SCIENCE MEETS LIFE

A tidy bioinformatics environment to the rescue!

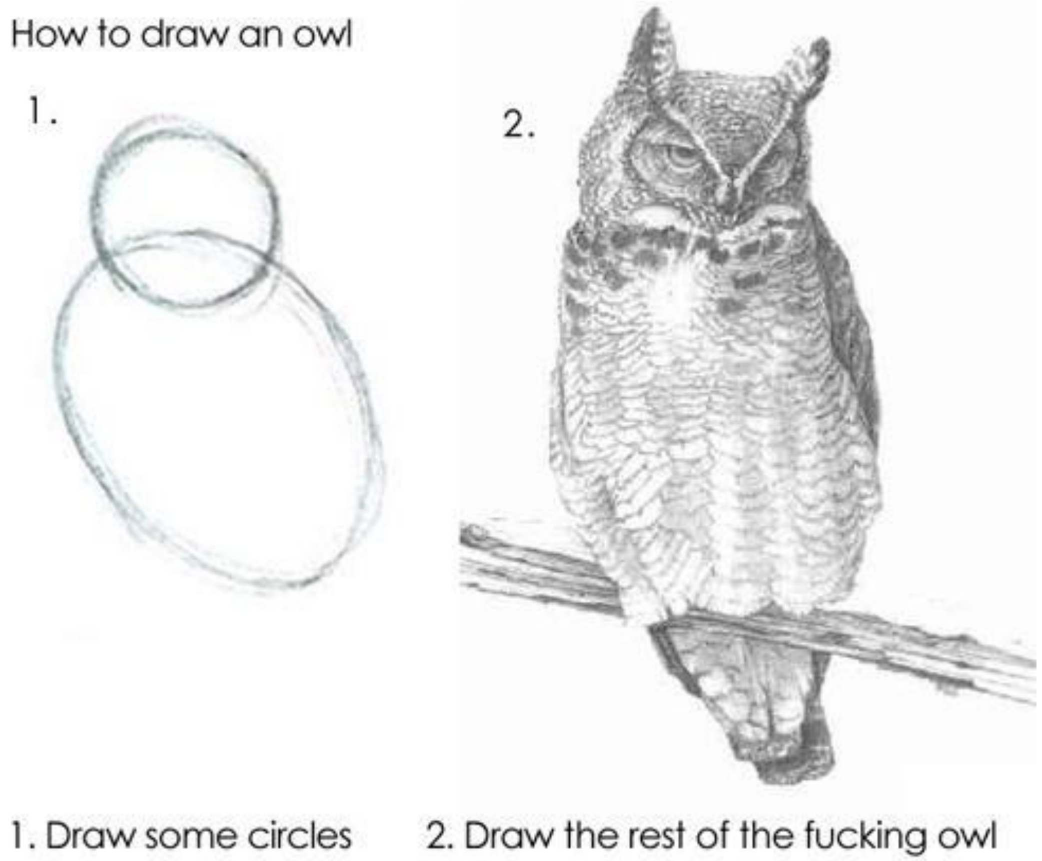
A little treat to your fellows and future self

24/02/2020

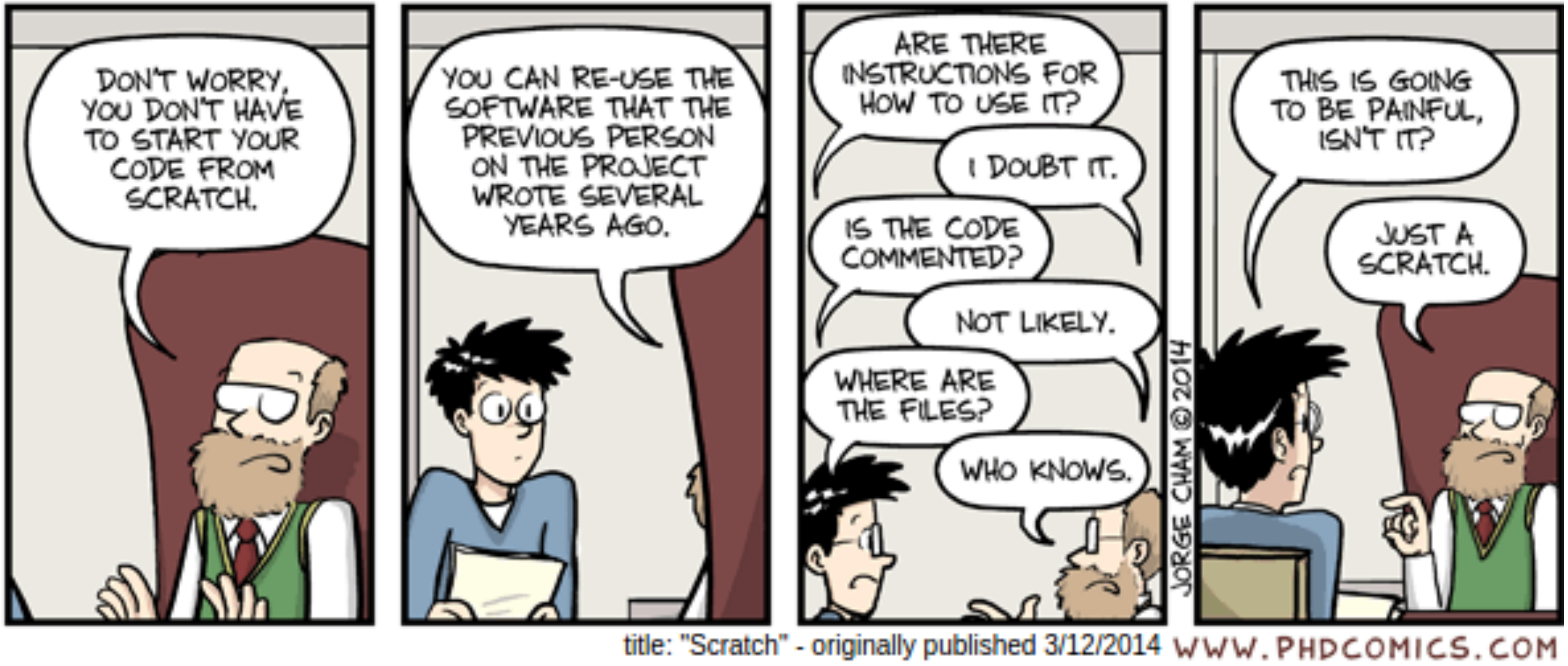
Alexander Botzki & Tuur Muyldermans
alexander.botzki@vib.be
tuur.muyldermans@vib.be

VIB

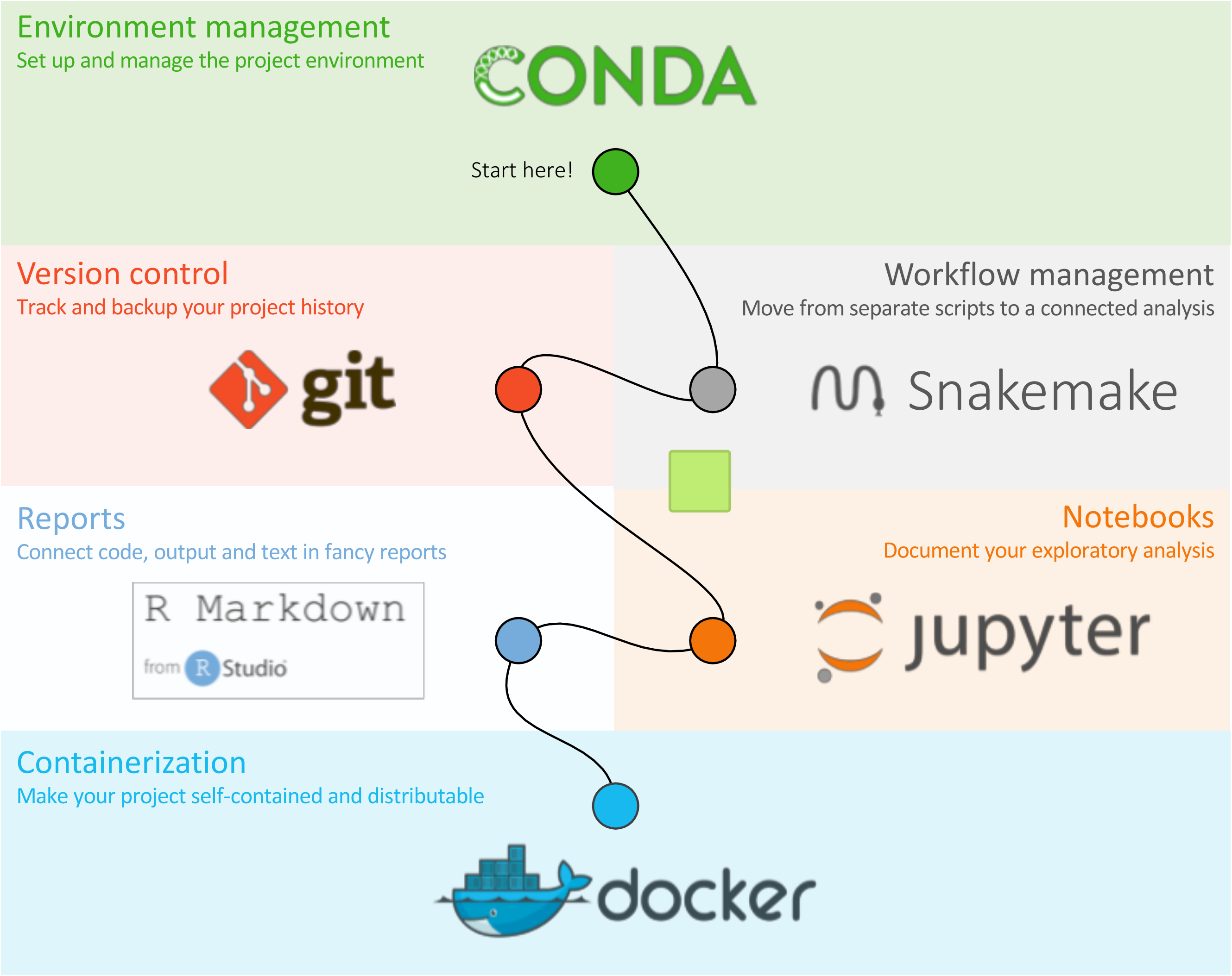
Introduction


How to draw an owl
1.
2.
1. Draw some circles
2. Draw the rest of the fucking owl

Piled Higher and Deeper *by Jorge Cham*


DON'T WORRY, YOU DON'T HAVE TO START YOUR CODE FROM SCRATCH.
YOU CAN RE-USE THE SOFTWARE THAT THE PREVIOUS PERSON ON THE PROJECT WROTE SEVERAL YEARS AGO.
ARE THERE INSTRUCTIONS FOR HOW TO USE IT?
I DOUBT IT.
IS THE CODE COMMENTED?
NOT LIKELY.
WHERE ARE THE FILES?
WHO KNOWS.
JORGE CHAM © 2014
THIS IS GOING TO BE PAINFUL, ISN'T IT?
JUST A SCRATCH.
WWW.PHDCOMICS.COM

title: "Scratch" - originally published 3/12/2014



Environment management

Set up and manage the project environment

Start here!

