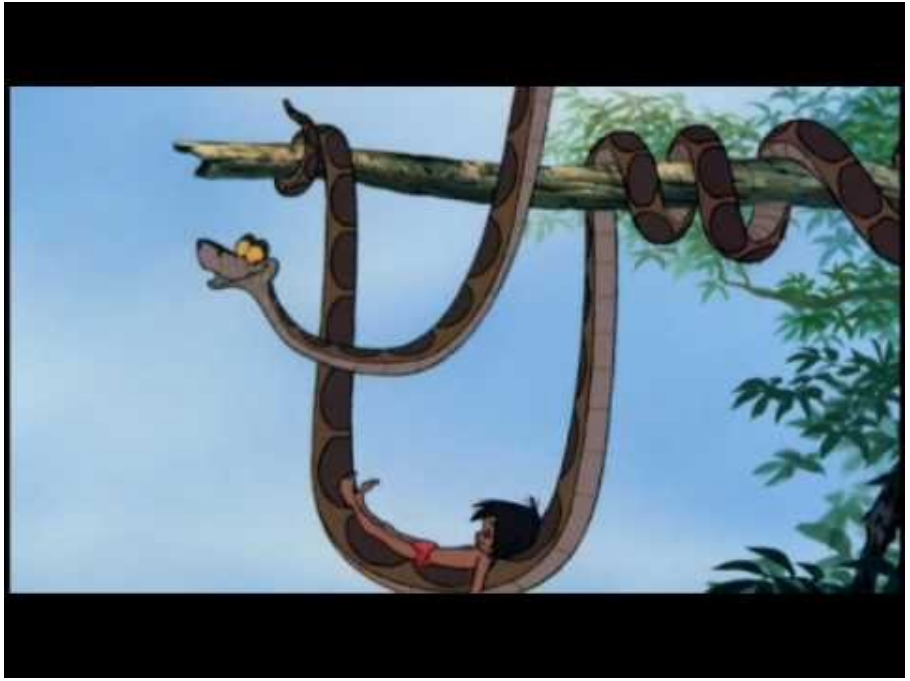
« Easy » bioinformatics with (ana/mini) CONDA



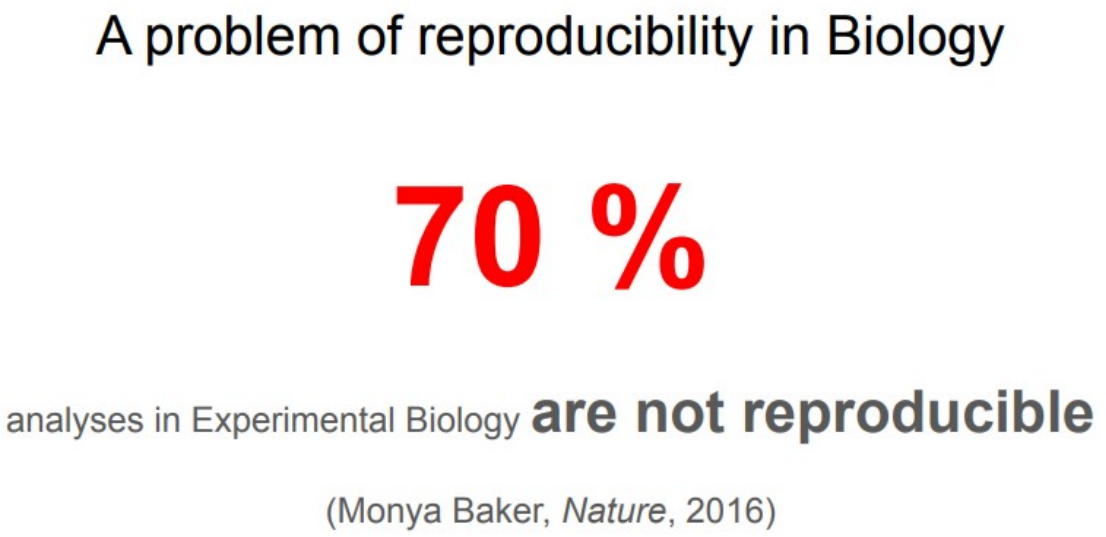
Club Bioinfo – Université Paris Diderot – Université de Paris
Olivier Kirsh, MCF, EDC-UMR 7216. 2019-09-19
https://tinyurl.com/yy4d8vyb

The FAIR data principles

Findable **A**ccessible **I**nteroperable **R**eusable

Credits : T .Denecker

- Last Thomas Denecker’s presentation :



http://www.normalesup.org/~vorgogoz/BioInfoCourses/2019-06-BioInfo-Reproducibility-Denecker.pdf

Because material and methods are (were ?) not FAIR

Gene expression data processing and normalization. The publicly available data sets with CRC tumor samples from the Gene Expression Omnibus (**Supplementary Table 3**) were normalized using the robust multi-array average (RMA) method as implemented in the 'affy' package[39]. Ove… …les in GSE14333 and GSE17536 were excluded from GSE… …and network analysis and training a consensus subtype classifier. … …and public Affymetrix data sets were renormalized using the single-sample frozen RMA method[33] as implemented in the 'frma' package for R/Bioconductor.

…mor sets were analyzed on full genome Agilent microar… …ara). Samples were hybridized against a common CRC reference pool, and … genome data was normalized using loess and local background subtraction ('limma' package). Details about sample processing and microarray analysis can be found in ref. 3.

Level 3 TCGA RNA-seq data for colon and rectal was downloaded from the TCGA data portal (January 2014). RSEM-normalized data[40] was further log-transformed, and non… …removed. … …onent analysis (PCA) revealed … …een … …es (data not shown), and sam… …thout … …showed a strong separation between genome analyzer (GA) and HiSeq samples and was batch-corrected using the ComBat method[32].

Versions? Parameters?

Versions? Parameters?

log2(n)?, log10(n)?, log(n+1)?, etc.

Versions? Parameters?

Could you redo the analysis ?

And even if you have a detailed protocol (workflow)

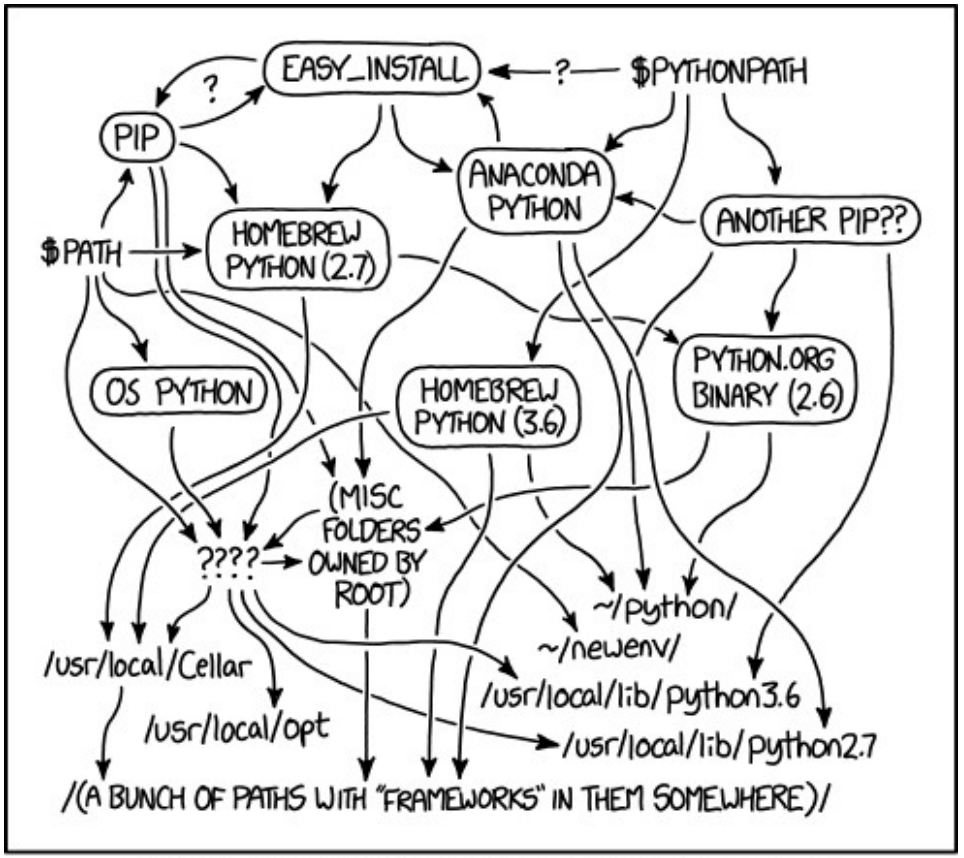
Problems still too frequent

- **Impossibility to install tools**
  - OS not compatible
  - Dependency no longer available
- **Update of the tool rendering the codes unusable**
  - Python 2 and Python 3 !
  - Changing the arguments of the functions used (R)
- **Impossibility to reproduce the results of the computational analysis**
  - IDE : stable version of the language different according to the OS (Rstudio)
  - Package versions

Credits : T .Denecker

Most of the time we don’t have skills to solve these problems

And if you think you have skills, your configuration can look like this



MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

Not really FAIR

Credits : xkcd

For a given biological question addressed with bioinformatics.

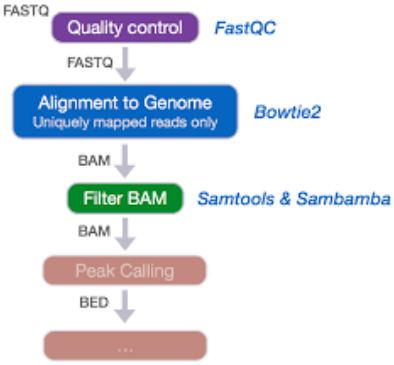
You need the right :

ressources **tools** **methods**

BOW TIE

STAR 2.7

HISAT2



Each choice has concequences

Same problems in wet biology

Many choices and solutions to set up your working environment

??

