

Andrea Telatin  
twitter: [@telatin](https://twitter.com/telatin)  
CPAN: **PROCH**




---

**conda install my-module**

---

---

# How did I get here?

- I'm **not** a computer scientist (I'm a molecular biologist by training)
- I learned Perl  in 2004, when it was the most popular scripting language among **bioinformaticians**
- Slides and example code on GitHub  
<https://github.com/telatin/TPC2020>

---

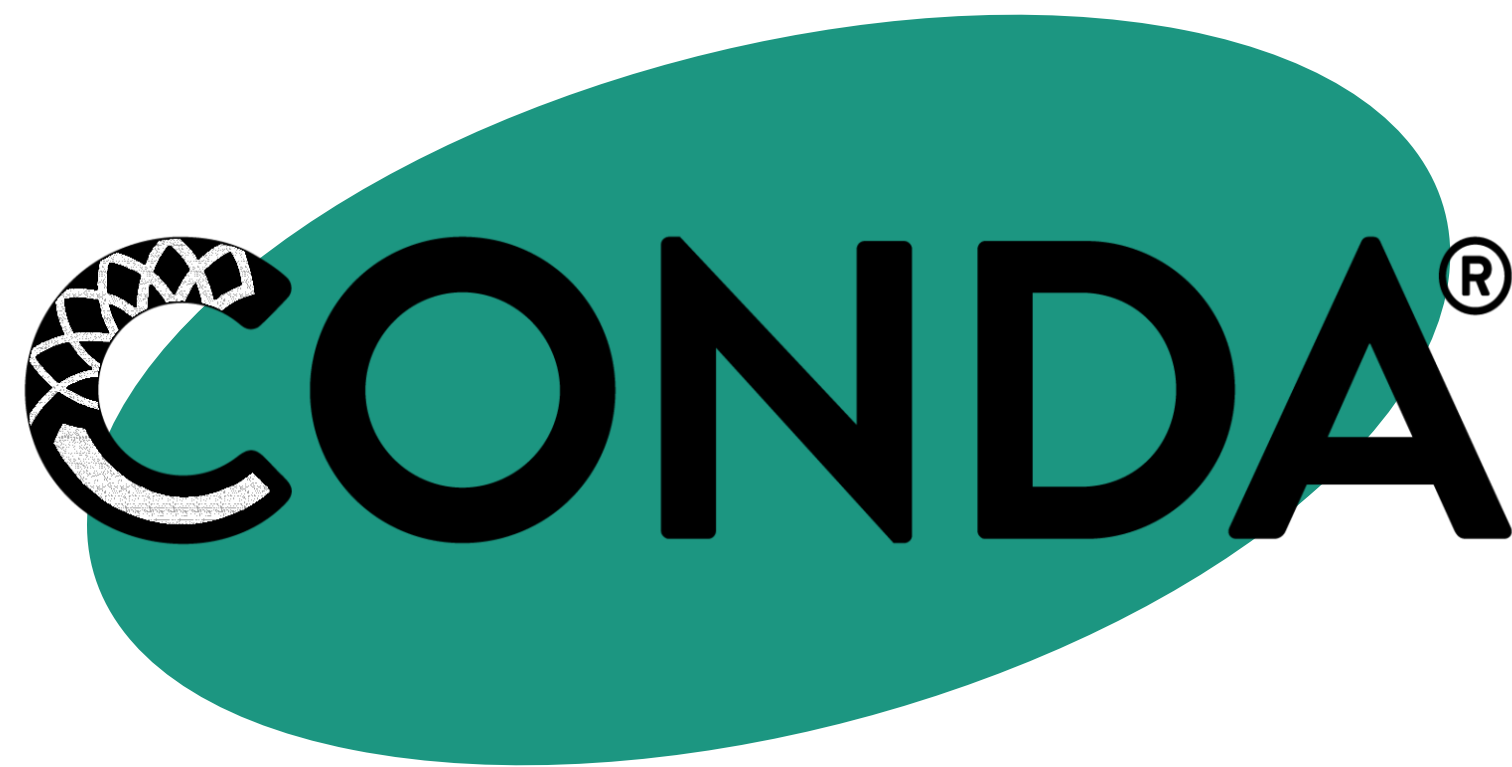
# The problem

- For **BIOINFORMATICIANS**:
  - Our analyses rely on a vast number of software tools
  - Sometimes incompatible versions of tools / modules etc.
- For **Perl programs**:
  - Non developers would struggle installing modules (and cpanm...)



---

# The solution



<https://anaconda.org>

- **PACKAGE MANAGER:**
  - Will install packages and their dependencies picking the right versions
  - `conda install package-name`
- **ENVIRONMENT MANAGER:**
  - Allows to use different versions of modules/tools in isolated “environments”
  - `conda create -n environment_1`
  - `conda activate environment_1`

---

# Conda “channels”

- **CONDA CHANNELS**

- Conda channels are the locations where packages are stored
- Any user signing up can create their own channel

- e.g. `conda install -c conda-forge htop=2.2.0`

- **The BioConda channel** <https://bioconda.github.io/>

- >7.000 packages (of which >700 Perl modules and several Perl programs)

