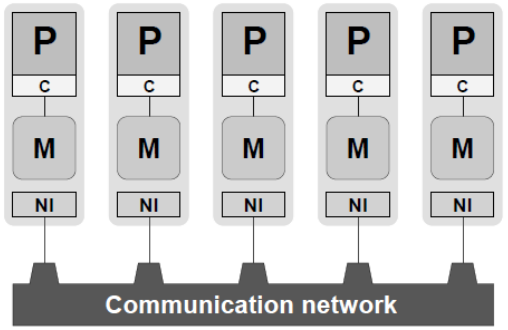
Parallel Computing: Shared Memory vs. Distributed Memory

- Shared Memory



Massively Parallel Computers

- Distributed Memory



[2] Introduction to High Performance Computing for Scientists and Engineers

- **A shared-memory parallel computer is a system in which a number of CPUs work on a common, shared physical address space**
- **A distributed-memory parallel computer establishes a 'system view' where no process can access another process' memory directly**

➢ **Complementary HPC course offers more details on shared & distributed memory architectures**

Python Programming Language

- Selected Benefits
  - Simple & flexible programming language
  - Is an interpreted powerful programming language
  - Has Efficient high-level data structures
  - Provides a simple but effective approach to object-oriented programming
  - Powerful libraries like ‘math’ library for inputs of functions with real numbers
  - Python script support is offered in almost every cloud computing environment as a programming language today

[16] Webpage Python

```
# usual start script hello world

print('Hello World!')
print("2 + 2 =", 2 + 2)
print("3 * 4 is", 3 * 4)
print('Goodbye!')
```

```
Hello World!
2 + 2 = 4
3 * 4 is 12
Goodbye!
```

- **Python is an ideal language for fast scripting and rapid application development that in turn makes it interesting for the machine learning modeling process and easy access to cloud resources**

➢ **Our course assignments take advantage of Python, but nobody needs to be a full Python Expert**

Powerful NumPy Python Library

- Selected features
  - Useful linear algebra and random number capabilities
  - Supports powerful N-dimensional array objects
  - Interesting broadcasting functions
  - Tools for integrating C/C++ and Fortran code
  - Particularly nicely supports work on vectors & matrices that are useful when working with machine learning & big data

[17] NumPy Library Web Page

```
import numpy as np

vector = np.random.randn(1,3)
print(vector)
```

```
[[1.01803478 0.21455826 0.5541203 ]]
```

(small number of code lines to create & print a 1x3 row vector with 3 random values)

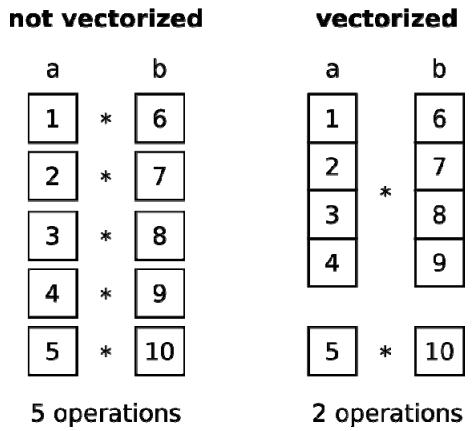
- **NumPy is one of the most important Python libraries used in cloud computing and big data analysis**
- **NumPy is used as an efficient multi-dimensional container of generic data, vectors, matrices, etc.**
- **Instead of math library used for functions with real numbers we use NumPy for vectors & matrices**

➢ **Our course assignments take advantage of NumPy for various aspectcs like working with vectors**

Simple Python/NumPy & Vectorization

- ‘Small-scale example‘ of the power of ‘parallelization‘
  - Enables element-wise computations at the same time (aka in parallel)
  - ‘small-scale‘ since we are still within one computer – but perform operations in parallel on different data
  - Supported on thehardware-level (i.e. register, etc.) & less lines of code
  - Impact matters much for ‘big data‘ computing (e.g. Machine learning & deep learning : matrix/vector multiplications)

[18] Blog



- **Vectorization in Python uses optimized & pre-compiled C code to perform operations over data sequences of the same data type (i.e. numpy array/vector instead of Python tuples or lists)**
- **Vectorized functions are multiple times faster than using operations in explicit for-loop statements**
- **Avoid explicit for-loops via vectorized Single Instruction Multiple Data (SIMD) functions in NumPy**

➢ **Lecture 2 shows more examples of using vectorization for numpy vectors in machine learning**

Simple Python/NumPy & Vectorization Example

```
import numpy as np
import time

vector_a = np.random.rand(100000)
vector_b = np.random.rand(100000)

t_start = time.time()
vector_c = np.dot(vector_a, vector_b) # vectorized function
t_end = time.time()

print(vector_c)
print("Computing time using vectorization: " + str(1000*(t_end-t_start)) + " ms")
```

```
25013.11080224216
Computing time using vectorization: 15.622138977050781 ms
```

```
t_start = time.time()
vector_c = 0
for i in range (100000):
  vector_c += vector_a[i] * vector_b[i]
t_end = time.time()

print(vector_c)
print("Computing time using explicit for-loops: " + str(1000*(t_end-t_start)) + " ms")
```

```
25013.110802242478
Computing time using explicit for-loops: 55.59992790222168 ms
```

Terminologies: Distributed Computing & Cloud Computing

- Distributed Computing ('parallel computing across computers')
  - Field of computer science/engineering that studies distributed systems
  - Distributed systems consist of multiple autonomous computers (each having its own memory and stroage; communication via network)
  - Information exchange in a distributed system is done via message passing
  - Program that runs in a distributed system is a distributed program
  - Process of writing distributed programs is known as distributed programming

towards Internet of Things (IoT)

[1] Distributed & Cloud Computing Book

- **An Internet Cloud of resources can be either a centralized or a distributed computing system**
- **Clouds apply parallel or distributed computing or a combination of both**
- **Clouds are using physical or virtualized resources over large centralized/distributed data centers**

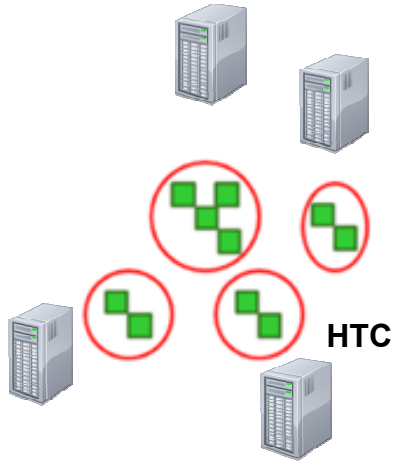
Distributed Computing – BOINC Example

- Tool for distributed computing

  [3] BOINC middleware tool

  - BOINC is one middleware tool in distributed computing, e.g. SETI@Home
  - Provides functionalities to work with compute and big data sets
  - Different geographically distributed nodes in a distributed system consist of a large number of heterogenous nodes used for computing
  - Architecture consists of a large number of rather ordinary desktop computers

- Unique selling proposition
  - BOINC implements concept of using unused resources
  - Use ‘free’ unused computing power or storage during the night or during longer inactive usage periods