Version control

Track and backup your project history

Workflow management

Move from separate scripts to a connected analysis

Reports

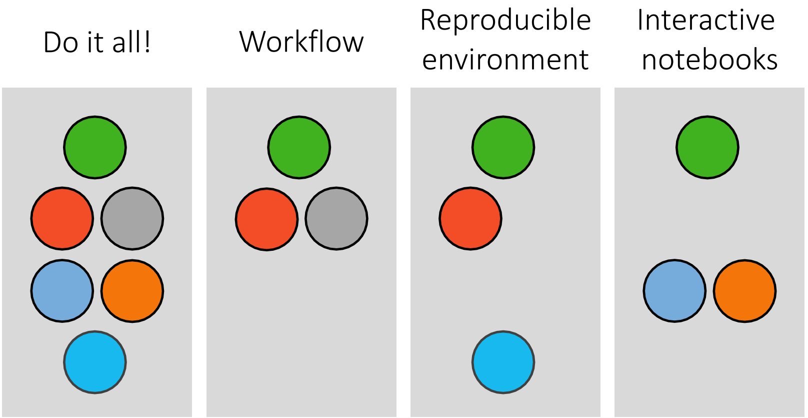
Connect code, output and text in fancy reports

Notebooks

Document your exploratory analysis

Containerization

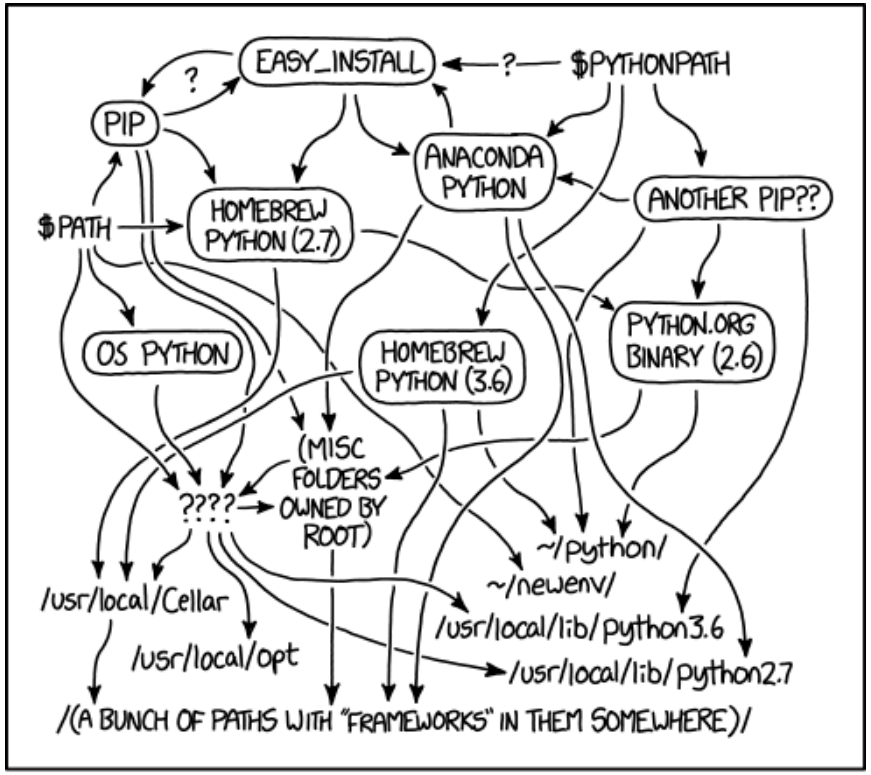
Make your project self-contained and distributable



Package and environment managers

Pip & Conda

Package dependency problems

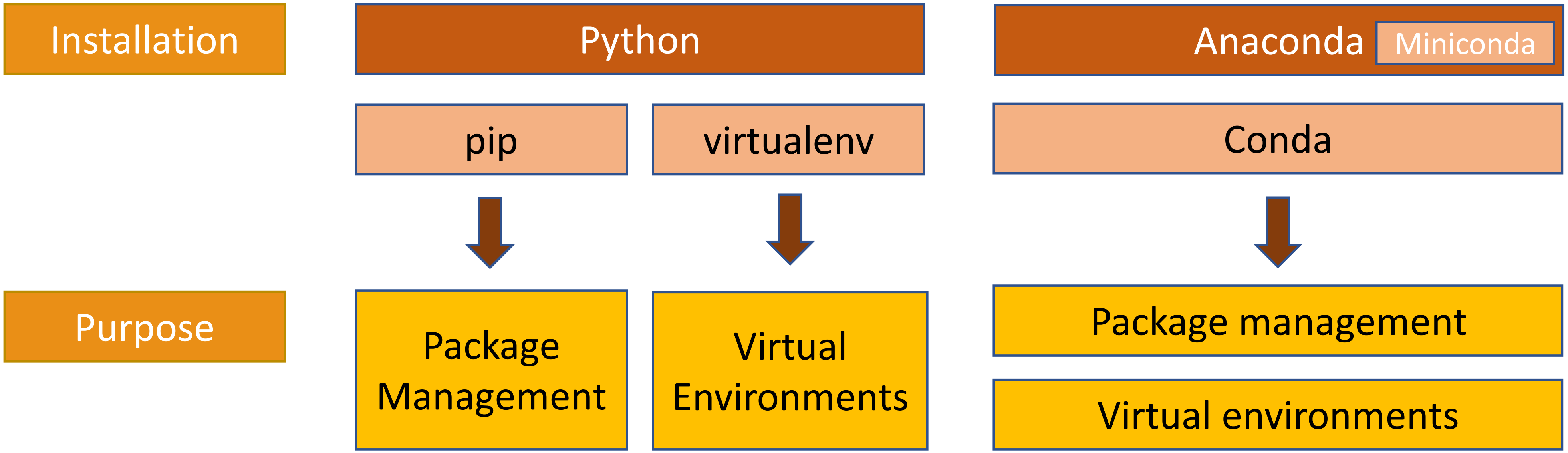


MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

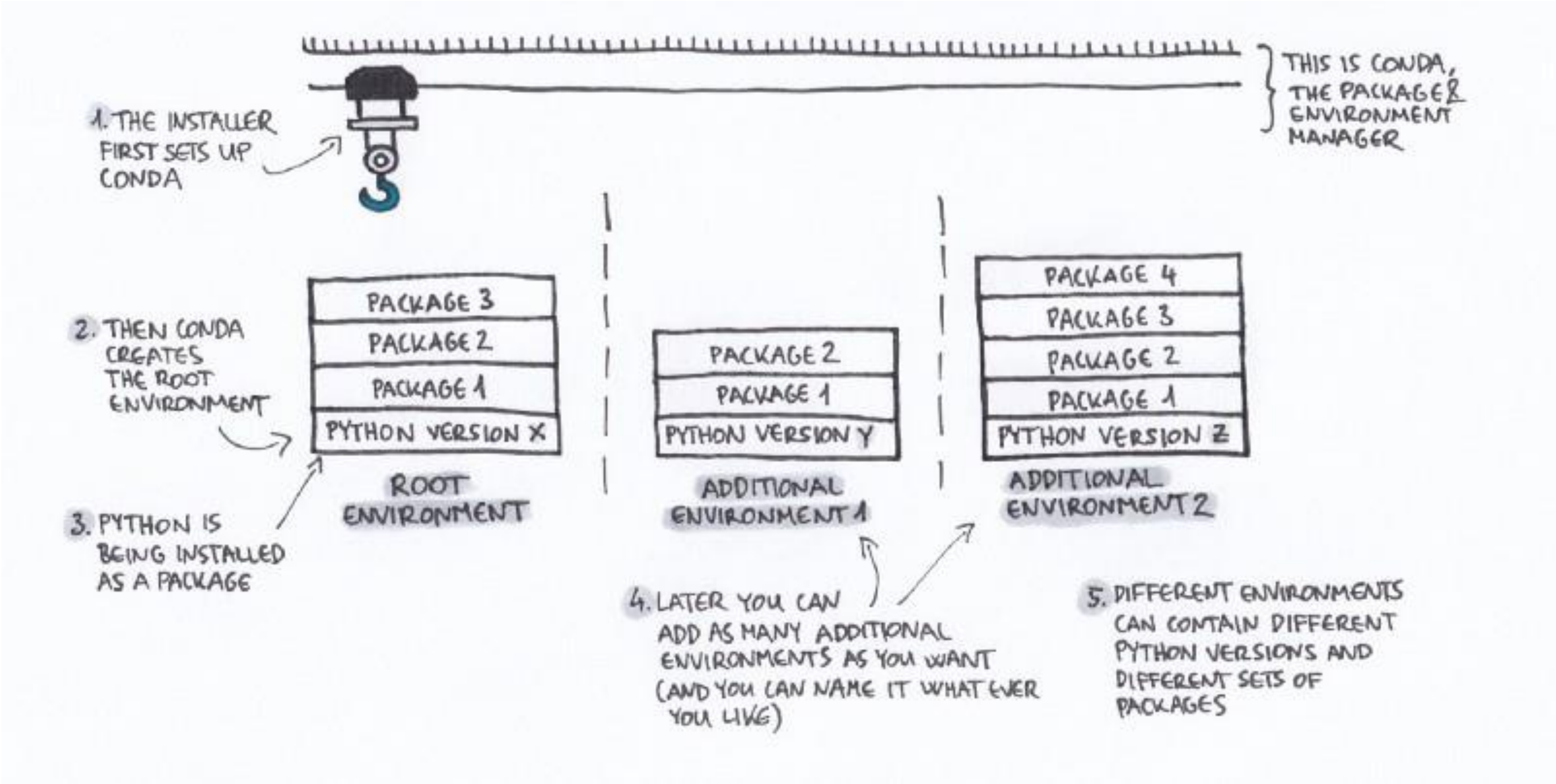
tech.instacart.com

YOU BEST START BELIEVING IN DEPENDENCY HELL
BECAUSE YOU'RE IN IT
memegenerator.net

Package and environment managers



academind.com


1. THE INSTALLER FIRST SETS UP CONDA
THIS IS CONDA, THE PACKAGE & ENVIRONMENT MANAGER
2. THEN CONDA CREATES THE ROOT ENVIRONMENT
3. PYTHON IS BEING INSTALLED AS A PACKAGE
PACKAGE 3
PACKAGE 2
PACKAGE 1
PYTHON VERSION X
ROOT ENVIRONMENT
PACKAGE 2
PACKAGE 1
PYTHON VERSION Y
ADDITIONAL ENVIRONMENT 1
PACKAGE 4
PACKAGE 3
PACKAGE 2
PACKAGE 1
PYTHON VERSION Z
ADDITIONAL ENVIRONMENT 2
4. LATER YOU CAN ADD AS MANY ADDITIONAL ENVIRONMENTS AS YOU WANT (AND YOU CAN NAME IT WHAT EVER YOU LIKE)
5. DIFFERENT ENVIRONMENTS CAN CONTAIN DIFFERENT PYTHON VERSIONS AND DIFFERENT SETS OF PACKAGES

freecodecamp.org

CONDA

Package, dependency and environment management for any language

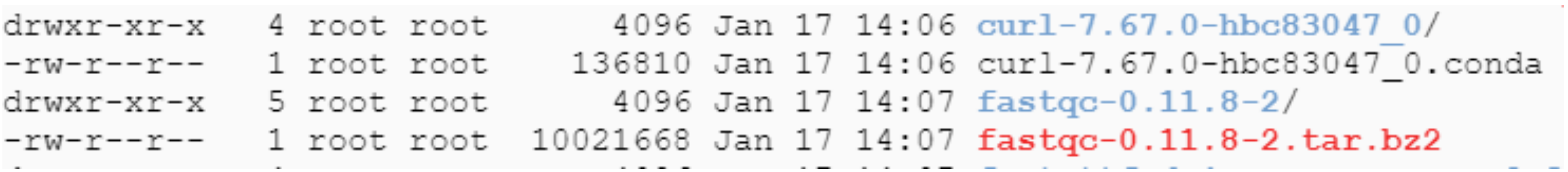
Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, FORTRAN, and more.

What is Conda?