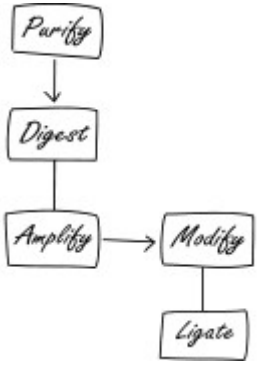
Life
Roche
Agilent
Qiagen
Applied
...

NEW ENGLAND BioLabs Inc.

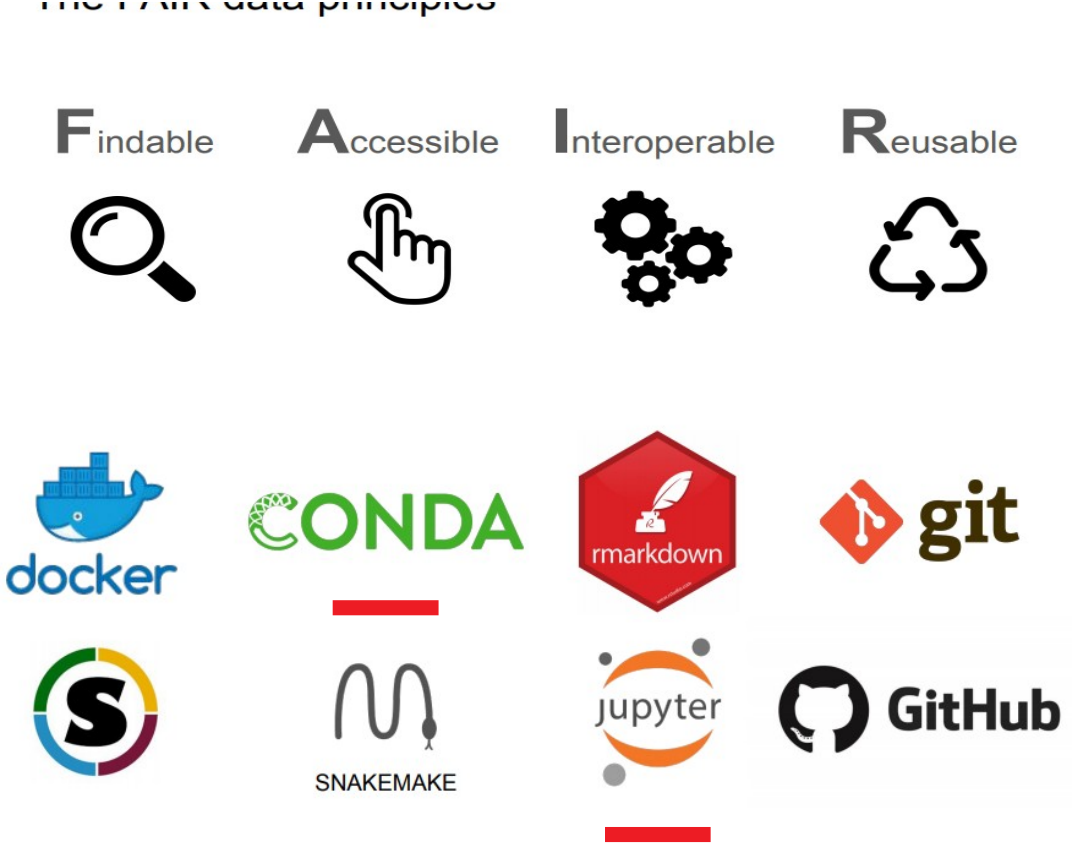
ThermoFisher SCIENTIFIC

Promega

Power supply
Waste
Gaz
Fluids
Etc...



How to fit the FAIR data principles in bioinformatics ?



Credits : T .Denecker

Conda (package manager)

From Wikipedia, the free encyclopedia

Conda is an open-source,[1] cross-platform,[2] language-agnostic package manager and environment management system. It was originally developed to solve difficult package management challenges faced by Python data scientists, and today is a popular Python/R package manager.[3][4] It is released under the Berkeley Software Distribution License by *Anaconda Inc*.[1][5][6][7][8][9]

- To install / desinstall programs
- To manage software environments
- To deal with dependancies (and OS)
- To manage program versions
- To get and distribute software environments
- To keep a « clean » machine
- ….

CONDA

Learn

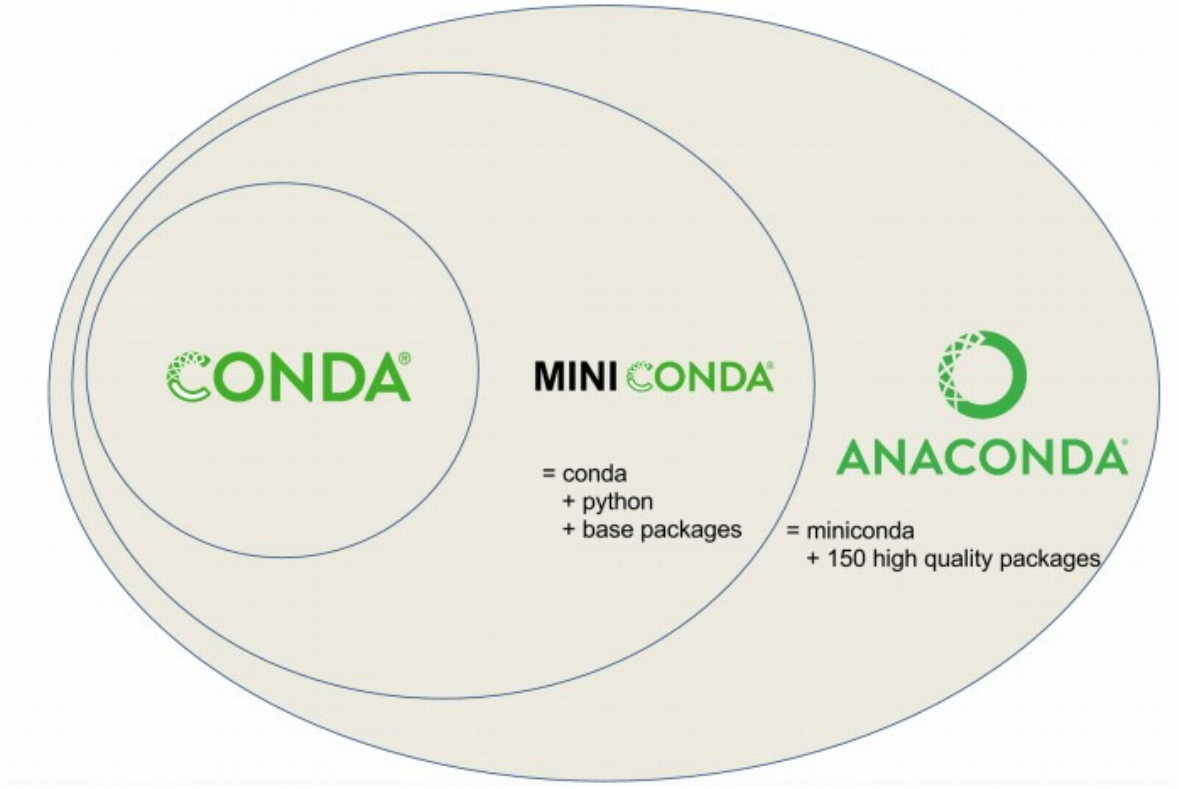
Try & developp

Keep tools accessible

Easy Uninstall

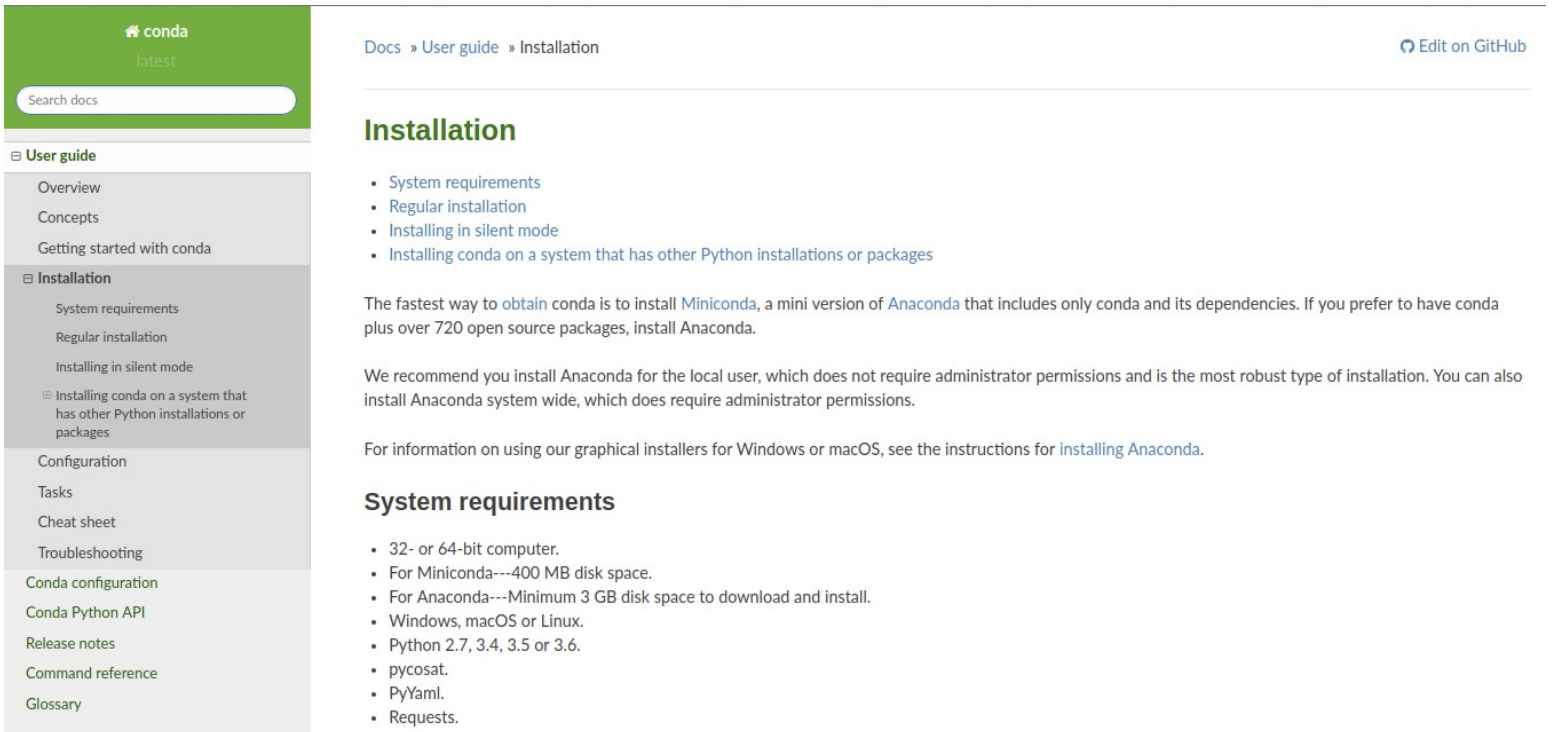
Which CONDA do we need?

