

# FAIR\_bioinfo for bioinformaticians

## Introduction to the tools of reproducibility in bioinformatics

C. Hernandez<sup>1</sup> T. Denecker<sup>1</sup> J.Sellier<sup>2</sup> C. Toffano-Nioche<sup>1</sup>

<sup>1</sup>Institute for Integrative Biology of the Cell (I2BC)  
UMR 9198, Université Paris-Sud, CNRS, CEA  
91190 - Gif-sur-Yvette, France

<sup>2</sup>Institut de Génétique et de Biologie Moléculaire et Cellulaire (IGBMC)  
CNRS UMR 7104 - Inserm U 1258  
67404 - Illkirch cedex, France

Sept. 2020



## A (not-so-uncommon) nightmare



What changed?

## A (not-so-uncommon) nightmare






## What changed?

- Software version
- Libraries version
- OS version
- ..?

# Different levels of encapsulation

Goal : capture the system environment of applications (OS, packages, libraries, . . . ) to control their execution.

- Hardware virtualisation (virtual machines) 
- OS virtualisation (images and containers) 
- Environment management  **CONDA**

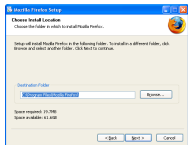
# Encapsulation

Let's say we want to install Firefox...

Windows

MacOS

Unix-based



```
ubuntu@ubuntu:~$ sudo apt-get install firefox
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  fonts-lyx
The following packages will be upgraded:
  firefox
1 upgraded, 0 newly installed, 0 to remove and 696 not upgraded.
Need to get 42.0 MB of archives.
After this operation, 88.3 MB of additional disk space will be used.
Get:1 http://security.ubuntu.com/ubuntu/ trusty-security/main firefox 1386-44.0
i-buld2-@ubuntu14.04.1 [42.0 MB]
1% [3] firefox 579 kB/42.0 MB 3% 75.0 kB/s Delta 12s
```

# Encapsulation

We started with a computer using a specific OS...

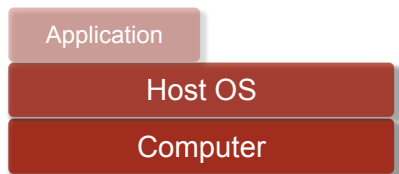


Host OS

The diagram consists of two stacked, rounded rectangular boxes. The top box is dark red and contains the text 'Host OS'. The bottom box is a slightly lighter shade of red and contains the text 'Computer'. Both boxes have a subtle drop shadow effect.

Computer

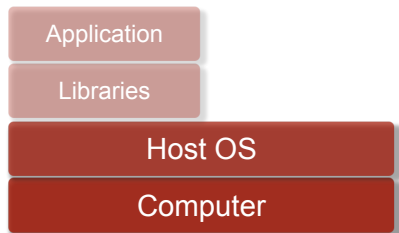
# Encapsulation



We started with a computer using a specific OS...

And inside this environment, we installed a new application.

# Encapsulation



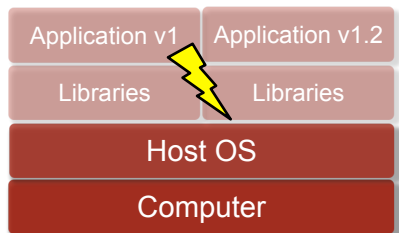
We started with a computer using a specific OS...

And inside this environment, we installed a new application.

Applications rely on dependencies, e.g. external libraries.

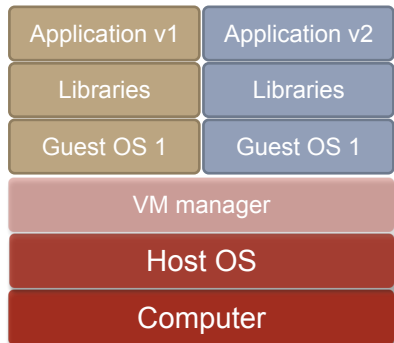


# Encapsulation



Usually dependencies of different applications don't interfere.  
But what if we want to test the latest version of our favourite tool?  
There might be conflicts. . .