- Finds, installs and updates packages
- Switch between environments for different versions
- Few commands make a totally separate environment with different versions of packages
- Combined with CI systems to provide frequent and automated testing of code

Packages

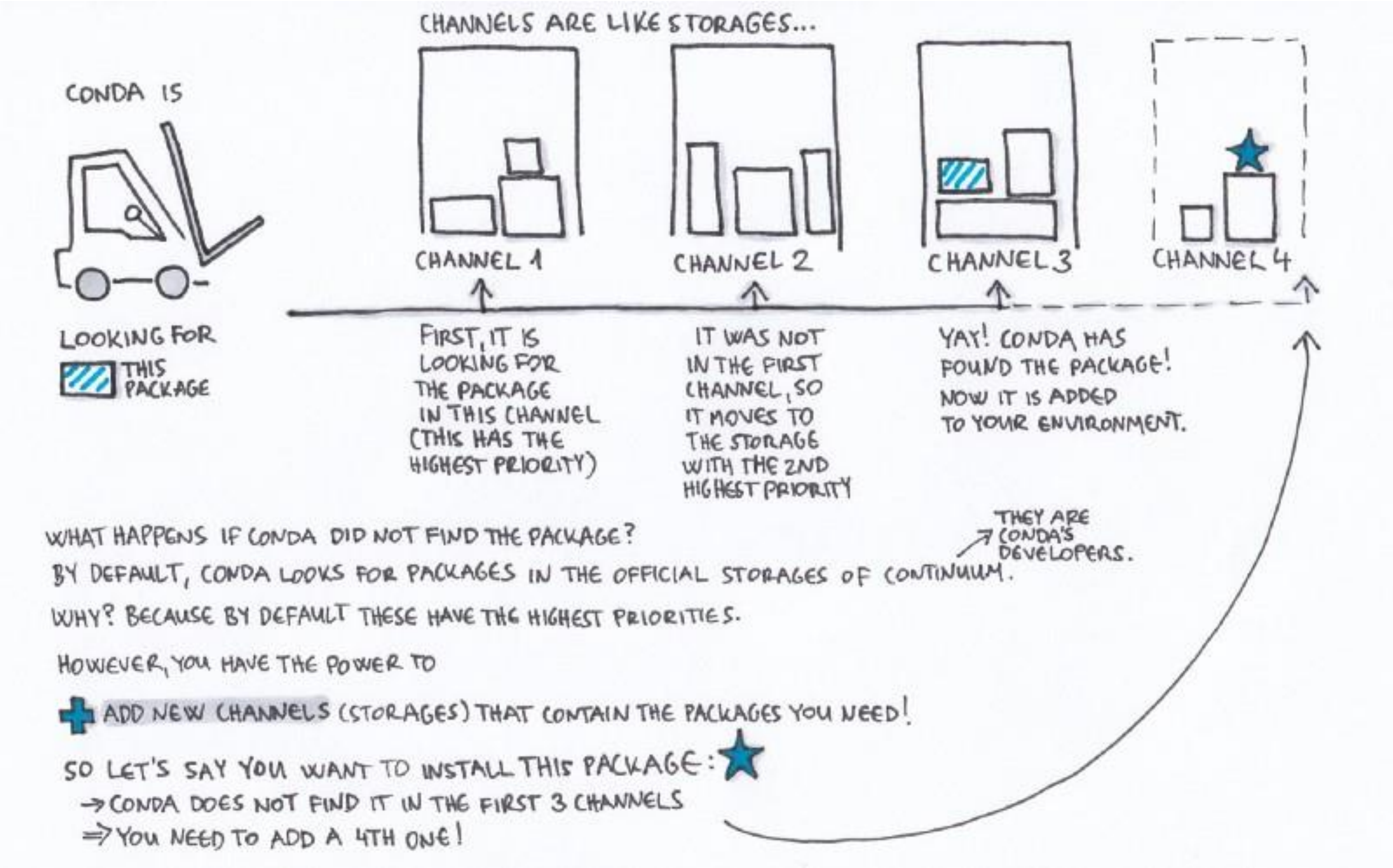
- Compressed tarball file (.tar.bz2) or .conda file
  - system-level libraries,
  - Python or other modules,
  - Executable programs, or other components
  - Metadata
  - Installation files



```
drwxr-xr-x   4 root root      4096 Jan 17 14:06 curl-7.67.0-hbc83047_0/
-rw-r--r--   1 root root    136810 Jan 17 14:06 curl-7.67.0-hbc83047_0.conda
drwxr-xr-x   5 root root      4096 Jan 17 14:07 fastqc-0.11.8-2/
-rw-r--r--   1 root root  10021668 Jan 17 14:07 fastqc-0.11.8-2.tar.bz2
```

- Format is identical across platforms and operating systems

Channels


CHANNELS ARE LIKE STORAGES...
CONDA IS
LOOKING FOR ▨ THIS PACKAGE
CHANNEL 1
CHANNEL 2
CHANNEL 3
CHANNEL 4
FIRST, IT IS LOOKING FOR THE PACKAGE IN THIS CHANNEL (THIS HAS THE HIGHEST PRIORITY)
IT WAS NOT IN THE FIRST CHANNEL, SO IT MOVES TO THE STORAGE WITH THE 2ND HIGHEST PRIORITY
YAY! CONDA HAS FOUND THE PACKAGE! NOW IT IS ADDED TO YOUR ENVIRONMENT.

WHAT HAPPENS IF CONDA DID NOT FIND THE PACKAGE?

BY DEFAULT, CONDA LOOKS FOR PACKAGES IN THE OFFICIAL STORAGES OF CONTINUUM. → THEY ARE CONDA'S DEVELOPERS.

WHY? BECAUSE BY DEFAULT THESE HAVE THE HIGHEST PRIORITIES.

HOWEVER, YOU HAVE THE POWER TO

✚ ADD NEW CHANNELS (STORAGES) THAT CONTAIN THE PACKAGES YOU NEED!

SO LET'S SAY YOU WANT TO INSTALL THIS PACKAGE: ★

→ CONDA DOES NOT FIND IT IN THE FIRST 3 CHANNELS

⇒ YOU NEED TO ADD A 4TH ONE!

Channels

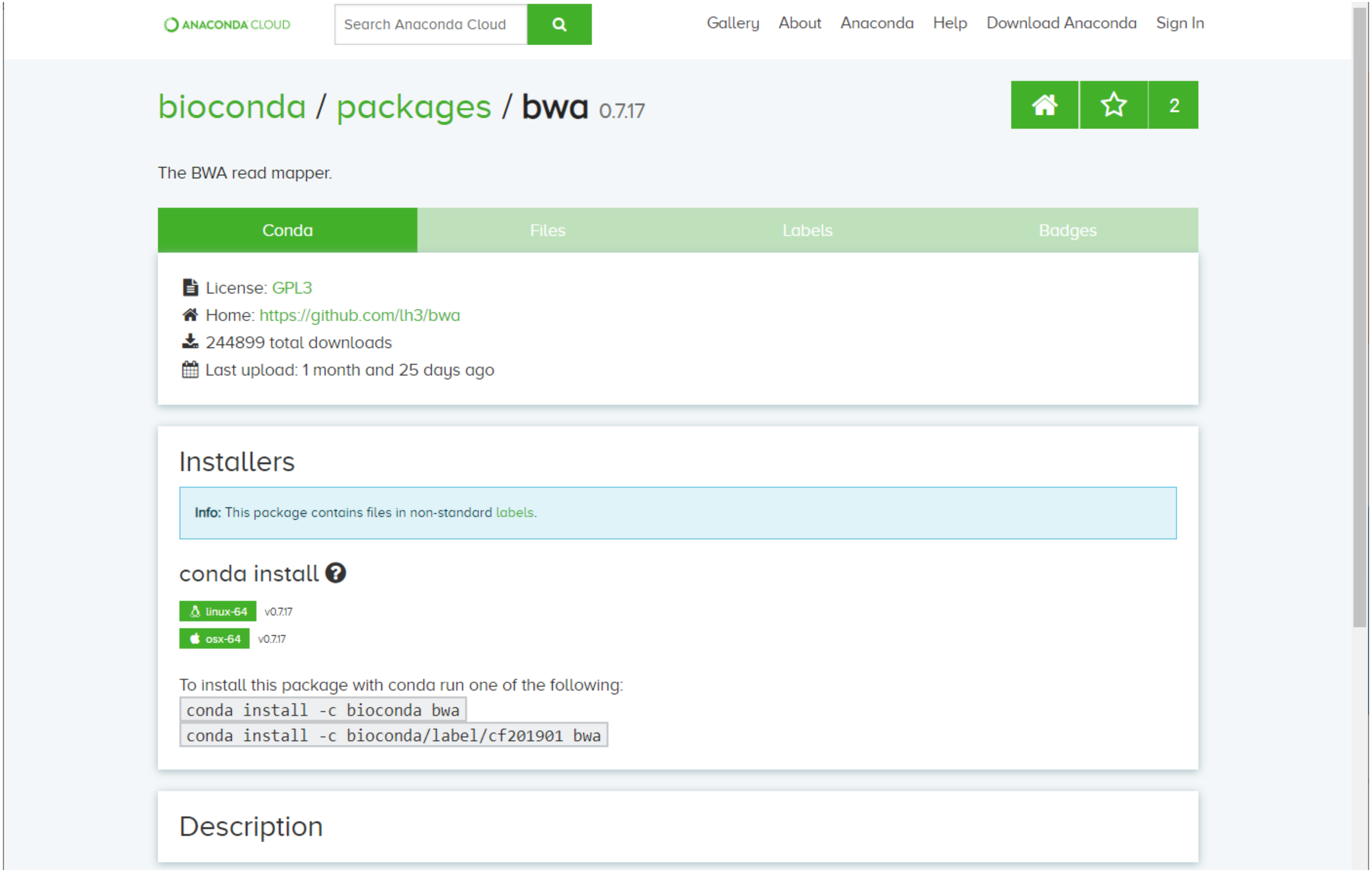
CONDA-FORGE

- >7000 packages
- A community-led collection of recipes, build infrastructure and distributions for the conda package manager
- e.g.: numpy, Scipy, CRAN packages, etc.

BIOCONDA®

- >6000 packages
- Specializing in bioinformatics software
- e.g. samtools, fastqc, salmon, cutadapt, etc.

- Add your own channel



bioconda / packages / bwa 0.7.17

The BWA read mapper.

License: GPL3
Home: https://github.com/lh3/bwa
244899 total downloads
Last upload: 1 month and 25 days ago

Installers

Info: This package contains files in non-standard labels.

conda install

linux-64 v0.7.17
osx-64 v0.7.17

To install this package with conda run one of the following:

```
conda install -c bioconda bwa
conda install -c bioconda/label/cf201901 bwa
```

Description

Environments

- Directory with specific collection of packages
- Switch between environments with *activate* and *deactivate*
- Directory structure
  - ROOT_DIR : where Ana/Miniconda was installed
  - /pkgs : decompressed packages
  - /envs : system location for additional conda environments

Pinning

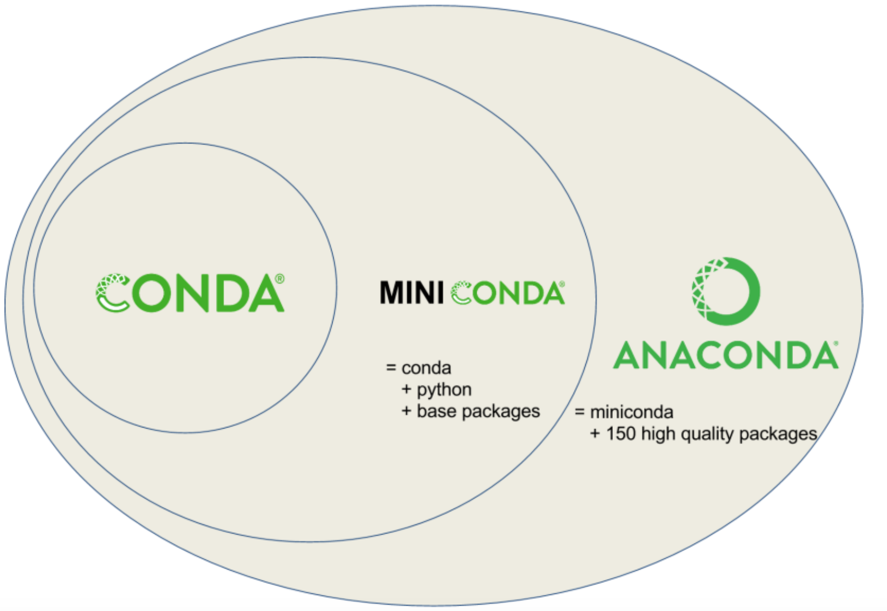
- Preventing packages from updating
- In the environment's conda-meta directory
    - File named *pinned* that includes a list of the packages that you do not want updated.

