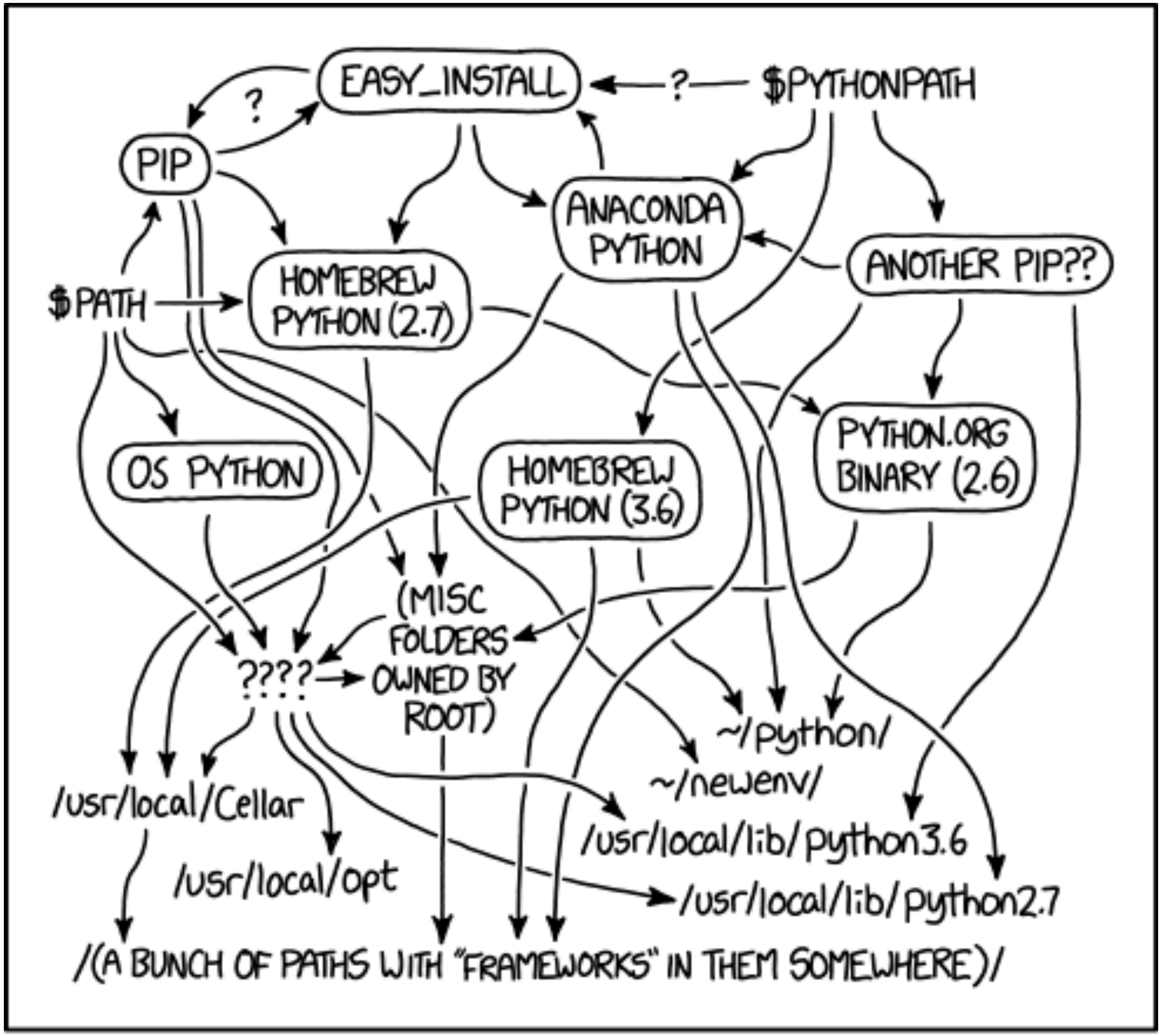
Python Tutorial Series

Don't get lost in the Python jungle

PhD Bruno C. Quint
bquint@ctio.noao.edu
Resident Astronomer at SOAR Telescope



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

Table of Contents

- Installing Packages
  - Download, unpack, and install
  - Search and get packages
- Virtual Environments
  - Concept
  - Again… a lot of options
- Using Conda
  - conda vs Anaconda vs Miniconda
  - Managing virtual environments
  - Astroconda
  - Channels and packages

Installing packages

Download, unpack, and install

```
(project root)
+-- sample/
|    +-- __init__.py
|    +-- core.py
|    +-- helpers.py
+-- docs/
|    +-- conf.py
|    +-- index.rst
+-- tests/
|    +-- test_basic.py
|    +-- test_advanced.py
+-- README.rst
+-- LICENSE
+-- setup.py
+-- requirements.txt
```

Installing packages

Download, unpack, and install

```
(project root)
+-- sample/
|    +-- __init__.py
|    +-- core.py
|    +-- helpers.py
+-- docs/
|    +-- conf.py
|    +-- index.rst
+-- tests/
|    +-- test_basic.py
|    +-- test_advanced.py
+-- README.rst
+-- LICENSE
+-- setup.py
+-- requirements.txt
```

