

FAIR_bioinfo for bioinformaticians

Introduction to the tools of reproducibility in bioinformatics

C. Hernandez¹ T. Denecker¹ J.Sellier² C. Toffano-Nioche¹

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

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

Sept. 2020



Conda

CONDA: an environment manager

Why using an environment manager?

- avoid compilation and dependencies problems: an environment manager will take care of everything!
- have several environments in parallel each with their own set of tools
- useful when cross-tools dependencies are incompatible with each other

Conda concepts, objets

- Environment: a set of packages/tools in a directory (added to our PATH)
- Conda: an open source package + a general-purpose environment management system (installation, execution, upgrade). For any programming language, multi-platform (Windows, MacOS, Linux).
- Conda package: a compressed tarball



Conda distribution

- Anaconda: a data science platform, comes with a lot of packages
- Miniconda: come without installed packages

Anaconda cloud, the "conda hub"

- [Anaconda cloud](#) (private company) relies on the community of developers, concerns many domains (Machine Learning, Data Visualization, Dashboarding-web, Image Processing, Natural Language Processing, etc)
- Anaconda cloud: made up of channels/owners. Each channels contains one or more conda packages
- be careful when downloading any packages from an untrusted source, always inspect before installation

Some conda channels

- defaults
- bioconda: bioinformaticians contributions
- conda-forge: many popular python packages (analogous to PyPI but with a unified, automated build infrastructure and more peer review of recipes)
- r: for packages in R language

Channels list order

- when different channels have the same package \Rightarrow collisions
- collisions resolved following the order of your channels list \Rightarrow put supplemental channels at the bottom of your channel list

simple commands

```
1 conda create env -n myenv # creation of a conda environment
2 conda info --envs # list environments (* for the active one)
3 conda activate myenv # active the myenv environment
4 conda deactivate # inactivate the environment
5 conda list # list packages (only in an active environment)
6 conda install package # installation of a tool/package
7 conda remove package # suppress the tool from the package
8 conda env remove -n myenv # suppress the myenv environment
```

miniconda3

With the miniconda3 distribution and by default, environments are installed in a miniconda3/envs/ repository

interactive

- create an environment
- activate the environment
- install some conda packages

configuration file

- list all conda packages in a configuration file (yaml or json format)
- create the environment based on the configuration file (option `-f`)
- activate the environment

reproducibility

- good practice: use a configuration file
- to specify a precise version of a package use the syntax
`<channel>::<package>=<version>`

Conda Exercise

Objective

Reminder

- We have already used (blind) conda and its strategy for defining environments!
- We are now going to explain what we used before

Example of a conda configuration file

envfair.yml

```
1 channels:
2   - conda-forge
3   - bioconda
4   - main
5   - default
6 dependencies:
7   - python=3.7.6 # specify python version (not required but
8     can help with downstream conflicts)
9   - snakemake-minimal=5.10.0 # workflow manager
10  - graphviz=2.42.3 # for visualisation
11  - xorg-libxrender
12  - xorg-libxpm
13  - wget=1.20.1 # for downloading files
14  - fastqc=0.11.9 # for the RNAseq analysis
15  - bowtie2=2.4.1
16  - samtools=1.10
17  - subread=2.0.1
```

Conda setup

How to access conda?

- Conda is so used that it could even be installed by default to your machine. To test this: `conda --version`
- if not, may install it or got it by a docker image:

```
1 docker run -i -t -v ${PWD}:/data continuumio/miniconda3
```

- on the IFB cluster, with modules: `module load conda`

Conda environment

We have already use a conda configuration file in the workflow session:

```
1 conda env create -n envfair -f envfair.yml
2 conda activate envfair
```

We will next detail the content of the configuration file, `envfair.yml`

How to access tools?

Manage Conda environment

- 1 create the working environment:

```
1 conda create env -n myenv
```

- 2 activate it:

```
1 conda activate myenv
```

- 3 if not yet done, install packages (specify the channel):

```
1 conda install -c bioconda bowtie2
```

- 4 work with the tools

- 5 quite the environment:

```
1 conda deactivate
```

Install Snakemake

Objective

Create a conda configuration file to install the snakemake tool.

Hint

- Search the channel in the Anaconda cloud web pages
- the "minimal" environment is sufficient

Install Snakemake

condaEnvSnakemake.yml

```
1 channels:
2   - conda-forge
3   - bioconda
4   - main
5 dependencies:
6   - snakemake-minimal=5.10.0
```

run

```
1 conda create env -n condaEnvSnakemake -f condaEnvSnakemake.
   yml
2 conda activate condaEnvSnakemake
3 snakemake ...
```