How it works

1. Installation & config
2. Channels
3. Environment
4. Packages
5. Reference & further reading

Installation

- Three different installers:
  - Miniconda
  - Anaconda
  - Anaconda Enterprise platform



- Miniconda
  - Conda (package & environment management system)
  - “root environment” with certain version of Python and few basic packages
- Anaconda
  - All of the above, and
  - 150+ packages
  - Navigator (GUI)

```
wget https://repo.continuum.io/miniconda/Miniconda2-latest-Linux-
x86_64.sh && bash Miniconda2-latest-Linux-x86_64.sh
```

.condarc

- Conda configuration file
  - `conda config --show`

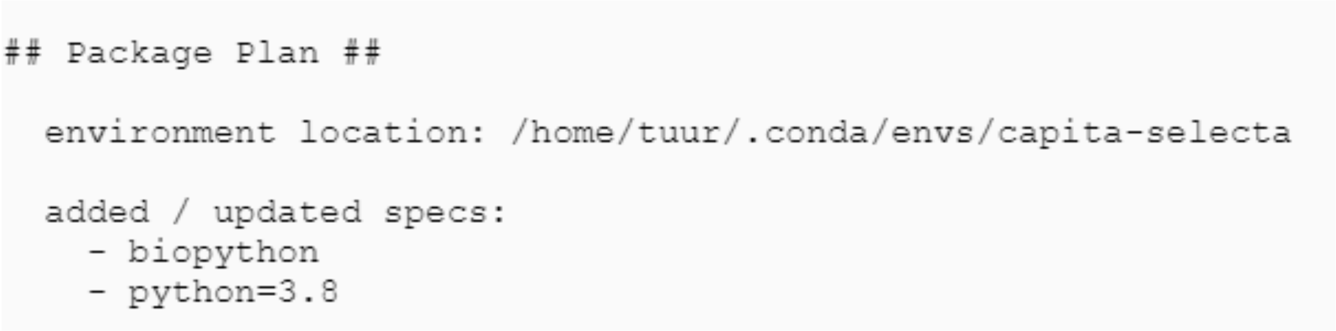
- Add channels
  - `conda config --add channels conda-forge`
  - `conda config --add channels bioconda`

Channels

- List channels
  - `conda config --get channels`
- Add a channel with lowest priority
  - `conda config --append channels newchannel`
- Add a channel with highest priority
  - `conda config --prepend channels newchannel`
- In order to install a package from a channel:
  - `conda install -c <channel> <package>`
- In order to automatically select channels you need to change your `.condarc`:
  - `conda config --add channels <my_channel>`

Environments

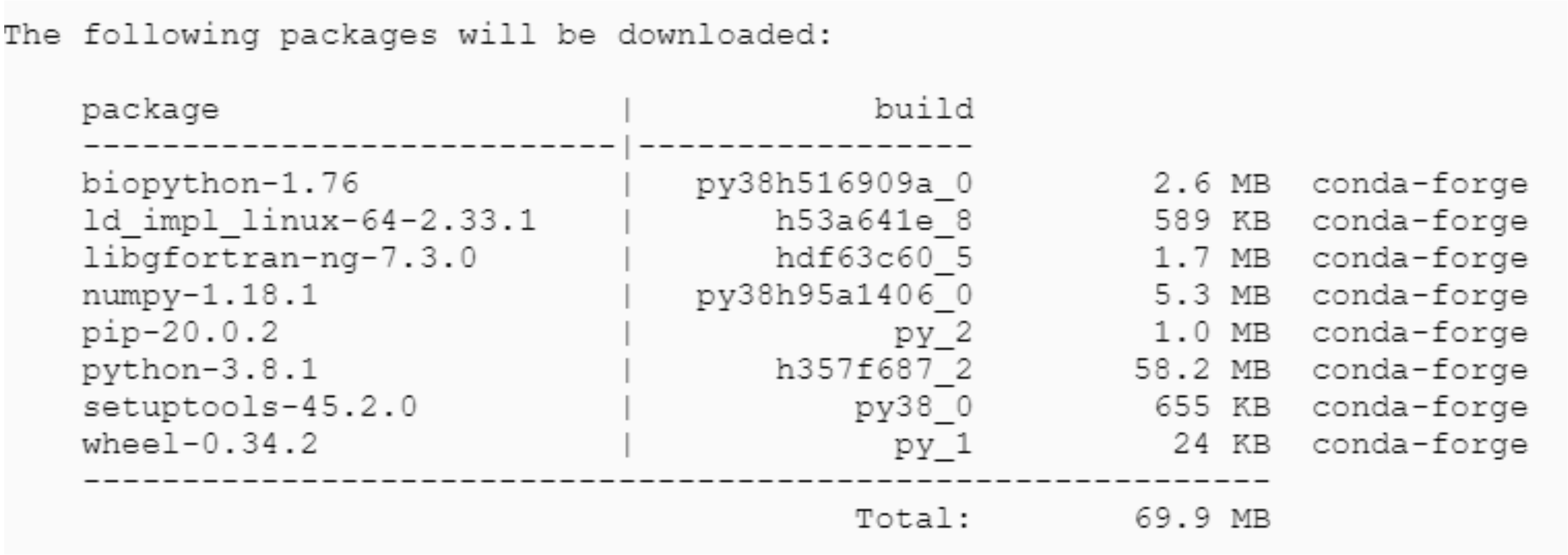
- Create a new environment and install a package in it
  - `conda create -n capita-selecta python=3.8 biopython`



```
## Package Plan ##

  environment location: /home/tuur/.conda/envs/capita-selecta

  added / updated specs:
    - biopython
    - python=3.8
```

Environments

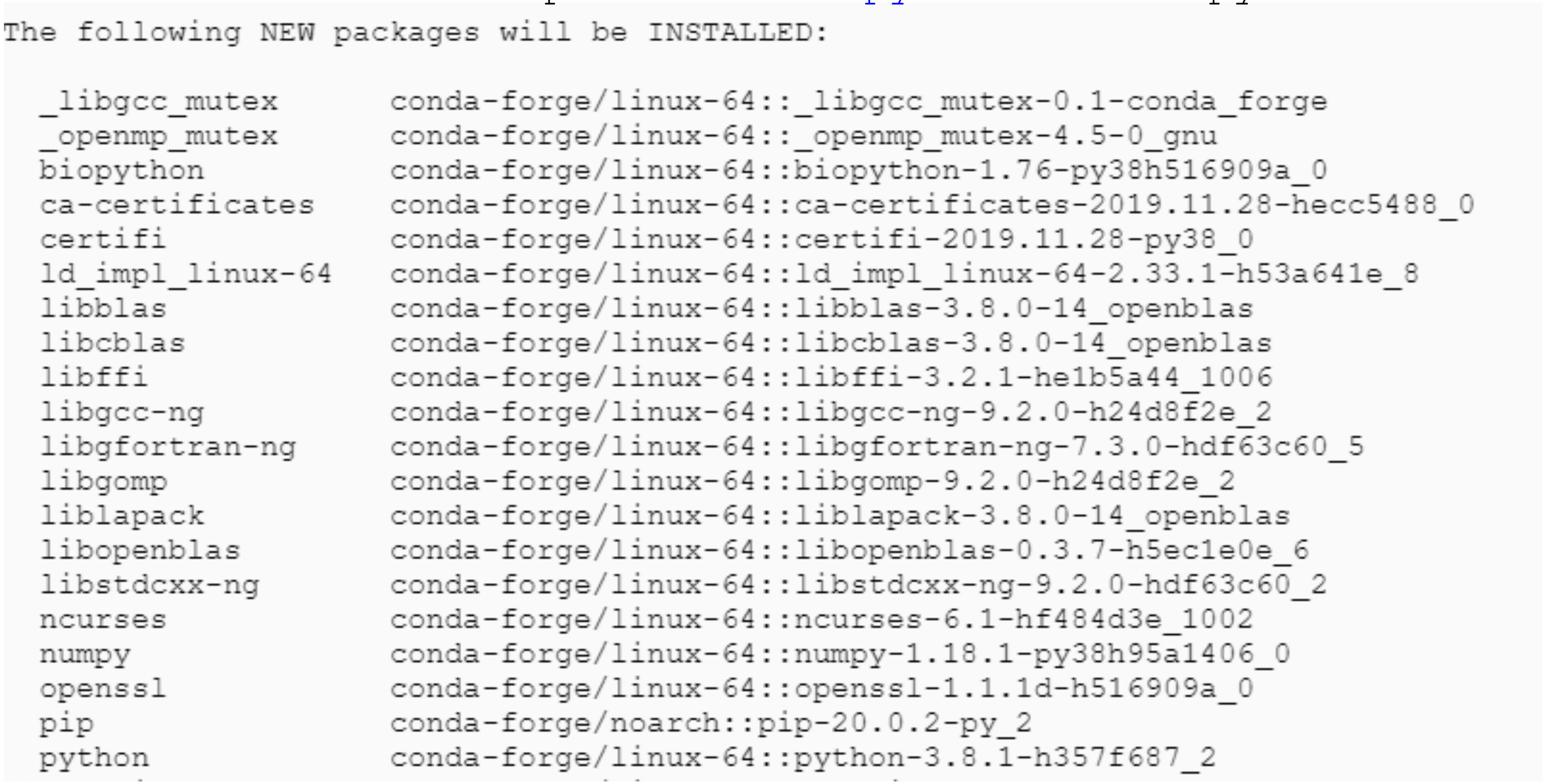
- Create a new environment and install a package in it
  - `conda create -n capita-selecta python=3.8 biopython`



```
The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    biopython-1.76             |   py38h516909a_0         2.6 MB  conda-forge
    ld_impl_linux-64-2.33.1    |       h53a641e_8         589 KB  conda-forge
    libgfortran-ng-7.3.0       |       hdf63c60_5         1.7 MB  conda-forge
    numpy-1.18.1               |   py38h95a1406_0         5.3 MB  conda-forge
    pip-20.0.2                 |             py_2         1.0 MB  conda-forge
    python-3.8.1               |       h357f687_2        58.2 MB  conda-forge
    setuptools-45.2.0          |          py38_0         655 KB  conda-forge
    wheel-0.34.2               |             py_1          24 KB  conda-forge
    ------------------------------------------------------------
                                           Total:        69.9 MB
```

Environments

- Create a new environment and install a package in it
  - `conda create -n capita-selecta python=3.8 biopython`



```
The following NEW packages will be INSTALLED:

  _libgcc_mutex      conda-forge/linux-64::_libgcc_mutex-0.1-conda_forge
  _openmp_mutex      conda-forge/linux-64::_openmp_mutex-4.5-0_gnu
  biopython          conda-forge/linux-64::biopython-1.76-py38h516909a_0
  ca-certificates    conda-forge/linux-64::ca-certificates-2019.11.28-hecc5488_0
  certifi            conda-forge/linux-64::certifi-2019.11.28-py38_0
  ld_impl_linux-64   conda-forge/linux-64::ld_impl_linux-64-2.33.1-h53a641e_8
  libblas            conda-forge/linux-64::libblas-3.8.0-14_openblas
  libcblas           conda-forge/linux-64::libcblas-3.8.0-14_openblas
  libffi             conda-forge/linux-64::libffi-3.2.1-he1b5a44_1006
  libgcc-ng          conda-forge/linux-64::libgcc-ng-9.2.0-h24d8f2e_2
  libgfortran-ng     conda-forge/linux-64::libgfortran-ng-7.3.0-hdf63c60_5
  libgomp            conda-forge/linux-64::libgomp-9.2.0-h24d8f2e_2
  liblapack          conda-forge/linux-64::liblapack-3.8.0-14_openblas
  libopenblas        conda-forge/linux-64::libopenblas-0.3.7-h5ec1e0e_6
  libstdcxx-ng       conda-forge/linux-64::libstdcxx-ng-9.2.0-hdf63c60_2
  ncurses            conda-forge/linux-64::ncurses-6.1-hf484d3e_1002
  numpy              conda-forge/linux-64::numpy-1.18.1-py38h95a1406_0
  openssl            conda-forge/linux-64::openssl-1.1.1d-h516909a_0
  pip                conda-forge/noarch::pip-20.0.2-py_2
  python             conda-forge/linux-64::python-3.8.1-h357f687_2
```

Environments

- Create a new environment and install a package in it
  - `conda create -n capita-selecta python=3.8 biopython`
- To use or “activate” the new environment
  - `conda activate capita-selecta`

Environments

- To see a list of all your environments
  - `conda info --envs`

```
# conda environments:
#
capita-selecta /home/tuur/.conda/envs/capita-selecta
base * /usr/local/Miniconda3-4.7.12.1-Linux-x86_64
```

