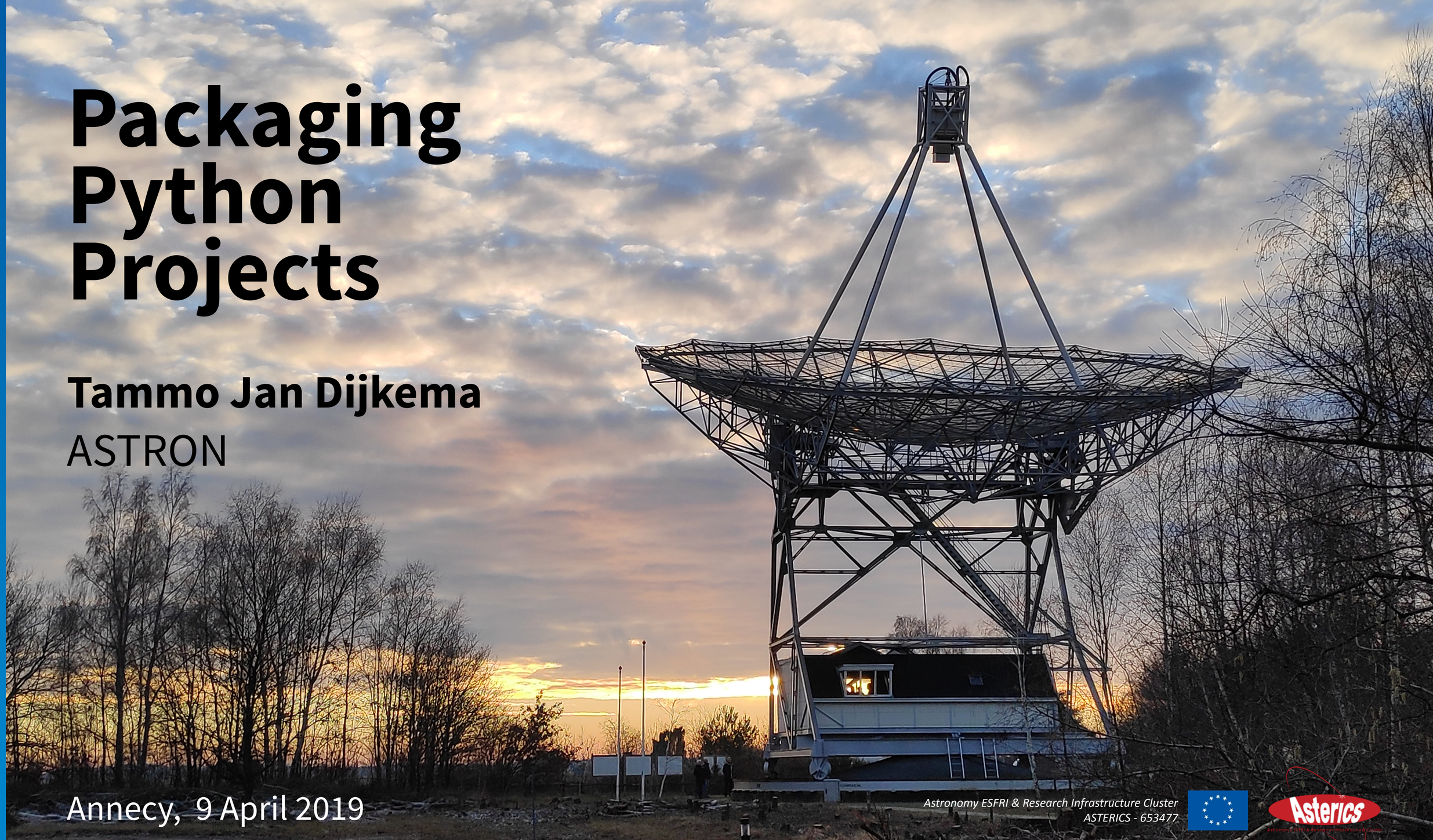
Packaging Python Projects

Tammo Jan Dijkema
ASTRON



Annecy, 9 April 2019

Astronomy ESFRI & Research Infrastructure Cluster
ASTERICS - 653477

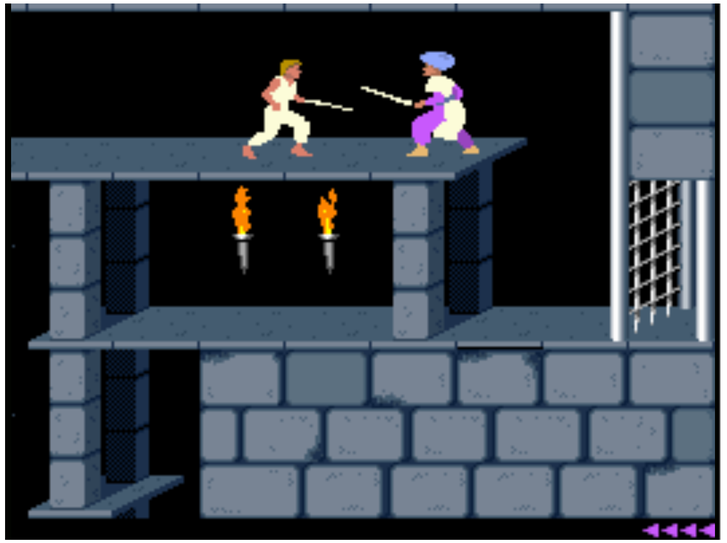
whoami

- 1999 - 2005: Math + Scientific Computing, Utrecht University
- 2005 - 2009: PhD numerical analysis (wavelets for PDEs), Utrecht

- now: Scientific Software Engineer, ASTRON
- now: volunteer Operator, Dwingeloo Radio Telescope

Astronomers writing code vs. programmers developing for astronomy (**different skills**)

Science-savvy programmers bridge the gap :)



Plan for this session

- Some improvements to yesterday's SkyViewBot
- A bit of pycharm demo
- Look at what docstrings can do for you

- Making this package 'pip' installable by adding setup.py

- Write your own basic package

Improvements to skyviewbot

- Add `--dry-run` option for testing without too much spam
- Specify object name instead of coordinates
- Use (well-supported) python packages where possible:
  `os.system('java …')` → `from astroquery.skyview import SkyView`
  `os.system('curl …')` → `requests.post`
- Adhere to PEP8 style guide (check with pycodestyle)
- Make command-line interface (even) more separated function

PEP8 code style

- Indent with 4 spaces
- Have a maximum line length
- See e.g. https://realpython.com/python-pep8/
  (or the https://www.python.org/dev/peps/pep-0008/)

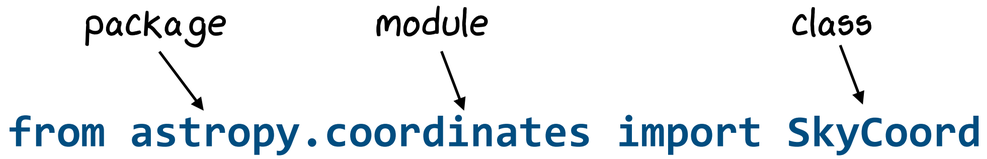
