

FAIR_bioinfo : Open Science and FAIR principles in a bioinformatics project

How to make a bioinformatics project more reproducible

C. Hernandez¹ T. Denecker² J. Sellier² G. Le Corguillé²
C. Toffano-Nioche¹

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

²IFB Core Cluster taskforce

June 2021



Conda

CONDA: an environment manager

Conda definitions

- 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 of a tool

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 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




- default
- conda-forge: many popular python packages (analogous to PyPI but with a unified, automated build infrastructure and more peer review of recipes)
- bioconda: bioinformaticians' contributions
- private

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

Conda and R

The R interpreter is included in the `r-essentials` package (200 r packages). Add `r-` before the regular `r` package name (eg. `r-ggplot2`)

⬇ Favorites	⬇ Downloads	⬇ Package (owner / package)	Platforms
18	304565	 <code>r / r-essentials</code> 3.6.0 Some essential packages for working with R	linux-32 linux-64 osx-64 win-32 win-64 conda
11	198513	 <code>skyblued / r-essentials</code> 3.5.1 Some essential packages for working with R	linux-64 osx-64 win-32 win-64 copy conda
1	66845	 <code>conda-forge / r-essentials</code> 4.1 Some essential packages for working with R. This was migrated from the 'r' channel.	linux-64 noarch osx-64 conda

Mamba

A fast drop-in alternative to conda, using `libsolv` for dependency resolution

```
conda install -c conda-forge mamba
```

Next, replace `conda` by `mamba` to use it

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 list # list packages (only in an active environment)
5 conda install package # installation of a tool/package
6 conda remove package # suppress the tool from the
   environment
7 conda env remove -n myenv # suppress the myenv environment
8 conda deactivate # inactivate the 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
- specify a precise version of a package:
`<channel>::<package>=<version>`

Conda Exercise

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
```

- already activated on the IFB cluster (otherwise with module: `module load conda`)



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 with conda

Objective

Create a conda configuration file to install the snakemake tool.

Hint

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

Install snakemake with conda

condaEnvSmk.yml

```
1 channels:  
2   - conda-forge  
3   - bioconda  
4   - main  
5 dependencies:  
6   - snakemake-minimal=6.5.0
```

run

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