# Encapsulation : hardware virtualisation

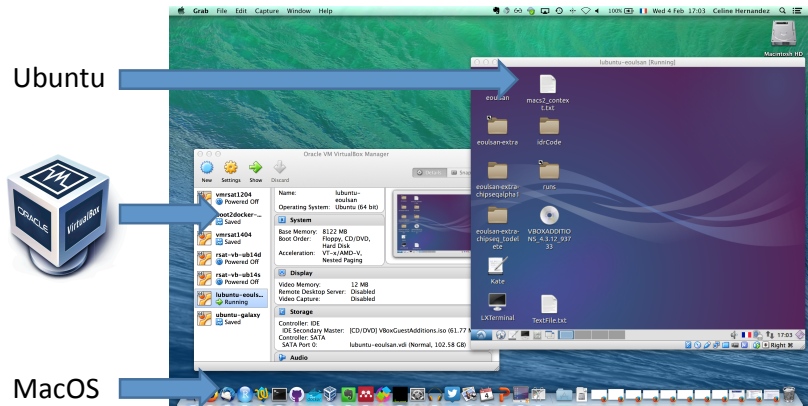


Idea: use virtual machines

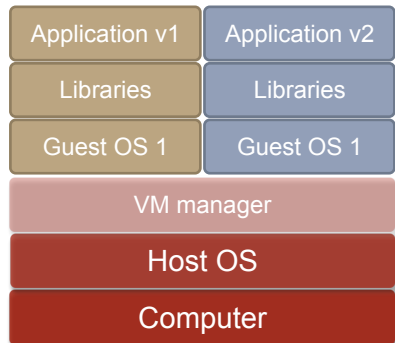
Pros:

- Each application gets a completely different and independent environment
- Virtual machines can be transferred to another computer (using the same manager)

# Encapsulation : hardware virtualisation



# Encapsulation : hardware virtualisation



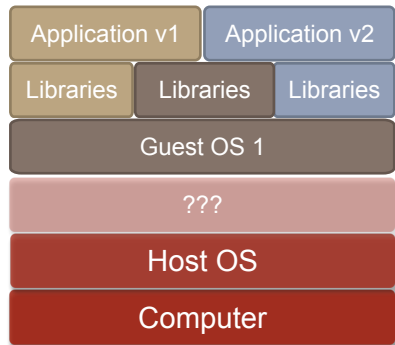
Idea: use virtual machines

Pros: transferable independent environments

Cons:

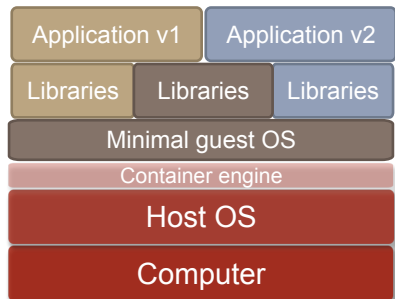
- Redundancy between VMs
- Heavy to set up
- No automation

# Encapsulation : OS virtualisation



Idea: "trick" applications into believing that they are in a different OS than the host's  
Avoid redundancy.

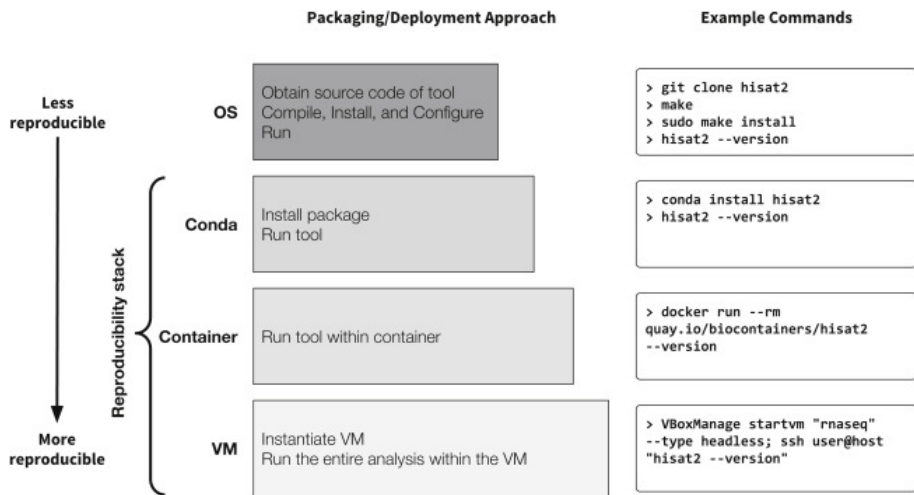
# Encapsulation : OS virtualisation



Idea: "trick" applications into believing that they are in a different OS than the host's  
Avoid redundancy.