Installing packages

Download, unpack, and install

```
(project root)
+-- sample/
|   +-- __init__.py
|   +-- core.py
|   +-- helpers.py
+-- docs/
|   +-- conf.py
|   +-- index.rst
+-- tests/
|   +-- test_basic.py
|   +-- test_advanced.py
+-- README.rst
+-- LICENSE
+-- setup.py
+-- requirements.txt
```

```
$ python setup.py install
```

Installing packages

Download, unpack, and install

```
(project root)
+-- sample/
|    +-- __init__.py
|    +-- core.py
|    +-- helpers.py
+-- docs/
|    +-- conf.py
|    +-- index.rst
+-- tests/
|    +-- test_basic.py
|    +-- test_advanced.py
+-- README.rst
+-- LICENSE
+-- setup.py
+-- requirements.txt
```

```
$ python setup.py install
```

```
$ pip install .
```

Installing packages

Download, unpack, and install

```
(project root)
+-- sample/
|    +-- __init__.py
|    +-- core.py
|    +-- helpers.py
+-- docs/
|    +-- conf.py
|    +-- index.rst
+-- tests/
|    +-- test_basic.py
|    +-- test_advanced.py
+-- README.rst
+-- LICENSE
+-- setup.py
+-- requirements.txt
```

```
$ python setup.py install
```

```
$ pip install .
```

```
$ easy_install install .
```

Installing packages

Download, unpack, and install

```
(project root)
+--- sample/
|    +--- __init__.py
|    +--- core.py
|    +--- helpers.py
+--- docs/
|    +--- conf.py
|    +--- index.rst
+--- tests/
|    +--- test_basic.py
|    +--- test_advanced.py
+--- README.rst
+--- LICENSE
+--- setup.py
+--- requirements.txt
```

```
$ python setup.py install --user
```

```
$ pip install . --user
```

```
$ easy_install install . --user
```

Installing packages

pip vs easy_install

	pip	easy_install
Uninstall Packages	Yes (pip uninstall)	No
Dependency Overrides	Yes (Requirements Files)	No
List Installed Packages	Yes (pip list and pip freeze)	No
Multi-version installs	No	Yes
Exclude scripts during install	No	Yes
per project index	Only in virtualenv	Yes, via setup.cfg
Search?	Yes	No

More on pip vs easy_install

Installing packages

pip vs easy_install

	pip	**easy_install**
Uninstall Packages	Yes (pip uninstall)	No
Dependency Overrides	Yes (Requirements Files)	No
List Installed Packages	Yes (pip list and pip freeze)	No
Multi-version installs	No	Yes
Exclude scripts during install	No	Yes
per project index	Only in virtualenv	Yes, via setup.cfg
Search?	**Yes**	**No**

More on pip vs easy_install

Installing packages

Using pip and 3rd parties packages

1. Search

```
$ pip search package_name
```

Installing packages

Using pip and 3rd parties packages

1. Search

```
$ pip search package_name
```

2. Install

```
$ pip install package_name --user
```

Installing packages

Using pip and 3rd parties packages

1. Search

```
$ pip search package_name
```

2. Install

```
$ pip install package_name --user
```

3. Update/upgrade

```
$ pip install –U/--upgrade --force-reinstall package_name
```

Installing packages

Using pip and 3rd parties packages

1. Search

```
$ pip search package_name
```

2. Install

```
$ pip install package_name --user
```

3. Update/upgrade

```
$ pip install –U/--upgrade --force-reinstall package_name
```

4. Uninstall

```
$ pip uninstall package_name
```

Virtual Environments

Concept

> “A self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.
>
> Python Official venv page