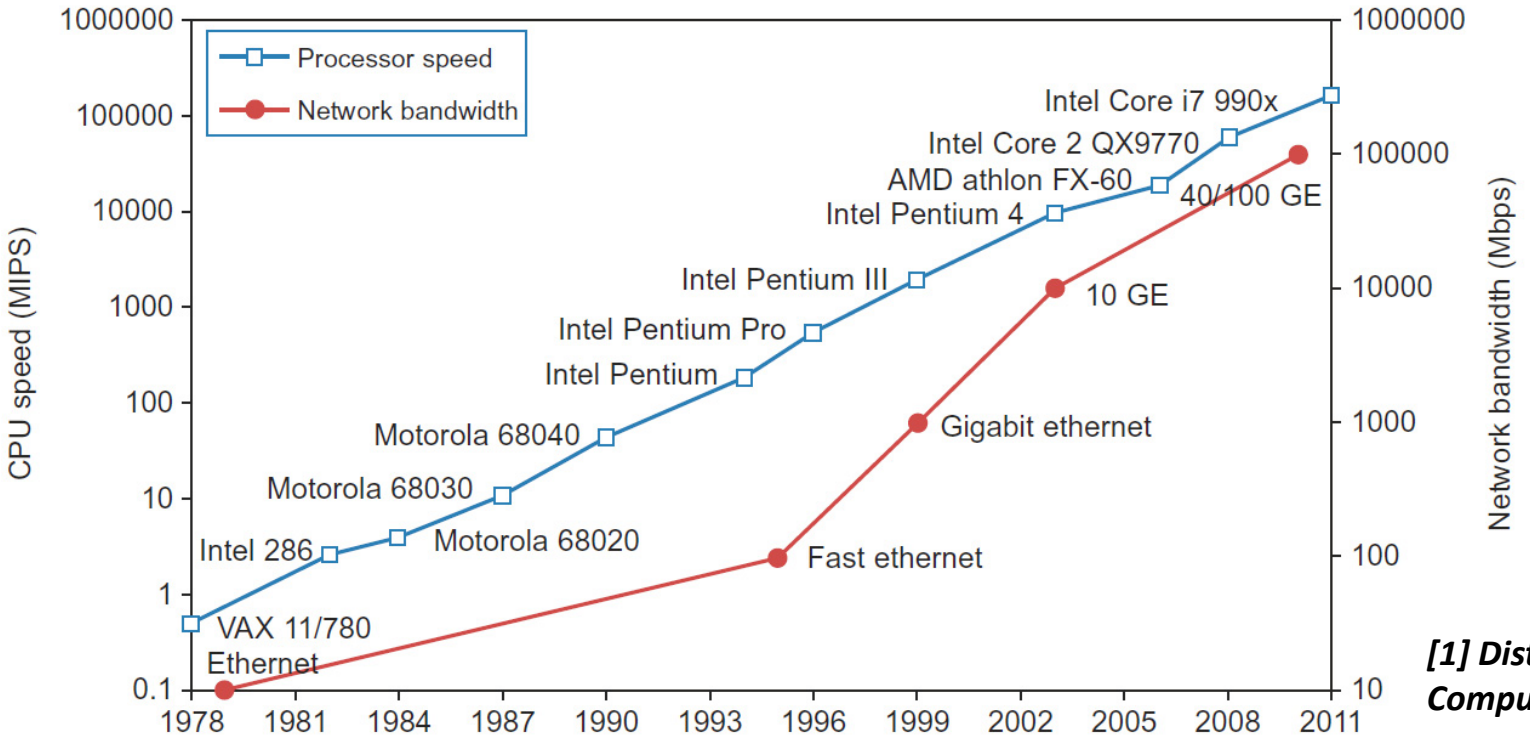


[1] Distributed & Cloud Computing Book

- **Berkely Open Infrastructure for Network Computing (BOINC) is distributed computing framework**
- **BOINC implements CPU scavenging that means using unused resources in distributed computing**

Cloud Enabling Technology Advances – CPU / Network

- Improvement in processor and network technologies

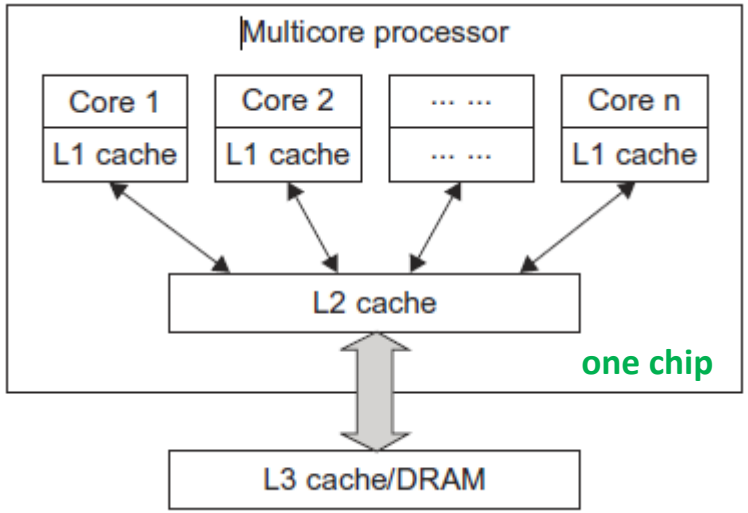


[1] Distributed & Cloud Computing Book

- **Millions of instructions per second (MIPS) is a measure for the processor / CPU speed**
- **Megabits per second (Mbps) and Gigabits per second (Gbps) are measures for network bandwidth**

Multi-core CPU Processors

- Significant advances in CPU (or microprocessor chips)
  - Multi-core architecture with dual, quad, six, or n processing cores
  - Processing cores are all on one chip
- Multi-core CPU chip architecture
  - Hierarchy of caches (on/off chip)
  - L1 cache is private to each core; on-chip
  - L2 cache is shared; on-chip
  - L3 cache or Dynamic random access memory (DRAM); off-chip



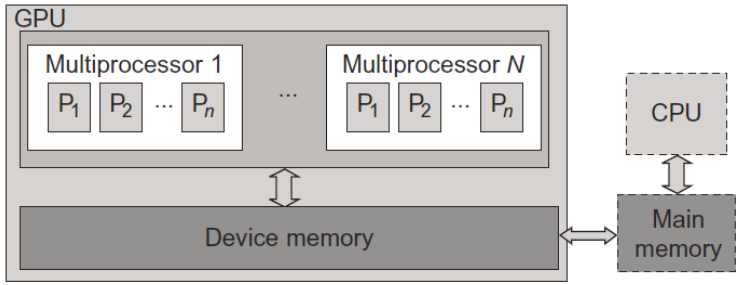
[1] Distributed & Cloud Computing Book

- **Clock-rate for single processors increased from 10 MHz (Intel 286) to 4 GHz (Pentium 4) in 30 years**
- **Clock rate increase with higher 5 GHz unfortunately reached a limit due to power limitations / heat**
- **Multi-core CPU chips have quad, six, or n processing cores on one chip and use cache hierarchies**

Many-core GPUs

- **Graphics Processing Unit (GPU) is great for data parallelism and task parallelism**
- **Compared to multi-core CPUs, GPUs consist of a many-core architecture with hundreds to even thousands of very simple cores executing threads rather slowly**

- Use of very many simple cores
  - High throughput computing-oriented architecture
  - Use massive parallelism by executing a lot of concurrent threads slowly
  - Handle an ever increasing amount of multiple instruction threads
  - CPUs instead typically execute a single long thread as fast as possible
- Many-core GPUs are used in large clusters and within massively parallel supercomputers today
  - Named General-Purpose Computing on GPUs (GPGPU)



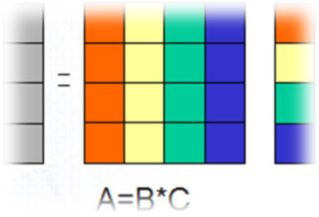
[1] Distributed & Cloud Computing Book

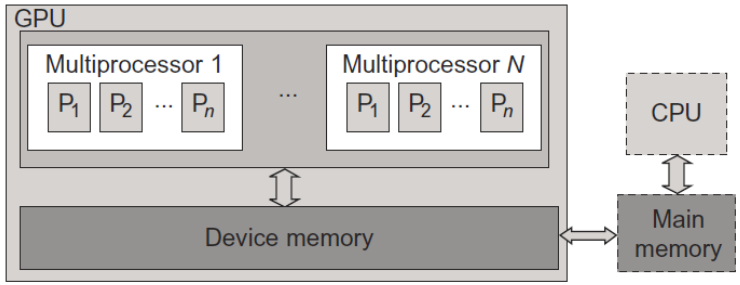
➢ **Lecture 6 & 7 provide more details on how GPUs are used as key technology in deep learning**

GPU Acceleration

- **CPU acceleration means that GPUs accelerate computing due to a massive parallelism with thousands of threads compared to only a few threads used by conventional CPUs**
- **GPUs are designed to compute large numbers of floating point operations in parallel**