Conda : package and environment manager for any language



Miniconda installation

Follow instructions https://conda.io/projects/conda/en/latest/user-guide/install/index.html



conda
latest

Search docs

- User guide
  - Overview
  - Concepts
  - Getting started with conda
  - Installation
    - System requirements
    - Regular installation
    - Installing in silent mode
    - Installing conda on a system that has other Python installations or packages
  - Configuration
  - Tasks
  - Cheat sheet
  - Troubleshooting
- Conda configuration
- Conda Python API
- Release notes
- Command reference
- Glossary

Docs » User guide » Installation

Edit on GitHub

Installation

- System requirements
- Regular installation
- Installing in silent mode
- Installing conda on a system that has other Python installations or packages

The fastest way to obtain conda is to install Miniconda, a mini version of Anaconda that includes only conda and its dependencies. If you prefer to have conda plus over 720 open source packages, install Anaconda.

We recommend you install Anaconda for the local user, which does not require administrator permissions and is the most robust type of installation. You can also install Anaconda system wide, which does require administrator permissions.

For information on using our graphical installers for Windows or macOS, see the instructions for installing Anaconda.

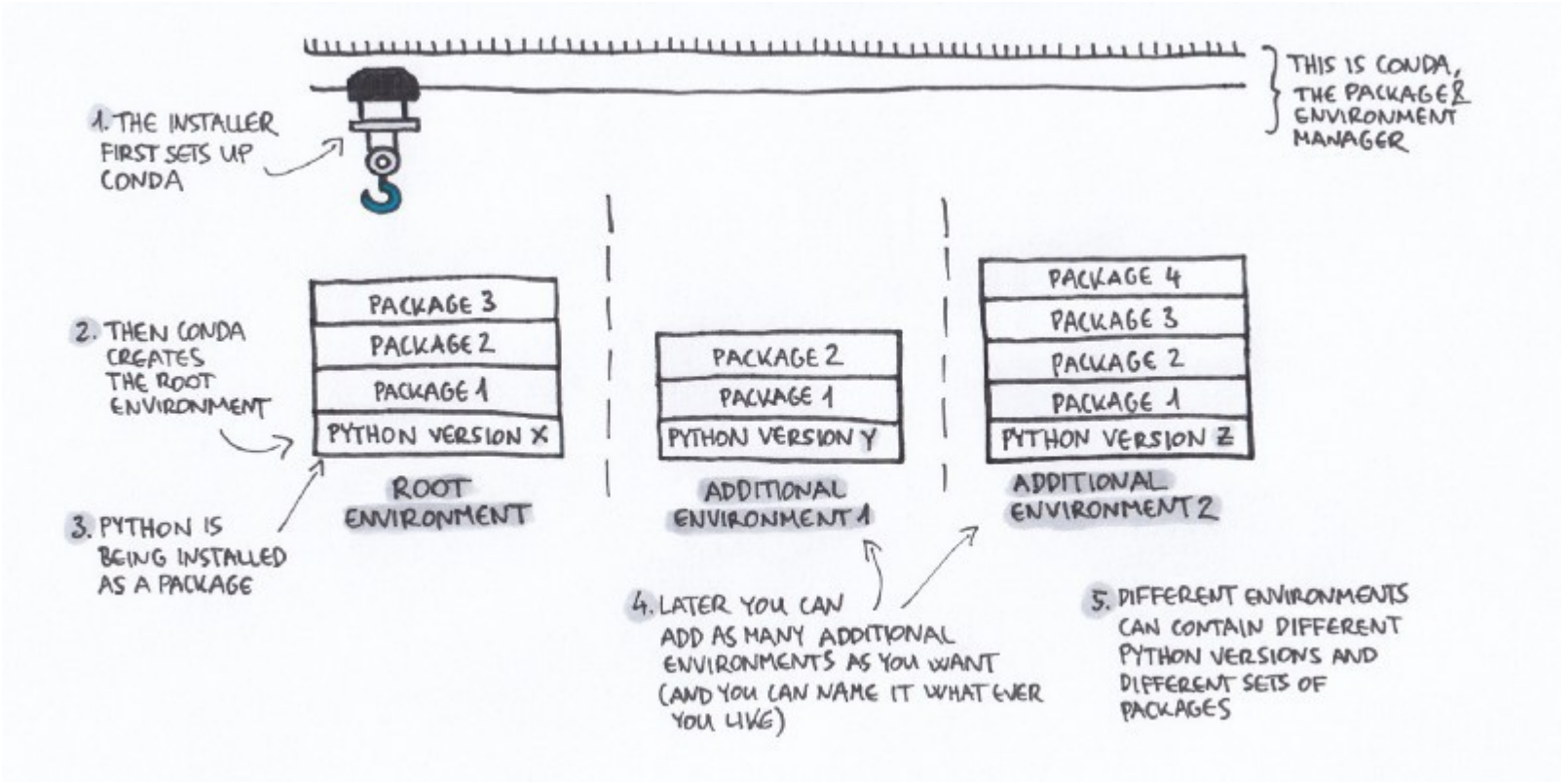
System requirements

- 32- or 64-bit computer.
- For Miniconda---400 MB disk space.
- For Anaconda---Minimum 3 GB disk space to download and install.
- Windows, macOS or Linux.
- Python 2.7, 3.4, 3.5 or 3.6.
- pycosat.
- PyYaml.
- Requests.

Miniconda installation

Go to https://docs.conda.io/en/latest/miniconda.html and download installer file

HOW CONDA WORKS ?



Credits : https://www.freecodecamp.org/news/why-you-need-python-environments-and-how-to-manage-them-with-conda-85f155f4353c/

Hands-on

- We will use a Docker container which run linux and contains the required files

  - Pros :
    - What is done in docker stays in docker !
    - Everybody share the same environment
  - Cons :
    - You need to install Docker !

Launch Docker

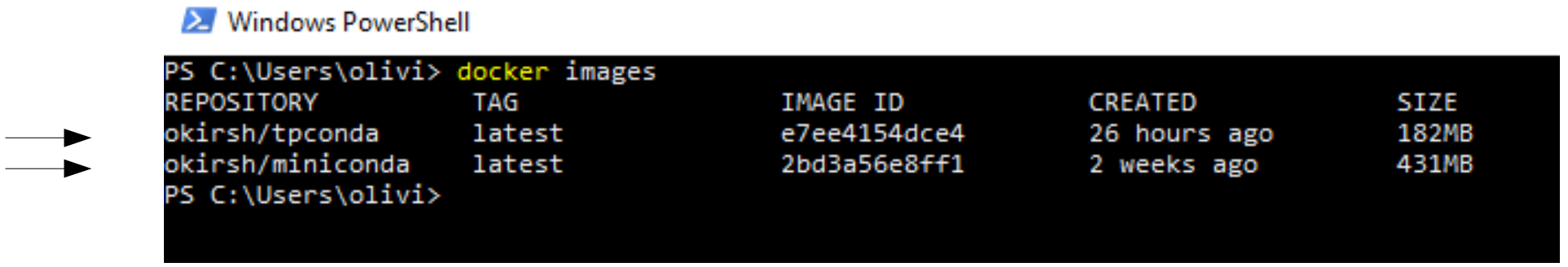
- PC : click on Docker-desktop icon
- MAC : click on Docker-desktop icon
- Linux : wait

Open a terminal

- PC : type « cmd » or « powershell » in search bar
- MAC : Control + Option + Shift + T
- Linux : Ctrl + Alt + t

Check docker images

docker images



It should be like me If you read the mail and did your homework !

