

Introduction to Big Data Processing Infrastructures

Introduction

- This course is not about Big Data, it is about infrastructures
- We should have visited a datacenter, but... Corona
- We need SSH, a little bit of C/C++, vim, Python
- We will use AWS and Google Cloud computing!
- Of course we will make a project and an oral exam
- Prof of course is a physicist of the IIFN
- The IIFN-CNAF hosts the Tier-1 datacenter in Bologna

Setting up AWS

- It is a pay for use model
- You have at the beginning a 50\$ credit

Big data

- They tend to be non structured
- They are characterized by 4 Vs: Volume, Variety, Veracity (can I trust them?), Velocity
- Another important point is Value, what I want to extract from the data

Computational challenge

- Find a substring in a string
- Doing it brute force is really slow
- I can create an index once and then I can search every time much faster
 - It is blast essentially
 - The BWA algorithm is another possibility
 - * It is used for more similar sequences
 - * It is faster when I have many reads to be aligned
- Using the right approach is much more effective than increasing computational power
- If possible use open source code
 - You avoid vendor lockin and the approach can be scaled more easily
- Creating a new approach from scratch is usually wrong
 - You will not create a good system from scratch c* Our challenge: align 554k sequences per patient against the human genome
 - The aim of this course is creating a cloud based model for computing this
 - I also want to estimate the time required

Some notes

- A checksum is essential when we are moving data
 - It is a small string used to detect error in data transmission or storage
 - It is possible to set an extended file attribute with the checksum (if the file system allows it)
- Files are moved compressed, moving uncompressed files is a crime

CPU

- 1 byte is 8 bits (but we can also use 10 to make it easier)
- A multicore processor has more than 1 processing unit
 - A core appears to the processor as a different CPU
 - A socket is the physical CPU, and it can have many cores
- A core is composed of an ALU, processor registers and control unit

- The register gives data to the ALU and stores its output
 - The control unit fetches data from memory and coordinates the operations of ALU and register
- Hyper-threading is a technology from Intel that represents each physical core with two logical cores at the software level
 - The OS sees a double number of cores
 - It maximises the exploitation of the processor by running at the same time operations that can be run at the same time
 - * An example can be doing an ALU operation and a logical operation at the same time
 - * When a process is waiting for something the processor can run another process
 - It can improve performances but it depends on the application
 - It uses more die area (space on the silicon chip)
 - It can also degrade performance by trashing the cache
 - * Memory is virtual in modern systems, in the sense that the address space used by applications does not refer to the physical memory address
 - * The MMU (memory management unit) translates the virtual addresses to physical addresses on the fly
 - * Virtual memory is the virtual address space, which can map to physical memory but also to disk
 - * Memory is managed in chunks called pages by the OS
 - * When a page is requested by an application, if its real location is not in memory it raise a page fault error
 - * The OS then swaps the requested page in memory from the swap area of the disk
 - * If there is no space in memory another page is swapped out to disk
 - * If the memory is constantly full this swapping results in a lot of time spent moving data back and forth, degrading performances
- `top` shows the status of all the logical cores in the system
 - `wa` shows the time spent waiting for I/O
 - `load` average shows the number of processes waiting for enetering in CPU
 - * It should be at most equal to the number of cores, otherwise we are in an overloading
- A good source of info for the system is `cat /proc/cpuinfo`
 - `flags` shows the capabilities of the processor (instructions)

Memory

- Memory is RAM, it is volatile and fast
- We typically have a memory hierarchy
 - The first memory is the CPU register
 - L1, L2 and L3 cache in the processor
 - * L1 is subdivided in L1i (instructions) and L1d (data)
 - * L3 is shared among cores (in some cases), while L1 and L2 are core-specific
 - * I have 128 KiB of L1i and 128 KiB of L1d cache, 1 MiB L2 and 8 MiB L3
 - Main system memory (RAM)
 - Swap as a last resort
- Different pieaces of memory have different latency
 - L1 has 4 cycles latency, 0.5 ns
 - L2 11 cycles, 7 ns
 - L3 39 cycles
 - RAM 107 cycles, 100 ns
- Data is transferred between pieces of memory in cache lines
 - It is a data chunk, typically 64 bytes
 - We can write code that optimizes the use of cache lines
 - * Keep physically close in memory data which is accessed together!
- Because of this latency differences, an $O(n)$ algorithm can perform better that a $O(1)$ if it cause less memory access

