Tools for Reproducible Research

Managing dependencies

Managing and executing analysis workflow

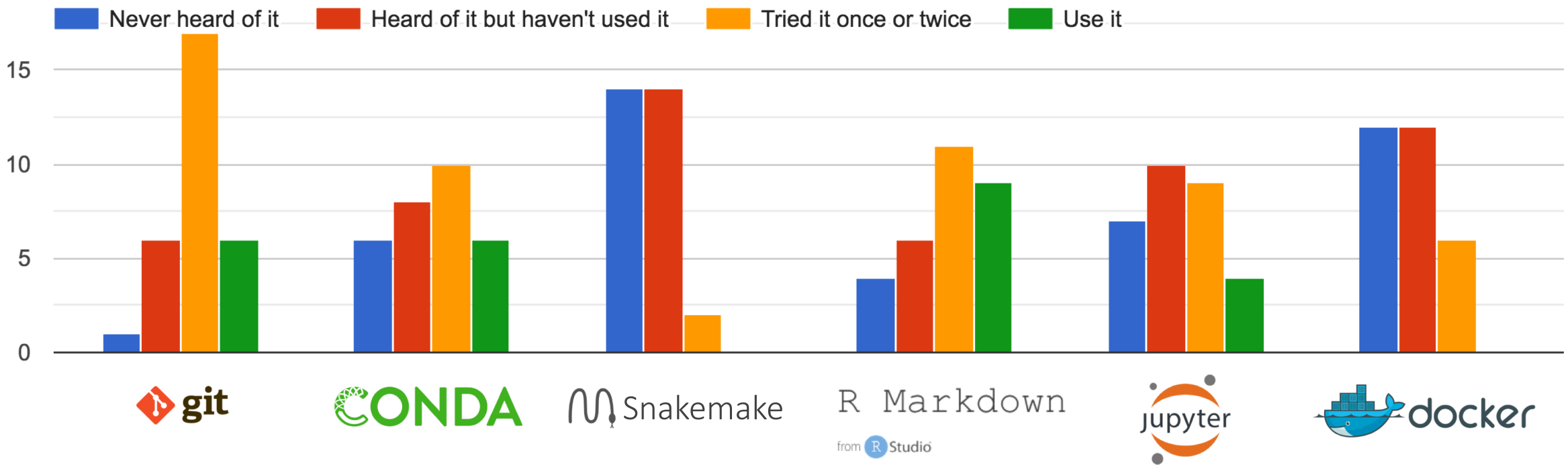
Versioning and collaborating on code (and some other files)

Connecting code and reporting

and…

Isolating and exporting environment

Student experience



Managing dependencies

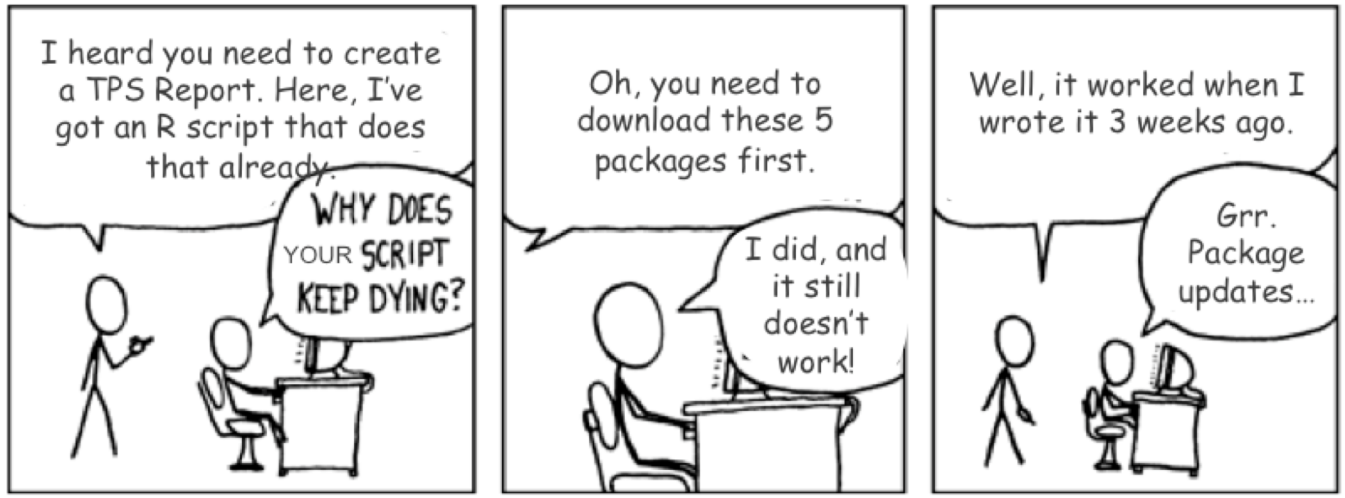
Managing and executing analysis workflow

