Large-scale machine-learning

ENSAI 2019-2020 Arthur Katossky & Rémi Pépin

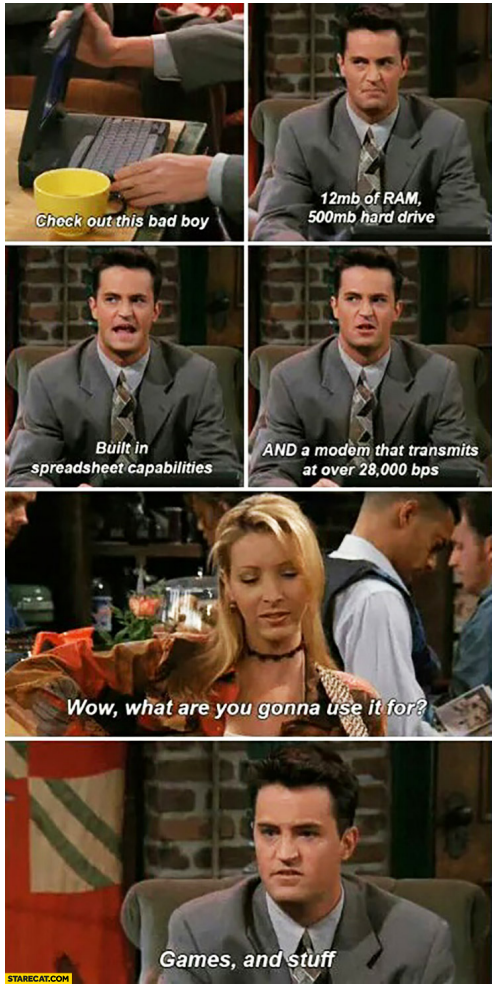
A quick history of IT hardware in enterprise

- 1940 - mid 1960 : Mainframe era
  - Commercial, scientific, and large-scale computing only.
  - Only big enterprise can buy mainframes.
  - No direct user interaction (scren, keyboard), but card punches
  - Data and computing are centralised

WarGames, John Badham, 1983

A quick history of IT hardware in enterprise

- mid 1960 - mid-1990 : Fat client era.
  - Developement of micro-computing, personnal computers become cheaper and cheaper
  - Instead of one mainframe for data and computing, one server is for the data, and computation are executed on personnal computer (**fat client**)
  - Fat client are autonomous of each other. They need the data server only to download the data



Friends, S2 E8

A quick history of IT hardware in enterprise

- mid-1990 - 2010 : In-house datacenter era.
  - mid 1990 : democratization of the Internet
  - Communication speed increase quickly
  - Centralisation of data and computing in data datacenter.
  - Personnal computer for daily tasks (mail, office etc)
  - Server for computation

Note : You can go on step further, like at Ensai, and put everything in the server

- mid-1990 - 2010 : In-house datacenter era.
  - mid 1990 : democratization of the Internet
  - Communication speed increase quickly
  - Centralisation of data and computing in data datacenter.
  - Personnal computer for daily tasks (mail, office etc)
  - Server for computation

Note : You can go on step further, like at Ensai, and put everything in the server

A quick history of IT hardware in enterprise

- mid-1990 - 2010 : Grid Computing
  - what enteprise could afford, could not be done by scientists
  - idea: using idle machines and make them work together on bigger tasks (**CPU scavenging**)
  - parallel ideas could now be applied *between* computers
  - on last Saturday (2020-01--18) the best grid's 495 820 computers was computing on average at the rate 30 petaflops (BOINC, Berkeley, link), same as the 5th fastest supercomputer
  - for comparison the fastest super computer today (Summit, IBM, link) delivers ~200 petaflops

Source: Top 500, the List (link)

A quick history of IT hardware in enterprise

- 2010 - ? : cloud computing era.

Cloud computing definition

Wikipedia : Cloud computing is **Internet-based computing**, whereby **shared resources, software, and informationare** provided to computers and other devices **on demand**, like the electricity grid. Cloud computing is a style of computing in **which dynamically scalable** and often **virtualized resources** are provided as a **service** over the Internet.

In a nutshell :

- shared physical ressources (mutalization of cost) in remote location
- Access via the Internet
- On demand scalable services

Why cloud computing ?