Virtual Environments

Concept

System Python

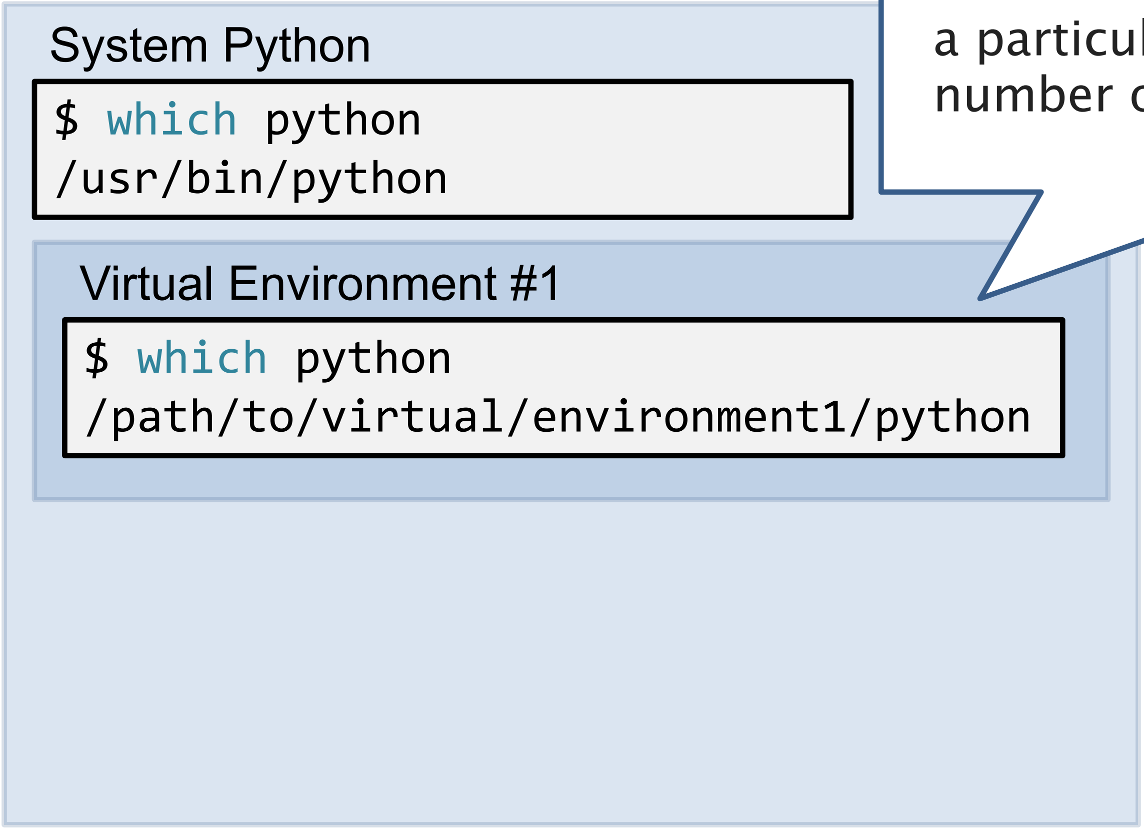
```
$ which python
/usr/bin/python
```

> “A self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.
>
> Python Official venv page

Virtual Environments

Concept

> A self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.
>
> [Python Official venv page]



System Python

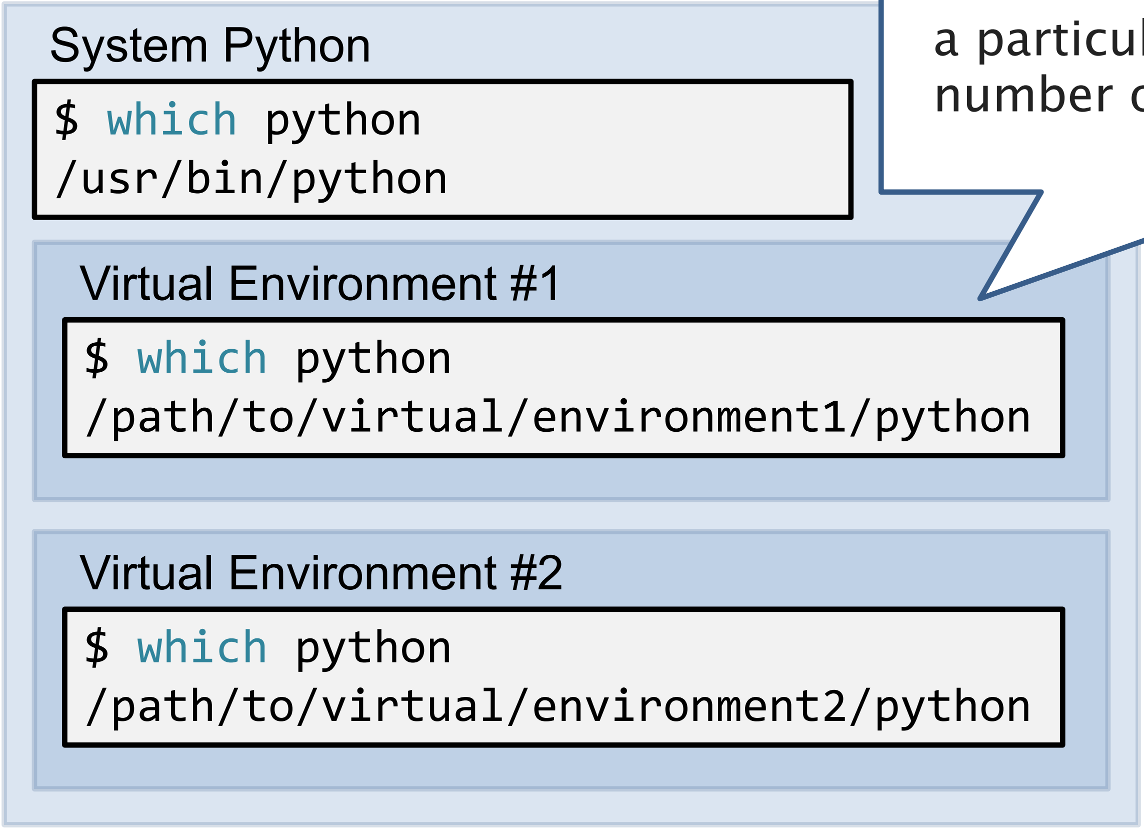
```
$ which python
/usr/bin/python
```

Virtual Environment #1

```
$ which python
/path/to/virtual/environment1/python
```