- GPU accelerator architecture example (e.g. NVIDEA card)
  - GPUs can have 128 cores on one single GPU chip
  - Each core can work with eight threads of instructions
  - GPU is able to concurrently execute 128 * 8 = 1024 threads
  - Interaction and thus major (bandwidth) bottleneck between CPU and GPU is via memory interactions
  - E.g. applications that use matrix – vector multiplication



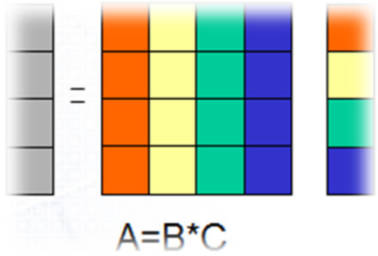


[1] Distributed & Cloud Computing Book

➢ **Lecture 6 & 7 provide more details on how GPUs are used as key technology in deep learning**

Parallel Matrix-Vector Multiplication Example on GPUs

- PO – P4 are processes on four GPU cores



(nice parallelization possible via independent computing)

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} b_{0,0}c_0 + & b_{0,1}c_1 + & b_{0,2}c_2 + & b_{0,3}c_3 \\ b_{1,0}c_0 + & b_{1,1}c_1 + & b_{1,2}c_2 + & b_{1,3}c_3 \\ b_{2,0}c_0 + & b_{2,1}c_1 + & b_{2,2}c_2 + & b_{2,3}c_3 \\ b_{3,0}c_0 + & b_{3,1}c_1 + & b_{3,2}c_2 + & b_{3,3}c_3 \end{bmatrix}$$

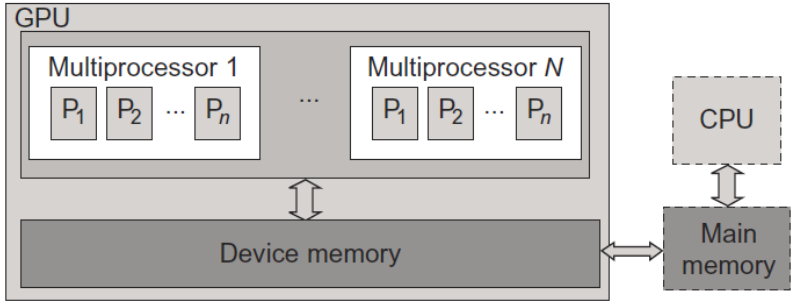
P0 P1 P2 P3

- **Step one: each GPU core has a column of matrix B (named as Bpart)**
- **Step one: each GPU core has an element of column vector C (named Cpart)**

- **Step two: Each GPU core performs an independent vector-scalar multiplication (based on their Bpart and Cpart contents)**

- **Step three: Each GPU core has a part of the result vector A (named Apart) and is written in device memory**

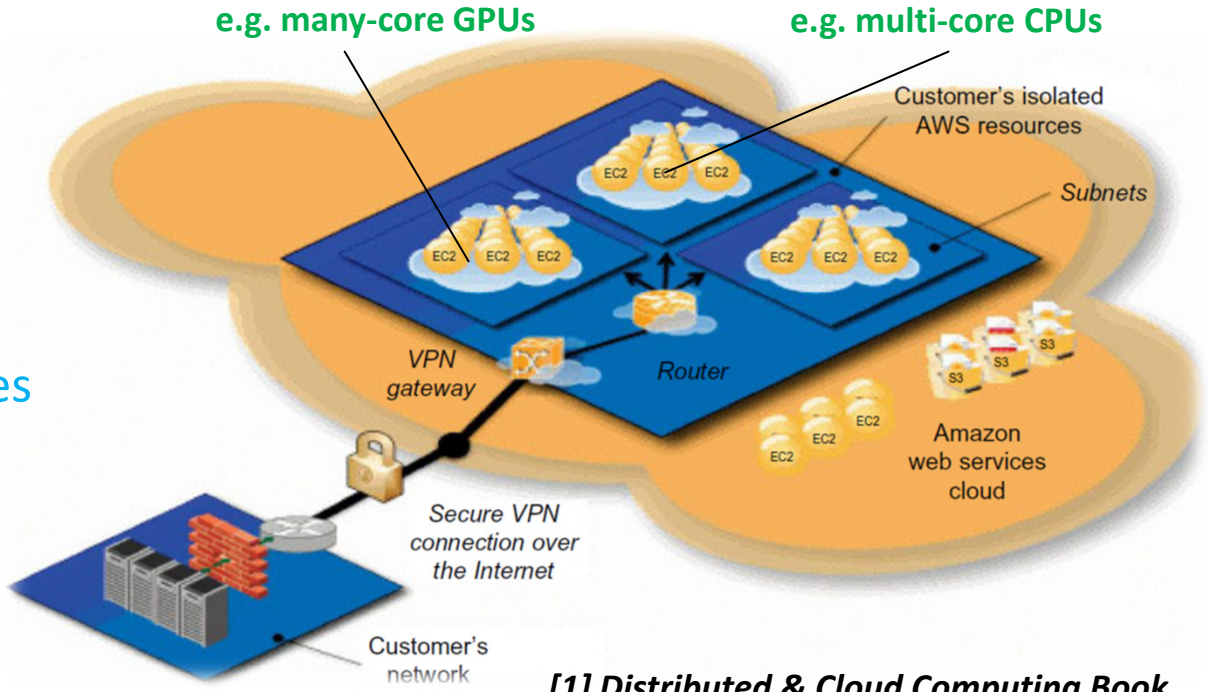
(GPUs are designed to compute large numbers of floating point operations in parallel)



[1] Distributed & Cloud Computing Book

Amazon EC2 Example

- Amazon EC2 provides an elastic compute cloud (EC2)
  - Elastic load balancing services and so-called auto-scaling
  - E.g. great during peak times in business (e.g. x-mas shopping, etc.)
  - Ensures that a sufficient number of EC2 instances are provisioned to meet expected performance
  - E.g. New York Times use it to quickly retrieve pictorial information from millions of articles



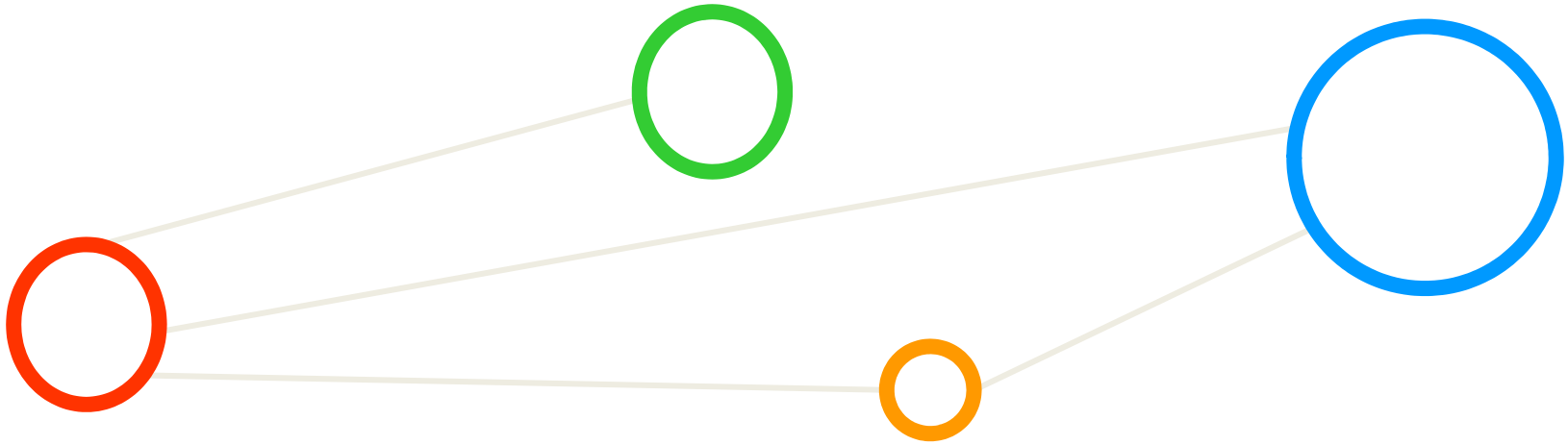
[1] Distributed & Cloud Computing Book

➢ **Lecture 8 provides more details about Amazon EC2 and its Infrastructure-as-a-Service models**

[Video] Cloud Computing Explained

[4] YouTube, Three Ways to Cloud Compute

Scalability driven by Big Data



What is Big Data?

