

Python Packaging

Michael Meffie

September 15, 2018

Python packaging

- How do we make it easy to install our code?
- How do we publish our packages?

Scope of this session - pure python - assume no extra files need to be installed - assume os specific packaging is not needed - assume we have pip and setuptools

What is a module

- basic unit python code
- importable
- single python file
- python extension implemented in c

What is a package?

The name package is overloaded.

- ① native install package; rpm, deb, dmg, msi
- ② importable directory of modules
- ③ distributable unit of code and metadata
 - name
 - version
 - source distribution (sdist)
 - built distributions (bdist)
 - wheel format
 - egg format (legacy)

Package index

- PyPI.org pie-pee-eye
- aka the cheese shop
- originally just links to contributed packages
- new pypi hosts packages
- test site available for dry runs
- pip uses the pypi by default
- create a private index for internal use

- use setuptools instead of distutils
- use pip instead of easy-install
- hassle factor of older systems
 - pip is not installed
 - setuptools not installed

What we need

- pip
- setuptools
- wheel
- twine

Directory layout

```
package-name/  
  README  
  LICENSE  
  setup.py  
package-name/  
  __init__.py  
  foo.py  
  bar.py  
  sub-package/  
    __init__.py  
  ...
```


More files

- tests - unit test files
- docs - documents
- Makefile (optional)
- Manifest.in - include other files

- defines the meta data for the package
- a python file

```
from setuptools import setup
setup(
    name='hello',
    version='1.2.0',
    author='Michael Meffie',
    author_email='mike@meffie.org',
    description='A tiny example package',
    url='https://github.com/CantonPython/packaging',
    classifiers=[
        'Programming Language :: Python :: 3',
        'License :: OSI Approved :: BSD License',
        'Operating System :: OS Independent',
        'Environment :: Console',
    ],
```

Create distribution files

```
$ python setup.py sdist  
$ python setup.py bdist  
$ python setup.py bdist_wheel
```

Upload dist with twine

Register a test account:

```
https://test.pypi.org/account/register/
```

Upload to the test index:

```
$ twine upload --repository-url https://test.pypi.org/legacy/
```

Test install

Use the index-url option to install your test package:

```
$ pip install --index-url https://test.pypi.org/simple/ yo
```

so long and thanks for all the fish