Start a linux container

`docker run -it okirsh/tpconda`

Your machine env

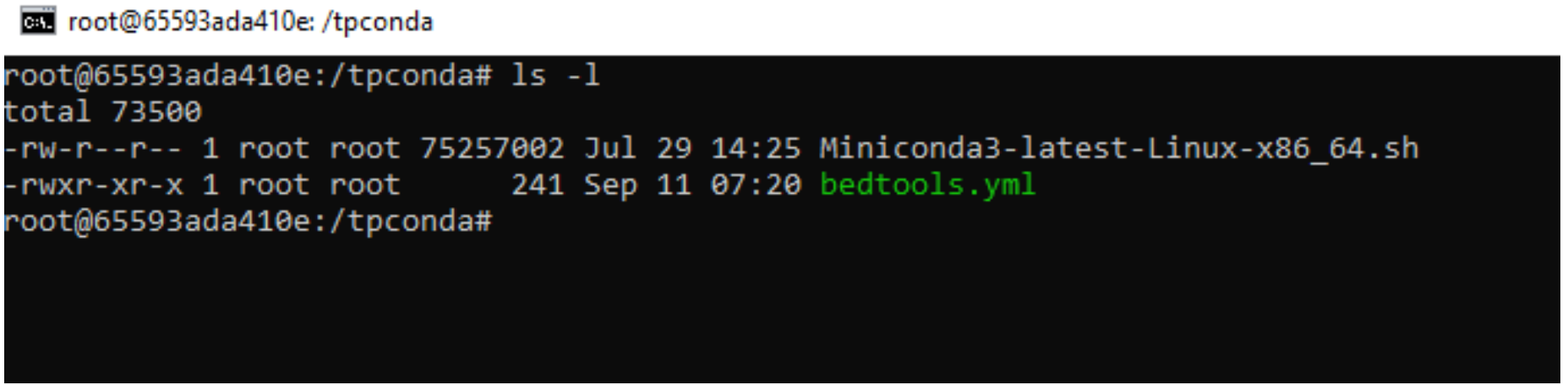
Inside docker container

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\olivi> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
okirsh/tpconda      latest              e7ee4154dce4        7 days ago          182MB
okirsh/miniconda    latest              2bd3a56e8ff1        3 weeks ago         431MB
PS C:\Users\olivi> docker run -it okirsh/tpconda
root@535c2857d260:/#
```

Install miniconda

`cd /tpconda`

`ls -l`

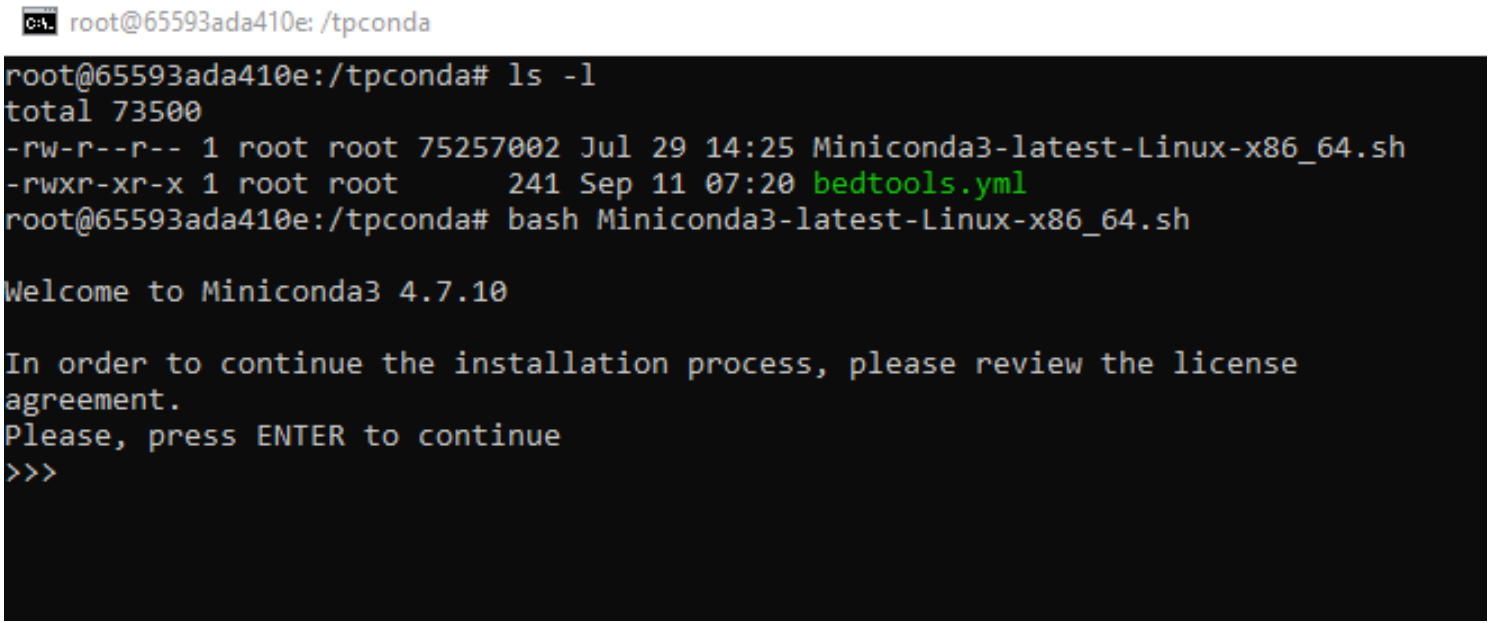


Install miniconda

`bash Miniconda3-latest-Linux-x86_64.sh`

Tip : use tab completion !! no need to type everything