- A RAM disk is a portion of RAM used as a storage device
- Cache lines operate on 84 GB/s between register and L1, 60 GB/s between L1 and L2, 30 GB/s between L2 and L3 and 10 GB/s between L3 and RAM
- Registers typically use SRAM, while other caches use SRAM or DRAM and memory use DRAM
 - SRAM is static RAM, it does not need refreshing but it is still volatile
 - * It is made with a bistable circuitry called flip-flop
 - * It is more expensive than DRAM since it uses 6 transistors per bit (a flip-flop circuit!)
 - * 1 Gb can cost 5000\$
 - DRAM needs refreshing and it is made with capacitors
 - * Each bit is managed by one transistor and one capacitor
 - * The transistor manages read and write operations on its capacitor
 - * 1 Gb can cost 50\$
 - * SDRAM is a DRAM which operates in sync with the clock
 - * DDR is a type of SDRAM
- Memory status can be seen with **free**

Network

- A computer network is an infrastructure that shares resources between nodes
- Network topology can affect reliability and throughput
 - Bus: everything connected to the backbone
 - Star: everything connected to a central node
 - Ring: each node connected to the ones at its sides
 - Mesh: each node is connected to an arbitrary number of nodes but guaranteeing that all nodes can be reached
 - Tree: hierarchical
 - Fully connected network
- OSI (open system interconnection) model: a series of layers from physical to abstract
 - Layer 1, physical layer: concerns the raw bit transmission
 - * It codes and decodes bits into the physical transmission protocol (voltage levels, frequencies, ...)
 - Layer 2, data link: node to node data transfer
 - * It catches and corrects errors in layer 1 transmission
 - * It initiates, maintains, and terminates node to node connections
 - * The MAC is part of this level
 - Layer 3, network layer: packets of data and transmission across networks
 - * It splits the data in packets
 - * It manages the route data to an address in the network
 - * It can provide reliable data transfer, but not necessarily
 - Layer 4, transport layer: control data reliability
 - * It re-transmits damaged packets
 - * It includes TCP and UDP
 - Layer 5, session layer: ports and sessions
 - Layer 6, presentation layer: encryption and make the data usable
 - Layer 7, application layer: human-computer interaction layer, it interacts with applications and with the presentation layer
- OSI is a general networking model and reference framework
- The Internet does not strictly follow the OSI model, but it uses the internet protocol suite (TCP/IP)
- The TCP/IP model (not the TCP protocol!) is an alternative to OSI
 - It collapses OSI 7 to 5 in the application level and OSI 1 and 2 to the network interface level
- LAN (local area network) are small and localised networks
 - They are present in houses (made by a router), universities, businesses
- WAN (wide area network) cover cities, nations, etc.
 - It is made up of connected LANs