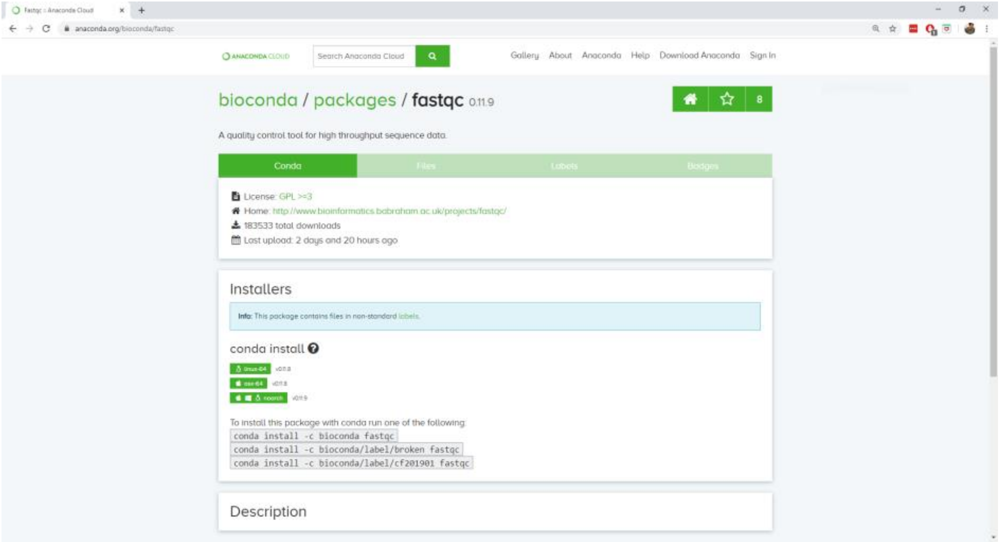
Environments

- Export
  - In order to export current environment:
    - `conda env export > capita-selecta.yml`
  - Or, to export any other environment:
    - `conda env export -n capita-selecta > capita-selecta.yml`

- Import
  - New environment from an environment definition:
    - `conda env create -n capita-selecta-from-file -f capita-selecta.yml`

Packages

- List all installed packages
  - `conda list`
- Search for a package:
  - If you're not sure if your package is available from conda, just google it!
  - `conda search fastqc`



bioconda / packages / fastqc 0.11.9

A quality control tool for high throughput sequence data.

License: GPL >=3
Home: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
183533 total downloads
Last upload: 2 days and 20 hours ago

Installers

Info: This package contains files in non-standard labels.

conda install

To install this package with conda run one of the following:

```
conda install -c bioconda fastqc
conda install -c bioconda/label/broken fastqc
conda install -c bioconda/label/cf201901 fastqc
```

Description

VIB

SCIENCE MEETS LIFE

Packages

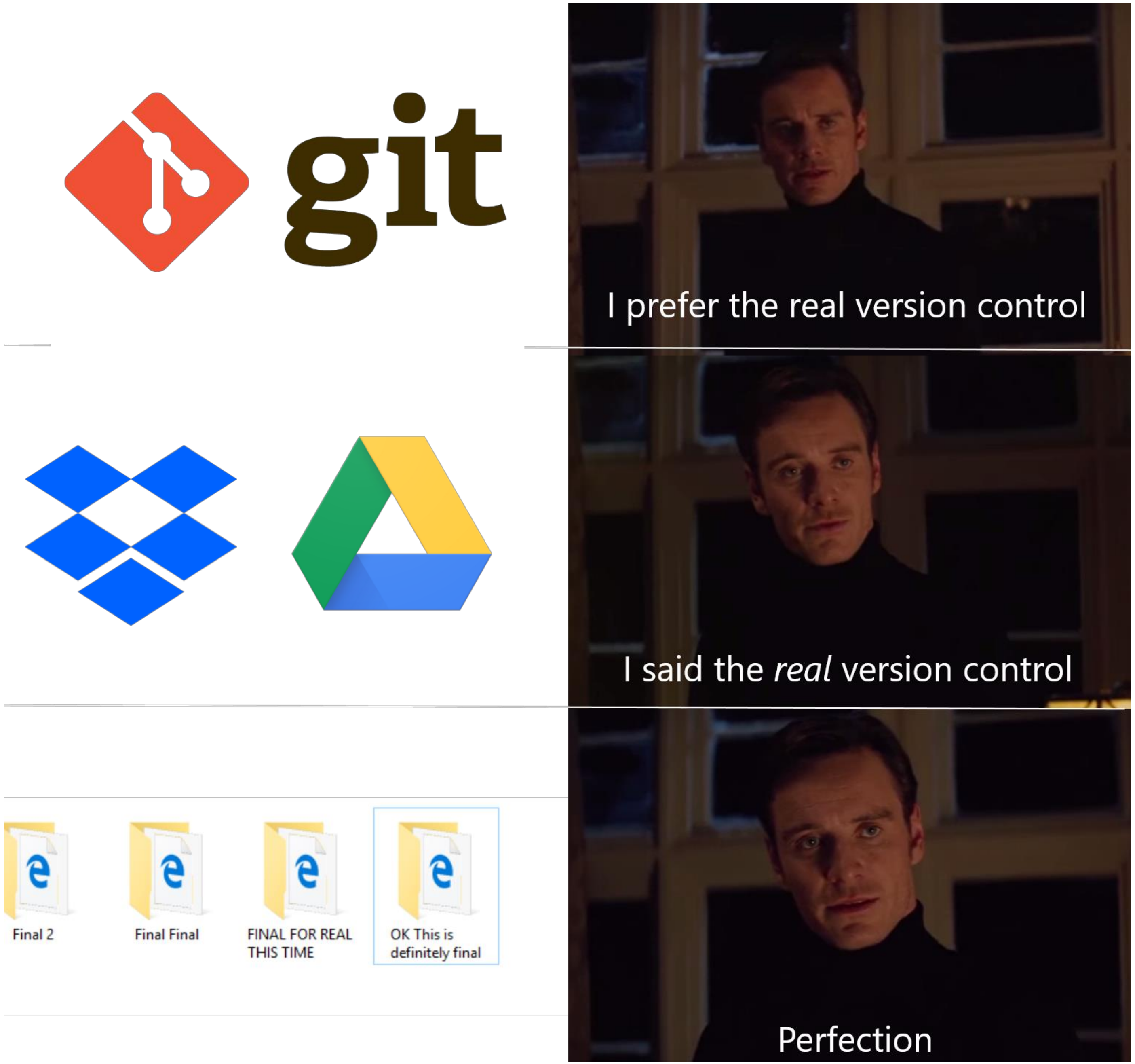
- List all installed packages
  - `conda list`
- Search for a package:
  - If you're not sure if your package is available from conda, just google it!
  - `conda search fastqc`
- Installing
  - If any other package is required, can be installed using conda:
    - `conda install seaborn`
    - `conda install fastqc=x.y.z`
  - Not in channel list:
    - `conda install -c conda-forge fastqc`

References and reading:

- https://docs.conda.io/en/latest/
- https://github.com/ifosch/conda-intro
- https://www.freecodecamp.org/news/why-you-need-python-environments-and-how-to-manage-them-with-conda-85f155f4353c/

Version controlling

Git & GitHub


git
I prefer the real version control
I said the *real* version control
Final 2
Final Final
FINAL FOR REAL THIS TIME
OK This is definitely final
Perfection

git

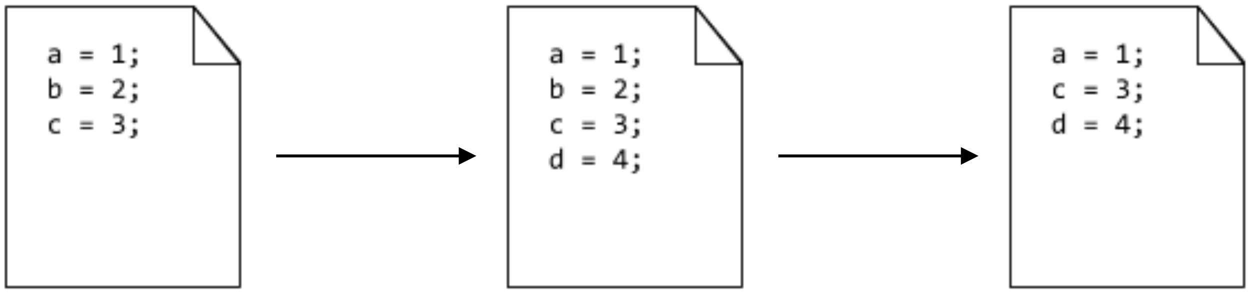
GitLab

GitHub

Bitbucket

Introduction

- What is Git used for?
  - Keep track of changes to your code



```
a = 1;
b = 2;
c = 3;
```

Create a file

```
a = 1;
b = 2;
c = 3;
d = 4;
```

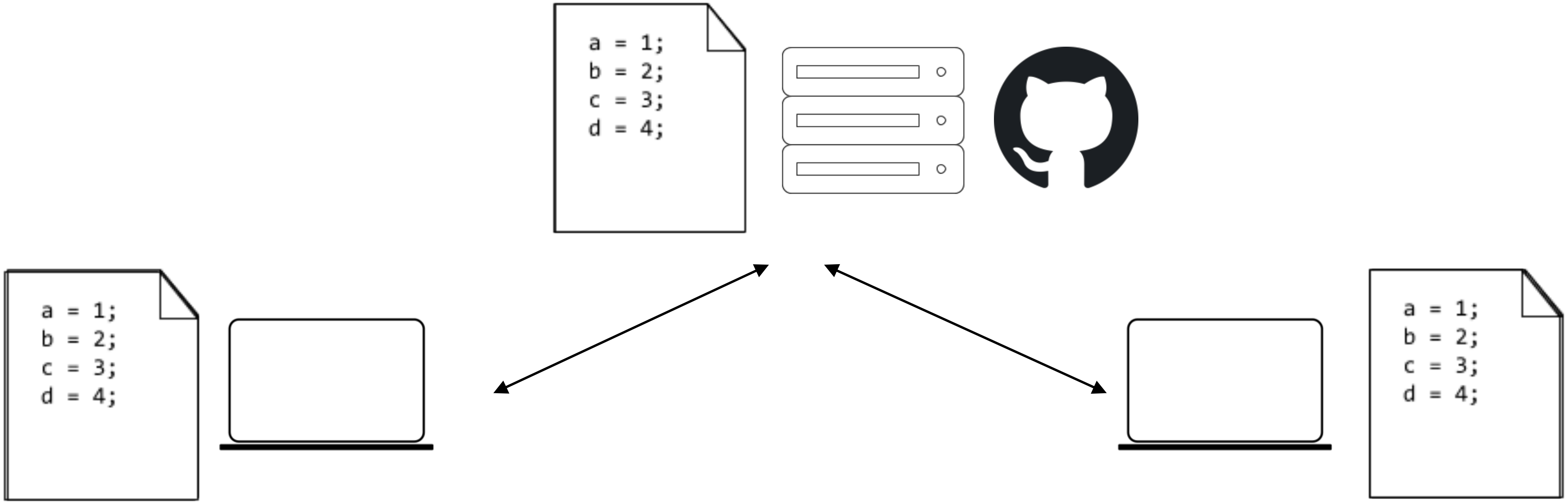
Add a line

```
a = 1;
c = 3;
d = 4;
```

Remove a line

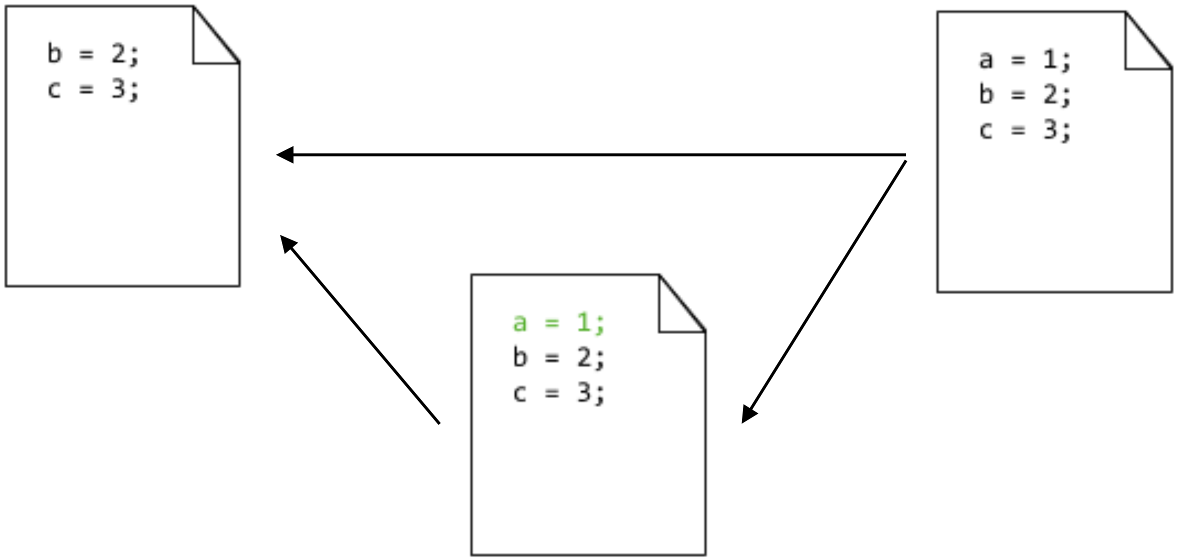
Introduction

- What is Git used for?
  - Keep track of changes to your code
  - Synchronize code between different people