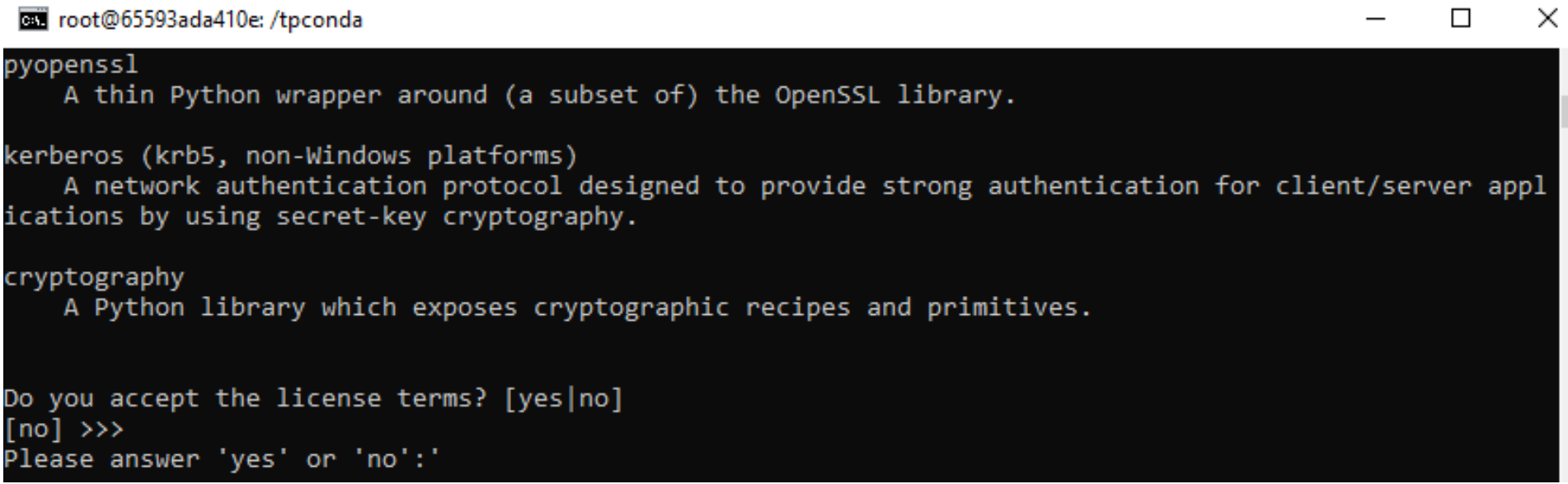
And press ENTER



Install miniconda

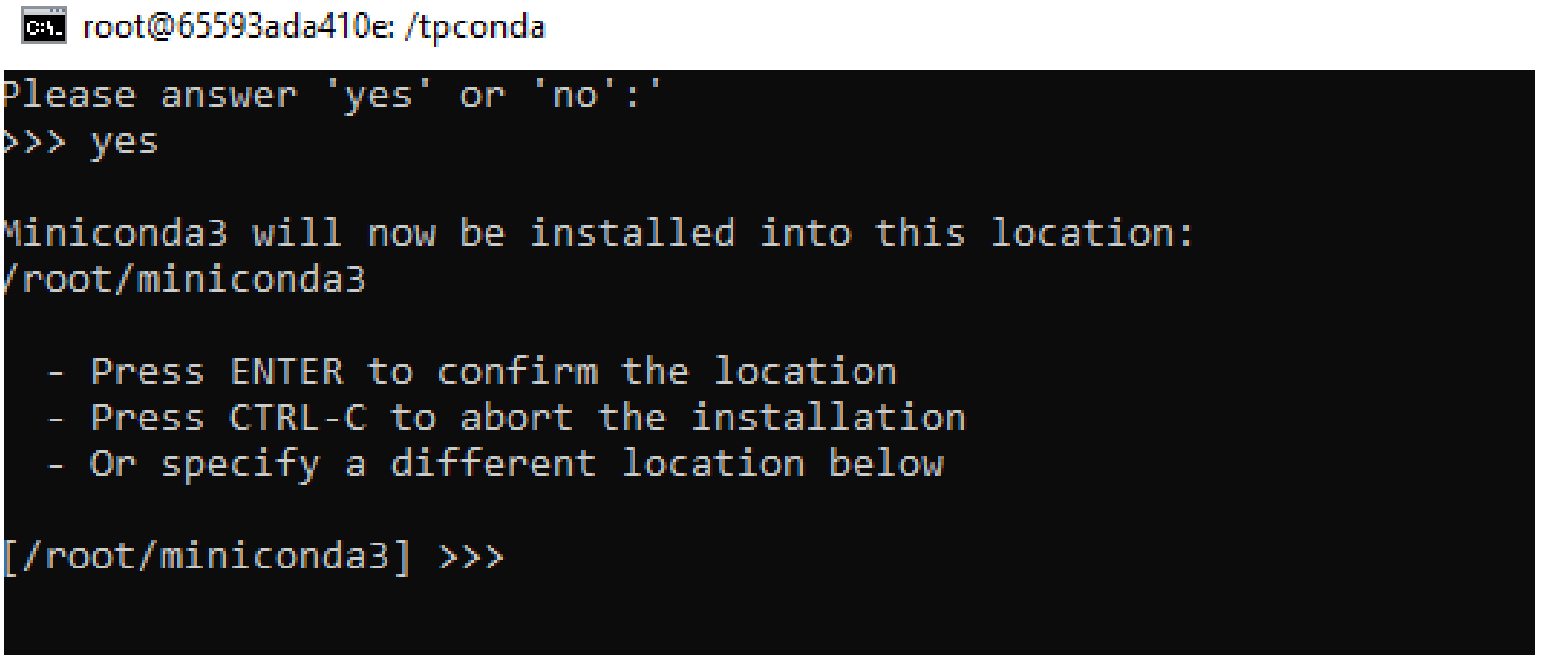
Read the license terms ...

yes



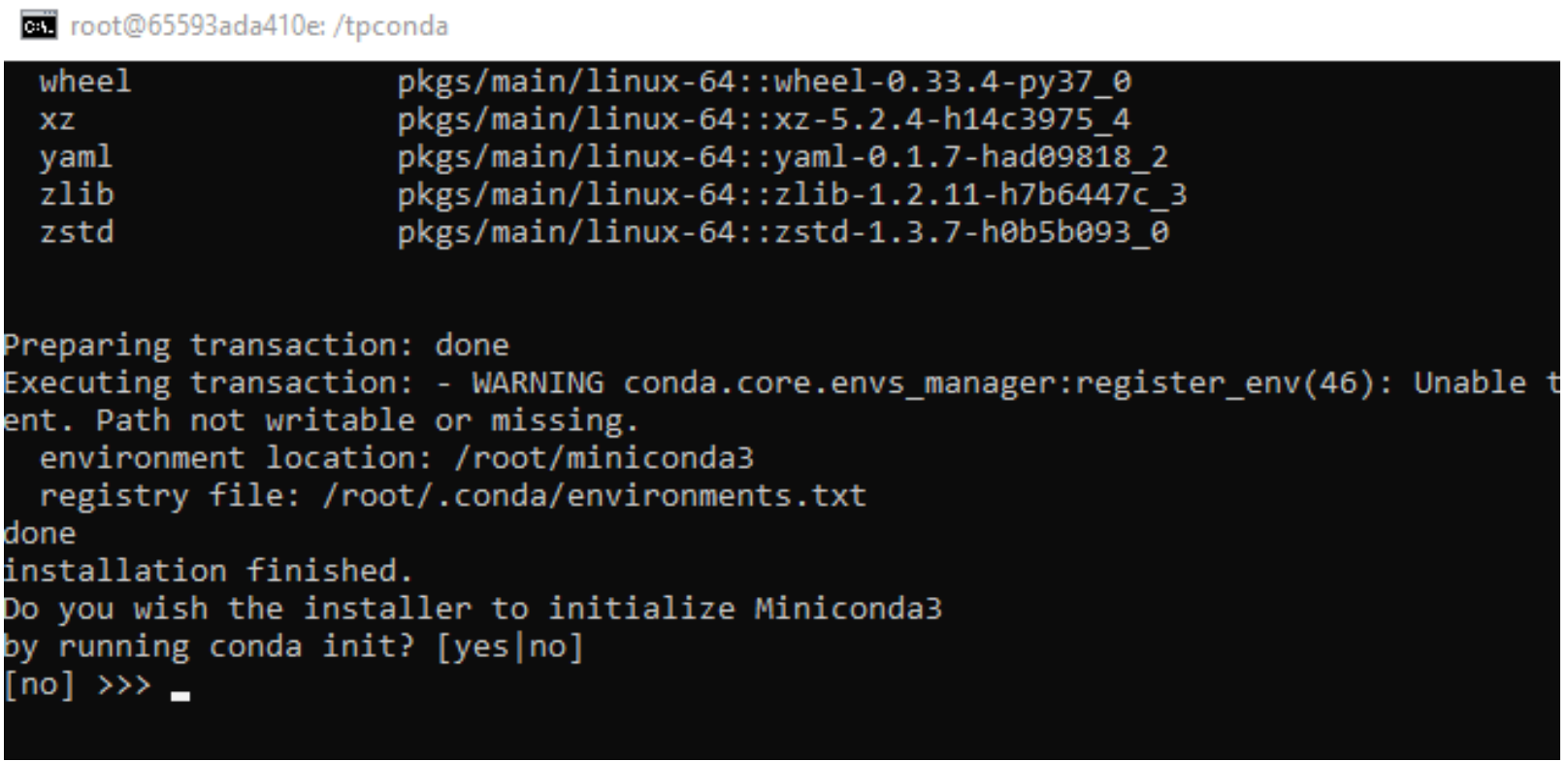
Install miniconda

Press ENTER

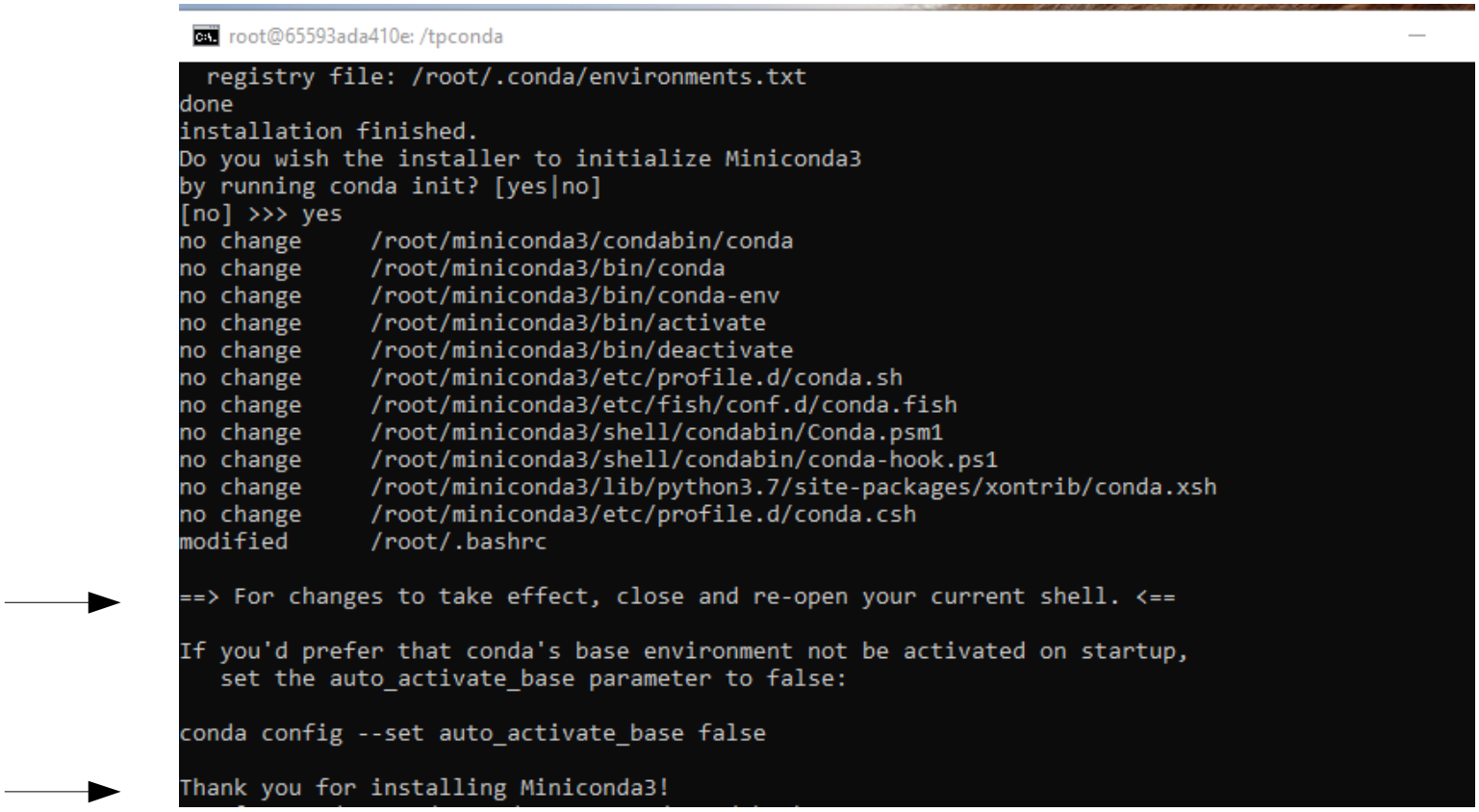


Install miniconda

yes

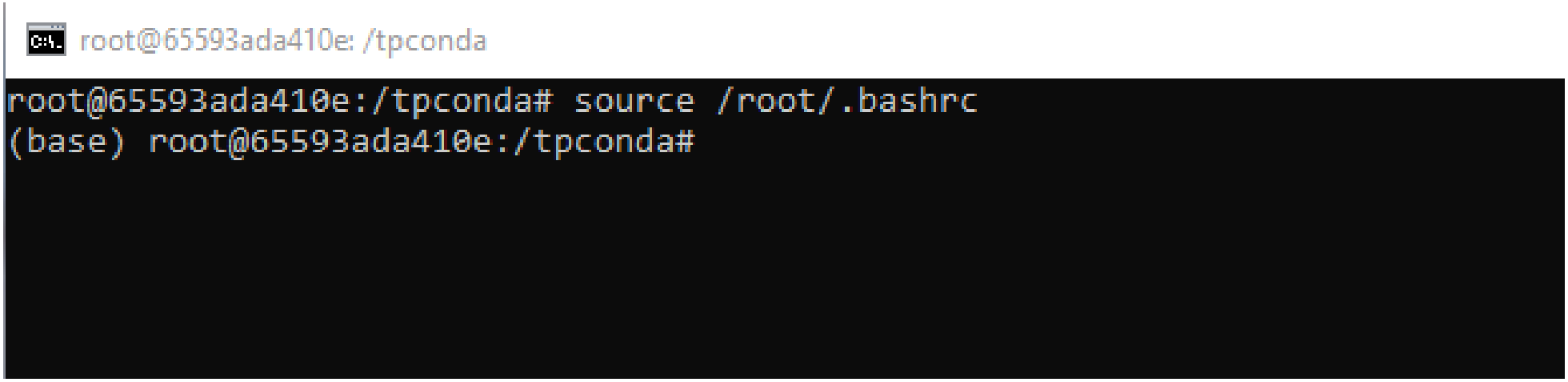


Install miniconda



Install miniconda

source /root/.bashrc



Et voilà !! miniconda is installed, added in your $PATH and loaded

(Mini) conda exploration

Type :

`conda`

Or

`conda --help`

```
olivier@dellolinux:~$ conda
usage: conda [-h] [-V] command ...

conda is a tool for managing and deploying applications, environments and packages.