Versioning and collaborating on code (and some other files)

Connecting code and reporting

and…

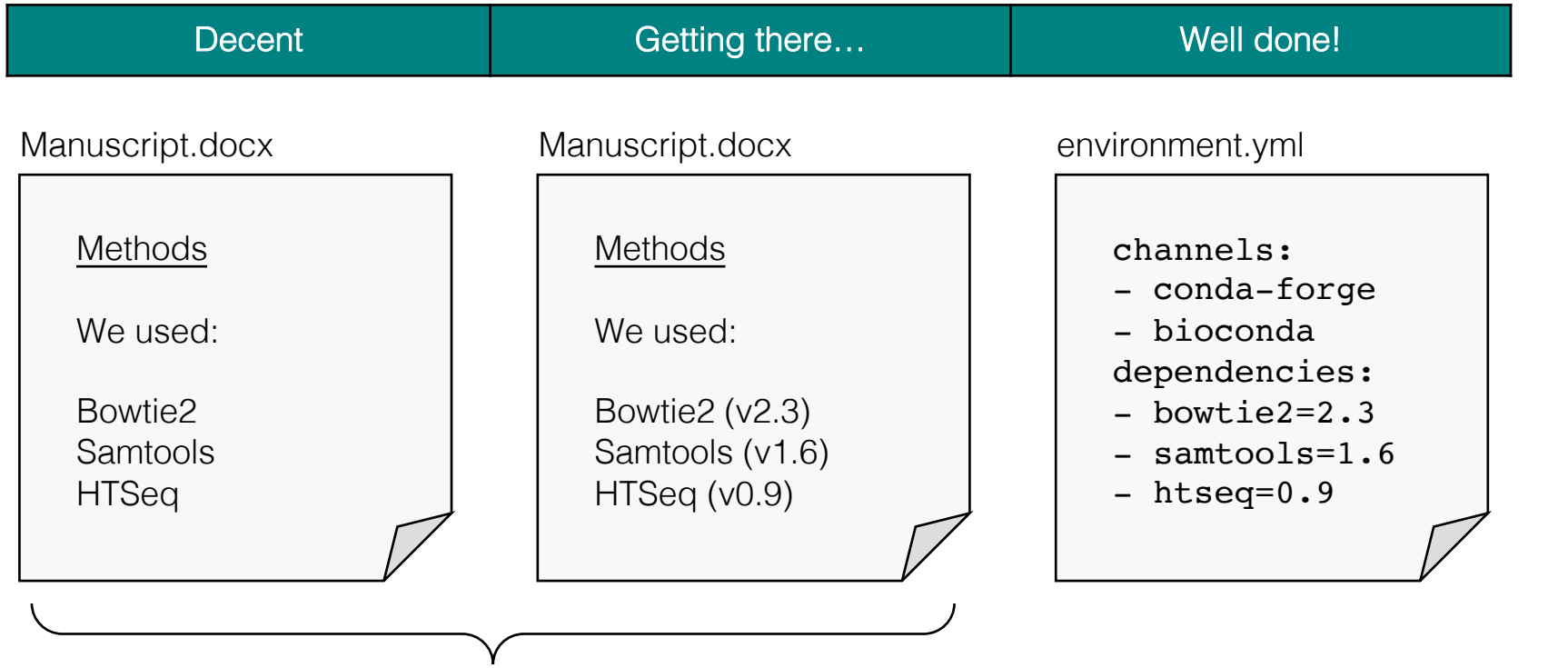
Isolating and exporting environment



Full reproducibility requires the possibility to recreate the system that was originally used to generate the results.

What is Conda?

- Conda is a package, dependency, and environment manager.
- Works for software developed in any programming language.



