Resen: Towards a Reproducible Software Environment using Python and Docker

Ashton Reimer[1], Asti Bhatt[1], Leslie Lamarche[1], Pablo Reyes[1], Todd Valentic[1]

[1] Center for Geospace Studies
SRI International

CEDAR Workshop

17/06/2019

Abstract

Credibility of scientific results depends on the reproducibility of those results, thus reproducibility is a key concept of the scientific process. We present Resen, a tool built to facilitate the scientific process in the digital age. Specifically, Resen enables reproducible data analysis and allows scientists to more easily work with community built analysis tools. Resen combines Python and Docker to provide a containerized JupyterHub interface, which allows scientists to perform analysis using Jupyter notebooks and/or a command line interface via a webbrowser. Additionally, Resen ships with a variety of common python packages preinstalled, including tools built by the space physics community, such as apexpy, davitpy, and spacepy. We will present Resen and describe current capabilities and future features.

Resen

Example Usage

Integrated Geoscience Workshop

Resen

Example Usage

Integrated Geoscience Workshop

Integrated Geoscience Observatory (InGeO): An EarthCube project supported by the NSF Cyberinfrastructure for Sustained Scientific Innovation program with 2 main goals:

* Provide tools that facilitate geospace researcher's ability to collaborate, share work, reproduce results, and access data
* Help educate the geospace community on best practices to facilitate reproducible scientific data analysis

InGeO Tool: Resen

* REproducible Software ENvironment
* Implemented using python and docker
* Simplify software installation process
* Provide a portable/reproducible analysis environment

Need

* Installing software sucks and reproducing analysis is hard
* There are many barriers to accessing data

Approach

* Provide a super easy to use tool
* Use a cross platform containerized environment with software pre-installed

Benefits

* Completely encapsulates the analysis environment
* Pre-install/package community tools
* Containers are cross platform across linux, macos, windows
* Can save completed analysis container and upload to Zenodo where others can download and run

Competition

Many existing tools that solve the installation and reproducibility problems. For example:

1) Installing software:
   * scientific linux operating systems
   * anaconda

But what about community tools?

2) Software tools for reproducibility
   * reprozip, sciunit2
   * Google colab, Mozilla iodide
   * JupyterWith, Binder, etc.

However, nothing exists that does both 1 and 2.

Resen

Resen is a command line tool for creating, importing, and exporting software environments. Provides a simplified and abstracted interface (users don't have to work with docker directly)

2 key concepts:

* Buckets - A bucket contains all of the software tools, analysis, and data for a study
* Cores - A particular software suite packaged with a bucket

Current Status

Beta version 2019.1.0rc2:

* Standard python packages like numpy, matplotlib, scipy, pandas, etc.
* Community tools such as: apexpy, davitpy, madrigalweb, spacepy, etc.
* Command line interface to create/remove buckets and start/stop jupyterlab servers
* Jupyterlab provides both notebook and command line access for analysis

Try it out yourself and send us feedback!

Resen

Example Usage

Integrated Geoscience Workshop

Installation and Usage

Instructions are available on readthedocs:

https://resen.readthedocs.io/en/latest/

Installation

Generally, a 2 step process:

1. Install docker
   - Easy on Linux and Macos. For Windows, follow installation instructions on readthedocs
2. Install the ‘‘resen’’ python package (available on github)
   - pip install git+https://github.com/EarthCubeInGeo/resen.git@v2019.1.0rc2