Virtual Environments

Concept



System Python

```
$ which python
/usr/bin/python
```

Virtual Environment #1

```
$ which python
/path/to/virtual/environment1/python
```

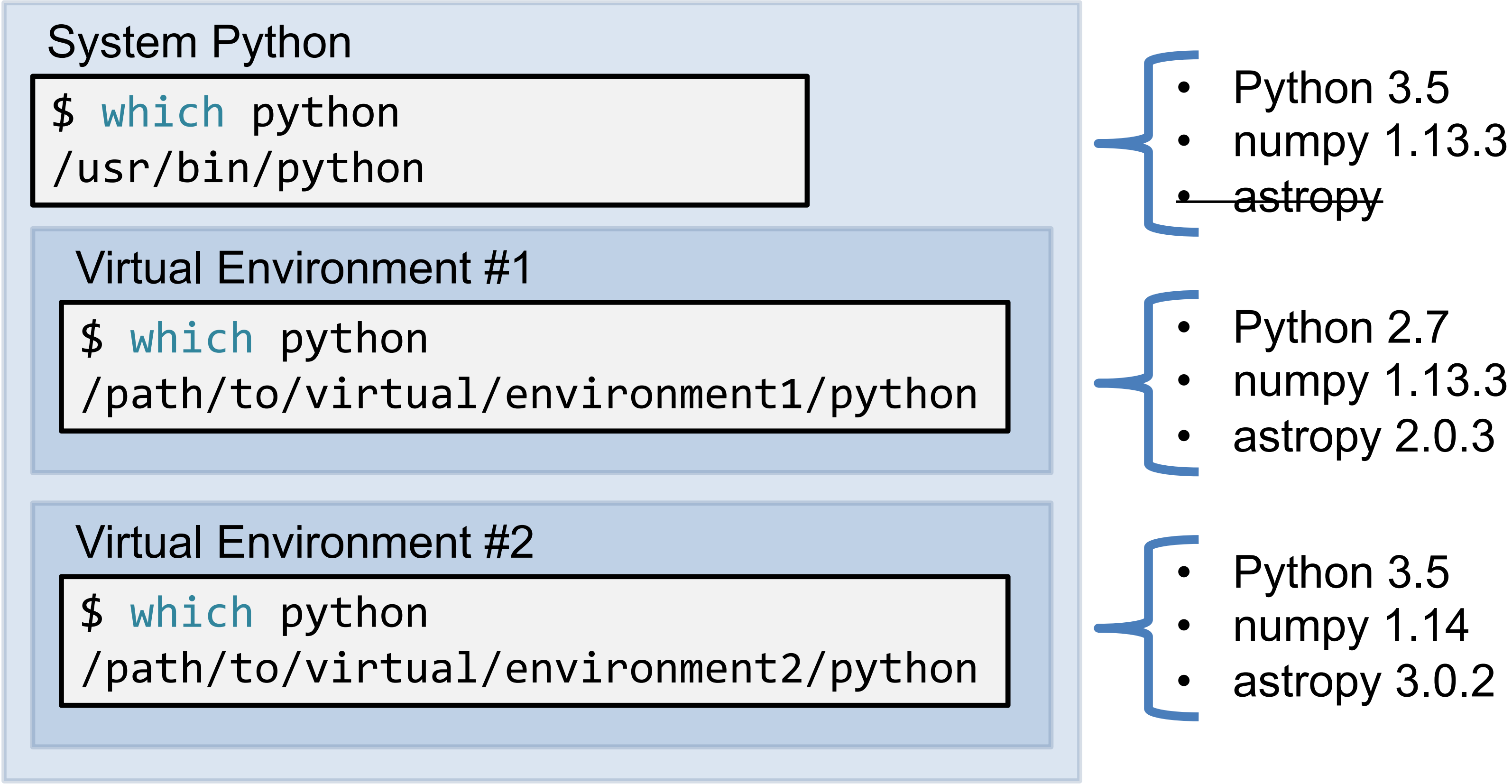
Virtual Environment #2

```
$ which python
/path/to/virtual/environment2/python
```

> “A self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.
>
> [Python Official venv page]

Virtual Environments

Concept



System Python

```
$ which python
/usr/bin/python
```

- Python 3.5
- numpy 1.13.3
- ~~astropy~~

Virtual Environment #1

```
$ which python
/path/to/virtual/environment1/python
```