- Some attempts of definitions for ‘Big Data’ :

  - **‘Big Data’ is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications.**

  [5] Wikipedia ‘Big Data’ Online

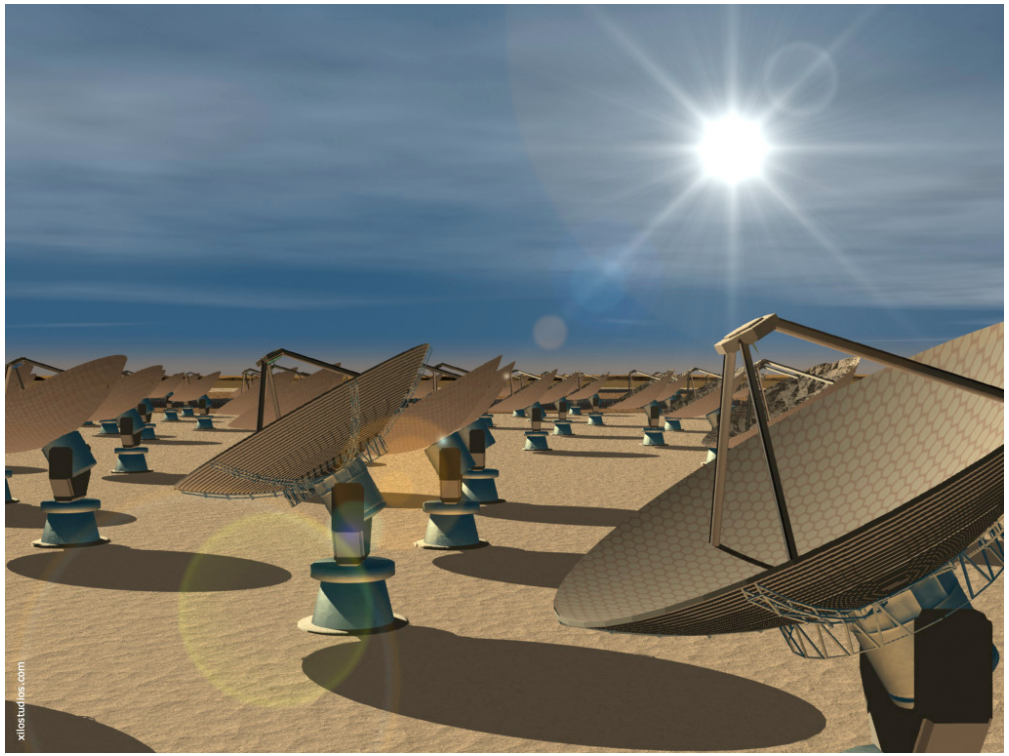
  - **‘Big Data’is data that becomes large enough that it cannot be processed using conventional methods.’**

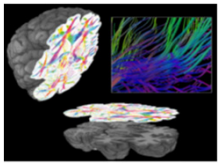
  [6] O’Reilly Radar Team, ‘Big Data Now: Current Perspectives from O'Reilly Radar’

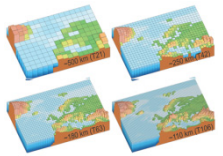
- Buzzword in science and engineering – what does this mean?
  - When does ‘big data’ start – with hundreds of MB / GB / TB / PB … EB?
  - Exact definition (in terms of volume) of big data is hard to find…
  - We have to look on concrete examples to find answers in Cloud context
  - Initially referred to VVV (Volume, Velocity, Variety) ***[7] Top 10 Big Data V’s***
  - Being constantly extended to n ‘Big Data Vs‘ (Veracity, Validity, …)

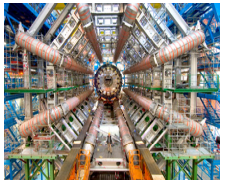
Search for Concrete ‘Big Data’– Examples & Challenges

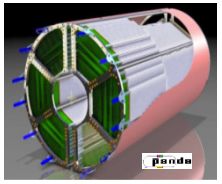
- **In commercial environments Big Data is all about Volume – Variety – Velocity, but concrete ‘sizes’ are rarely given**





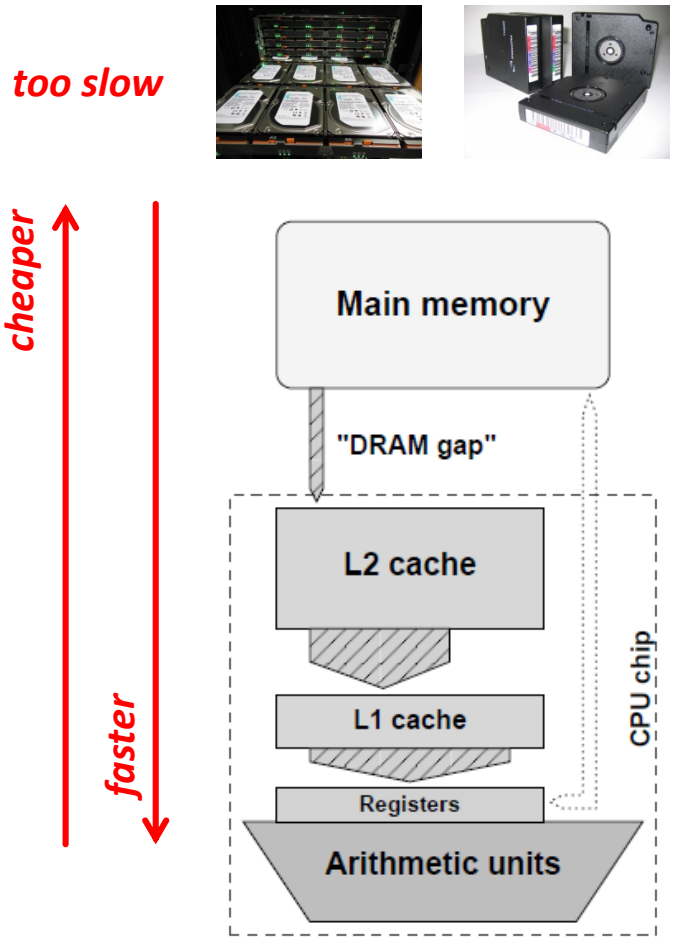






- **In ccience environments the term ‘Big Data’ is often related to one concrete scientific experiment: e.g. square kilometre array →1 PB / 20 seconds**