- Routing a packet in a WAN means finding to which LAN it should go
 - ARPANET (US defense) was the first WAN and the first network to implement TCP/IP
- LHCone network: the WAN of the LHC
- Packets: organising data
 - The alternative is to transmit bit streams
 - It is a formatted list of bits of 10 bits to some kbs
 - They include control information like destination, source, checksums
 - The actual data transmitted is the payload
 - Increasing the payload increases speed at the cost of accuracy
- MTU (maximum transmission unit) is the size of the largest protocol data unit (PDU) that can be transmitted in a single network layer transaction
 - Larger MTU reduces overhead but can increase delay
 - It should be set so to respect the properties of the physical network
- The MAC (media access control) address is a unique ID assigned to a NIC (network interface controller)
 - It is at OSI 2
 - It is used for communications within a network segment
- The IP (internet protocol) address is a numerical label assigned to the machine
 - It is at layer OSI 3
 - IPv4 is a 32 bit number
 - To make it more human-readable, each byte is represented with a decimal number from 0 to 255
- The IP is subdivided into a subnet and a rest field using the CIDR notation or the subnet mask
 - The network prefix identifies the network, while the rest field is host-specific
 - The CIDR notation is represented with a slash after the IP that says the number of bits to be used for the subnet
 - * 192.168.1.1/24 means that the first 24 bits of the 32 total (192.168.1) are the subnet, while 1 is the rest field
 - The subnet mask (255.255.255.0) means that the first 3 bytes are the subnet while the last one is the rest field
 - * The subnet is obtained with a bitwise AND between IP and subnet mask
- IPv4 uses private addresses, that do not to be globally unique
 - They are not exposed to the internet
 - The private address is translated to/from a global address with NAT (network address translation)
- There are 3 non-overlapping address ranges reserved for private use
 - 24 bit block: CIDR 10.0.0.0/8
 - * 8 bits is the subnet so 24 bits (8-32) is the rest field
 - 20 bit block: CIDR 172.16.0.0/12
 - 16 bit block: CIDR 192.168.0.0/16
 - * 8 bit block: CIDR 192.168.0.0/24
- IPv6 has a 128 bit address space
 - Actually it is much different from IPv4 in terms of routing
 - Its size allows to reserve large address blocks for specific use
 - The unique local address (ULA) is reserved for local use
 - * It is fc00::/7
- `ifconfig` can be used to configure the network interface
- The loopback network device: a connection to the same machine
 - It is entirely managed by the OS and does not send any packet to network devices
 - It is assigned to the block 127.0.0.0/8 in IPv4
 - In practice it is used almost always 127.0.0.1
 - `localhost` is mapped to 127.0.0.1
 - It is used for diagnostic and by applications to reach resources on the same machine
- DNS is a decentralised mapping system between IP addresses and mnemonic names
- A network protocol is a set of rules for exchanging packets over a network
 - In a protocol stack each protocol uses services from the one below it
 - HTTP runs over TCP which runs over IP

- There are various kinds of bandwidth
 - Goodput is the actual data transmitted
 - Channel bandwidth includes also the overhead control information
- Latency is measured as end-to-end delay (OWD) or round-trip-time (RTT)
 - It can be assessed with `ping`
 - RTT is usually 2*OWD, but not necessarily (upload and download can differ)
 - Latency cannot be smaller than what the speed of light requires!
- Network adapters are physical devices that connects a computer to a network⁷
 - They can be part of the motherboard or can be plugged in as expansion cards
 - They implement OSI 1 and 2
 - Network adapters are Ethernet, WiFi, Fiber Channel
 - There are also high-performance adapters used for high-performance computing like Omnipath and Infiniband
 - * Infiniband (Mellanox) and Omnipath (Intel) are high bandwidth and low latency network interfaces
- Hub: it broadcast the same packet to all the ports
- Switch: delivers a packet to a specific machine
 - It knows the MAC of all the devices attached to it
 - It works at OSI2 (and in some case at OSI3)
 - It does not increase significantly network response times
- Router: delivers a packet to an IP address
 - It can route to other LANs
 - It works at OSI3
- Top-of-the-rack switching: a switch that delivers packets to a rack of servers
 - Usually there is also a switch that delivers to the right rack