Introduction

- What is Git used for?
  - Keep track of changes to your code
  - Synchronize code between different people
  - Testing new code



```
b = 2;
c = 3;
```

```
a = 1;
b = 2;
c = 3;
```

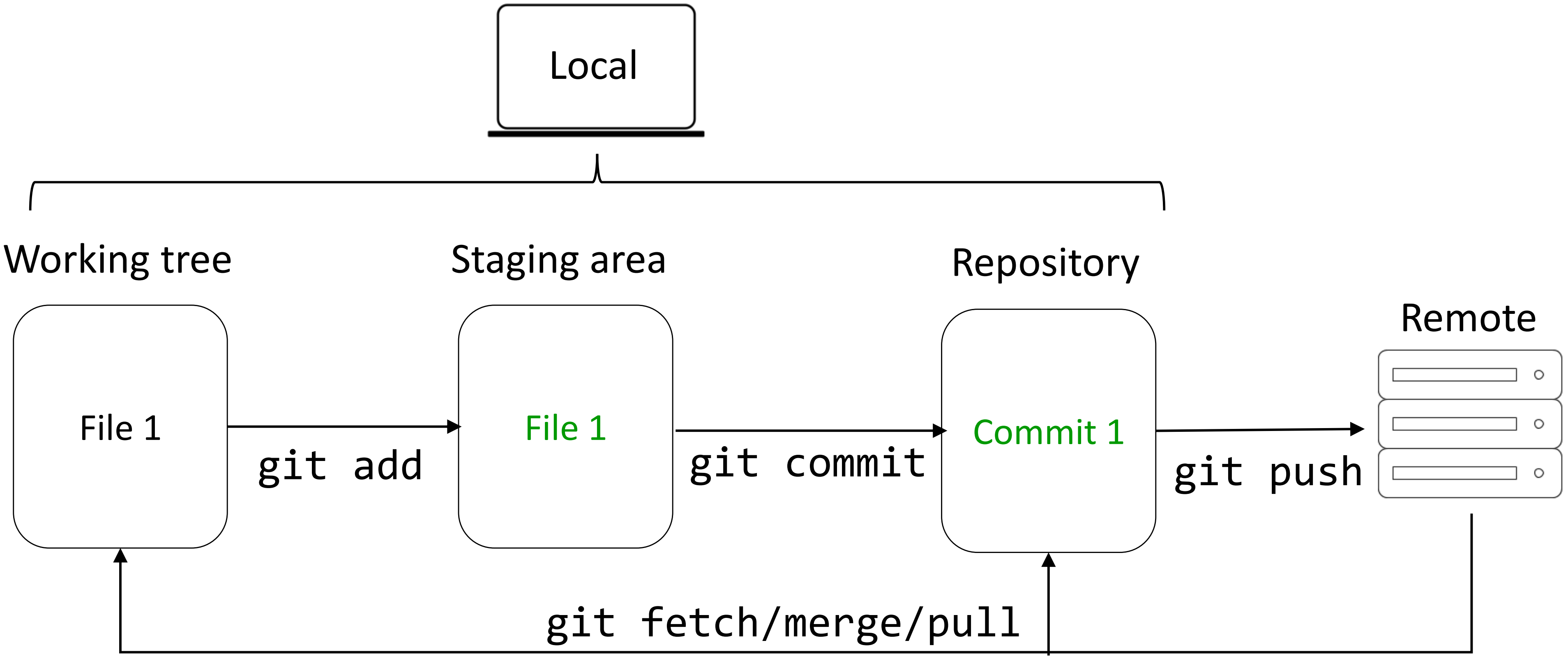
```
a = 1;
b = 2;
c = 3;
```

Introduction

- What is Git used for?
  - Keep track of changes to your code
  - Synchronize code between different people
  - Testing new code
  - Reverting back changes 🟩

Save changes

- Three conceptual areas of a repository



Local

Working tree

Staging area

Repository

Remote

File 1

File 1

Commit 1

git add

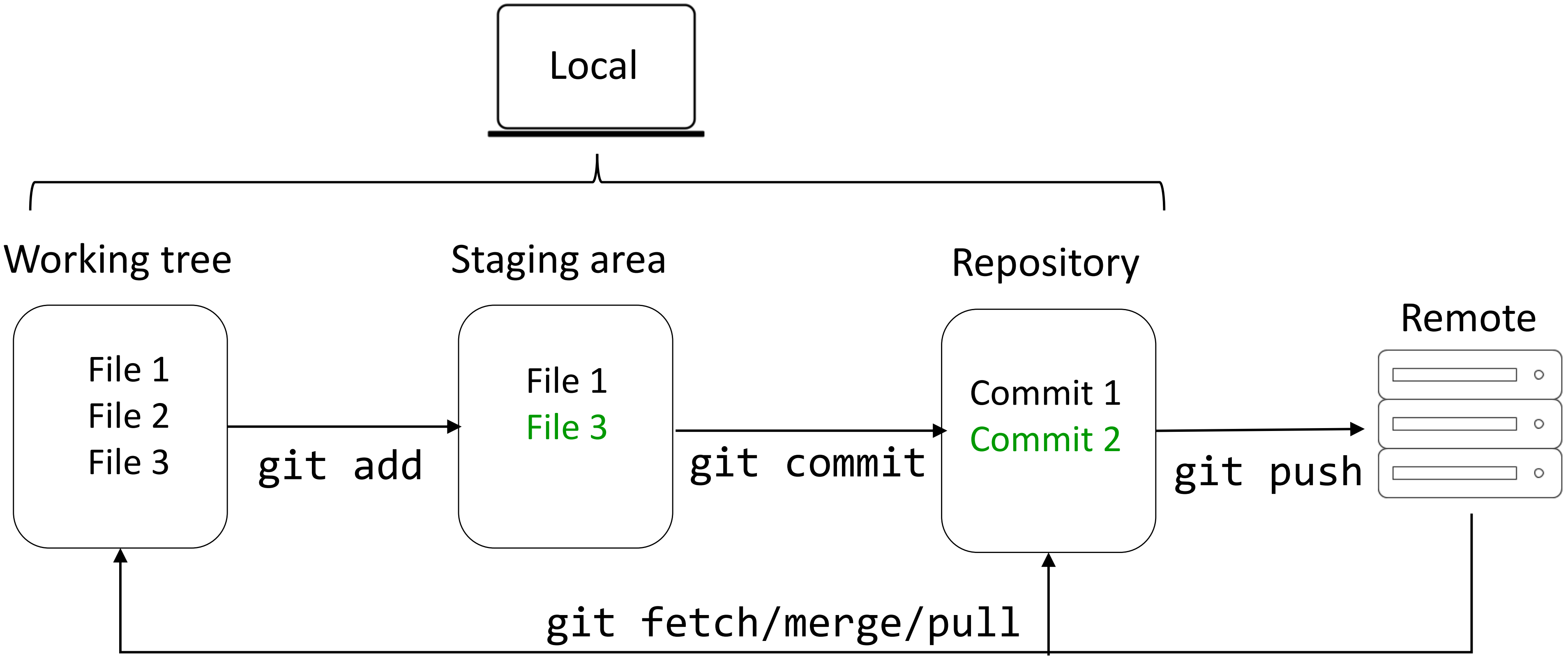
git commit

git push

git fetch/merge/pull

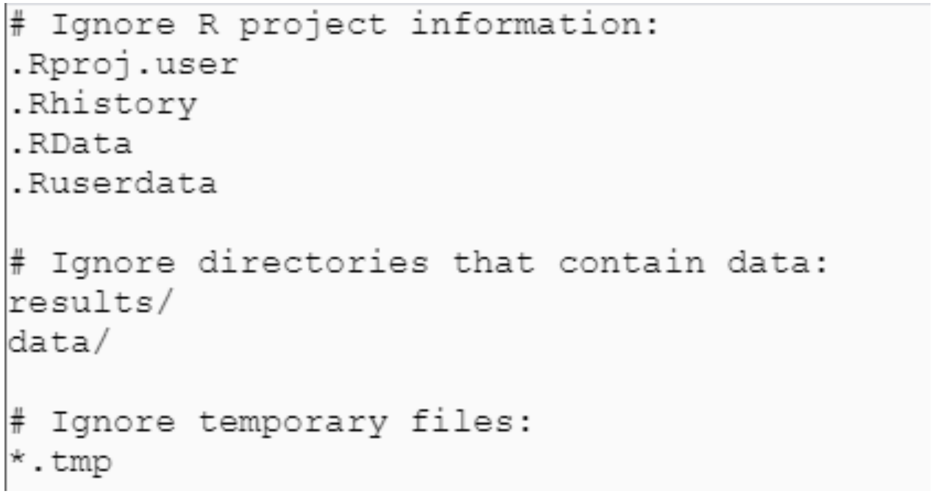
Save changes

- Three conceptual areas of a repository



.gitignore

- Ignore certain files or directories in repository
- E.g.: data files, results files, temporary files
- * wildcards



```
# Ignore R project information:
.Rproj.user
.Rhistory
.RData
.Ruserdata

# Ignore directories that contain data:
results/
data/

# Ignore temporary files:
*.tmp
```

SCIENCE MEETS LIFE

Workflow pipelines

Nextflow

November 2019

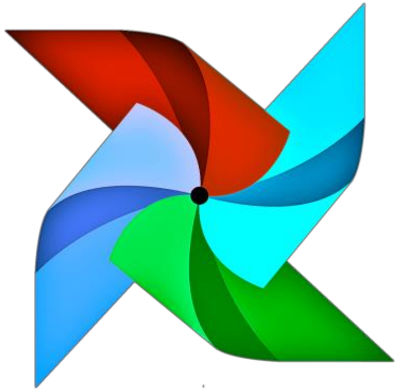
Tuur Muyldermans

Bash pipeline

```
#!/bin/bash
blastp -query sample.fasta -outfmt 6 \
      | head -n 10 \
      | cut -f 2 \
      | blastdbcmd -entry - > sequences.txt
```

Galaxy PROJECT

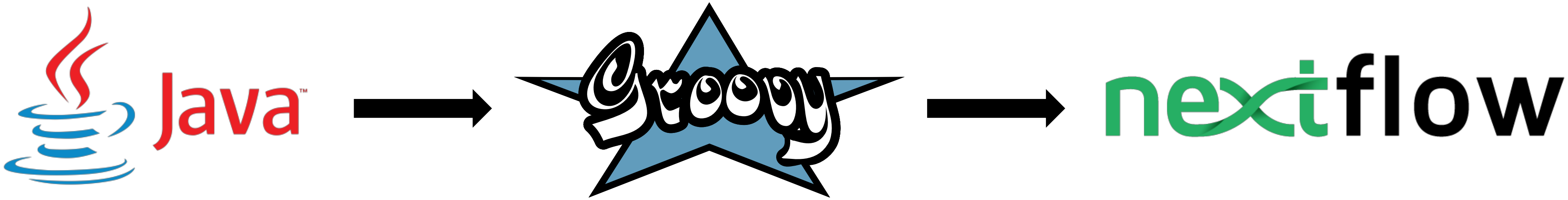
nextflow



Snakemake

Nextflow

- Reactive workflow framework and a programming DSL that eases the writing of data-intensive computational pipelines.
- Scripting language:

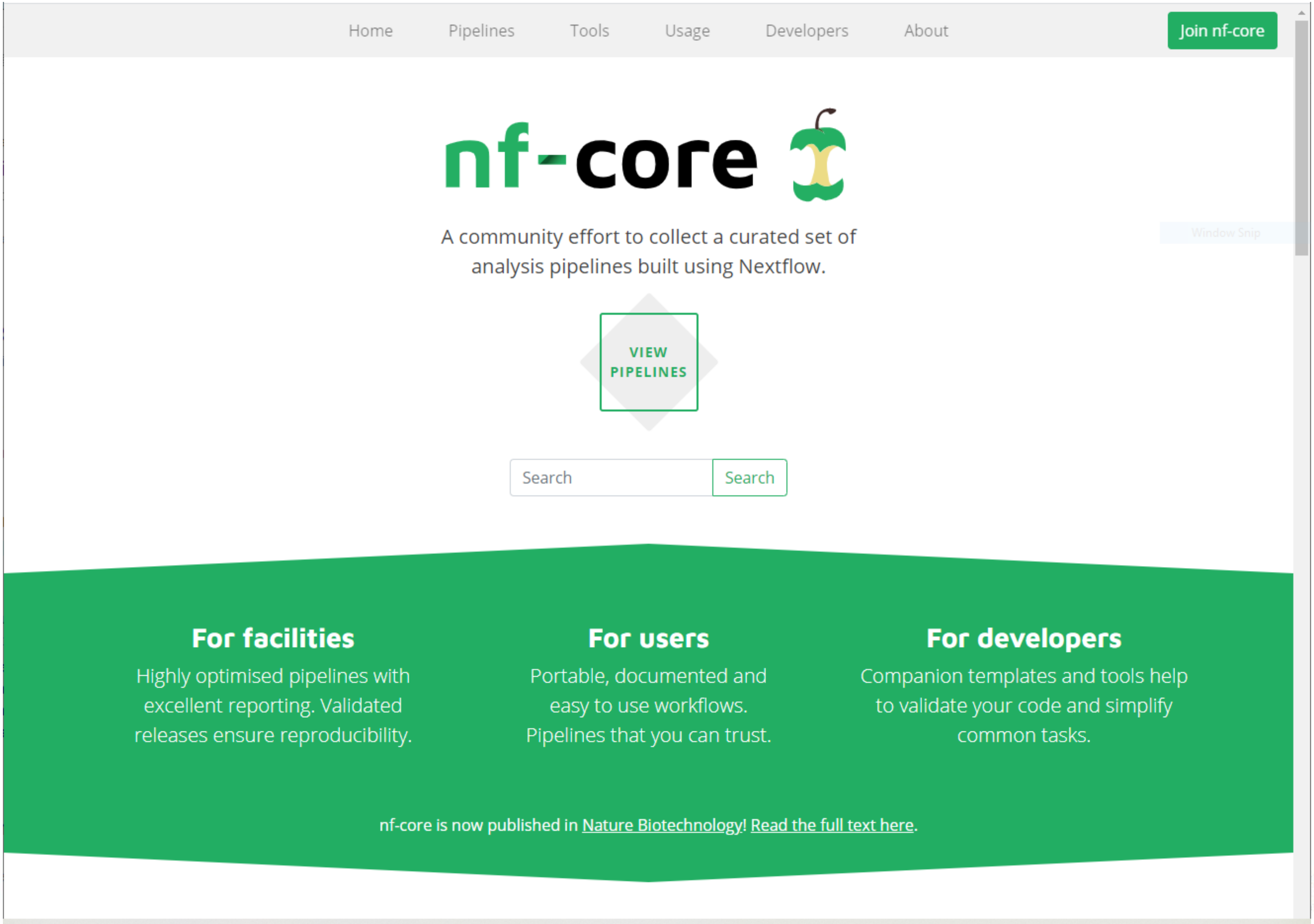


Why (not)?

nextflow

+ Parallelization
+ Highly scalable and portable
+ Reproducible (native support of containers)
+ Continuous checkpoints for resuming / expanding pipelines

- Groovy
- Not made for simple pipelines



A community effort to collect a curated set of analysis pipelines built using Nextflow.

For facilities

Highly optimised pipelines with excellent reporting. Validated releases ensure reproducibility.

For users

Portable, documented and easy to use workflows. Pipelines that you can trust.

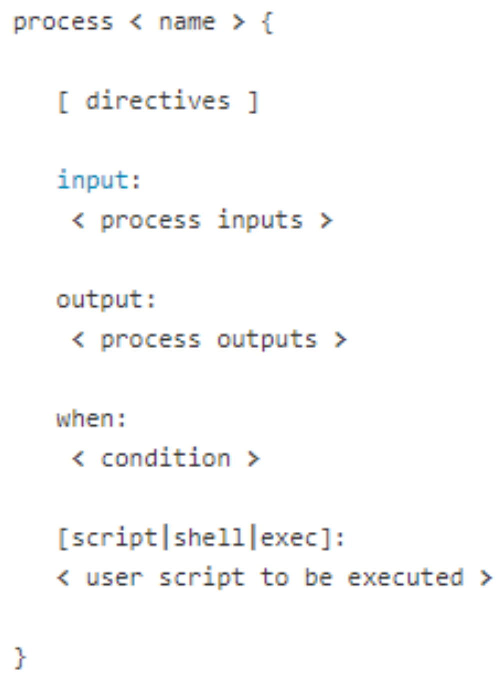
For developers

Companion templates and tools help to validate your code and simplify common tasks.

nf-core is now published in Nature Biotechnology! Read the full text here.

Processes

- Five definition blocks
  - Directives
  - Inputs
  - Outputs
  - When clause
  - Process script

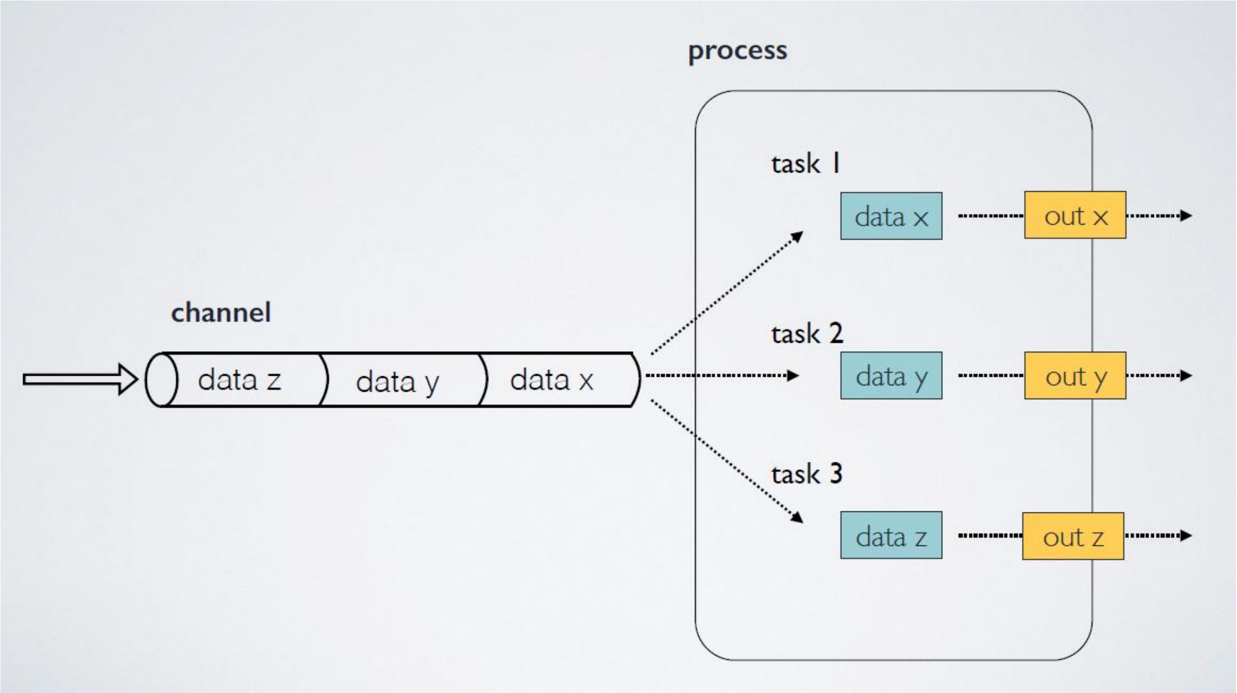


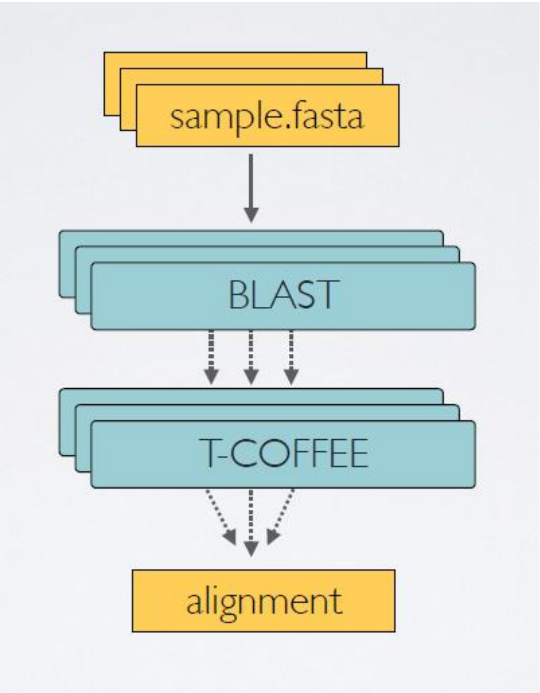
```
process < name > {

   [ directives ]

   input:
    < process inputs >

   output:
    < process outputs >

   when:
    < condition >

   [script|shell|exec]:
   < user script to be executed >

}
```

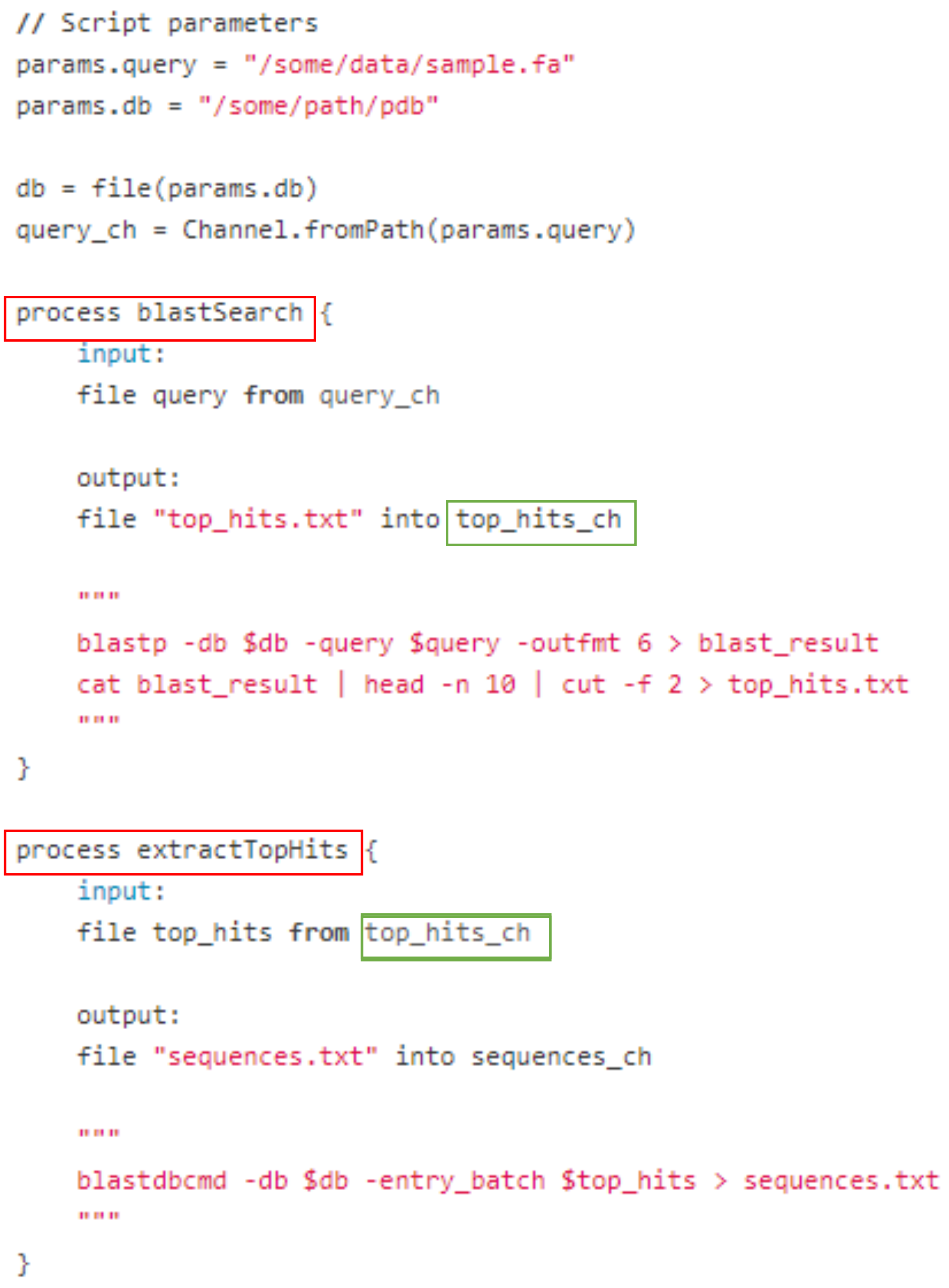
- In any language (Bash, Python, Perl, Ruby, etc.)
- Executed independently & isolated
- Processes communicate via asynchronous FIFO queues = *channels*