- e.g. `conda install -c bioconda perl-capture-tiny`

Yes, it's Capture::Tiny



---

# The “recipe”

- **INGREDIENTS**

- **meta.yaml** Metadata (package name, URL, dependencies)
  - **build.sh** Compilation / installation script (not needed for Python)
  - **test.sh** Optional test script (basic tests can be added in meta.yaml)
  - All inside a `package-name` directory
- 
- Fork `bioconda-recipes`, add the new recipe and make a PR

```
{% set name = "perl-module-name" %}
{% set version = "0.01" %}
{% set sha256 = "de31386013dc32f46f4c00d19230eecca0c33ed1f1b13403a08e087a0278a05" %}
```

```
package:
  name: {{ name }}
  version: {{ version }}
```

```
source:
  url: https://cpan.metacpan.org/authors/id/N/NI/NICKNAME/Module-Name-{{ version }}.tar.gz
  sha256: {{ sha256 }}
```

```
build:
  noarch: generic
  number: 0
```

```
requirements:
  host:
    - perl
    - perl-module-build
    - perl-test-more
```

```
run:
  - perl
```

```
test:
  imports:
    - Module::Name
```

# meta.yaml example

- **BIOCONDA MAGIC**
  - When you release a new version of the module on CPAN, the BioConda Bot will automatically fetch it and make it available

---

# build.sh template

```
#!/bin/bash
```

```
# If it has Build.PL use that, otherwise use Makefile.PL
```

```
if [ -f Build.PL ]; then
```

```
    perl Build.PL
```

```
    perl ./Build
```

```
    perl ./Build test
```

```
    # Make sure this goes in site
```

```
    perl ./Build install --installdirs site
```

```
elif [ -f Makefile.PL ]; then
```

```
    # Make sure this goes in site
```

```
    perl Makefile.PL INSTALLDIRS=site
```

```
    make
```

```
    make test
```

```
    make install
```

```
else
```

```
    echo 'Unable to find Build.PL or Makefile.PL. You need to modify build.sh.'
```

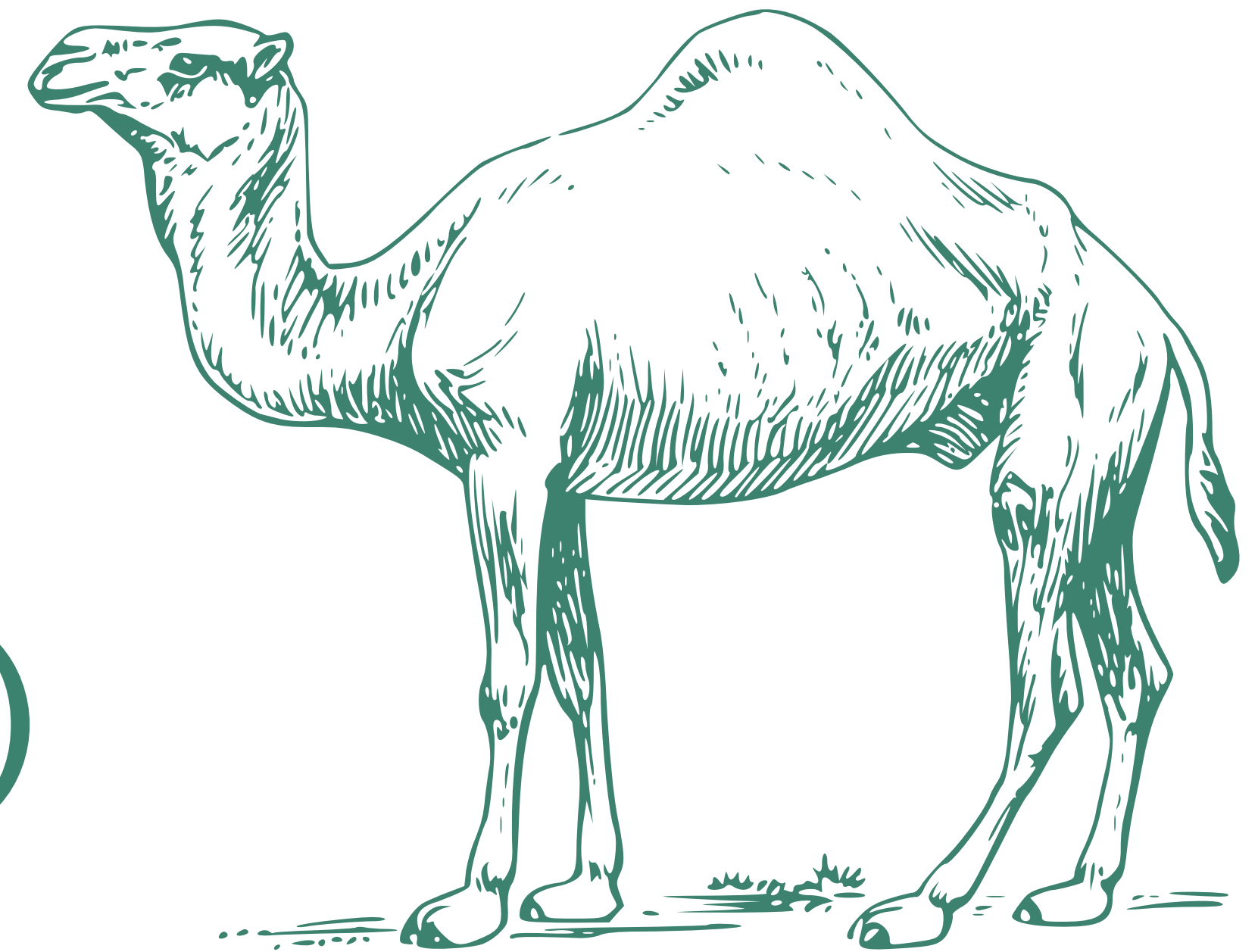
```
    exit 1
```

```
fi
```



---

We can finally have Python  
serving the **Perl community** ;)







**Thanks for  
your attention**