Harnessing the Power of Python in ArcGIS Using the Conda Distribution

Shaun Walbridge
Clinton Dow

https://4326.us/esri/conda-2017

https://github.com/scw/conda-devsummit-2017-talk

Conda

Getting to Packages

Why Packages?

- Software is composed of many smaller components, often called *packages* or *libraries*.
- It's often better to reuse code that solves a problem well rather than recreating it
- But, sharing code is a **hard problem**. Do you have the same packages of the same versions as the developer did?

Package Management for Python

Why not pip, wheels, virtualenvs?

- Don’t handle the harder problem of system dependencies, considered out of scope by Python packagers – does it end up in site-packages?
- Package devs: On OSX and Linux, ‘easy’ to get the deps! Use a system package manager (e.g. apt, brew, yum) and the included compiler (e.g. clang, gcc).
- It’s still not easy to make reproducible builds, and what about *that other* platform?

Why Conda?

- Scientific Python community identified that there was a gap not being addressed by the core Python infrastructure, limiting their ability to get packages into the hands of users
- Industry standard built by people who care about this space — *Continuum Analytics*

Why Conda?

- It solves the hard problem:
  - Handles dependencies for many languages
  - Built for Python first, but it really solves a much broader infrastructural issue.
- Gateway to data science — scientific, analytics, integrated software ecosystem for organizations

Interlude: *Reviewing Reviews*

Reviewing Reviews

```
import pandas
df_raw = pandas.read_excel('data/DevSummit Survey Results.xls')

# find all sessions with me as a speaker, skip missing
df = df_raw[df_raw['Speakers'].str.contains('Walbridge', na=False)]
df_conda = df[df['Event Title'].str.contains('Conda')]

# likert scaled
ratings_columns = [
    'Title and level of the workshop were consistent with the content ',
    'Content of the workshop was relevant to my work',
    'The workshop provided information or techniques I can apply to my work right away',
    'The presentation was organized and easy to understand',
    'The presenter exhibited strong public speaking skills'
    ]
binary_columns = ['I would recommend this workshop to a colleague']

# collect ratings
ratings = df_conda.loc[:, ratings_columns]
inverse_ratings = df_conda.loc[:, binary_columns]

# score by adding ratings + rescaling recommended
score = (ratings.sum(axis=1) + inverse_ratings.sum(axis=1)*5)
```

Reviewing Reviews

```
score = (ratings.sum(axis=1) + inverse_ratings.sum(axis=1)*5)
'''''
1041    30.0    1034    30.0    1036    29.0
1039    27.0    1038    26.0    1037    26.0
1040    25.0    1042    24.0    1035    15.0
'''''
# 1035, let's see if they left us a comment
df_conda.loc[1035]['Comments']
```

There was no real basic explanation of what conda does… just a package and env manager.

Reviewing Reviews

If you don't agree with someone, try walking a mile in their shoes…

…because then you'll be a mile away from them, and you'll have their shoes. — Terry Pratchett

Let's walk a mile

Fundamentals

Fundamentals

- Consistent building of packages (Windows, Linux, Mac OS), public and private sharing
- Cross platform, and *cross-langauge* — handles *C/C++*, *R*, *Java*, *Scala*, *Javascript* and many more
- Ultimately realized as a collection of files in an archive, and rules which dictate package dependencies
- Open source: Esri is using it, you can use it in your own projects for other contexts

- *Environments* —
  - A collection of packages and Python install is called an *environment*, the building block for managing Python with Conda
  - Flexibly make changes without affecting installed software
  - Can create multiple environments and switch seamlessly
- *Requirements* — include explicit state information, not just the package name.

Where do I get packages?

- Conda packages can come from a variety of locations:
  - Public repositories hosted on Anaconda Cloud
  - Public repositories self-hosted
  - Private repositories
  - Anaconda Enterprise
  - On disk
- *Channels* —
  - A collection of packages owned by a user or organization
  - Configure Conda to look at these locations (.condarc file)

Demo: Using Packages

Conda Basics

Activating environments, a couple ways:

- Use the shortcuts included in Pro
- Manually activate the environment:

```
cd C:\ArcGIS\bin\Python\Scripts
activate arcgispro-py3
```

Conda Basics

```
conda --help

conda info
```

Conda info is the starting point — it tells you the state of the environment.