Challenge with Big Data: Fast Data Access for Processing

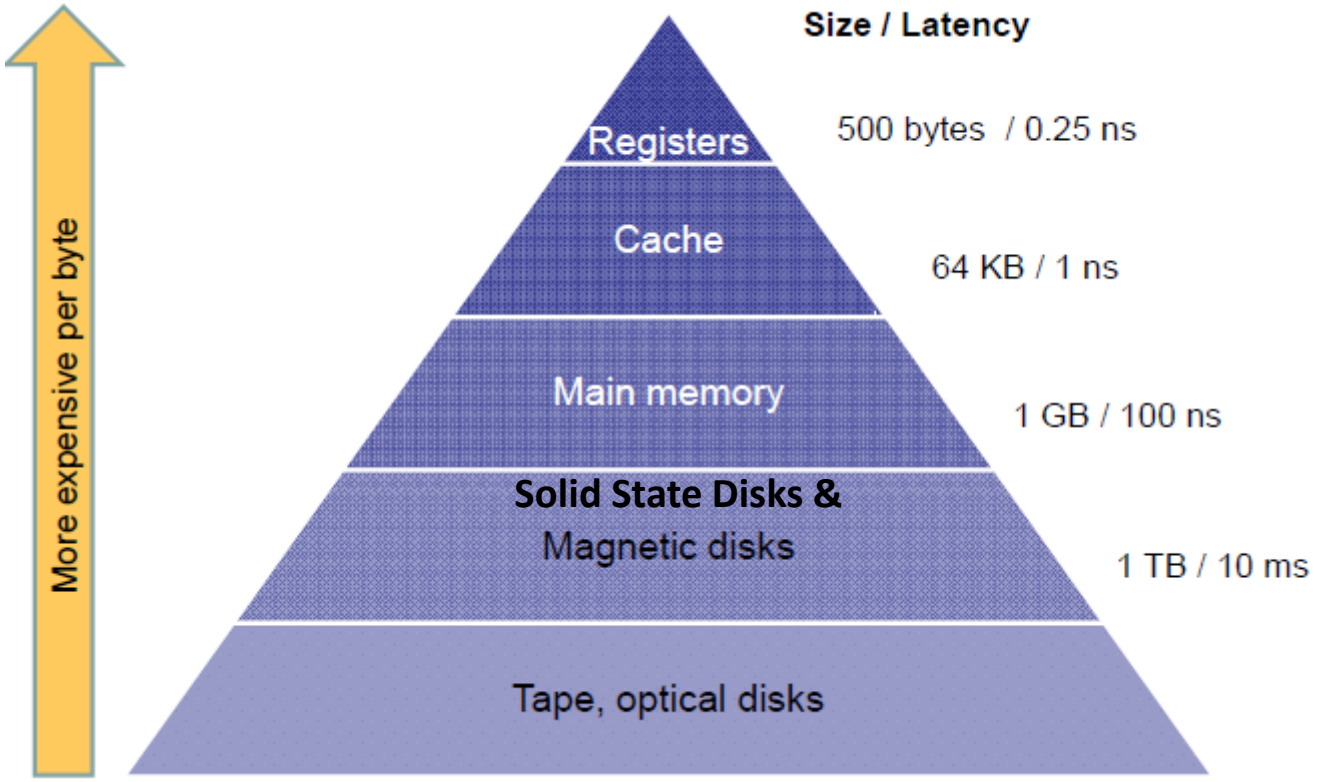


- Processor core elements & Memory
  - Compute: floating points or integers
  - Arithmetic units (compute operations)
  - Registers (feed those units with operands)
- 'Data access' for application/levels
  - Registers: 'accessed w/o any delay'
  - L1D = Level 1 Cache – Data (fastest, normal)
  - L2 = Level 2 Cache (fast, often)
  - L3 = Level 3 Cache (still fast, less often)
  - Main memory, Dynamic Random Access Memory (DRAM), slow, but larger in size
  - Too slow: storage like harddisk, tapes, etc.

[2] Introduction to High Performance Computing for Scientists and Engineers

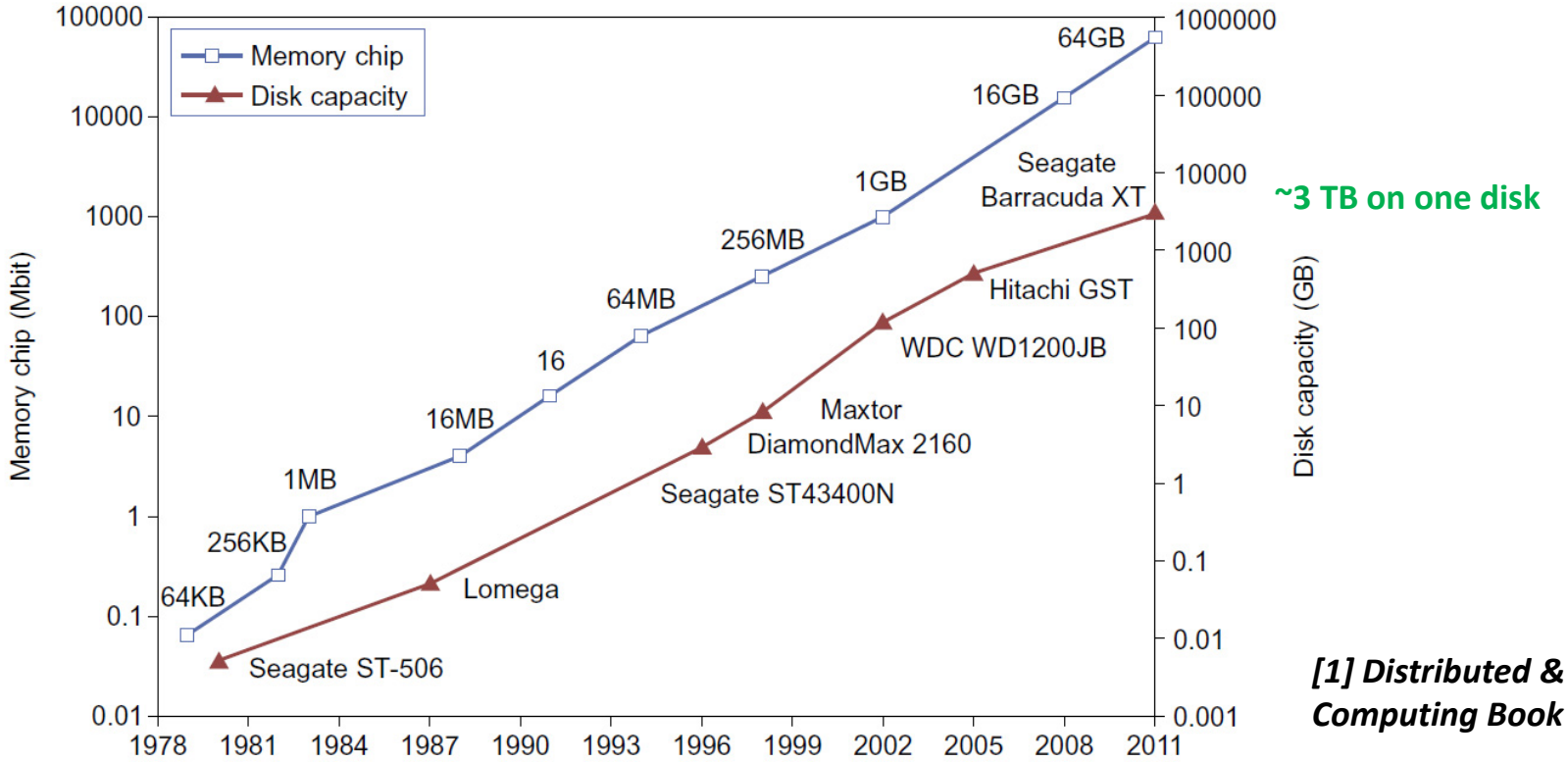
- **The DRAM gap is the large discrepancy between main memory and cache bandwidths**

Storage Devices as Storage Hierarchy



Cloud Enabling Technology Advances – Memory / Disks

- Improvement in memory and disk storage technologies

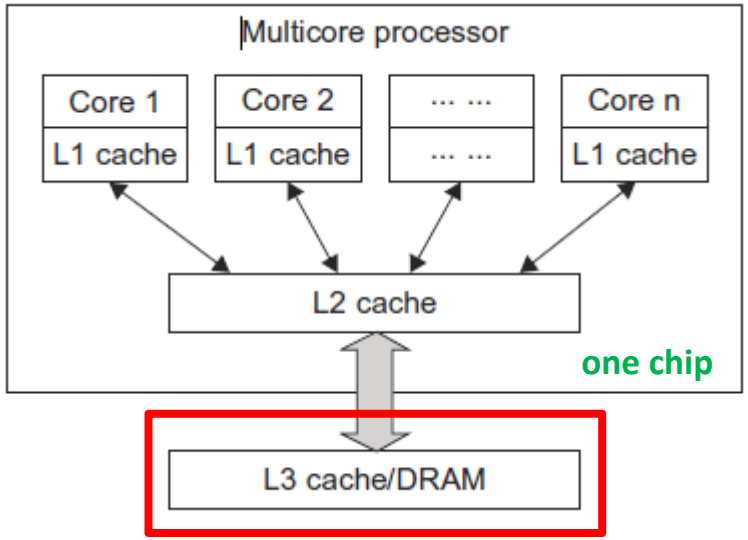


[1] Distributed & Cloud Computing Book

- **Fast memory chip capacities are measured in Kilobyte (KB), Megabyte (MB), Gigabyte (GB)**
- **Slower disk storage capacities are measured in Terabyte (TB), Petabyte (PB), and Exabyte (EB)**