- The world is changing faster than ever
- Need quick reponse to new problem
- Building a new datacenter, buying new servers or create new architectures take too much time !
- The on demand nature of cloud computing make possible to create new services in no time

Why cloud computing

- Less investment : don't have to buy anything, just pay for what you use
- Easy scalable : "just" pay for more ressources
- Flexible : create and delete ressource in no time
- No maintenance cost : it's the resposability of your cloud provider
- Reliable : it's the resposability of your cloud provider
- Innovation : new ressources are accessible in no time, and you pay for what you use.

Different services

For example you want to have somewhere to sleep. You can

- Buy a site to build your house.

  -> But you have to keep everything in good condition all by yourself. Your investement is important, and before your house is built you can't sleep in your house.

- Rent a site to build a house.

  -> Your landlord keep in good condition the site and you your home. Your investemet is lower but you still have to wait for your house.

- Rent an already built empty house.

  -> Your landlord keep in good condition everything. You still have to buy a bed.

- Go to the hotel.

  -> You do not have to do anything BUT you control nothing

Different services

For example you want to have somewhere to sleep. You can

- Buy a site to build your house **In house datacenter**.

High investement, but you have full control on your infrastructure.

- Rent a site to build a house. **Infrastrucre as a Service**.

You pay for an infrastructure but you have to build up your own IT system. You have a full controll on what you do with your ressources.

- Rent an already built empty house. **Platform as a Service**.

You pay for an already installed platform, you do not manage any lower level resource management and can develop you own IT system on it

- Go to the hotel. **Software as a Servce**.

You directly use some existed IT solution. Little control on the software.

Infrastrucre as a Service, IaaS

- Provide to the consumer
  - Processing capacity
  - Storage
  - Network
  - and other fundamental computing ressources
- The consumer do not manage physical infrastructure but has control over the provided ressources
- Exemples : Amazon Elastic Compute Cloud

Platform as a Service, IaaS

- Provide to the consumer already configured infrascture to deploy it's own application
- The consumer do not manage physical infrastructure neither has control over the provided ressources

- The consumer control the deployed application

- Exemples : Amazon Relational Database Service, Amazon Elastic Map Reduce

Software as a Service, SaaS

- Provide to the consumer the capability to use a running application manage by the cloud provider
- The consumer do not control anything, excepte some user-specific settings
- Exemples : all the Google Apps

Some exemples of cloud computing : New York Times

- The need
  - conversion from TIFF to pdf of all articles from 1851 to 1922
  - 4TB of input (405,000 very large TIFF images, 3.3 million articles in SGML and 405,000 xml files mapping articles to rectangular regions in the TIFF's)
- The realization
  - Run around 100 EC2 instances to create an Hadoop cluster for 24h (estimate cost 800$)
  - Realize they made a mistake
  - Rerun the cluster with the fix for 24h (estimate cost 800$)
- Conclusion
  - Convert to pdf 70 years of newspapers for less than 2000$!

Some exemples of cloud computing : Animoto

Animoto is a cloud-based video creation service (Saas)