# Encapsulation : OS virtualisation



Practical Computational Reproducibility in the Life Sciences - Björn Grüning et al (2018)



# What is Docker?

Docker is not very “old”

- First commit January 2013
- First version March 2013
- Version 1.0 in June 2014

But its adoption was fast

- Officially packaged in Ubuntu since 2014 (v14.04)



# What is Docker?

Image



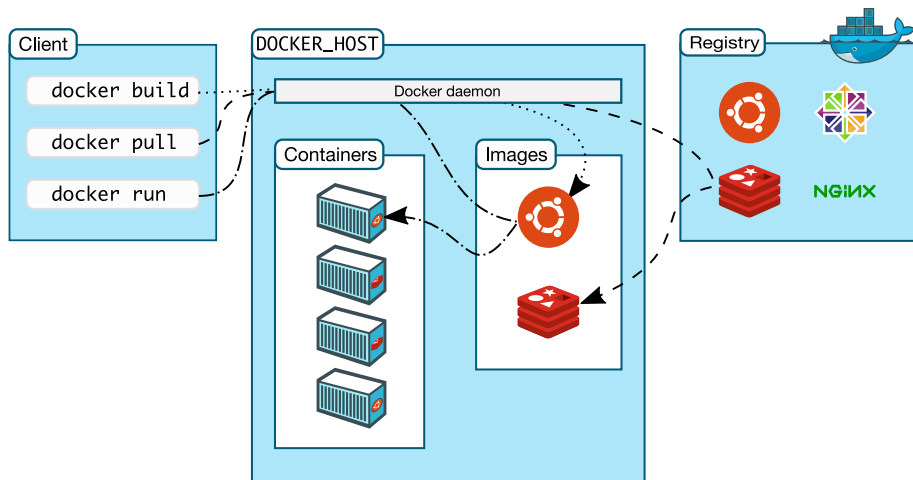
- Set of libraries and functions
- Fixed. Cannot be modified
- Can be stored/shared online
- Can be automatically built

Container



- "Active image"
- Can be modified (interactive)
- Can be turned into an image
- One image, many containers

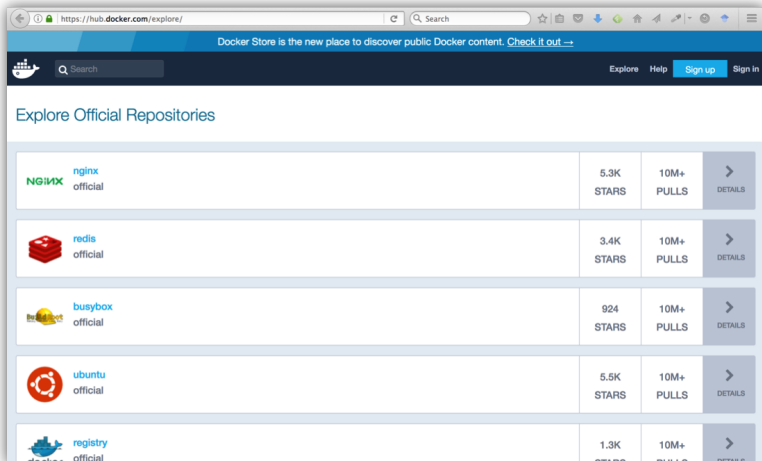
# What is Docker?



(<https://docs.docker.com/get-started/overview/>)

# What is Docker?

## DockerHub



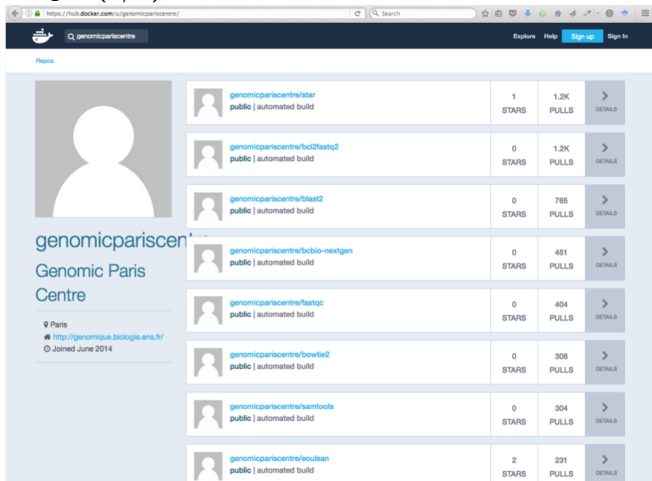
The screenshot shows the DockerHub website interface. At the top, there is a navigation bar with the text "Docker Store is the new place to discover public Docker content. Check it out →". Below this is a search bar and navigation links for "Explore", "Help", "Sign up", and "Sign in". The main content area is titled "Explore Official Repositories" and displays a list of five repositories in a table format.

