Pex: Theory and Practice

Moshe Zadka – https://cobordism.com

Pyninsula April 2018

Python start-up

```
$ python /some/script.py
```

Causes

```
"/some" in sys.path
```

Python start-up

```
$ python /some/script/
```

Same as

```
$ python /some/script/__main__.py
```

Python start-up

```
$ python /some/script/__main__.py
```

Causes

```
"/some/script" in sys.path
```

Python start-up

```
/some/script/__main__.py
          foo.py
          bar/__init__.py
               something.py
```

Python start-up

```
$ python /some/script.zip
```

Treats zip archive like directory.

Python start-up

Zip is an ending-oriented format:

```
$ ( echo " hello "; cat foo . zip ) > bar . zip
$ unzip −t bar . zip
Archive :    bar . zip
warning [ bar . zip ] :    6 extra bytes at beginning or w
   ( attempting to process anyway )
```

Python start-up

```
$ HERE=$(pwd)
$ (cd /some/script/ && zip $HERE/stuff.zip -r .)
$ (echo '#!/usr/bin/python3'; cat stuff.zip) > runna
$ chmod +x runnable
```

Productionizing – hard

- Lots of edge cases!
- Reimplement? No!

Pex – roll up your libraries

Supports a lot!

Pex – recipe

```
$ pip wheel --wheel-dir /some/dir requirements.txt
$ pex --no-index --find-links /some/dir requirement
    -m ...
    -o ...
```

Pex – Python recipe

```
$ pip wheel --wheel-dir /some/dir requirements.txt
```

Pex – Python recipe

```python
builder = pex_builder.PEXBuilder()
builder.set_entry_point(...)
builder.set_shebang('/usr/bin/python3')
for dist in os.listdir('/some/dir'):
    dist = os.path.join(wheelhouse, dist)
    builder.add_dist_location(dist)
builder.build(...)
```

Pex only wheels

Pex dependency resolution different from pip: why worry?

Caveats

No manylinux wheels

Caveats

Python installed in shebang

Caveats

No windows support

Caveats

No PyPy support