Package manager

- Conda package: compressed tarball (system-level libraries, Python or other modules, executable programs, or other components).
- Conda keeps track of the dependencies between packages and platforms.
- Conda packages are downloaded from remote channels.

```
$ conda install -c conda-forge matplotlib

Fetching package metadata ...........
Solving package specifications: ..........

Package plan for installation in environment /Users/varemo/Applications/miniconda2/envs/test-r2:

The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    sqlite-3.13.0              |                1         1.4 MB  conda-forge
    libpng-1.6.24              |                0         338 KB  conda-forge
    python-2.7.12              |                1        11.8 MB  conda-forge
    certifi-2016.8.31          |           py27_0         218 KB  conda-forge
    freetype-2.6.3             |                1         782 KB  conda-forge
    functools32-3.2.3.2        |           py27_1          16 KB  conda-forge
    numpy-1.11.1               |           py27_0         3.1 MB  defaults
    pyparsing-2.1.8            |           py27_0          89 KB  conda-forge
    pytz-2016.6.1              |           py27_0         183 KB  conda-forge
    six-1.10.0                 |           py27_0          18 KB  conda-forge
    cycler-0.10.0              |           py27_0          13 KB  conda-forge
    python-dateutil-2.5.3      |           py27_0         236 KB  conda-forge
    setuptools-26.1.1          |           py27_0         346 KB  conda-forge
    matplotlib-1.5.3           |      np111py27_0         4.1 MB  conda-forge
    wheel-0.29.0               |           py27_0          81 KB  conda-forge
    pip-8.1.2                  |           py27_0         1.5 MB  conda-forge
    ------------------------------------------------------------
                                          Total:        24.2 MB

The following NEW packages will be INSTALLED:

$ python
>>> import matplotlib
```

Environment manager

- Conda environment: directory that contains a specific collection of Conda packages that you have installed.
- Packages are symlinked between environments to avoid duplication.

```
$ conda create --name env1 -c bioconda fastqc
$ fastqc --version
-bash: fastqc: command not found
$ source activate env1
$(env1)fastqc --version
FastQC v0.11.5
$(env1)source deactivate
$ conda create --name env2 -c bioconda python=3 snakemake
$ python --version
Python 2.7.12 :: Continuum Analytics, Inc.
$ snakemake --version
-bash: snakemake: command not found
$ source activate env2
$(env2)python --version
Python 3.4.3 :: Continuum Analytics, Inc.
$(env2)snakemake --version
3.7.1
$(env2)
```

Defining and sharing environments

environment.yml

```
channels:
- conda-forge
- bioconda
dependencies:
- fastqc=0.11
- sra-tools=2.8
- snakemake=4.3.0
- multiqc=1.3
- bowtie2=2.3
- samtools=1.6
- htseq=0.9
- graphviz=2.38.0
```

- Create an environment from specifications in a file.
- All additional dependencies will be included.
- The environment.yml file can be shared with others and used to recreate the environment on other systems.

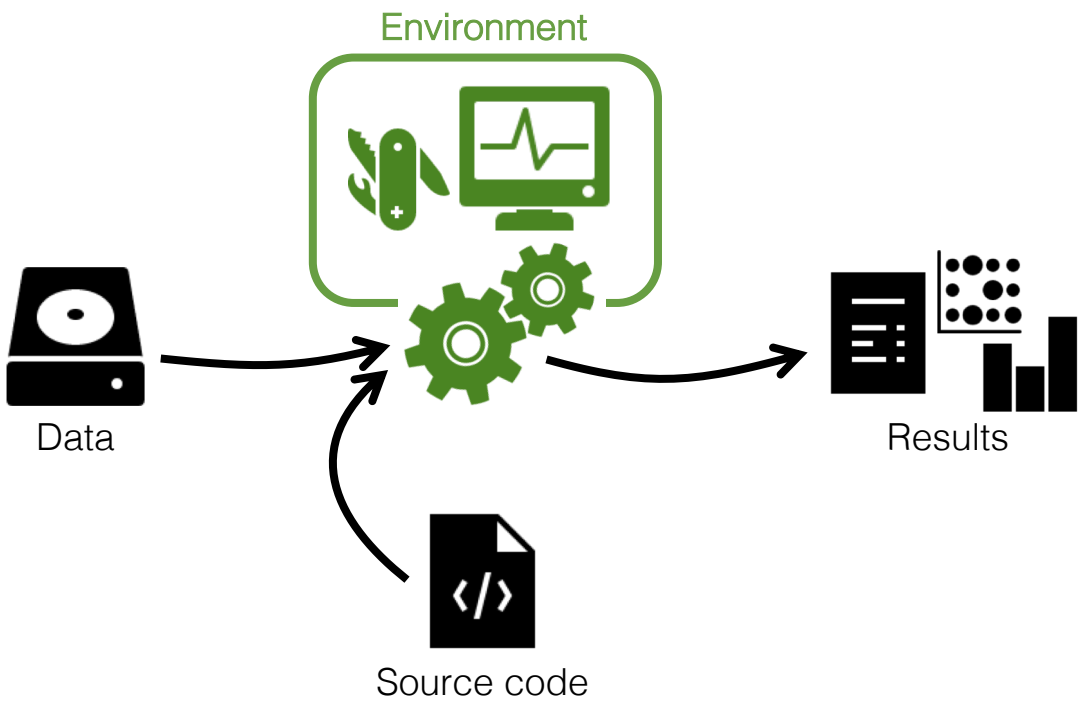
```
$ conda env create --name project_a -f environment.yml
```