- Python 2.7
- numpy 1.13.3
- astropy 2.0.3

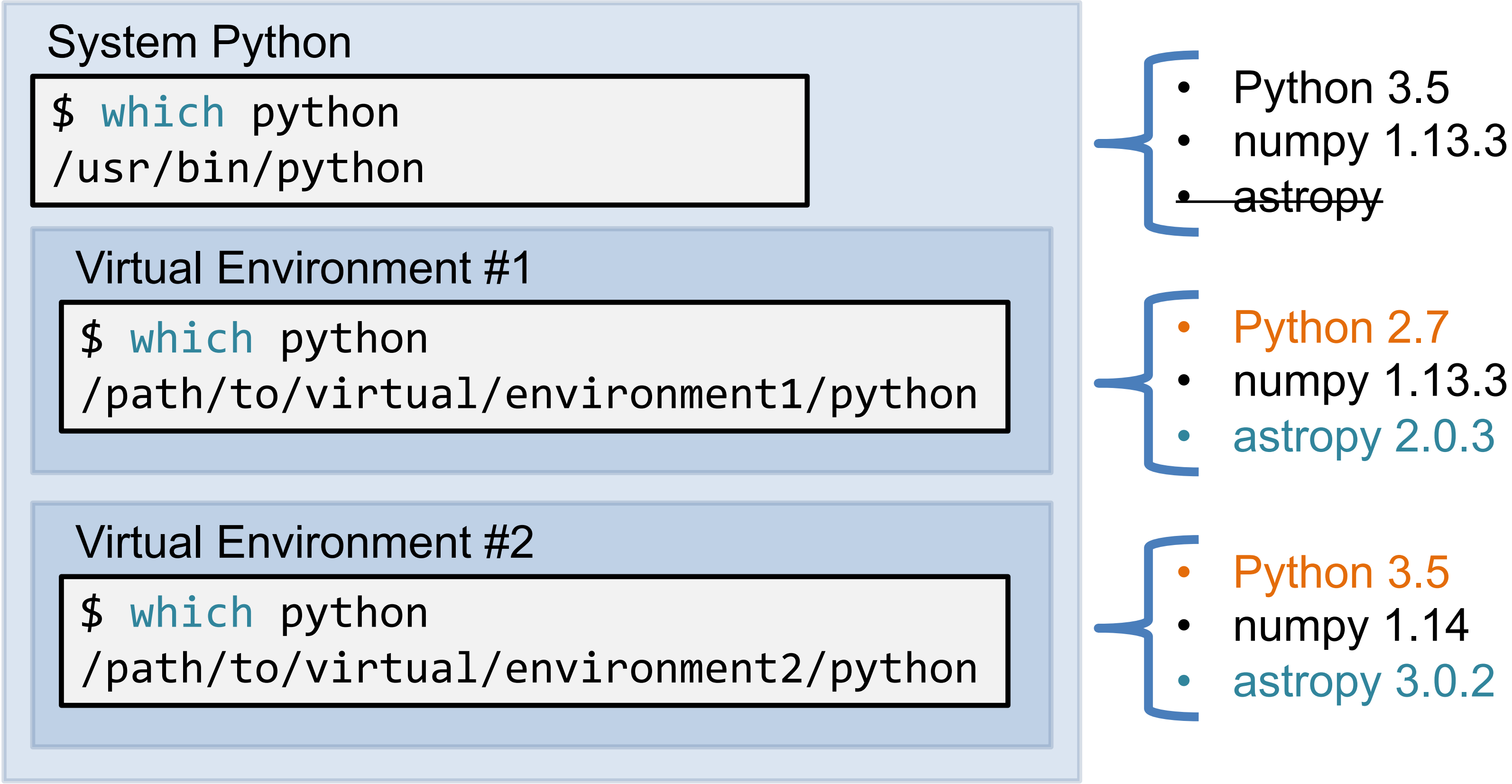
Virtual Environment #2

```
$ which python
/path/to/virtual/environment2/python
```

- Python 3.5
- numpy 1.14
- astropy 3.0.2

Virtual Environments

Concept



System Python

```
$ which python
/usr/bin/python
```

Virtual Environment #1

```
$ which python
/path/to/virtual/environment1/python
```

Virtual Environment #2

```
$ which python
/path/to/virtual/environment2/python
```