Conda Basics

```
conda list

# packages in environment at C:\ArcGIS\bin\Python\envs\arcgispro-py3:
#
colorama                  0.3.7                    py35_0    defaults
cycler                   0.10.0                   py35_0    defaults
future                   0.15.2                   py35_0    defaults
matplotlib                1.5.3           np111py35_0e  [arcgispro]  esri
mpmath                     0.19                    py35_1    defaults
netcdf4                   1.2.4                 py35_0e  [arcgispro]  esri
nose                      1.3.7                   py35_1    defaults
numexpr                   2.6.1           np111py35_0e  [arcgispro]  esri
numpy                     1.11.2                py35_0e  [arcgispro]  esri
pandas                    0.19.0             np111py35_0    defaults
pip                      8.1.2                  py35_0    defaults
py                       1.4.31                 py35_0    defaults
pyparsing                 2.1.4                   py35_0    defaults
pypdf2                   1.26.0                   py_0    esri
pytest                    2.9.2                  py35_0    defaults
```

Deeper Dive

Conda Packaging

- OK, so how do we make a new package?
- Create a recipe which describes the instructions to build the software
  - Where do I get the code?
  - What are we building, and what does it depend on?
  - Run `conda build` to create a package from this recipe

conda packaging

meta.yaml

```
package:
  name: conda-devsummit-2017-talk
  version: "1.0"

source:
  git_url: https://github.com/scw/conda-devsummit-2017-talk.git

requirements:
  run:
    - python
    - scikit-learn
```

conda packaging

Build the package:

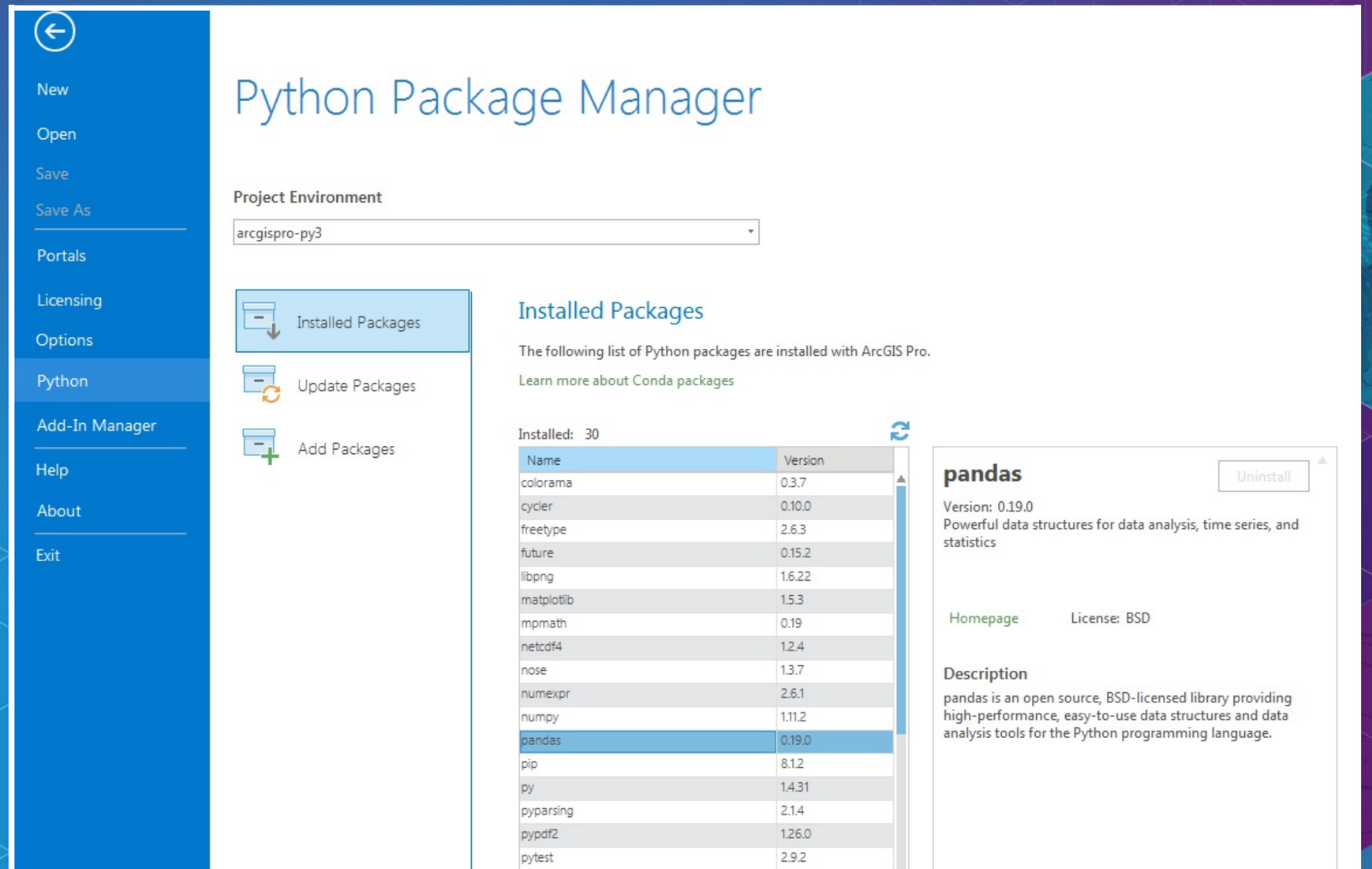
```
conda build c:\example\mypackage
```