- Update existing environment after adding new packages to environment.yml:

```
$ conda env update -f environment.yml
```

- Export existing environment as new yaml file (also includes dependencies):

```
$ conda env export > environment_full.yml
```




```
project
|- doc/
|
|- data/
|   |- raw_external/
|   |- raw_internal/
|   |- meta/
|
|- code/
|- notebooks/
|
|- intermediate/
|- scratch/
|- logs/
|
|- results/
|   |- figures/
|   |- tables/
|   |- reports/
|
|- Snakefile
|- config.yml
|- environment.yml
|- Dockerfile
```

The tutorials

A few practical notes…

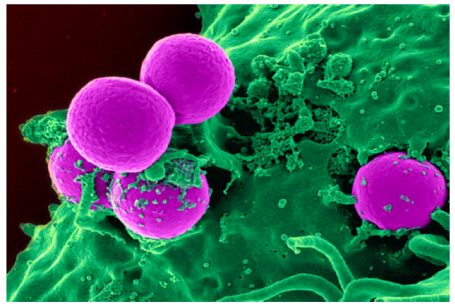
OPEN ACCESS Freely available online

PLOS ONE

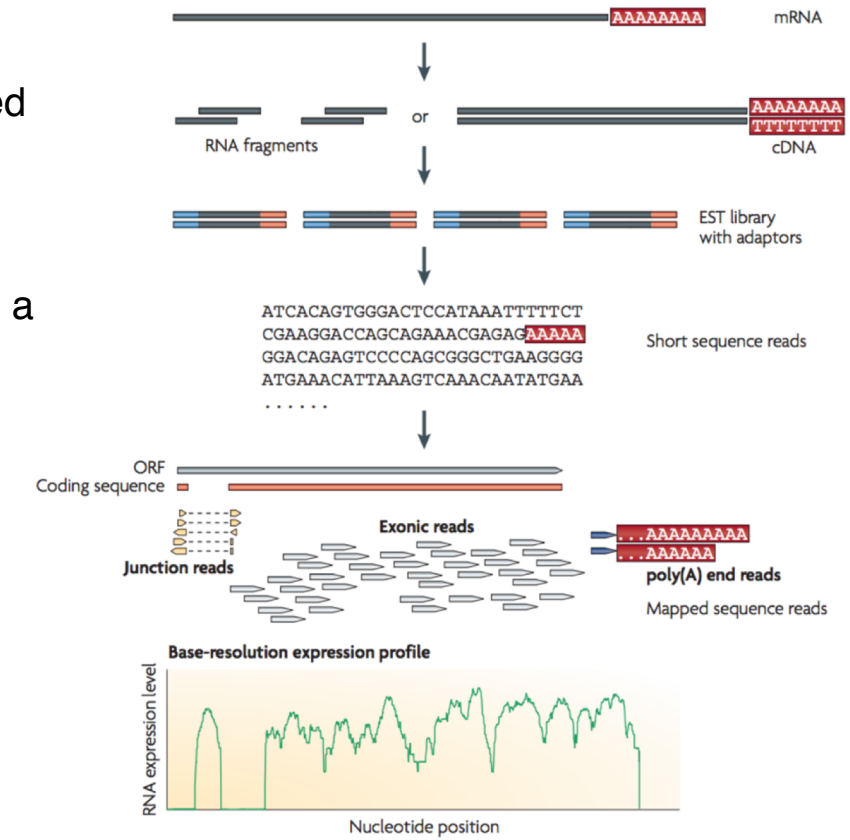
RNA-Seq Reveals Differential Gene Expression in *Staphylococcus aureus* with Single-Nucleotide Resolution