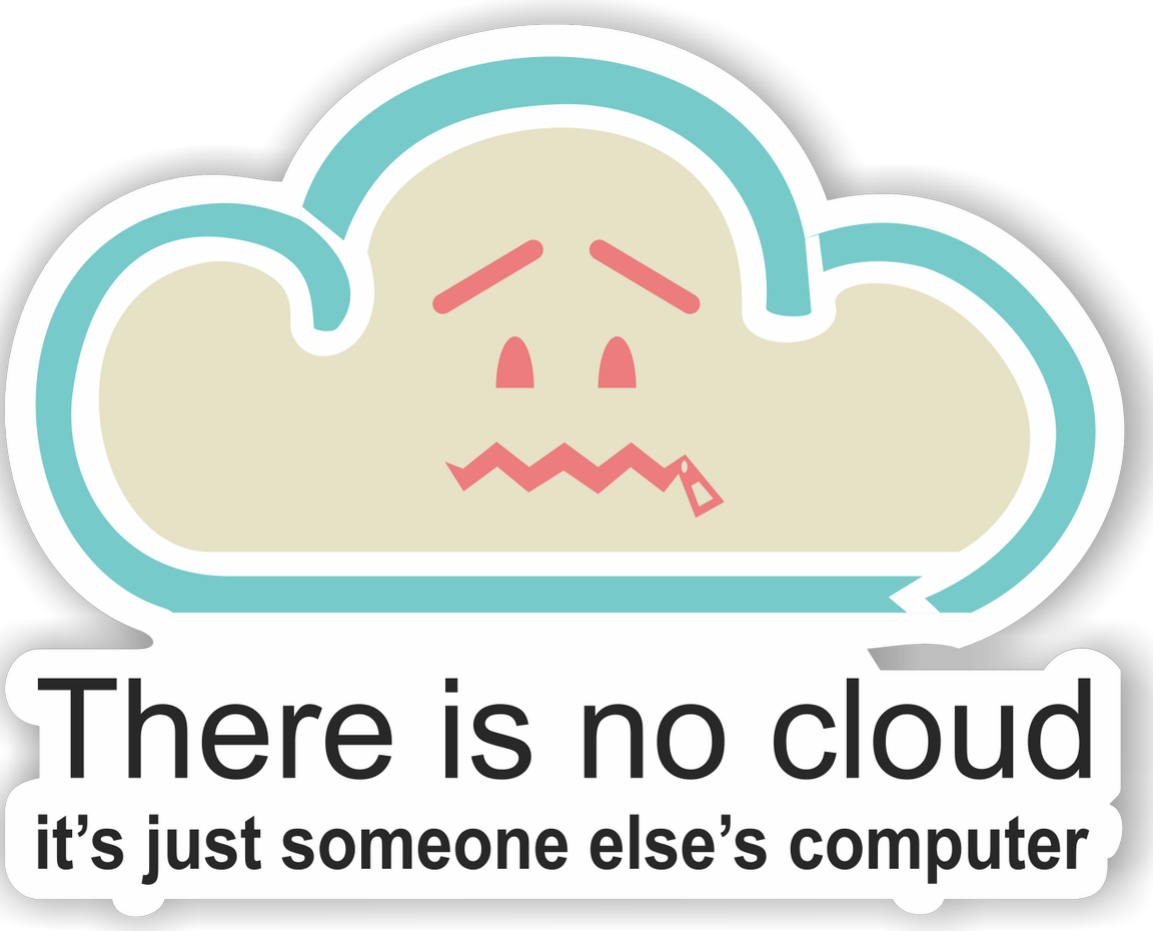
- Start-up without any physical server
- Rents servers (IaaS) to deploy its application
- Starts with 40 servers
- Scales up from 40 to 5000 servers in 4 days to meet the growing number of subscriber !

Cannot be done with on site datacenter

Some cloud providers

- Amazon web service
- Google cloud plateform
- Microsoft azure
- OVHcloud

Cloud and ecology



Cloud and ecology

- It's not because you don't own the server that the server does not pollute.
- The internet consume about 2% of the worldwide electricity.
- Theoretically, with the mutualization of ressource cloud computing could be greenner than in site datacenter
- But, because cloud computing become cheaper, more people use it, and are less carful with the cost. So more servers are running, maybe over dimensionned one, and cloud computing can me more polluting than old school datacente.
- No real evidence of that.

Principles of IaaS

Principles of IaaS

When it comes to infrastructure as a service, all cloud-providers basically share the same two basal products:

1. normal storage
2. virtual machines

Normal storage

This is the place where you put all of your files, be they programs, data or else.

You create as many "buckets" as you wish, where a "bucket" is a unit of storage.

"Normal", in "normal storage" means:

- slower access than directly from the virtual machine
- faster than recovery storage

On the other hand, the provider takes charge of preventing potential corruption of your data, by keeping usually 3 distinct copies of your data, in different racks situated in different rooms or even locations.

Each resource you store in such a way gets a unique URL, that you can refer to from inside the cloud-provider ecosystem (for instance, from a virtual machine) or, if you allow it, from the outside world. You can control who and how people get access to these resources.

Data is usually encrypted server-side (meaning that stealing physically disks from a Google server is useless) but by default, the provider stores the encryption key themselves somewhere.

Normal storage

Storage solutions have different names, but similar pricing schemes:

|----------|-----------------|------------|----------------------|-----------| | Provider | Service's name | Store | Write | Read* | |---------|-----------------|------------|----------------------|-----------| | Google | Cloud Storage | 0,03 \$/Go | free | free | | Amazon | S3 | 0,02 \$/Go | free | 0,09 \$/Go | | Azure | Blob Storage | 0,02 \$/Go | 0,00 \$ / 10 000 op. | 0.02 \$ / 10 000 op. | | OVH | Object Storage | 0,01 €/Go | free | 0,01 €/Go | |----------|-----------------|------------|----------------------|-----------|

Prices are given monthly.