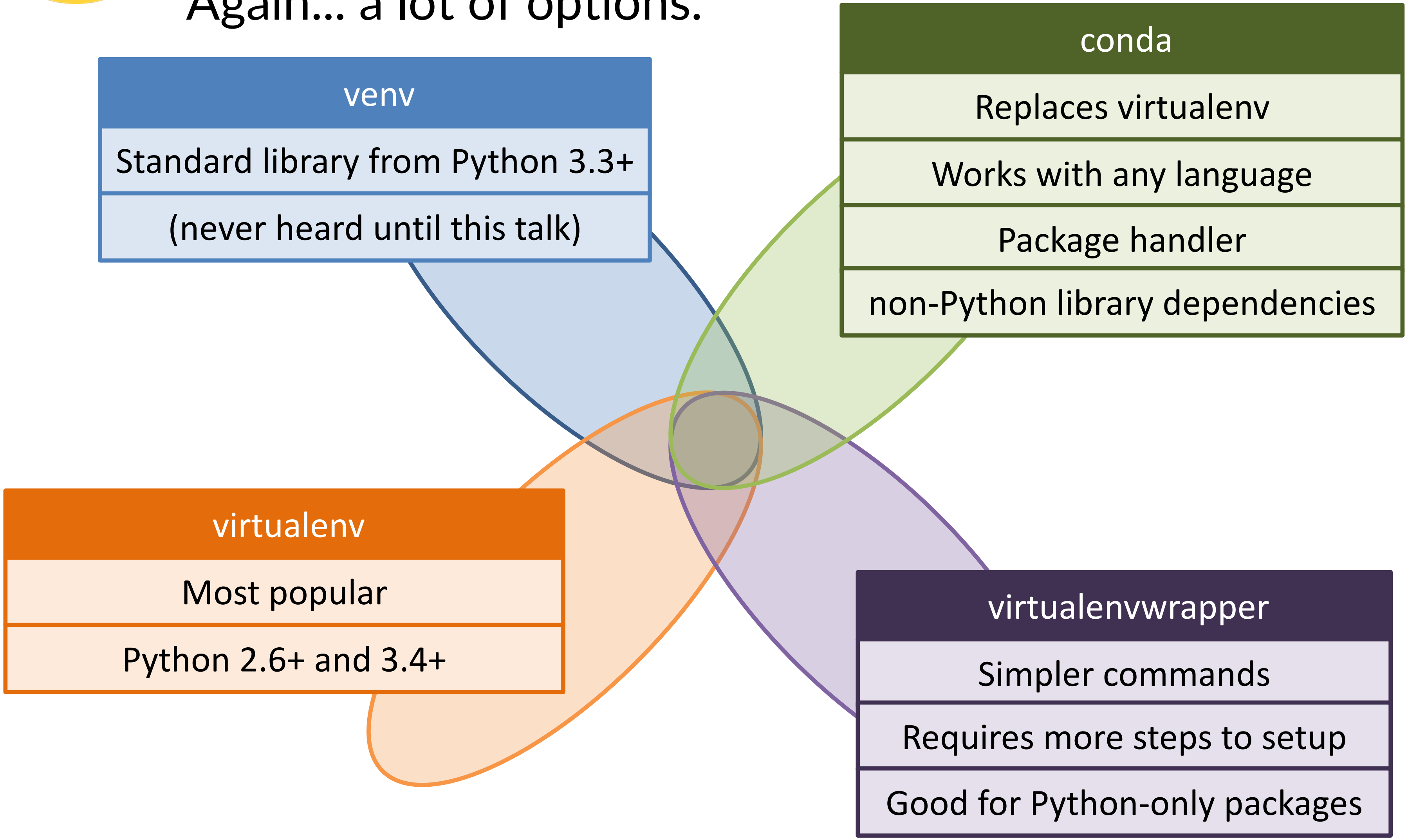
- Python 3.5
- numpy 1.13.3
- ~~astropy~~

- Python 2.7
- numpy 1.13.3
- astropy 2.0.3

- Python 3.5
- numpy 1.14
- astropy 3.0.2

Virtual Environments

Again... a lot of options.



venv

- Standard library from Python 3.3+
- (never heard until this talk)

conda

- Replaces virtualenv
- Works with any language
- Package handler
- non-Python library dependencies