Joseph Osmundson[1*], Scott Dewell[2], Seth A. Darst[1]

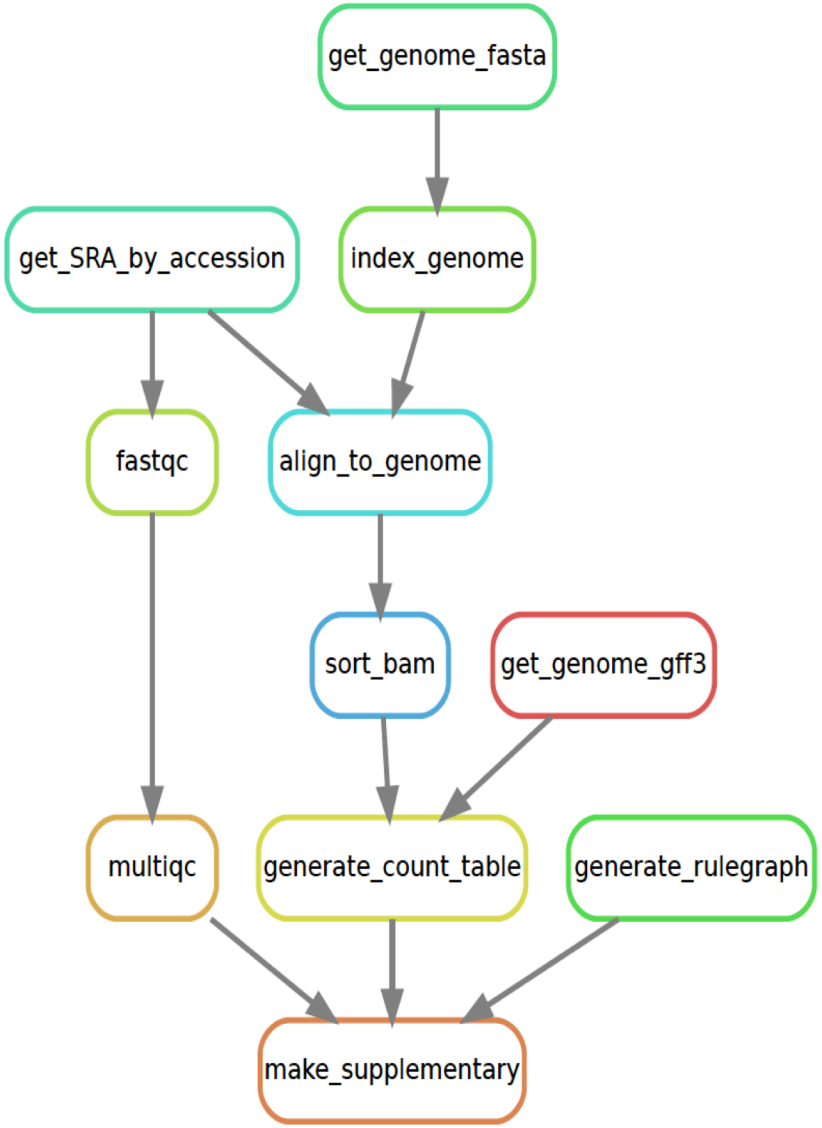
- Methicillin-resistant *Staphylococcus aureus* (MRSA):
  - is resistant to broad spectrum beta-lactam antibiotics
  - lead to difficult-to-treat infections in humans
- Lytic bacteriophages have been suggested as potential therapeutic agents, or as the source of novel antibiotic proteins or peptides.
- One such protein, gp67, was identified as a transcription-inhibiting transcription factor with an antimicrobial effect.
- To identify *S. aureus* genes repressed by gp67, the authors expressed gp67 in *S. aureus* cells.
- RNA-seq was performed on *S. aureus* strains:
  - RN4220 with pRMC2 with gp67
  - RN4220 with empty pRMC2
  - NCTC8325-4