- Local reads are made locally inside *one* provider's cluster. For instance, you may be charge extra for an external reader, or for reading data from a cluster situated in an other place (ex: reading data sotred in Europe from a virtual machine located in Australia).

Sources: GCP (link), AWS (link), Windows (link)OVH (link), visited on 2020/01/18.

Normal storage

Other forms of storage, beyonf "normal" storage, include:

- disks that can be "attached" to a virtual machine (faster, but more expensive)
- storage whose physical location adapts to usage
- so-called "cold" storage for unfrequently-accessed storage (less expensive, but you pay reads and writes more)
- archive solution (inexpensive, but takes min. to hours for access)

Virtual machines

The other component of a typical IaaS environement are **virtual machines** (VMs).

A VM is made of a combination of components:

- memory
- processing units
- local storage (hard disk)
- an operating system (FR: système d'exploitation), typically Linux

The VM is... **virtual** which does **not** mean that there is no physical memory, physical processing units or physical storage, but rather that your virtual computer has rights over pooled ressources situated on neighbooring servers of the same cluster.

VMs come in a fixed (but high) number of pre-configured options. Some providers offer tailored configuration, but they are usually more expensive that the closest pre-configured option.

Virtual machines

Virtual machines (VMs) come with little pre-configured software, and *you* are the one to perform all the rest of the configuration. You may use such machines to

VMs **may be launched and stopped for specific tasks**. Staticians are typically concerned by this king of virtual machines, for instance for training one model, for exploring one large dataset, for using software that need specific configuration, etc. **VMs may be launched and stopped programatically.**

VMs, on the other hand, **may also be used continuously**. Think of a mail server, a live trafic visualisation, a back-end for a web-service, etc. Statisticians are less often concerned with this kind of usage.

BEWARE! Holding the pysical resource and doing nothing with it comes at a cost for the provider. Thus, **you will pay for a VM even you do not use it.** It is a good reflex to shut down all your VMs before quitting a session.

Virtual machines

|----------|------------------|-------------|-----------|------------| | Provider | Service's name | Processing / *Memory* / ***Storage*** | |----------|------------------|-------------|-----------|------------| | Google | Compute Engine | ~0,03 \$ | ~0,004 \$ | ~0,04 \$ | | Amazon | EC2 | ~0,05 \$ | ~0,005 \$ | ? | | Azure | Virtual Machines | ~0.05 \$ | ~0.005 \$ | ~0.00 \$ | | OVH | Public Cloud | ~0,04 € | ~0,002 € | ~0,00 € | |----------|------------------|-------------|-----------|------------|

- (Average) cost per processor (CPU or GPU) per hour **(Average) cost per Go (CPU or GPU) per hour *** (Average) cost per Go per month

Sources: GCP, Iowa cluster (links for processing and memory and for storage) ; AWS, London cluster, own calculation (link) ; Azure, France South cluster (link) ; OVH, Gravelines vluster, own calculation (link) ; all visited on 2020/01/18.

Virtual machines

Depending on the providers, you may have extra features such as:

- preenptible machines / spot instances (for tasks you don't need to run at a specific moment, you can get much cheaper machines that can be stopped at any time, and relaunched later)
- redundancy (you get exact copies of a given VM) ; this may be useful to scale a process such as a web server
- specialised VMs such as high CPU-to-memory ratio (e.g. for batch processing) or on the contrary high memory-to-CPU ratio (e.g. for in memory computation) or tailored configurations

Infrastructure as a service: billing

BEWARE! All what you book is billed, even what you end up not using! If there is something you do not use, *you* have to shut it down. It is good practice to start small, and scale up when you need.

Also, in order for customers to mind their architectural choices, **most provider will bill communication between you virtual machines and your storage spaces** — unless sometimes if they are situated in precisely the same cluster.

Other services

There are other services of pure infrastructure that the cloud providers offer, but that you won't have to worry about as a statistician:

- network
- security
- orchestration

Platform as a service

Besides storage and VMs, you will probably get to use services, without having to install and configure yourself a computer from the bottom up. This is **platform as a service** (**PaaS**).

As a statician, services you may come to use are:

- databases
- data analytics
- machine-learning
- serverless computing

Pros: scale automatically, less maintenance Cons: less configurable, possibly more expensive