Computing Infrastructures

- A computing farm is a collection of servers and it can have millions of cores
 - Network devices manage communication between servers and the interaction with users
- Intel Xeon is used in datacenters since it is reliable and can work continuously
- The link between sockets in the same motherboard is called qlink
- Servers are organised in hot islands which are separated by a cooling system
- A computing farm is usually shared among paying customers
 - The resources must be allocated efficiently among users
 - The batch scheduler manages resource access
- A batch system takes care of scheduling non-interactive jobs
 - There are many batch systems: HTCondor, OpenLava, LSF, ...
 - It dynamically allocates jobs so to maximise cluster use, minimise latency and respect fairshare on a time window
 - It provides a single point of control for jobs submitted to the CPU farm
- Jobs are composed of
 - Job type
 - Prologue: initial checks
 - Input sandbox: the list of needed files
 - Requirements: the hardware required
 - Executable: the actual code to run
 - where stdout/stderr should be directed
 - Output sandbox: the files that need to be produced
 - Epilogue: final cleanup, file uploads, updates, ...
 - Error recovery: what to do if the job fails
- A job can be a single batch job that occupies a single slot and it is executed in one core
- A DAG workflow is a series of jobs dependent from each other described by a directed acyclic graph
 - It is essentially a pipeline

- A collection of jobs can be run in parallel
- A parallel job needs more than 1 core to run
- Reservation: the batch scheduler can reserve cores for 1 job that is waiting for something to be executed
 - This is typical of parallel jobs that are waiting for enough cores to be available
- Backfill: while the reserved core are idle they can be used by other jobs
 - Only jobs that will finish before the job that reserved the cores will start are permitted

Storage

- The performance of a storage device is described in terms of bandwidth and IOPS
 - The bandwidth is the data transfer speed
 - The IOPS is the number of I/O operations per second
 - * It is important when operating on small files
- SSDs use NAND technology, it is much slower than RAM but non-volatile
 - It can withstand a limited number of write cycles
 - NAND SSDs use NAND logical gates
 - The NAND used in USB drives tend to be less performant, cheaper and less durable than the one used in SSDs
 - Speed is 400/500 MB/s, 100k/400k IOPS
 - Capacity 500 GB
 - The biggest advantage of SSDs is in terms of IOPS
- Hard disks are slow since they have moving parts
 - Speed is 100/150 Mb and 100/200 IOPS
 - Capacity 8 TB
 - They have spinning disks and a moving head
- Tape cartridge are expensive since they require an infrastructure (tape library) but really scalable
 - Speed is 250 MB/s, sequential access (no IOPS)
 - Capacity 8.5 TB
- Data maintenance is a huge responsibility, since it could be impossible to recreate lost data
- RAID (Redundant Array of Independent Disks) systems allow to virtualize data storage
- RAID0: file striping at the block level
 - A single file is striped into several drives
 - I collect n physical disks into 1 virtual disk
 - I get higher performances but no redundancy
 - Losing only 1 disk means losing all the data
- RAID1: file mirroring
 - Performances can improve for reading
 - 2 disk that operate as 1, mirroring their contents
- RAID2 is rarely used and involves striping at the bit level
 - It uses an Hamming code for error correction
- RAID3 is rarely used and involves striping at the byte level with a parity disk
 - A parity disk contains parity bits for each block, that are used in error correction
 - It requires the rotation of disks to be in sync!
- RAID4: block-level striping with parity disk
- RAID5: block-level striping with distributed parity
 - In each disk I have all the blocks but 1 plus the parity of the missing block
 - It requires at least 3 disks
 - It is similar to RAID0 in term of performances (but a little bit less), but tolerant to the loss of an entire disk
 - It is more reliable of RAID0 because of parity, but also less performant
- RAID6: block-level striping with double distributed parity
 - It is similar to RAID5 but it uses 2 parity blocks
 - It can tolerate the failure of 2 disks
 - It requires a minimum of 4 disks