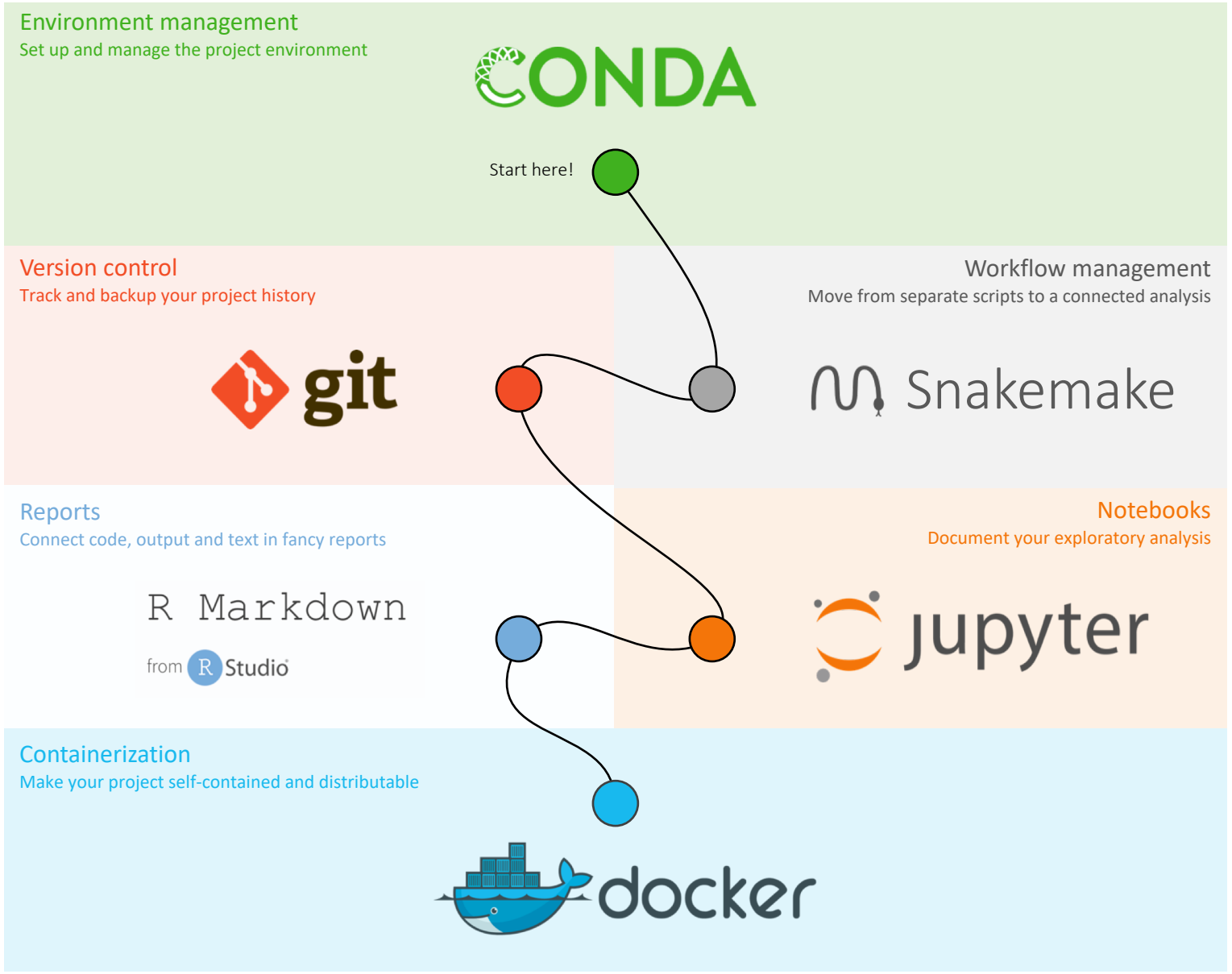
Scanning electron micro-graph of a human neutrophil ingesting MRSA