Some names

- Function: / Class:

```
def bla():        class Bla():
```

- Module: file consisting of python code

- Package: file directory structure



Python path

- If you type `import grapje`, python will look for:
  - a module `grapje` in the current working directory
    (a module should contain a file `__init__.py`)
  - a file `grapje.py` in the current working directory
  - a module `grapje` in the system path (`sys.path`) (watch out for magic eggs)
- If you have imported grapje, `grapje.__file__` tells you where it's from

Doctests

- (Placeholder, view in terminal.)

Exercise for you

- Link to what we just made for reference: https://github.com/cosmicpudding/skyviewbot
- Create a file `joke.py` which prints a joke to the command line.
  - Have the actual function in print_joke, and use `if __name__ == “__main__”`
- Put this file in a directory `joke`. Put an empty file `__init__.py` in it too. It’s now a module! Try to import it.
- Make a setup.py which installs your joke module, and creates a command-line tool joke. It’s now a package!
- Install your package with `pip install -e .`
- Bonus: add a doctest.
- Bonus: upload your package to github, install it with pip straight from github:
  `pip install` https://github.com/tammojan/nosuchrepo/archive/master.tar.gz
- Bonus bonus: add functionality to post this joke to Slack!
- Solution (without the bonuses): https://asciinema.org/a/O9bDTZOIQ1rPVH0yS4tUYoksa