- A file system controls how data are stored and retrieved in the physical medium
 - It can support file metadata like owner, permissions, access time
- POSIX (Portable Operating System Interface) is a family of standards for maintaining compatibility between OS
- A POSIX filesystem basically is one that supports open operations
- Cables are important, especially for DAS (direct access storage) systems
- A DAS system does not incorporate any network capability
- A JBOD (just a bunch of disks) is a container containing disks that can be plugged into a server
- A NAS (network access storage) is a storage device exposed to the network
 - It has file level access for external clients
 - It uses protocols like NFS and SAMBA
- The storage area network (SAN) is a dedicated network for storage access at the block level
 - I am not exposing the file system, but the disk itself
- The SAN is composed of an host layer, a fabric layer and Storage layer
 - The host layer is composed of servers that allow access to the SAN
 - The fabric layer is composed of cables, routers, switches
 - The storage layer is composed of the storage devices
 - * Every partition of every storage device is identified by a logic unit number (LUN)
- Important data are backed up frequently and possibly moved to a different geographical location
- A parallel filesystem is a single filesystem across multiple storage devices
 - It usually supports multiple clients (thousands of them!)
 - It is built on a SAN, with block level access
 - It maintains metadata servers with information about the location of the blocks for each file and their metadata
- Metadata are essential and cannot be lost: metadata servers have a lot of redundancy
- A distributed file system is built on a system with file-level access
 - It is built on top of the file systems of the disks!
 - They are typically slower than parallel file systems
- The tape area network (TAN) is a subsection of a SAN dedicated to tape devices
 - Tapes are organized in libraries containing tape drives and robotic arms that move the tape cartridges into the drives
 - A tape library can have 85 PB capacity
- Storage is organised in tiers according to their QoS
 - Tier 1 storage is used for hot data (frequently used)
 - * It is composed of large RAID systems and SSDs
 - Tier 2 is used for warm data
 - * It is mainly old disks and small RAID5s
 - Tier 3 is used for cold data like backups and rarely accessed data
 - * It is mainly tape storage
- Moving a file to an higher QoS is called recall, moving it to a lower one migration
- Tape storage is extremely safe: I can put the cartridge in a safe box and it is inaccessible
- Nowadays datacenters are adopting the HyperConverged architecture
 - Storage and computing happens on the same machines, no separate areas

Cloud Computing

- It is not the same as virtualization!
- It is a pay-per-use model, an utility service
- The infrastructure has a theoretical infinite capacity
 - But you need infinite money to use it!
- The model is focused on the concept of service, without the need of thinking about the underlying physical infrastructure
- Infrastructure as a service (IaaS): the customer receives a machine with an OS
- Platform as a service (PaaS): the customer manages data and software, but no direct access to the OS

- I can ask for a HTC-Condor cluster already functional, a parallel file system, an SQL database
- Software as a service (SaaS): pre-formed environment entirely managed by the vendor
 - Gmail infrastructure for companies
- Deployment: public, hybrid private
 - A private cloud is accessible only to insiders
 - * It is rarely with infinite resources
 - A public cloud is AWS, which can be assumed infinite
 - An hybrid system is a private system that redirects peak requests to a public system
 - I can also manage only sensitive data on a private system and non-sensitive data in a public cloud
- Isolation: the scope of the cloud
 - A dedicated cloud is dedicated to a scope (es. Bioinformatics)
 - A multi-tenant cloud is multi-purpose
 - I can also consider a tenant as a group of customers (es. research group), that payed for a specific QoS
- Web hosting is one of the most common cloud use cases
- Public clouds can be adapted much quicker to the demand!
- A stateless service does not store the state (non-interactive web server)
- A stateful service stores information about user interaction (a shopping cart)
 - It cannot be replicated easily!
- A cloud-friendly application is stateless and distributed
 - If there is a rise in requests I can just duplicate the system and create a new instance
- Cloud risks: security, privacy, vendor lock-in, data loss
 - In the terms of service for standard accounts there is no guarantee of continuity, reliability, etc
 - In many case cloud services are not completely mature

Databases

- Big data are typically stored in relational or non-relational databases
 - Non-relational databases require little coding skills
 - * They include NoSQL databases like MongoDB
 - Relational databases are massively parallel systems
 - * They typically require data scientists
- There is a huge amount of software in the big data ecosystem

MapReduce

- MapReduce is an algorithm developed by Google for parallel and decentralized processing and generation of big data
 - It usually uses a distributed filesystem like Hadoop
- It is based on the Map and Reduce actions
 - Map: convert a set of data into another, described by key/value tuples
 - * I can take a document and form a dictionary with single words as keys and their count in the document as value
 - Reduce: aggregate the data from a set of tuples
 - * I take several dictionaries from different documents and count the total word occurrence