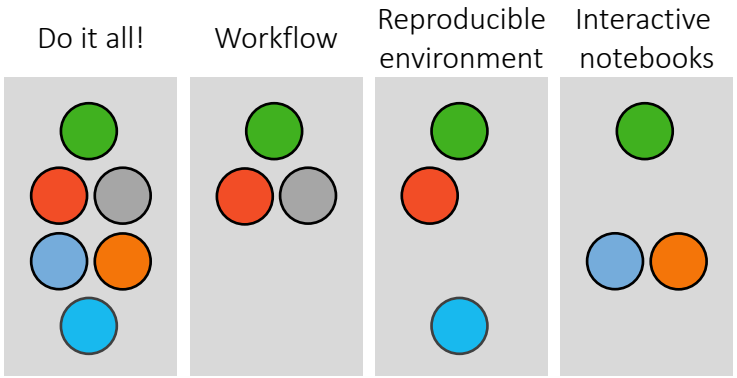


The analysis workflow



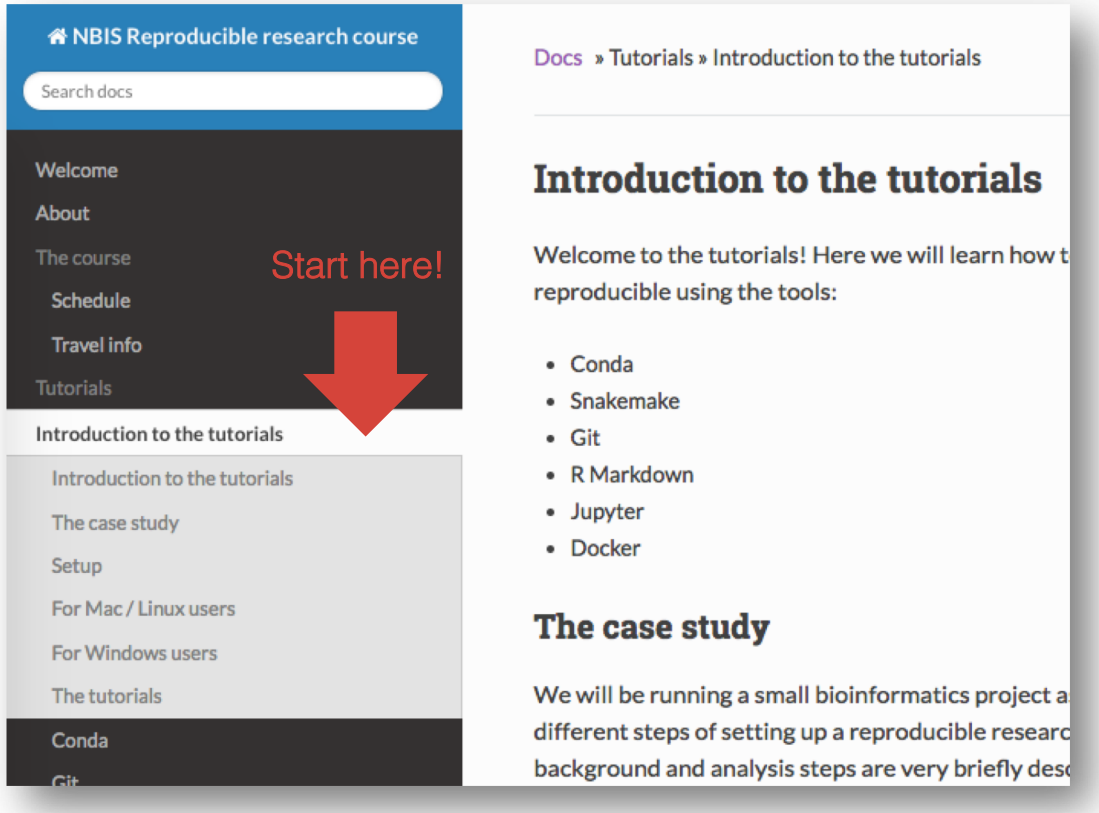
The tutorials





Getting started

- Clone course git repository to get all files needed for tutorials!
- Each tutorial will run in a specific subdirectory within `reproducible_research_course`, make sure you are running from the right place!
- Exception: the git tutorial will be run in a user-created directory outside of `reproducible_research_course`.



http://nbis-reproducible-research.readthedocs.io