

# Conda

## Easier Installs and Simpler Builds

Dr. Mike Müller

[Python Academy GmbH & Co. KG](http://python-academy.com)

[mmueller@python-academy.de](mailto:mmueller@python-academy.de)

@pyacademy

EuroPython 2016

Bilbao, Spain

# Conda is

- An installer similar to pip
- An environment manager similar to virtualenv
- Cross-platform
- Not limited to Python
- Strong in the scientific community
- Useful for all Python users

# Conda is

- BSD licensed
- Included in:
  - Miniconda
  - or Anaconda

# Miniconda

- Small bootstrap-like version
- Includes Python and conda, as well as dependencies and helpers (pip, wheel, setuptools, etc.)
- Provides access to many hundreds (or thousands) of mainly scientific packages
- They are just a conda install away

# Anaconda

- Large distribution of Python packages with focus on scientific applications
- Includes Python, conda, conda-build and about 200 scientific packages (new ones get added continuously)
- Needs about 2 GB of disk space
- One-stop install with all essential scientific Python tools

# Channels

- Locations of packages
- default = Anaconda server
- conda-forge
- Private channels
- `install -c my_channel package_name`

# Basic Tasks

- Install packages
- Create and administer environments
- Create packages

# Search

```

$ conda search pandas
Using Anaconda Cloud api site https://api.anaconda.org
Fetching package metadata .....
geopandas          0.1.1          py27_0  conda-forge/o
sx-64
...
                                0.2          py35_0  conda-forge/
osx-64
pandas            0.8.1          np16py26_0  defaults
...
. 0.16.2          np19py34_0  defaults
...
* 0.18.1          np110py35_0  defaults
                                0.18.1          np110py35_0  conda-forge/o
sx-64
                                0.18.1          np111py27_0  conda-forge/o
sx-64
                                0.18.1          np111py27_0  defaults
                                0.18.1          np111py34_0  defaults
                                0.18.1          np111py34_0  conda-forge/o
sx-64

```



# Search with Exact Match

```

conda search --full-name pandas
Using Anaconda Cloud api site https://api.anaconda.org
Fetching package metadata .....
pandas          0.8.1          np16py26_0 defaults
                 0.8.1          np16py27_0 defaults
...
. 0.17.1        np110py35_0 defaults
  0.18.0        np110py27_0 defaults
  0.18.0        np110py34_0 defaults
  0.18.0        np110py35_0 defaults
  0.18.0        np111py27_0 defaults
  0.18.0        np111py34_0 defaults
  0.18.0        np111py35_0 defaults
  0.18.1        np110py27_0 conda-forge/o
sx-64          * 0.18.1        np110py27_0 defaults

```

# Search for Platform and Version

```
conda search --platform win-32 --spec pandas=0.18.1
Using Anaconda Cloud api site https://api.anaconda.org
Fetching package metadata .....
pandas 0.18.1 np110py27_0 conda-forge/w
in-32 0.18.1 np110py27_0 defaults
0.18.1 np110py34_0 defaults
in-32 0.18.1 np110py34_0 conda-forge/w
0.18.1 np110py35_0 defaults
in-32 0.18.1 np110py35_0 conda-forge/w
in-32 0.18.1 np111py27_0 conda-forge/w
0.18.1 np111py27_0 defaults
0.18.1 np111py34_0 defaults
in-32 0.18.1 np111py34_0 conda-forge/w
in-32 0.18.1 np111py35_0 conda-forge/w
```

# Install a Package

```
conda install pandas
Using Anaconda Cloud api site https://api.anaconda.org
Fetching package metadata .....
Solving package specifications: .....
```

Package plan for installation in environment /Users/mike/anaconda/envs/mypy35:

The following NEW packages will be INSTALLED:

mk1:	11.3.3-0	
numpy:	1.11.1-py35_0	
pandas:	0.18.1-np111py35_0	conda-forge
python-dateutil:	2.5.2-py35_0	conda-forge
pytz:	2016.3-py35_0	conda-forge
six:	1.10.0-py35_0	conda-forge

Proceed ([y]/n)?

```
Linking packages ...
[ COMPLETE ] |#####| 100%
```

# Create an Environment I

```
$ conda create -n mypy35 python=3.5
Using Anaconda Cloud api site https://api.anaconda.org
Fetching package metadata .....
Solving package specifications: .....
```

Package plan for installation in environment /Users/mike/anaconda/envs/mypy35:

The following packages will be downloaded:

package	build	size	channel
setuptools-23.0.0	py35_0	461 KB	conda-forge

The following NEW packages will be INSTALLED:

```
ca-certificates: 2016.2.28-1 conda-forge
ncurses: 5.9-7 conda-forge
openssl: 1.0.2h-0 conda-forge
pip: 8.1.2-py35_0
python: 3.5.2-1 conda-forge
readline: 6.2-0 conda-forge
setuptools: 23.0.0-py35_0 conda-forge
sqlite: 3.13.0-1 conda-forge
tk: 8.5.19-0 conda-forge
wheel: 0.29.0-nv35_0
```