Options:

positional arguments:
  command
    clean        Remove unused packages and caches.
    config       Modify configuration values in .condarc. This is modeled
                 after the git config command. Writes to the user .condarc
                 file (/home/olivier/.condarc) by default.
    create       Create a new conda environment from a list of specified
                 packages.
    help         Displays a list of available conda commands and their help
                 strings.
    info         Display information about current conda install.
    init         Initialize conda for shell interaction. [Experimental]
    install      Installs a list of packages into a specified conda
                 environment.
    list         List linked packages in a conda environment.
    package      Low-level conda package utility. (EXPERIMENTAL)
    remove       Remove a list of packages from a specified conda environment.
    uninstall    Alias for conda remove.
    run          Run an executable in a conda environment. [Experimental]
    search       Search for packages and display associated information. The
                 input is a MatchSpec, a query language for conda packages.
                 See examples below.
    update       Updates conda packages to the latest compatible version.
    upgrade      Alias for conda update.

optional arguments:
  -h, --help     Show this help message and exit.
  -V, --version  Show the conda version number and exit.

conda commands available from other packages:
  env
```

(Mini) conda exploration

Type :

`conda env --help`

```
(base) root@65593ada410e:/tpconda# conda env
usage: conda-env [-h] {create,export,list,remove,update} ...

positional arguments:
  {create,export,list,remove,update}
    create              Create an environment based on an environment file
    export              Export a given environment
    list                List the Conda environments
    remove              Remove an environment
    update              Update the current environment based on environment
                        file

optional arguments:
  -h, --help            Show this help message and exit.

conda commands available from other packages:
  env
(base) root@65593ada410e:/tpconda#
```

(Mini) conda basic command

conda info

conda info -e

conda env list

conda list

conda deactivate

which python

conda activate

which python

conda list

```
olivier@dellolinux:~$ conda
usage: conda [-h] [-V] command ...

conda is a tool for managing and deploying applications, environments and packages.

Options:

positional arguments:
  command
    clean        Remove unused packages and caches.
    config       Modify configuration values in .condarc. This is modeled
                 after the git config command. Writes to the user .condarc
                 file (/home/olivier/.condarc) by default.
    create       Create a new conda environment from a list of specified
                 packages.
    help         Displays a list of available conda commands and their help
                 strings.
    info         Display information about current conda install.
    init         Initialize conda for shell interaction. [Experimental]
    install      Installs a list of packages into a specified conda
                 environment.
    list         List linked packages in a conda environment.
    package      Low-level conda package utility. (EXPERIMENTAL)
    remove       Remove a list of packages from a specified conda environment.
    uninstall    Alias for conda remove.
    run          Run an executable in a conda environment. [Experimental]
    search       Search for packages and display associated information. The
                 input is a MatchSpec, a query language for conda packages.
                 See examples below.
    update       Updates conda packages to the latest compatible version.
    upgrade      Alias for conda update.

optional arguments:
  -h, --help     Show this help message and exit.
  -V, --version  Show the conda version number and exit.