Dynamic Random Access Memory (DRAM) & Memory Wall

- Fast memory used for data or program code
  - Computer processor needs memory to function
  - Volatile memory that loses data when power is removed



[1] Distributed & Cloud Computing Book

- Towards 'memory wall problem'
  - Faster CPU processor speeds
  - Larger memory capacity but access time not much better
  - Result is a wide gap between them

- **Huge growth of DRAM chip capacity in the last 30 years; 16 KB (1976) to 64 GB (2011) and more**
- **DRAM access time did not improve much contributing to 'memory wall problem' with better CPUs**

Storage Technologies

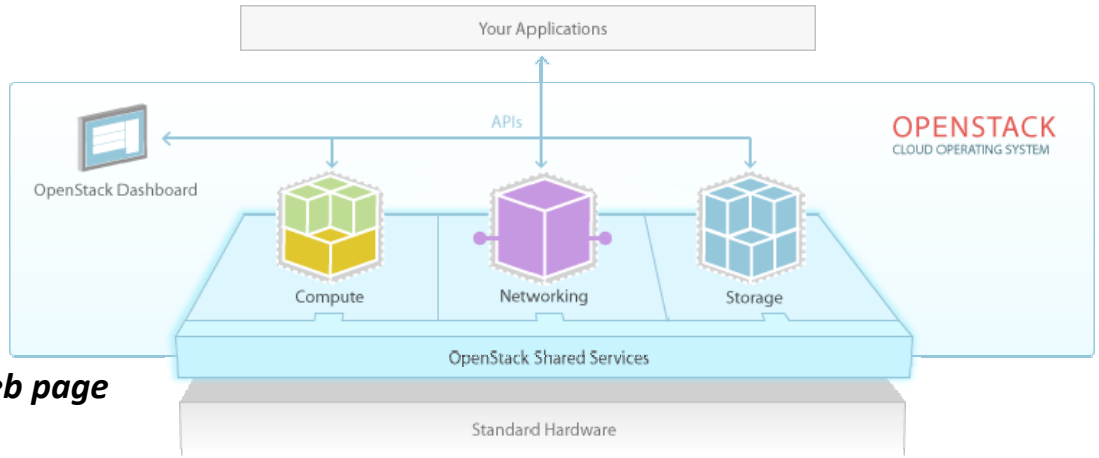
- Slower disk storage used for data used by program code
  - Magnetic disks in the past
  - Rapid growth of Solid State Disks (SDDs)
  - But SDDs are still quite expensive
  - Flash memory is a (solid-state) non-volatile storage for persistent data
- Huge disk storage capacity growth
  - Capacity increase last 30 years and continues for disk arrays in the future
  - Cloud data centres work with 'big data' in PBs, EBs, and above

array of disks used within cloud data centres

- **Huge growth of disk capacity in the last 30 years; 260 MB (1981), 250 GB (2004), 3 TB (2011)**
- **Rapid growth of flash memory and solid state disks (SDDs) impacts cloud computing storages**

OpenStack Swift Example

- OpenStack Cloud Operating System
  - Manages and controls cloud resources
- Swift is one of the core services
  - Manages and provides object storage as distributed system platform
  - Includes scale-out storage and highly fault tolerant features
  - E.g. storage tasks such as backup, archiving, data retention, etc.

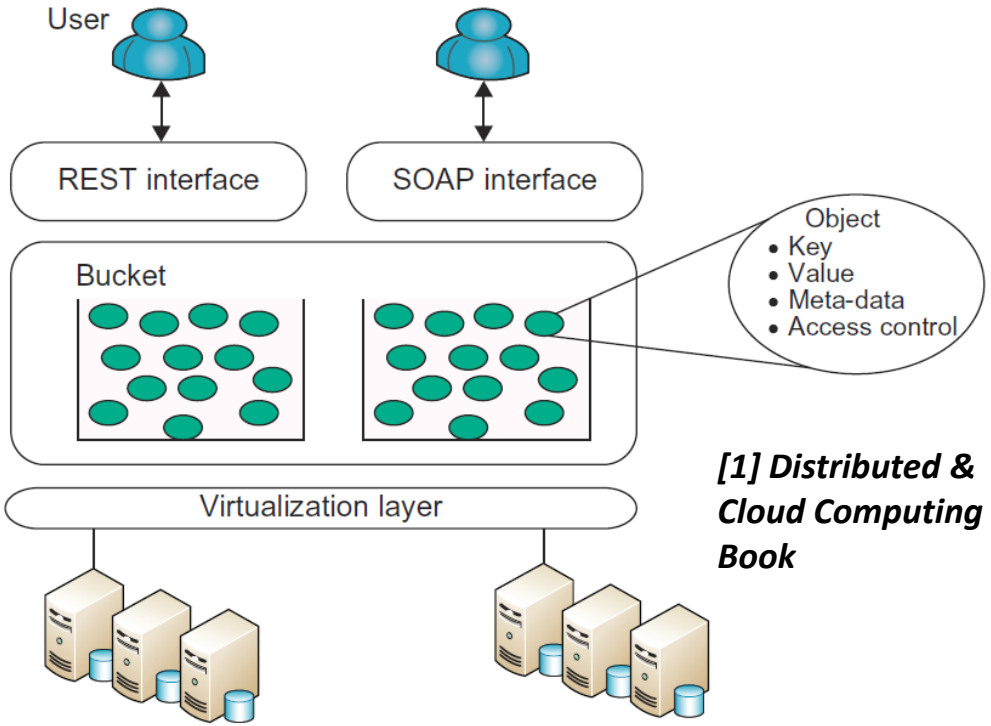


[8] OpenStack Web page

➢ **Lecture 13 provides in-depth insights into other features of the OpenStack cloud operating system**

Amazon S3 Example

- S3 is ‘storage as a service’ with a Web messaging interface
  - Using API with Representational State Transfer (REST)
  - Using API with Simple Object Access Protocol (SOAP)
- Remote object storage
  - Data considered objects to be named by end users
  - Objects alongside metadata are stored in bucket containers
  - Buckets enable the organization with namespace for user identification & accounting



[1] Distributed & Cloud Computing Book

➢ **Lecture 8 provides more details about Amazon S3 and its Infrastructure-as-a-Service models**

Big Data Source Example – Online Social Networking

1 | Facebook
3 - eBizMBA Rank | **1,100,000,000** - Estimated Unique Monthly Visitors | 3 - Compete Rank | 3 - Quantcast Rank | 2 - Alexa Rank | *Last Updated* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

2 | YouTube
3 - eBizMBA Rank | **1,000,000,000** - Estimated Unique Monthly Visitors | 4 - Compete Rank | 2 - Quantcast Rank | 3 - Alexa Rank | *Last Updated:* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

3 | Twitter
12 - eBizMBA Rank | **310,000,000** - Estimated Unique Monthly Visitors | 21 - Compete Rank | 8 - Quantcast Rank | 8 - Alexa Rank | *Last Updated* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

4 | LinkedIn
18 - eBizMBA Rank | **255,000,000** - Estimated Unique Monthly Visitors | 25 - Compete Rank | 19 - Quantcast Rank | 9 - Alexa Rank | *Last Updated* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

5 | Pinterest
22 - eBizMBA Rank | **250,000,000** - Estimated Unique Monthly Visitors | 27 - Compete Rank | 13 - Quantcast Rank | 26 - Alexa Rank | *Last Updated* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