Basic Usage

```
$ resen
```

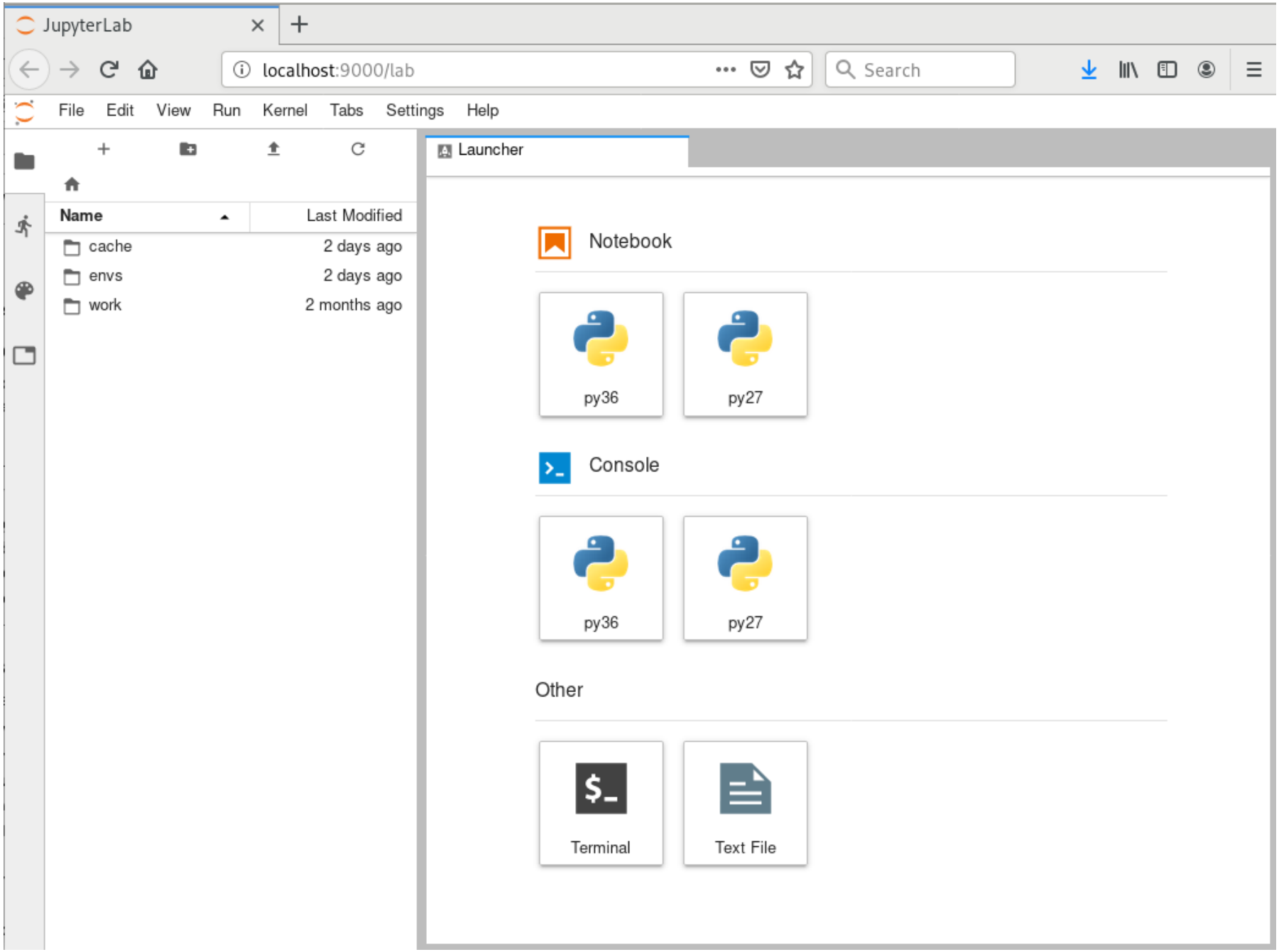
Create a Bucket and Start Analysis

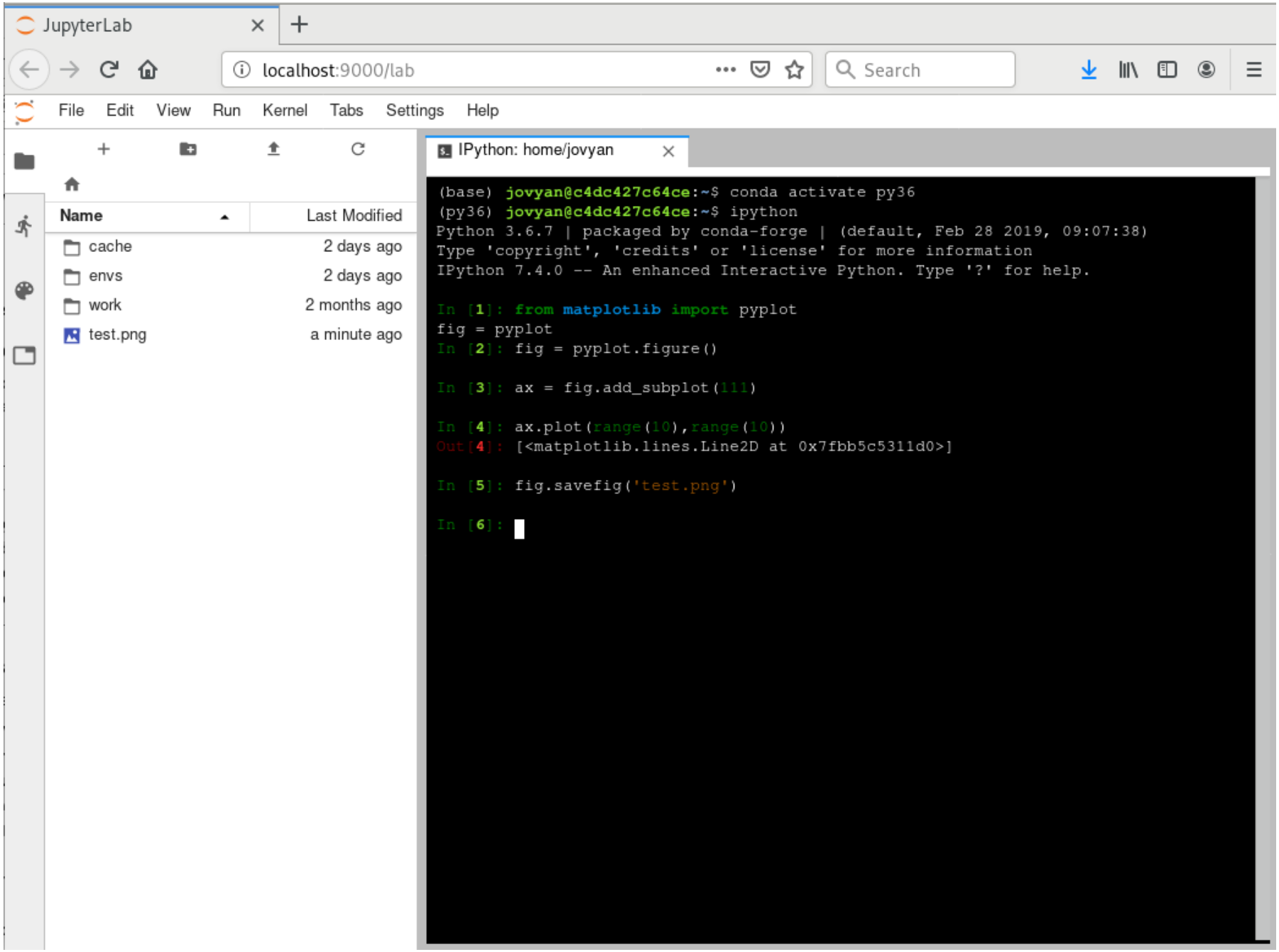
create_bucket

* Starts a guided Q&A process for creating a new analysis bucket

start_jupyter bucket_name

* Starts a jupyterlab server in the bucket and opens a tab in your default browser


JupyterLab
localhost:9000/lab
Search
File Edit View Run Kernel Tabs Settings Help
Name
Last Modified
cache 2 days ago
envs 2 days ago
work 2 months ago
Launcher
Notebook
py36
py27
Console
py36
py27
Other
Terminal
Text File


JupyterLab
localhost:9000/lab
Search
File Edit View Run Kernel Tabs Settings Help
Name
Last Modified
cache 2 days ago
envs 2 days ago
work 2 months ago
test.png a minute ago
IPython: home/jovyan
(base) jovyan@c4dc427c64ce:~$ conda activate py36
(py36) jovyan@c4dc427c64ce:~$ ipython
Python 3.6.7 | packaged by conda-forge | (default, Feb 28 2019, 09:07:38)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.4.0 -- An enhanced Interactive Python. Type '?' for help.
In [1]: from matplotlib import pyplot
fig = pyplot
In [2]: fig = pyplot.figure()
In [3]: ax = fig.add_subplot(111)
In [4]: ax.plot(range(10),range(10))
Out[4]: [<matplotlib.lines.Line2D at 0x7fbb5c5311d0>]