Repository Name	Stars	Pulls	Action
nginx official	5.3K STARS	10M+ PULLS	DETAILS
redis official	3.4K STARS	10M+ PULLS	DETAILS
busybox official	924 STARS	10M+ PULLS	DETAILS
ubuntu official	5.5K STARS	10M+ PULLS	DETAILS
registry official	1.3K STARS	10M+ PULLS	DETAILS

(<https://hub.docker.com/>)

# What is Docker?

## Usermade images (1/2)



The screenshot shows the Docker Hub profile for 'genomicpariscentre'. The profile includes a bio: 'Genomic Paris Centre', location 'Paris', website 'http://genomique.biologie.ens.fr', and 'Joined June 2014'. Below the bio is a table of Docker images:

Image Name	Stars	Pulls
genomicpariscentre/star public   automated build	1	1.2K
genomicpariscentre/bol2fastq2 public   automated build	0	1.2K
genomicpariscentre/blast2 public   automated build	0	765
genomicpariscentre/bcbio-nextgen public   automated build	0	451
genomicpariscentre/fastqc public   automated build	0	404
genomicpariscentre/bowtie2 public   automated build	0	308
genomicpariscentre/samtools public   automated build	0	304
genomicpariscentre/eoulsan public   automated build	2	231

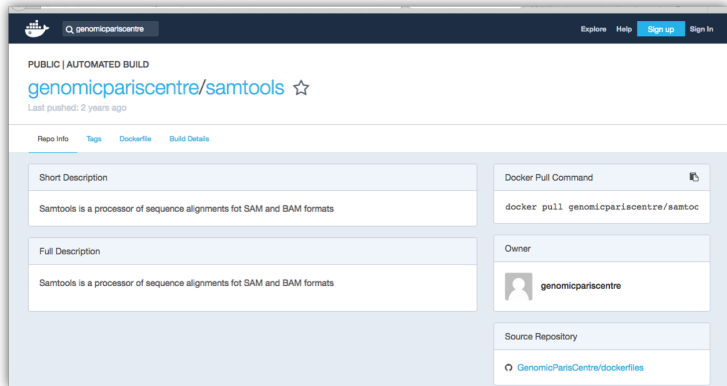
(url<https://hub.docker.com/u/genomicpariscentre/>)



# What is Docker?

Usermade images (2/2)

Be critical!