virtualenv

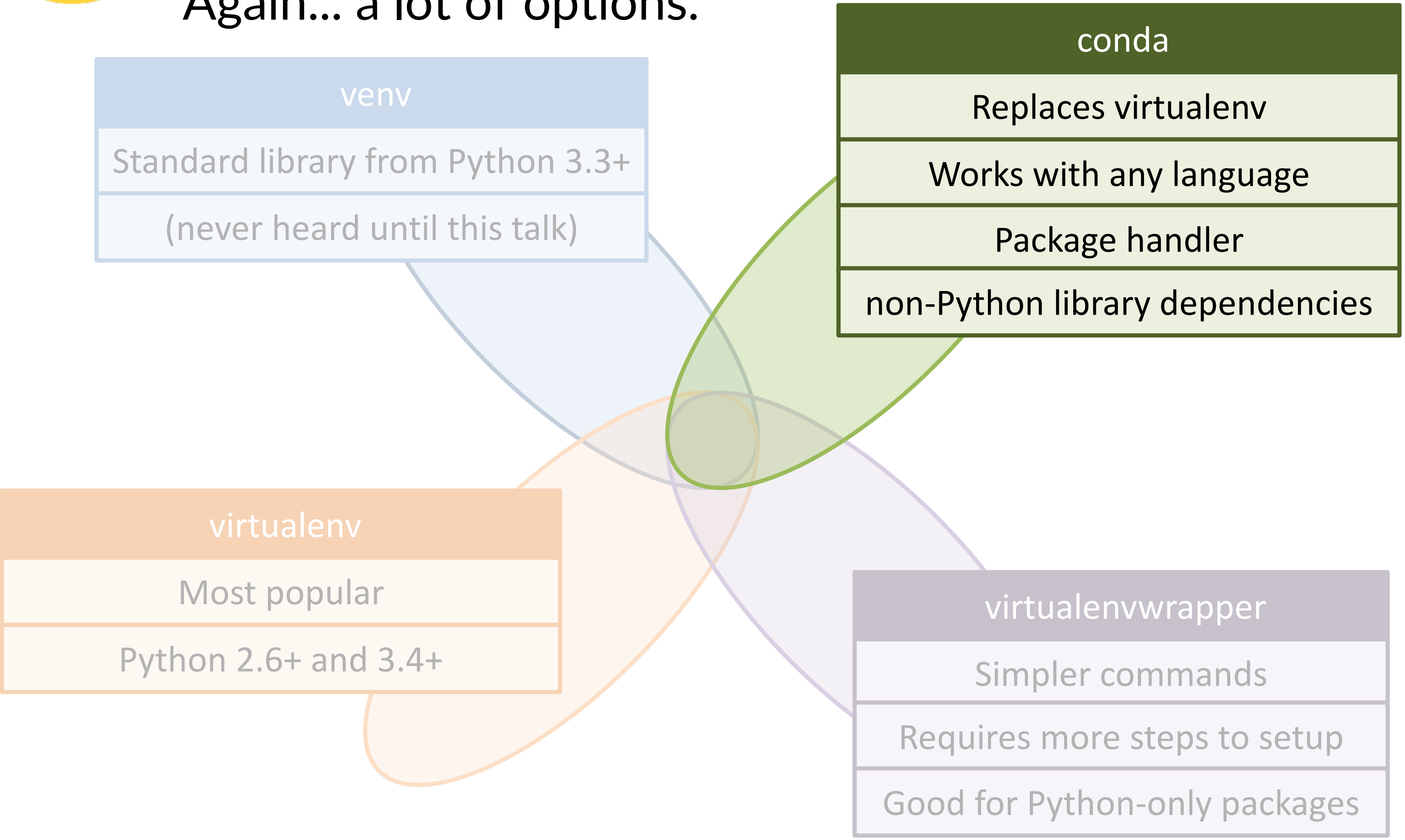
- Most popular
- Python 2.6+ and 3.4+

virtualenvwrapper

- Simpler commands
- Requires more steps to setup
- Good for Python-only packages

Virtual Environments

Again… a lot of options.



conda

- Replaces virtualenv
- Works with any language
- Package handler
- non-Python library dependencies

venv

- Standard library from Python 3.3+
- (never heard until this talk)

virtualenv

- Most popular
- Python 2.6+ and 3.4+

virtualenvwrapper

- Simpler commands
- Requires more steps to setup
- Good for Python-only packages

Using Conda

Managing virtual environments

Using Conda

Managing virtual environments

1. Download and Install Anaconda or Miniconda

Distributions

Anaconda

- New to conda or python
- Everything installed at once
- Time and disk space for installation

Miniconda

- Install packages individually
- Quicker installation
- Lighter installation

See more at Astroconda: The choice is yours

Using Conda

Managing virtual environments

1. Download and Install Anaconda or Miniconda

Distributions

Anaconda

- New to conda or python
- Everything installed at once
- Time and disk space for installation

Miniconda

- Install packages individually
- Quicker installation
- Lighter installation

conda

- Packaged manager (like pip)
- Wrapper for venv
- Dedicated for scientific and analytic packages

Command

See more at Astroconda: The choice is yours

Using Conda

Managing virtual environments

2.

Create new virtual environments

```
$ conda create -n env_name
```

See more on [Conda: Managing virtual environments]

Using Conda

Managing virtual environments

2. Create new virtual environments

```
$ conda create -n env_name
```

```
$ conda create -n env_name python=3.6 channel_name
```

Using Conda

Managing virtual environments

2. Create new virtual environments

```
$ conda create -n env_name
```

```
$ conda create -n env_name python=3.6 channel_name
```

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

See more on Conda: Managing virtual environments

Using Conda

Managing virtual environments

2. Create new virtual environments

```
$ conda create –n env_name
```

```
$ conda create –n env_name python=3.6 channel_name
```

```
$ conda env create –f environment.yml
```

environment.yml

```
name: my_new_env
channels:
··- astroconda
dependencies:
··- numpy
··- matplotlib
··- astropy
··- pip
- pip:
····- ccdproc
····- sphinx
····- sphinxcontrib.napoleon
```

Using Conda

Managing virtual environments

3. Checking existing environments

```
$ conda info --envs
```

See more on Conda: Managing virtual environments

Using Conda

Managing virtual environments

3. Checking existing environments

```
$ conda info --envs
```

```
$ conda env list
```

See more on Conda: Managing virtual environments

Using Conda

Managing virtual environments

3. Checking existing environments

```
$ conda info --envs
```

```
$ conda env list
```

```
# conda environments:
#
astroconda            *  /Users/Bruno/miniconda3/envs/astroconda
dev_pyqubes              /Users/Bruno/miniconda3/envs/dev_pyqubes
dragons                  /Users/Bruno/miniconda3/envs/dragons
goodman                  /Users/Bruno/miniconda3/envs/goodman
iraf27                   /Users/Bruno/miniconda3/envs/iraf27
samfp_gui                /Users/Bruno/miniconda3/envs/samfp_gui
superlists               /Users/Bruno/miniconda3/envs/superlists
tuna                     /Users/Bruno/miniconda3/envs/tuna
root                     /Users/Bruno/miniconda3
```

See more on Conda: Managing virtual environments

Using Conda

Managing virtual environments

4.