Hadoop

- Apache Hadoop is an open-source software ecosystem for big data analysis and storage
 - It is designed to scale well from a single server to thousands
 - It is highly fault tolerant
 - It is typically run on low-grade hardware, but it is highly reliable because of the software-level failure handling
 - New nodes can be added quickly without changing configurations
 - It does not require any schema, it can accept any kind of data

- When a node fails the system just redirects the load on the remaining nodes, without halting
- Hadoop is composed of
 - Hadoop Common: a series of common utilities that support the other modules
 - Hadoop Distributed File System (HDFS)
 - Hadoop YARN: framework for job scheduling and resource management
 - Hadoop MapReduce: a YARN-based system for parallel processing based on the MapReduce approach
 - Hadoop Ozone: object store
 - Hadoop Submarine: machine learning engine
- The Hadoop Distributed File System (HDFS) is the core component of the Hadoop ecosystem
 - It abstracts resources and allows to see the whole HDFS as a single unit
 - It stores data in the various nodes and maintains metadata
 - It is composed of 2 core components: NameNode and DataNode
 - NameNode is the metadata server: high computational requirements but low storage requirements
 - DataNode: commodity hardware with more storage resources
 - There is typically only 1 NameNode and many DataNode
- Writing in an Hadoop cluster requires 2 essential pieces of information to be submitted with the data: block size and replication (number of copies of the block that I want)
 - Typically a block size of 64 or 128 Mb and a replication of 3 are used
 - The client sends the request to the NameNode, which returns a sorted list of DataNodes in order of distance from the client
 - The client sends the data only to the first DataNode
 - The first DataNode sends to the second and so on for the desired replication level
 - All the DataNode send a confirmation for finished writing to the NameNode
 - When all the blocks have been written the NameNode stores the file metadata
 - The NameNode returns a success message to the client
- Reading data from an Hadoop cluster
 - The client request a file to the NameNode
 - The NameNode returns a list of blocks and of DataNodes having them
 - The client retrieves the blocks from the respective DataNodes
- Node failure in Hadoop is handled by the NameNode
 - The DataNodes every 3 seconds send a HEARTBEAT message to the NameNode, signaling their presence
 - If the NameNode does not receive the HEARTBEAT from a DataNode (because it is dead or there is a network failure) it considers it dead
- Network failure handling in Hadoop
 - Whenever data is sent the receiver returns an ACK message
 - If the ACK message is not received after several trials the sender assumes the receiver to be dead
- Corrupted data in Hadoop
 - Data always include a checksum when it is sent and when it is stored
 - DataNodes periodically send a BLOCKREPORT to the NameNode, a list of all the blocks in the node
 - * Before sending the BLOCKREPORT a DataNode checks all the block checksums
 - * If a block is corrupted it is not sent to the NameNode, that assumes the missing one to be corrupted
- The NameNode always maintains 2 essential tables: a list of blocks and a list of DataNodes
- When a DataNode is dead or assumed dead, it is skept for all transactions both by client and NameNode
 - If it is so the data will not reach the desired replication, but the NameNode will subsequently instruct the DataNodes to copy data from each other accordingly
- In order to be as resilient as possible, block replicas are kept at most as 1 per datanode and 2 per server rack, if possible
 - It is also possible to specify a custom placement algorithm
- YARN performs allocates resources and schedules jobs in the Hadoop cluster
- It is composed of 2 major components: ResourceManager and NodeManager

- The ResourceManager is the central node of the cluster
 - It receives requests and dispatches them to the relevant NodeManagers
- The NodeManagers are installed onto each DataNode and manages task execution of that node
- ApplicationManagers accept job submission and forwards them to the respective ApplicationMaster on the DataNode
- Hadoop MapReduce is the core processing component of the ecosystem
 - It is a software framework for writing applications that process large datasets using distributed and parallel algorithms in an Hadoop environment