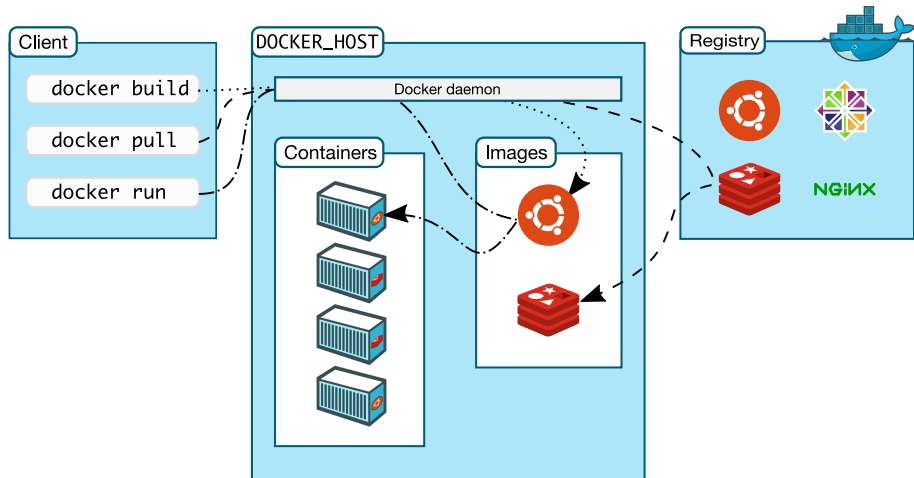


The screenshot shows the Docker Hub interface for the repository `genomicpariscentre/samtools`. The page is titled "PUBLIC | AUTOMATED BUILD" and includes a search bar with the text "genomicpariscentre". Navigation links for "Explore", "Help", "Sign up", and "Sign in" are visible in the top right. The repository name is displayed in large blue text with a star icon, and it notes "Last pushed: 2 years ago". Below the repository name are tabs for "Repo Info", "Tags", "Dockerfile", and "Build Details". The main content area is divided into two columns. The left column contains a "Short Description" and a "Full Description", both stating: "Samtools is a processor of sequence alignments for SAM and BAM formats". The right column contains a "Docker Pull Command" section with the command `docker pull genomicpariscentre/samtools`, an "Owner" section showing the profile of "genomicpariscentre", and a "Source Repository" section with a link to "GenomicParisCentre/dockerfiles".

(<https://hub.docker.com/r/genomicpariscentre/samtools/>)



# What is Docker?



(<https://docs.docker.com/get-started/overview/>)

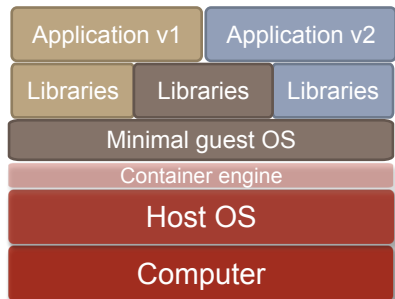
# What is Docker?

Other commands :

- `docker images` : list images available locally
- `docker ps` : status of containers
- `docker rm` : delete a container
- `docker rmi` : delete an image
- ...

(More details during the practical session.)

# Encapsulation : OS virtualisation



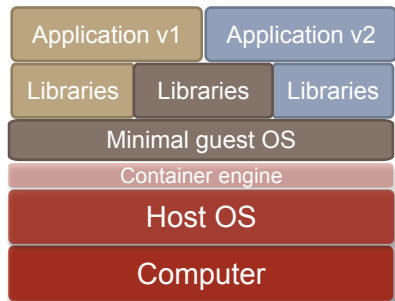
## OS virtualisation vs hardware virtualisation

### Pros:

- Speed
  - ▶ Installation is faster
  - ▶ No boot time
- Lightweight
  - ▶ Minimal base OS
  - ▶ Minimal libraries and application set
- Easy sharing of applications



# Encapsulation : OS virtualisation



## Cons:

- Needs root access (Singularity)
- Changes of policies of the Docker company

## Update of the Docker Image retention policy (13/08/2020)

### What is a container image retention limit and how does it affect my account?

Image retention is based on the activity of each individual image stored within a user account. If an image has not either been pulled or pushed in the amount of time specified in your subscription plan, the image will be tagged "inactive." Any images that are tagged as "inactive" will be scheduled for deletion. Only accounts that are on the **Free** individual or organization plans will be subject to image retention limits. A new dashboard will also be available in Docker Hub that offers the ability to view the status of all of your container images.

### What are the new container image retention limits?

Docker is introducing a container image retention policy which will be enforced starting November 1, 2020. The container image retention policy will apply to the following plans:

- Free plans will have a 6 month image retention limit
- Pro and Team plans will have unlimited image retention

<https://www.docker.com/pricing/retentionfaq>

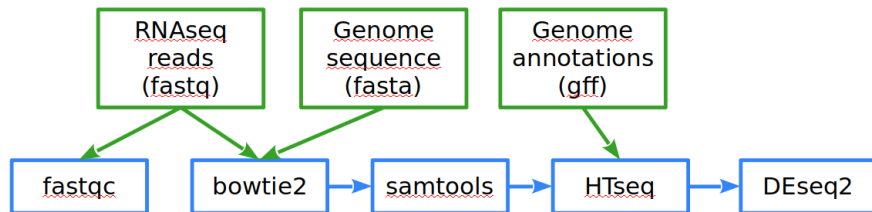
# Practical session

Practical session : Docker and Samtools.  
See companion document.



# Practical session

## Analysis workflow



green=input, blue=tool

**fastqc** control quality of the input reads

**bowtie2** reads mapping on the genome sequence

**samtools** mapped reads selection & formatting

**HTseq** count table of mapped reads on genes (annotations)

**DEseq2** statistical analysis: genes list having differential expression

# Practical session

## Savoir FAIRe

- (Installation de Docker)
- Learn the structure of a Docker command
- Pull a pre-defined image available on the DockerHub
- Start a container
- Bonus: build a Dockerfile