6 | Google Plus+
30 - eBizMBA Rank | **120,000,000** - Estimated Unique Monthly Visitors | *32* - Compete Rank | *28* - Quantcast Rank | NA - Alexa Rank | *Last Updated* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

7 | Tumblr
34 - eBizMBA Rank | **110,000,000** - Estimated Unique Monthly Visitors | 55 - Compete Rank | *13* - Quantcast Rank | 34 - Alexa Rank | *Last Updated* January 1, 2017.
The Most Popular Social Networking Sites | eBizMBA

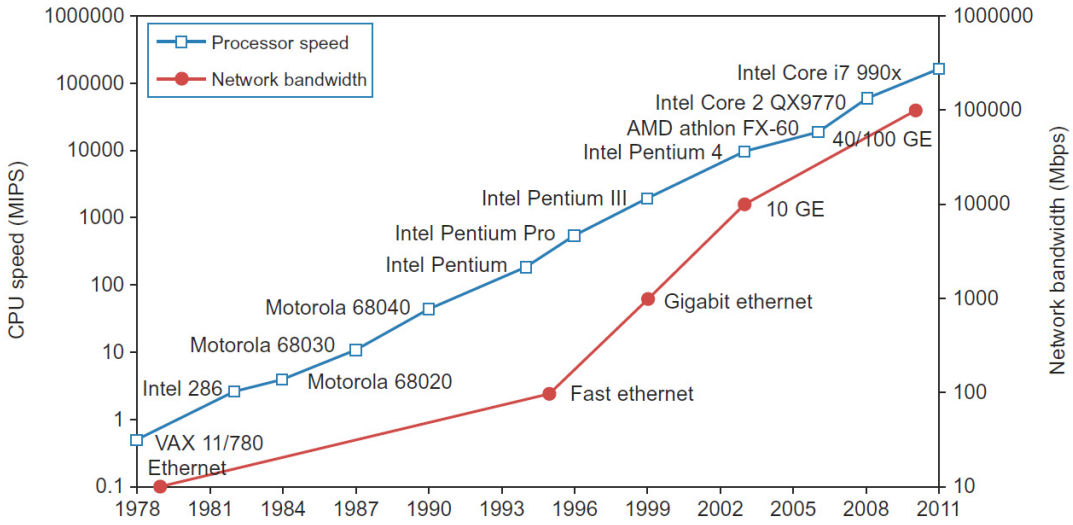
[9] Top 15 most popular social networking sites

- **Most of the popular Onlince Social Networking Web sites are installed with a client/server architecture where a large number of servers form a data center or using several data centres**

➢ **Lecture 14 provides insights into online social networking systems & their cloud & big data impact**

Role of Scalability & Wide-Area Networks

- High-bandwidth networking increases the capability of building massively distributed systems
  - Enables elastic computing and scalability beyond one server or one data center
  - Interconnected cloud data centers & servers



[1] Distributed & Cloud Computing Book

- **We observe tremendous price/performance ratio of commodity hardware that is driven by the desktop, notebook, and tablet computing markets today.**
- **Price/performance ratio has also driven the adoption and use of commodity hardware in large-scale and distributed computing including high-bandwidht network increases**

Networking & Big Data Impacts on Cloud Computing

- Requirements for scalable programming models and tools
  - CPU speed has surpassed IO capabilities of existing cloud resources
  - Data-intensive clouds with advanced analytics and analysis capabilities
  - Considering moving compute task to data vs. moving data to compute

➢ **Lecture 3 will provide details on data-intensive services & computing using Apache Spark**

- Requirements for Reliable Filesystems
  - Traditional parallel filesystems need to prove their 'big data' feasibility
  - Emerging new forms of filesystems that assume hardware error constantly
  - E.g. Hadoop distributed file system (HDFS)

[10] HDFS Architecture Guide

➢ **Lecture 5 will give in-depth details on Map-Reduce & implementation Hadoop & filesystem**

Big Data Analytics vs. Data Analysis

- **Analytics are powerful techniques to work on large data**
- **Data Analysis is the in-depth interpretation of research data**

- Data Analysis supports the search for 'causality'
  - Describing exactly WHY something is happening
  - Understanding causality is hard and time-consuming
  - Searching it often leads us down the wrong paths

- Big Data Analytics focussed on 'correlation'
  - Not focussed on causality – enough THAT it is happening
  - Discover novel patterns and WHAT is happening more quickly
  - Using correlations for invaluable insights – often data speaks for itself

➢ **Lecture 11 will give in-depth details on differences between data analysis & big data analytics**

Big Data Analytics Frameworks

- **Big Data analytics frameworks shift the approach from ‘bring data to compute resources‘ into ‘bring compute tasks close to data‘**

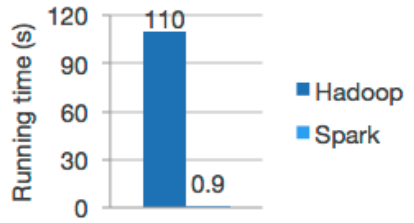
- Distributed Processing

[11] Map-Reduce

  - ‘Map-reduce via files‘: Tackle large problems with many small tasks
  - Advantage of ‘data replication‘ via specialized distributed file system
  - E.g. Apache Hadoop

➢ **Lecture 5 will give in-depth details on using the map-reduce paradigm for big data analytics**

- In-Memory Processing
  - Perform many operations fast via ‘in-memory‘
  - Enable tasks such as ‘map-reduce in-memory‘
  - E.g. Apache Spark, Apache Flink



Logistic regression in Hadoop and Spark

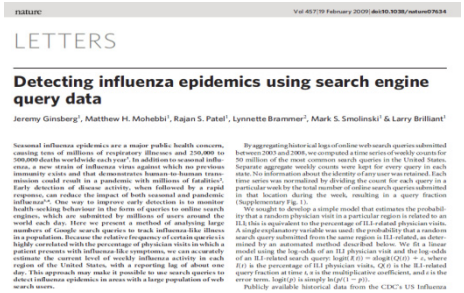
[12] Apache Spark

➢ **Lecture 3 will give in-depth details on using in-memory processing & Apache Spark & MLLib**

Big Data Analytics Applications – Example

- ~2009 – H1N1 Virus Made Headlines
  - Google using ‘logged big data‘
    → search queries
  - Explains how Google is able to predict fast winter flus
  - Not only on national scale, but down to regions
- ~2014 – The Parable of Google Flu
  - Large errors in flu prediction & lessons learned

- **Large errors are possible when working with ‘big data‘ to infer insights with ‘statistical data mining‘ methods**
- **(1) Dataset: Transparency & Replicability impossible**
- **(2) Study the algorithm since they keep changing in Google: making reproducability impossible**
- **(3) It‘s not just about the size of the data: elements like data quality and many other factors play a role too**

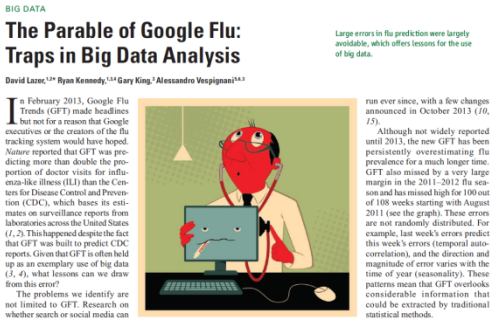

nature

LETTERS

Detecting influenza epidemics using search engine query data

Jeremy Ginsberg[1], Matthew H. Mohebbi[1], Rajan S. Patel[1], Lynnette Brammer[2], Mark S. Smolinski[3] & Larry Brilliant[1]

[13] Jeremy Ginsburg et al., ‘Detecting influenza epidemics using search engine query data’, Nature 457, 2009


BIG DATA

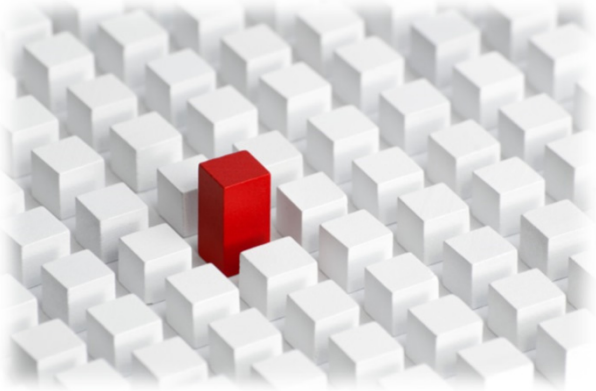
The Parable of Google Flu: Traps in Big Data Analysis

Large errors in flu prediction were largely avoidable, which offers lessons for the use of big data.