conda commands available from other packages:
  env
```

Create an environment

Conda is environment management system :

You can create insulated software envs (like different rooms in the lab)

The command is like `conda create -n envname toolname`

Or like `conda install  -n envname toolname` (if the env already exists)

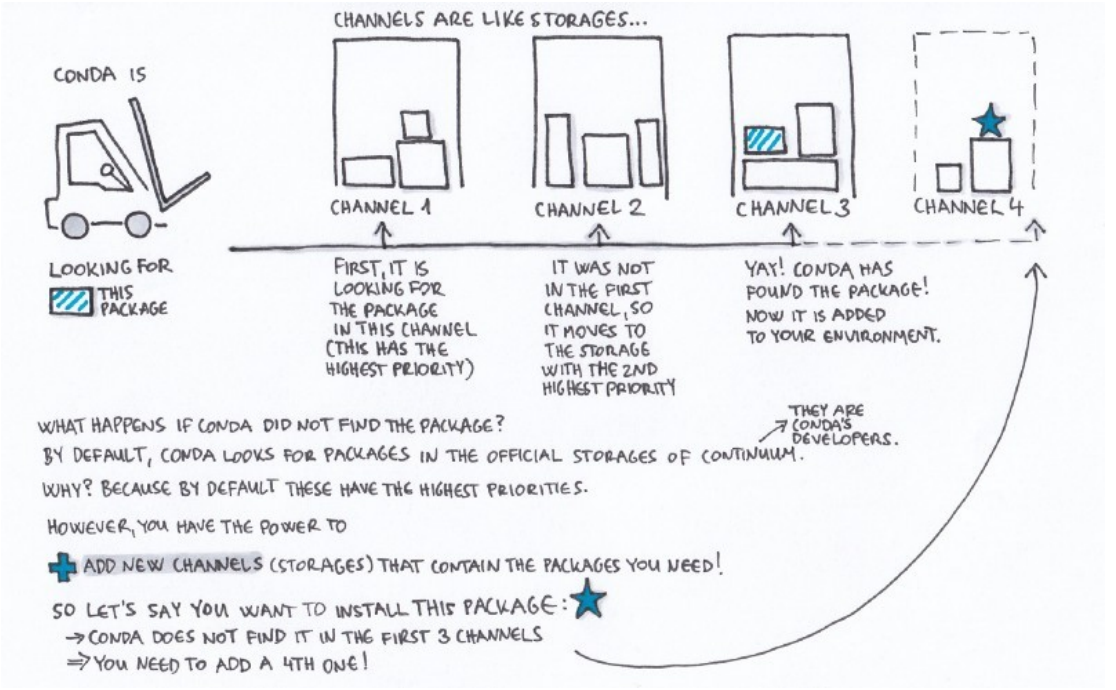
Important notice : the « base » env shouldn’t be used (or modified).
Always create a new or modify an existing env if you want to try something

How does it work ?

Conda is package manager system :

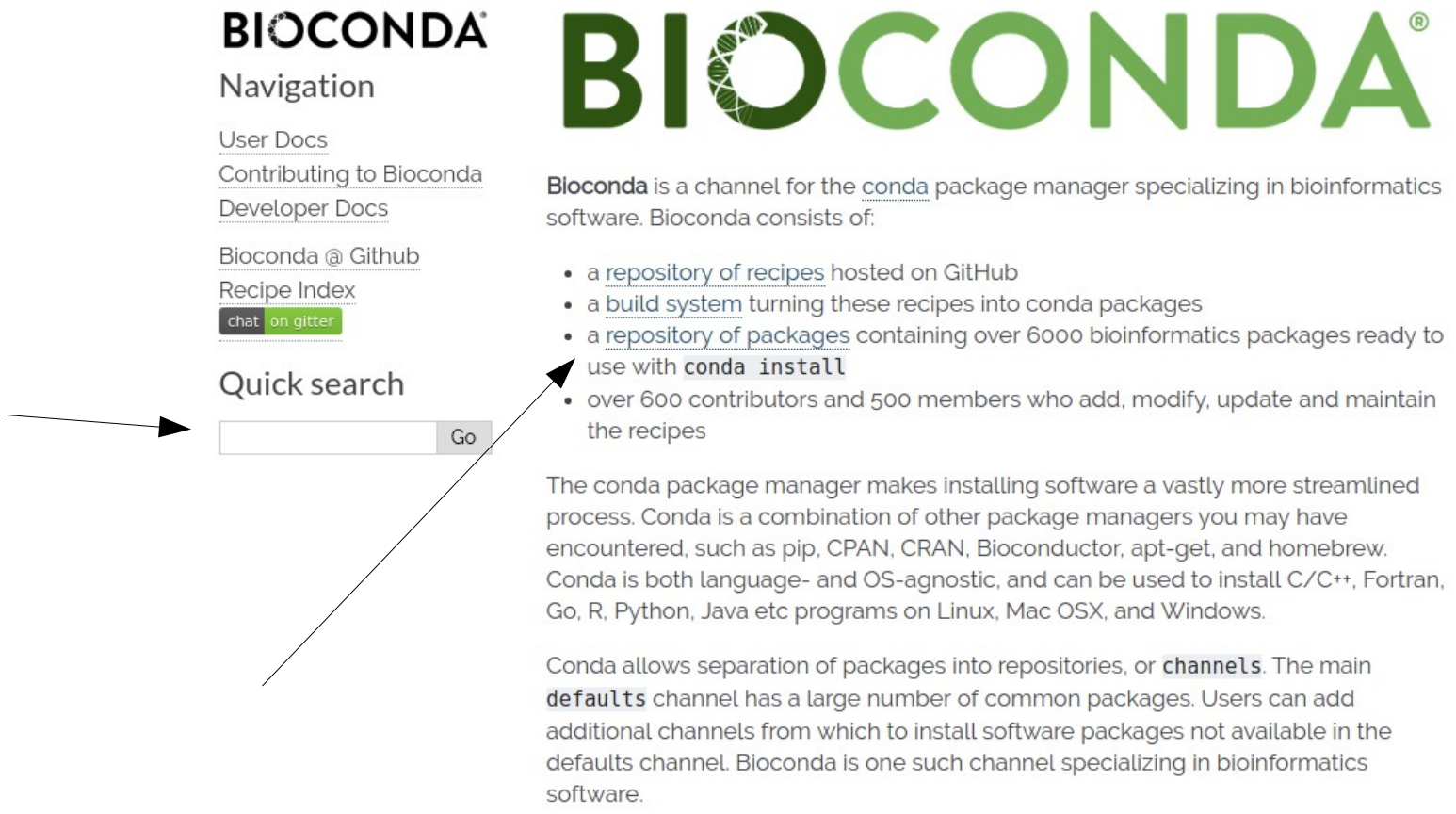
You can any program from dedicated repositories (channels)

Conda will find the required dependencies for your packages



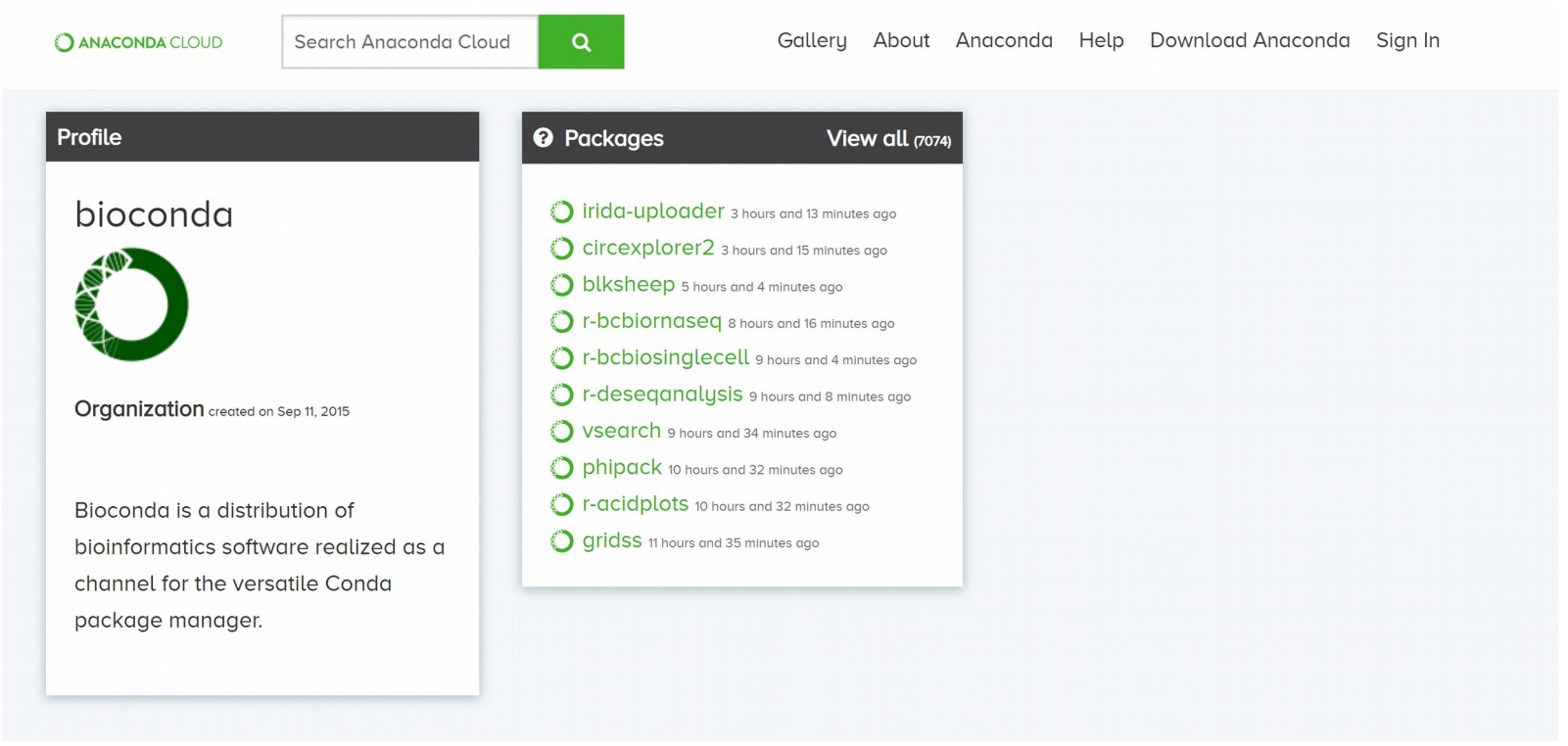
Credits : https://www.freecodecamp.org/news/why-you-need-python-environments-and-how-to-manage-them-with-conda-85f155f4353c/

Usefull channel / repository



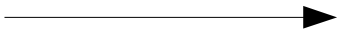
https://bioconda.github.io/

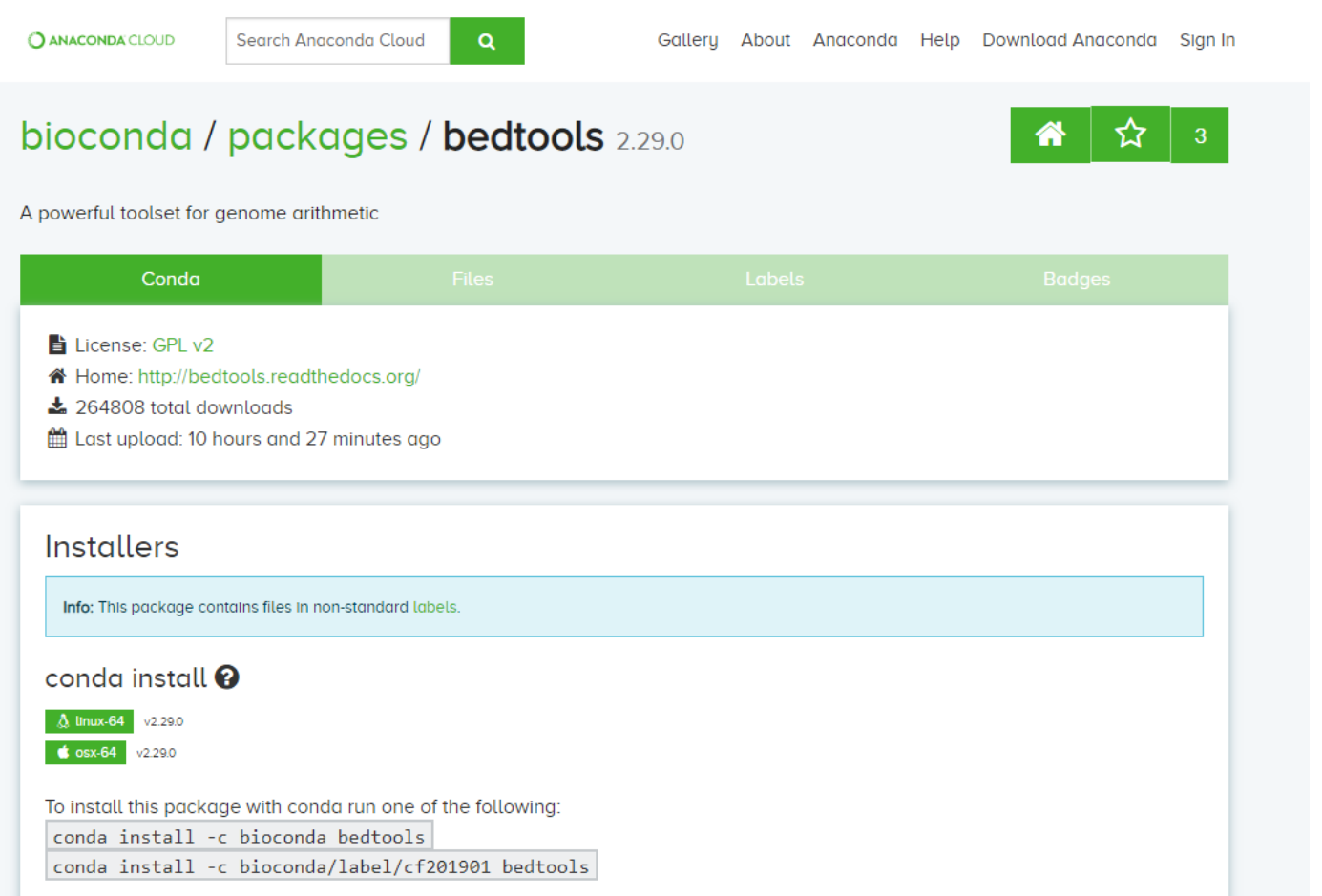
Repository of packages



Install bedtools

Search in anaconda-cloud
And click the link





Install bedtools in a new environment

`conda create -n bedtools229 -c bioconda bedtools`

Open & check your new environment

`conda activate bedtools229`

Check

`conda list`

`bedtools --version`

`bedtools`

```
(base) root@05b7772c32d1:/tpconda# conda create -n bedtools229 -c bioconda bedto
Collecting package metadata (current_repodata.json): done
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.7.10
  latest version: 4.7.11