Upload and share:

```
anaconda upload mypackage
```

Demo: Conda Packaging

Python Package Manager



How can I use this?

- We already ship you the SciPy stack — powerful and out of the box in all products
- Conda command and a Conda root Python install
- New modules (e.g. requests), environment with Pro
- Python Package Manager in Pro
    - Get packages, expand your possibility space
    - Package your work: this is an opportunity to distribute it

Where Can I Run This?

- ArcGIS Pro 1.3
  - Conda is *the* Python install, included for all
- ArcGIS Pro 1.4
  - Python Package Manager
  - Python 3.5 with current package set
- ArcGIS Enterprise 10.5
- ArcGIS API for Python

from future import *

- ArcGIS Pro 2.0:
  - Manage environments and channels
  - Resettable environment
  - “µConda”

Resources

Other Sessions

- Getting Data Science with R and ArcGIS
  - Weds 4:00PM, San Jacinto
- Deploying Your Geoprocessing Tools as Python Modules
  - Weds 4:00, Demo Theater 1
- Python Package Management Using Conda
  - Weds 4:30PM, Demo Theater 1
- Continuum Analytics: Exploring Continuum Analytics' Open-Source Offerings
  - Thurs 10:30AM, Mesquite G-H

Conda vs…

Name	Means	Included?
Conda	The command itself	✓
Miniconda	A minimum set of Python packages to build and run Conda.	✓
Anaconda	A distribution 200+ packages and run Conda	
Anaconda Workgroup	Self-hosted, distributed and HPC additions	

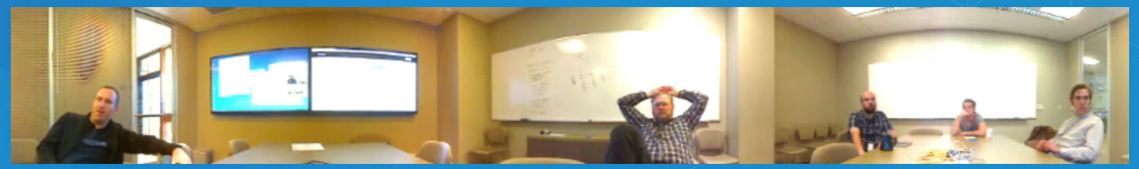
Resources

- Conda Cheatsheet
- Anaconda.org
- Conda Recipes
- Harnessing the Power of Python in ArcGIS Using the Conda Distribution
- Understanding Conda by Jake Vanderplas
- Security updates for Python depdendencies

Closing

Thanks

Esri Conda Team:



Continuum Analytics for creating and open sourcing Conda

Rate This Session

iOS, Android: Feedback from within the app

Be warned that we may incorporate feedback into next year’s session


esri
THE
SCIENCE
OF
WHERE