Processes & channels

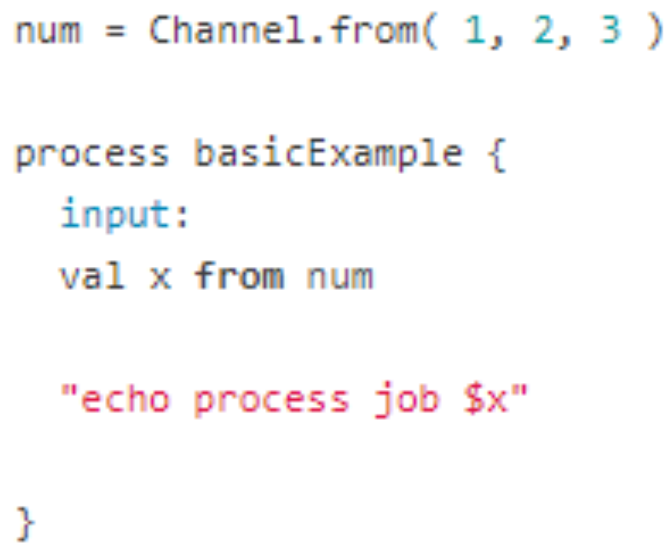
- Processes are linked via channels: one process will wait for the output of another and then runs reactively when the channel has contents



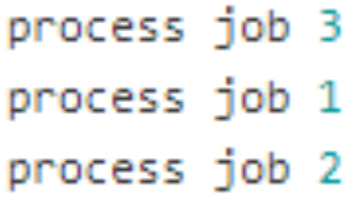




```
// Script parameters
params.query = "/some/data/sample.fa"
params.db = "/some/path/pdb"

db = file(params.db)
query_ch = Channel.fromPath(params.query)

process blastSearch {
    input:
    file query from query_ch

    output:
    file "top_hits.txt" into top_hits_ch

    """
    blastp -db $db -query $query -outfmt 6 > blast_result
    cat blast_result | head -n 10 | cut -f 2 > top_hits.txt
    """
}

process extractTopHits {
    input:
    file top_hits from top_hits_ch

    output:
    file "sequences.txt" into sequences_ch

    """
    blastdbcmd -db $db -entry_batch $top_hits > sequences.txt
    """
}
```



```
num = Channel.from( 1, 2, 3 )

process basicExample {
  input:
  val x from num

  "echo process job $x"

}
```



```
process job 3
process job 1
process job 2
```

Execution abstraction

- *Executor* determines *how* the script is run on the target system
- By default: locally
- Alternatively: HPC or cloud platforms

Schedulers

GRID ENGINE
Sun Grid Engine
(SGE)

slurm
workload manager

LSF
Platform Load Sharing
Facility

PBS Works™
Portable Batch System

HTCondor
High Throughput Computing

Cloud platforms

kubernetes

amazon
webservices™

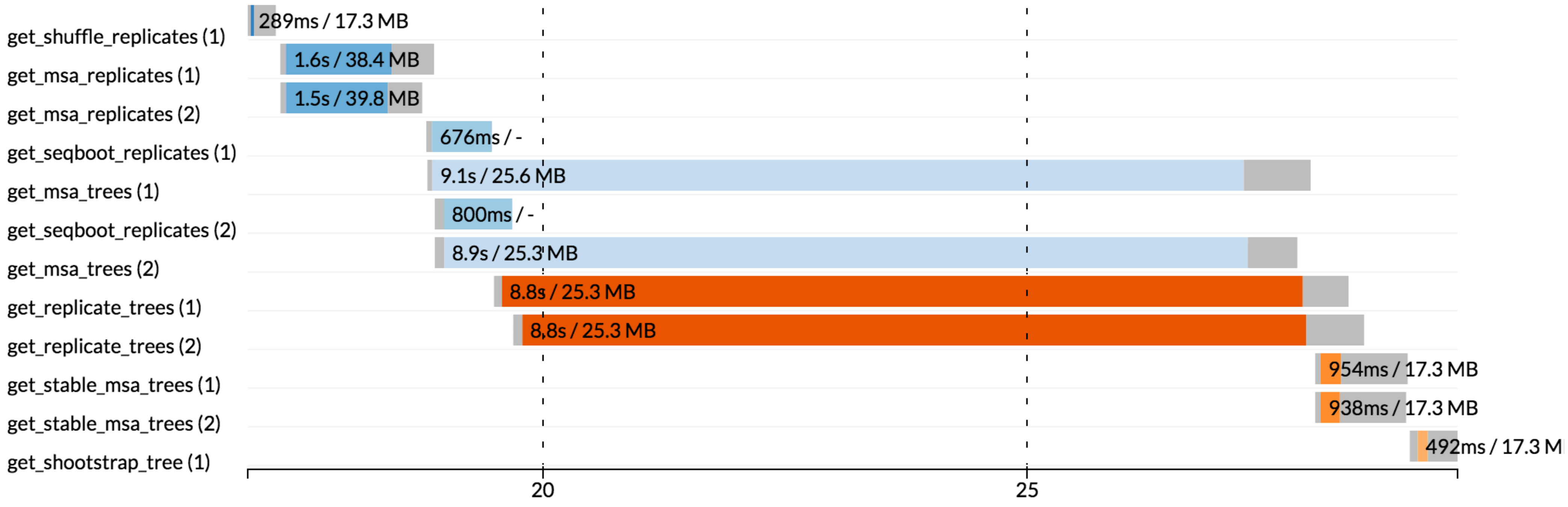
Google Cloud Platform

.config

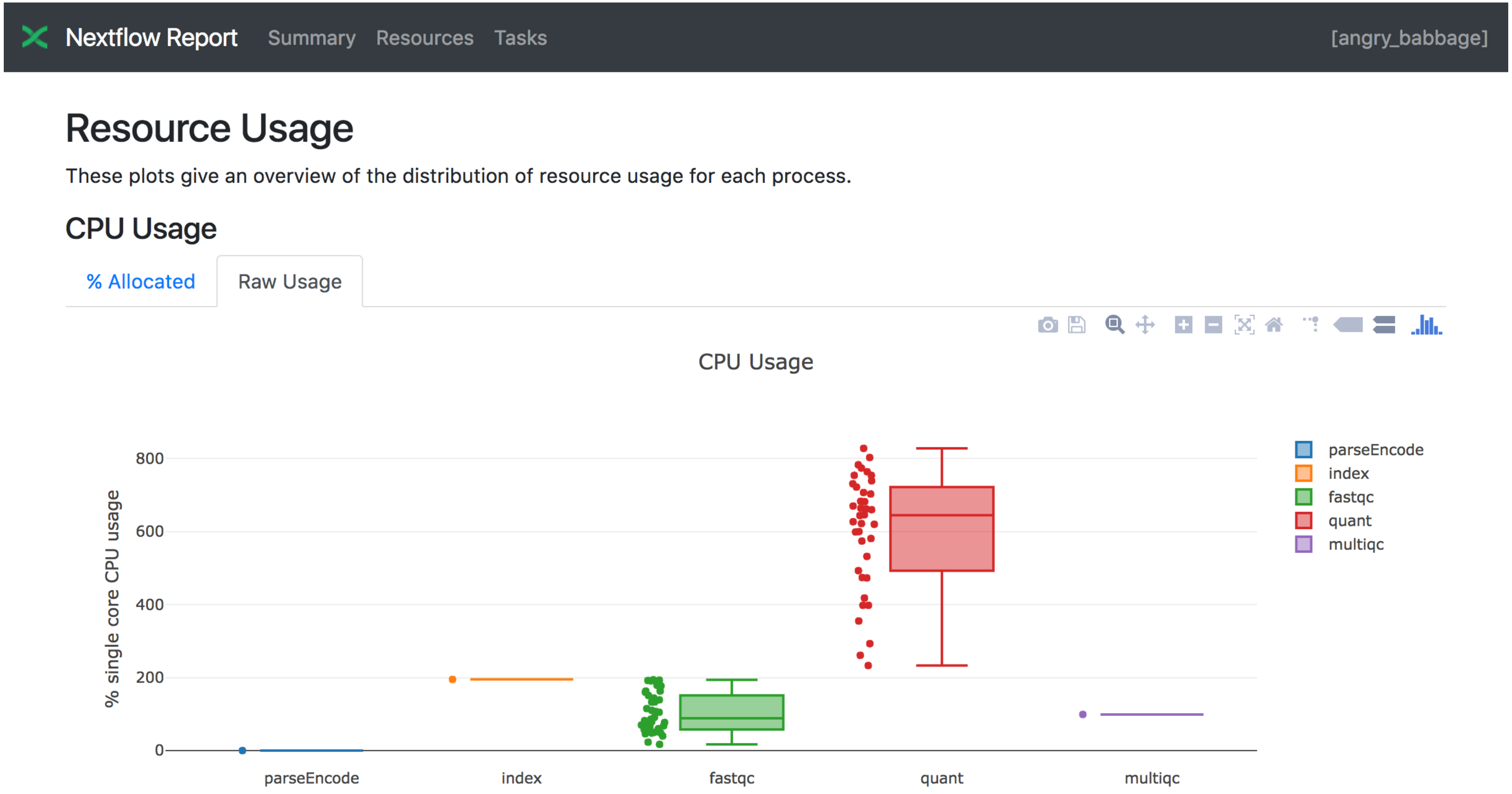
- Local or cluster usage

```
executor {
  cpus = 4
}

process {
  executor = 'sge’
  penv = 'smp'
 clusterOptions = { "-V -S /bin/bash " }
}
```

Output & report



Output & report



Resource Usage

These plots give an overview of the distribution of resource usage for each process.

CPU Usage

Galaxy PROJECT

- Web-based platform
- Built-in integration with many tools and datasets
- Little control over tasks parallelization
- Easy-to-use
- Suited for training/learning and non-experienced users

nextflow

- Command-line oriented tool
- Can incorporate any tool
- Fine control over parallelization and parameters
- Learning curve
- Suited for production workloads & experienced bioinformaticians

Snakemake

- Command-line oriented tool
- Pull model
- Python based
- Compute DAG ahead
- Support for sub-workflows
- …

nextflow

- Command-line oriented tool
- Push model
- Java/Groovy based
- Compute DAG at runtime
- Working on sub-workflows
- …

References & further reading

- https://www.nextflow.io/docs/latest/getstarted.html
- https://github.com/nextflow-io

SCIENCE MEETS LIFE

training.vib.be

Chan Zuckerberg Initiative

- Pip
- Bioconda
- Docker and R
- JupyterHub

SCIENCE MEETS LIFE

Containerization

Docker & Singularity

Slides:

- https://material.bits.vib.be/courses/?https://raw.githubusercontent.com/vibbits/material-liascript/master/slides-docker-introduction.md#12