In [5]: fig.savefig('test.png')
In [6]:

JupyterLab

localhost:9000/lab

Search

File Edit View Run Kernel Tabs Settings Help

Name	Last Modified
cache	2 days ago
envs	2 days ago
work	2 months ago
cedar2019.ipynb	a minute ago
test.png	4 minutes ago

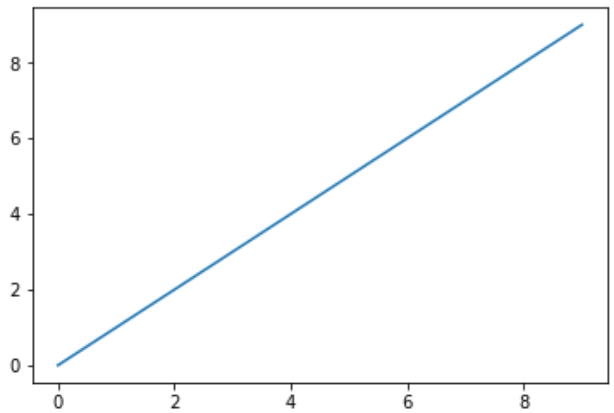
cedar2019.ipynb

Code py36

[1]:
```
%matplotlib inline
from matplotlib import pyplot
```

[2]:
```
fig = pyplot.figure()
ax = fig.add_subplot(111)
ax.plot(range(10),range(10))
```

[2]: `[<matplotlib.lines.Line2D at 0x7fb083612ac8>]`



[ ]:

Coming Soon

Adding the commands:

* **export_bucket** - export a bucket so that you can share it with others and upload it to online cite-able repositories such as Zenodo
* **import_bucket** - import a bucket that was shared with you or that was downloaded from a repository

Resen

Example Usage

Integrated Geoscience Workshop

Integrated Geoscience Workshop

Wednesday Night 7pm at 109 N Guadalupe St

* Resen Online
* Guided Tutorial
* Try Resen!

If you want to try Resen locally, please work through installation and usage instructions on readthedocs: https://resen.readthedocs.io/en/latest

Feedback, questions, suggest a package:

* ingeo-team@ingeo.datatransport.org