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: /root/miniconda3/envs/bedtools229

  added / updated specs:
    - bedtools


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    bedtools-2.29.0            |       h6ed99ea_1        13.8 MB  bioconda
    bzip2-1.0.8                |       h7b6447c_0          78 KB
    ------------------------------------------------------------
                                           Total:        13.9 MB

The following NEW packages will be INSTALLED:

  _libgcc_mutex      pkgs/main/linux-64::_libgcc_mutex-0.1-main
  bedtools           bioconda/linux-64::bedtools-2.29.0-h6ed99ea_1
  bzip2              pkgs/main/linux-64::bzip2-1.0.8-h7b6447c_0
  libgcc-ng          pkgs/main/linux-64::libgcc-ng-9.1.0-hdf63c60_0
  libstdcxx-ng       pkgs/main/linux-64::libstdcxx-ng-9.1.0-hdf63c60_0
  xz                 pkgs/main/linux-64::xz-5.2.4-h14c3975_4
  zlib               pkgs/main/linux-64::zlib-1.2.11-h7b6447c_3


Proceed ([y]/n)?
```

Install an older version of bedtools and add bwa

Check available versions

`conda search -c bioconda bedtools`

```
(base) root@05b7772c32d1:/tpconda# conda search -c bioconda bedtools
Loading channels: done
# Name                       Version           Build  Channel
bedtools                      2.16.2               0  bioconda
bedtools                      2.17.0               0  bioconda
bedtools                      2.19.1               0  bioconda
bedtools                      2.19.1      he860b03_2  bioconda
bedtools                      2.19.1      he941832_1  bioconda
bedtools                      2.20.1               0  bioconda
bedtools                      2.20.1      he860b03_2  bioconda
bedtools                      2.20.1      he941832_1  bioconda
bedtools                        2.22               0  bioconda
bedtools                        2.22      hdbcaa40_3  bioconda
```

You can create a new env or clone your env and downgrade software version

`conda create --clone bedtools229 -n bedtools222`

`conda install -c bioconda -n bedtools222 bedtools=2.22 bwa`

Export and share your envs

Check your env list

conda env list

```
(bedtools229) root@05b7772c32d1:/tpconda# conda env list
# conda environments:
#
base                  /root/miniconda3
bedtools222           /root/miniconda3/envs/bedtools222
bedtools229         *  /root/miniconda3/envs/bedtools229
```

Which env is « activated » ?

conda activate

Export environment configuration

conda env export -n bedtools222

conda env export –no- build -n bedtools222

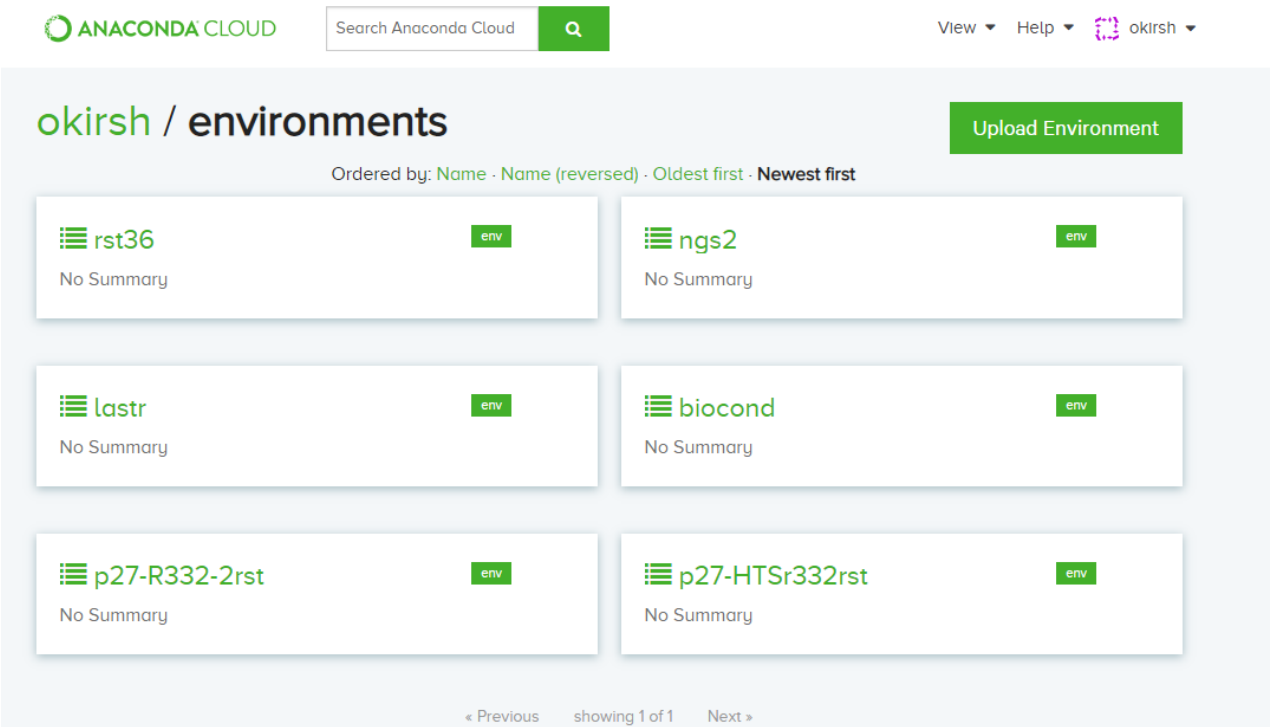
Save it in a yml file

conda env export --no-build -n bedtools222 > bedtools222.yml

Export and share your envs

Send yml file by email

Share them on anaconda cloud



Managing envs

> Creating an env from a yml

```
(base) [okirsh@clust-slurm-client ~]$ conda env list
# conda environments:
#
base                  *  /shared/mfs/data/software/miniconda
bcftools-1.9             /shared/mfs/data/software/miniconda/envs/bcftools-1.9
bedtools-2.26.0          /shared/mfs/data/software/miniconda/envs/bedtools-2.26.0
bedtools-2.27.1          /shared/mfs/data/software/miniconda/envs/bedtools-2.27.1
biokevlar-0.6.1          /shared/mfs/data/software/miniconda/envs/biokevlar-0.6.1
blast-2.7.1              /shared/mfs/data/software/miniconda/envs/blast-2.7.1
bowtie-1.2.2             /shared/mfs/data/software/miniconda/envs/bowtie-1.2.2
bowtie2-2.3.4.3          /shared/mfs/data/software/miniconda/envs/bowtie2-2.3.4.3
bwa-0.7.17               /shared/mfs/data/software/miniconda/envs/bwa-0.7.17
ceas-1.0.2               /shared/mfs/data/software/miniconda/envs/ceas-1.0.2
clustalo-1.2.4           /shared/mfs/data/software/miniconda/envs/clustalo-1.2.4
coreutils-8.25           /shared/mfs/data/software/miniconda/envs/coreutils-8.25
csvkit-1.0.3             /shared/mfs/data/software/miniconda/envs/csvkit-1.0.3
cutadapt-1.10            /shared/mfs/data/software/miniconda/envs/cutadapt-1.10
cutadapt-1.8.3           /shared/mfs/data/software/miniconda/envs/cutadapt-1.8.3
cytoscape-3.7.1          /shared/mfs/data/software/miniconda/envs/cytoscape-3.7.1
deeptools-3.1.2          /shared/mfs/data/software/miniconda/envs/deeptools-3.1.2
deeptools-3.2.0          /shared/mfs/data/software/miniconda/envs/deeptools-3.2.0
delly-0.7.9              /shared/mfs/data/software/miniconda/envs/delly-0.7.9
```

View of available conda env on nncr (IFB cluster)

Managing envs

> Creating an env from a yml

Type :

`ls -l`

`cat bedtools.yml`

```
root@76a9d2488003:/tpconda# ll
total 73508
drwxr-xr-x 2 root root     4096 Sep 11 07:23 ./
drwxr-xr-x 1 root root     4096 Sep 12 09:20 ../
-rw-r--r-- 1 root root 75257002 Jul 29 14:25 Miniconda3-latest-Linux-x86_64.sh
-rwxr-xr-x 1 root root      241 Sep 11 07:20 bedtools.yml*
```

`conda env create -n bedtoolsifb -f bedtools.yml`

Managing envs

Remove an env :Simply delete the folder, or

`conda env list`

`conda env remove -n bedtools229`

Set up « Channels »

- https://bioconda.github.io/user/install.html#set-up-channels

2. Set up channels

After installing conda you will need to add the bioconda channel as well as the other channels bioconda depends on. **It is important to add them in this order** so that the priority is set correctly (that is, conda-forge is highest priority).

The conda-forge channel contains many general-purpose packages not already found in the `defaults` channel.

```
conda config --add channels defaults
conda config --add channels bioconda
conda config --add channels conda-forge
```

* order matters

First check your channel config with : `conda config --get channels`

This step is not mandatory but it’s always good to « control » your machine

Is conda really FAIR ?

The FAIR data principles

Findable **A**ccessible **I**nteroperable **R**eusable

```
a bedto

## Package Plan ##
  environment location: /root/miniconda3/envs/bedtools229
  added / updated specs:
```

MENU

PARIS7 BIOINFO CLUB

Bioinformatics Club – IJM, Epigenetics, BFA

https://paris7bioinfo.wordpress.com/

Thursday September 26, 10am: Presentation of eLabFTW (Nicolas Carpi, Institut Curie). eLabFTW is a free and open source electronic lab notebook designed for researchers. Note that we are not sure whether it is possible to implement it at IJM. In any case, this presentation should give you an overview of what you can do with electronic lab notebooks.

Thursday October 3rd, 10am: Do-it-yourself Extraction of differentially-expressed genes from an RNAseq experiment (Flora Borne, IJM) *Please bring your own laptop!*