David Lazer,[1,2]* Ryan Kennedy,[1,3,4] Gary King,[2] Alessandro Vespignani[3,4,5]

[14] David Lazer, Ryan Kennedy, Gary King, and Alessandro Vespignani, ‘The Parable of Google Flu: Traps in Big Data Analysis’, Science Vol (343), 2014

Big Data Motivation: Intertwine Clouds & Machine Learning



- Rapid advances in data collection and storage technologies in the last decade
  - Extracting useful information is a challenge considering ever increasing massive datasets
  - Traditional data analysis techniques cannot be used in growing cases (e.g. memory, speed, etc.)

(yellow blocks very important to learn for exams)

- **Machine learning / Data Mining is a technology that blends traditional data analysis methods with sophisticated algorithms for processing large volumes of data**
- **Machine Learning / Data Mining is the process of automatically discovering useful information in large data repositories ideally following a systematic process**

modified from [11] Introduction to Data Mining

- Machine Learning & Statistical Data Mining
  - Traditional statistical approaches are still very useful to consider
  - Deep Learning tools become effective and are available in Clouds today

Relationship Machine/Deep Learning & Big Data & Clouds

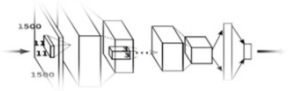
#GPUs	images/s	speedup	Performance per GPU [images/s]
1	55	1.0	55
4	178	3.2	44.5
8	357	6.5	44.63
16	689	12.5	43.06
32	1230	22.4	38.44
64	2276	41.4	35.56
128	5562	101.1	43.45

(large amounts of GPUs = Graphical Processing Units)

Processing Time

'small training sets'

Large Deep Learning Network

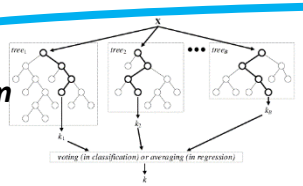


Medium Deep Learning Network

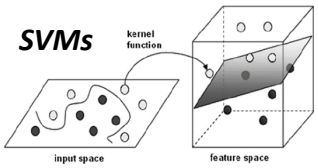
Small Neural Network

Traditional Learning Models

manual feature engineering changes ordering



Random Forests



SVMs

MatLab

Statistical Computing with R

scikit-learn *Weka* *Octave*

Model Performance / Accuracy

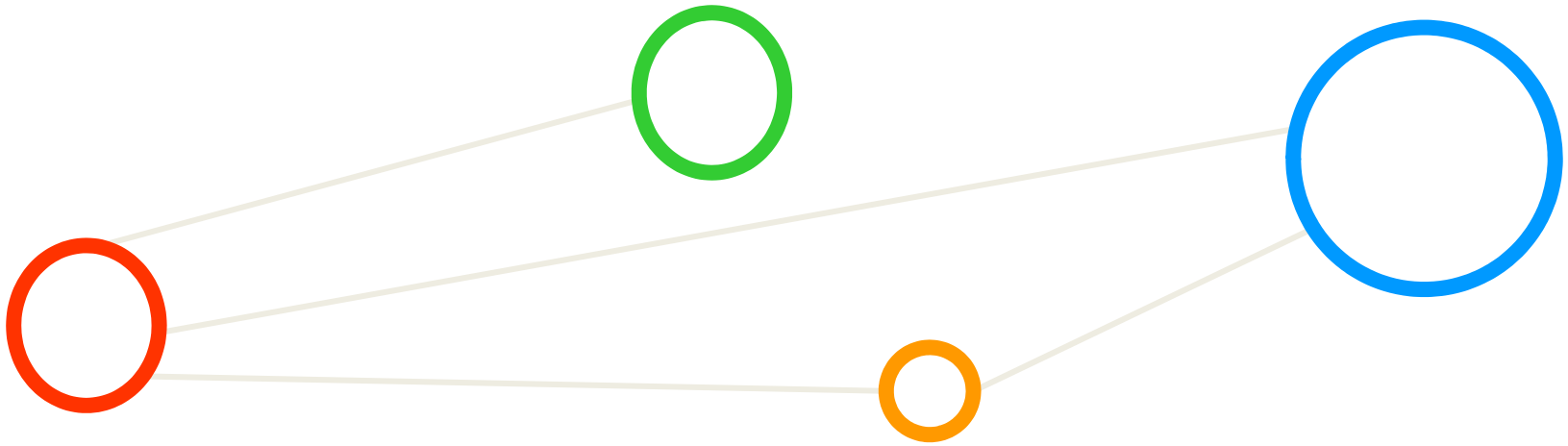
Dataset Volume

→ 'Big Data'

[Video] Big Data Analytics in Banking Industry

[15] IBM Big Data and Analytics

Lecture Bibliography



Lecture Bibliography (1)

- [1] K. Hwang, G. C. Fox, J. J. Dongarra, 'Distributed and Cloud Computing', Book, Online: http://store.elsevier.com/product.jsp?locale=en_EU&isbn=9780128002049
- [2] Introduction to High Performance Computing for Scientists and Engineers, Georg Hager & Gerhard Wellein, Chapman & Hall/CRC Computational Science, ISBN 143981192X
- [3] BOINC Middleware Tool, Online: https://boinc.berkeley.edu/
- [4] YouTube Video, 'The Three Ways to Cloud Compute', Online: https://www.youtube.com/watch?v=SgujaIzkwrE
- [5] Wikipedia 'Big Data', Online: http://en.wikipedia.org/wiki/Big_data
- [6] O'Reilly Radar Team, 'Big Data Now: Current Perspectives from O'Reilly Radar'
- [7] Top 10 Big Data V's, Online: http://www.datasciencecentral.com/profiles/blogs/top-10-list-the-v-s-of-big-data
- [8] OpenStack Web page, Online: https://www.openstack.org/software/
- [9] Top 15 Most Popular Social Networking Sites, Online: http://www.ebizmba.com/articles/social-networking-websites
- [10] HDFS Architecture Guide, Online: http://hadoop.apache.org/docs/stable/hdfs_design.html
- [11] J. Dean, S. Ghemawat, 'MapReduce: Simplified Data Processing on Large Clusters', OSDI'04: Sixth Symposium on Operating System Design and Implementation, December, 2004.

Lecture Bibliography (2)

- [12] Apache Spark,
  Online: http://spark.apache.org/
- [13] Jeremy Ginsburg et al., 'Detecting influenza epidemics using search engine query data',
  Nature 457, 2009
- [14] David Lazer, Ryan Kennedy, Gary King, and Alessandro Vespignani, 'The Parable of Google Flu: Traps in Big Data Analysis', Science Vol (343), 2014
- [15] 'Demo: IBM Big Data and Analytics at work in Banking', YouTube Video,
  Online: https://www.youtube.com/watch?v=1RYKgj-QK4I
- [16] Official Web Page for Python Programming Language,
  Online: https://www.python.org/
- [17] Numpy Python Library Web Page,
  Online: http://www.numpy.org/
- [18] Blog 'Vectorization and parallelization in Python with NumPy and Pandas',
  Online: https://datascience.blog.wzb.eu/2018/02/02/vectorization-and-parallelization-in-python-with-numpy-and-pandas/


Data
Science
Big
Research
science
analysis
methods
big
simulations
processing
technologies
computing
TB
European
Supercomputing
Scientific
computational
research
databases
Computational
HPC
Juelich
Climate
modelling
structures
datasets
performance
management
data-intensive
Service
EUDAT
ENES
storage
community
centers
nodes
Computing
climate
Centre
resources
services
year
System
access
images
often