# Create an Environment II

```
Pruning fetched packages from the cache ...
Fetching packages ...
setuptools-23. 100% |#####|
#####| Time: 0:00:05 89.34 kB/s
Extracting packages ...
[      COMPLETE
]|#####| 100%
Linking packages ...
[      COMPLETE
]|#####| 100%
#
# To activate this environment, use:
# $ source activate mypy35
```

# Show Environments

```
conda env list
# conda environments:
#
fipy_py26      /Users/mike/anaconda/envs/fipy_py26
fipy_py27      /Users/mike/anaconda/envs/fipy_py27
fipy_py34      /Users/mike/anaconda/envs/fipy_py34
fipy_py35      /Users/mike/anaconda/envs/fipy_py35
mypy35         /Users/mike/anaconda/envs/mypy35
py26           /Users/mike/anaconda/envs/py26
py27           /Users/mike/anaconda/envs/py27
py33           /Users/mike/anaconda/envs/py33
py34           /Users/mike/anaconda/envs/py34
py35           /Users/mike/anaconda/envs/py35
py35_test     /Users/mike/anaconda/envs/py35_test
pydatabin2016 /Users/mike/anaconda/envs/pydatabin2016
tensorflow     /Users/mike/anaconda/envs/tensorflow
root          *
```

# Activate an Environment

- Linux, OS X

```
# $ source activate mypy35
```

- Windows

```
# $ activate mypy35
```

# Environment Marked as Active

```
conda env list
```

```
..  
mypy35      *  /Users/mike/anaconda/envs/mypy35
```



# List All Installed Packages

```
conda list
# packages in environment at /Users/mike/anaconda/envs/mypy35:
#
ca-certificates      2016.2.28          1      conda-forge
ncurses              5.9                7      conda-forge
openssl              1.0.2h             0      conda-forge
pandas               0.18.1             np111py35_0  conda-forge
python               3.5.2              1      conda-forge
python-dateutil      2.5.2              py35_0      conda-forge
pytz                 2016.3             py35_0      conda-forge
readline             6.2                0      conda-forge
setuptools           23.0.0             py35_0      conda-forge
six                  1.10.0             py35_0      conda-forge
sqlite               3.13.0             1      conda-forge
tk                   8.5.19             0      conda-forge
xz                   5.2.2              0      conda-forge
zlib                 1.2.8              3      conda-forge
mkl                  11.3.3             0
numpy                1.11.1             py35_0
phreeqpy             0.2.0              <pip>
pip                  8.1.2              py35_0
wheel                0.29.0             py35_0
```

# Building a Package

1. From a package on PyPi
2. From scratch

# Build from PyPi with a Skeleton

```
conda install conda-build  
conda skeleton pypi mypackage  
conda build mypackage
```

# Result is a Tarball

```
/Users/mike/anaconda/conda-bld/osx-64/mypackage-0.1  
.0-py35_0.tar.bz2
```

# Install from Local File

```
conda install --use-local mypackage
```

- With full path

```
conda install /Users/mike/anaconda/conda-bld/osx-64  
/mypackage-0.1.0-py35_0.tar.bz2
```

# Specify a Python Version

```
conda build --python 3.4 mypackage
```

# Convert to Other Platforms

```
conda convert --platform all ./mypackage-0.1.0-py35  
_0.tar.bz2 -o outputdir/
```

# Upload to Anaconda Cloud

```
conda install anaconda-client
```

```
anaconda upload /Users/mike/anaconda/conda-bld/osx-64/mypackage-0.1.0-py35_0.tar.bz2
```



# Building from Scratch

- `meta.yaml`
- `build.sh` - Linux and Mac OS X
- `build.bat` - Windows
- `setup.py` - just as with `pip`

# The meta.yaml

```
package:  
  name:  
  version:  
  
source:  
  git_rev:  
  git_url:  
  
requirements:  
  build:  
    - python  
    - setuptools  
  
  run:  
    - python  
  
test:  
  imports:  
    -  
  
about:  
  home:  
  license:  
  license_file:
```

# Example

```
package:  
  name: mypackage  
  version: 1.0  
  
source:  
  path: ../..  
  
requirements:  
  build:  
    - python  
    - setuptools  
  
  run:  
    - jupyter  
    - libpython # [win]  
    - numpy  
    - pandas  
    - python  
    - pywin32 # [win]  
    - pyyaml  
  
about:  
  home: me  
  license: MIT
```

# The Build Files

- Windows build.bat

```
"%PYTHON%" setup.py install  
if errorlevel 1 exit 1
```

- Linux / OS X build.sh

```
$PYTHON setup.py install
```

- Add more commands as needed

# Build It

`conda build mypackage`

- Install and upload as with skeleton

# Conclusions

- conda is a great
  - Installer
  - Package manager
  - Environment manager
  - Build tool
- Works together with `pip`
- Well known in the scientific Python community
- Can be really useful for all Python programmers
- You should give it a try

**Thanks - Questions?**