Activate an existing virtual environment

```
$ source activate my_env
```

See more on [Conda: Managing virtual environments]

Using Conda

Managing virtual environments

4. Activate an existing virtual environment

```
$ source activate my_env
```

```
(my_env) $
```

See more on [Conda: Managing virtual environments]()

Using Conda

Managing virtual environments

4. Activate an existing virtual environment

```
$ source activate my_env
```

```
(my_env) $
```

```
(my_env) $ conda info --envs
```

See more on [Conda: Managing virtual environments]

Using Conda

Managing virtual environments

4. Activate an existing virtual environment

```
$ source activate my_env
```

```
(my_env) $
```

```
(my_env) $ conda info --envs
```

```
# conda environments:
#
my_env                *  /Users/Bruno/miniconda3/envs/my_env
my_other_env             /Users/Bruno/miniconda3/envs/my_other_env
root                     /Users/Bruno/miniconda3
```

See more on [Conda: Managing virtual environments]()

Using Conda

Managing virtual environments

4. Activate an existing virtual environment

```
$ source activate my_other_env
```

```
(my_other_env) $
```

```
(my_other_env) $ conda info --envs
```

```
# conda environments:
#
my_env                   /Users/Bruno/miniconda3/envs/my_env
my_other_env          *  /Users/Bruno/miniconda3/envs/my_other_env
root                     /Users/Bruno/miniconda3
```

See more on [Conda: Managing virtual environments]()

Using Conda

Managing virtual environments

5. De-activate an existing virtual environment

```
(my_env) $
```

See more on [Conda: Managing virtual environments]()

Using Conda

Managing virtual environments

5.

De-activate an existing virtual environment

```
(my_env) $
```

```
$ source deactivate
```

See more on Conda: Managing virtual environments

Using Conda

Managing virtual environments

5. De-activate an existing virtual environment

```
(my_env) $
```

```
$ source deactivate
```

```
$ conda info --envs
```

```
# conda environments:
#
my_env                   /Users/Bruno/miniconda3/envs/my_env
my_other_env             /Users/Bruno/miniconda3/envs/my_other_env
root                  *  /Users/Bruno/miniconda3
```

See more on [Conda: Managing virtual environments]()

Using Conda

Managing virtual environments

6. Removing an existing environment

```
$ source deactivate
```

```
$ conda remove --name my_env --all
```

```
$ conda info --envs
```

```
# conda environments:
#
my_other_env             /Users/Bruno/miniconda3/envs/my_other_env
root                  *  /Users/Bruno/miniconda3
```

See more on [Conda: Managing virtual environments]

Using Conda

Managing packages

1. Search

```
$ conda search package_name
```

2. Install

```
$ conda install package_name
```

```
$ conda install numpy
```

See more on Conda: Command Reference

Using Conda

Managing packages

1. Search

```
$ conda search package_name
```

2. Install

```
$ conda install package_name
```

```
$ conda install ccdproc
```

See more on [Conda: Command Reference]

Using Conda

Managing packages

1. Search

```
$ conda search package_name
```

2. Install

```
$ conda install package_name
```

```
$ conda install csdms-topoflow-meteorology
```

See more on Conda: Command Reference

Using Conda

Managing packages

1. Search

```
$ conda search package_name
```

2. Install

```
$ conda install -c channel package_name
```

```
$ conda install -c astropy ccdproc
```

See more on [Conda: Command Reference]

Using Conda

Managing packages

1. Search

```
$ conda search package_name
```

2. Install

```
$ conda install –c channel package_name
```

```
$ conda install –c astropy ccdproc
```

3. Add channels

```
$ conda config --add channels https://channel.address
```

```
$ conda config --add channels http://ssb.stsci.edu/astroconda
```

See more on [Conda: Command Reference]

Astroconda

A virtual environment with packages for astronomy.

1. Add astroconda channel

```
$ conda config --add channels http://ssb.stsci.edu/astroconda
```

2. Create astroconda virtual environment

```
$ conda create -n astroconda stsci
```

Astroconda

A virtual environment with packages for astronomy.

1. Add astroconda channel

```
$ conda config --add channels http://ssb.stsci.edu/astroconda
```

2. Create astroconda virtual environment

```
$ conda create -n my_strange_env stsci
```

conda vs/and pip

pip
Only packages
Only Python
Simpler/easier
More libraries

conda
Manages envs and packages
Works with any language
More options
non-Python library dependencies

Questions?