



U.S. DEPARTMENT OF  
**ENERGY**



# Developing Pegasus Workflows via Jupyter Notebooks

---

Rafael Ferreira da Silva  
rafsilva@isi.edu



USC Viterbi  
School of Engineering  
Information Sciences Institute

<http://pegasus.isi.edu>

# Jupyter Notebooks

## From Jupyter.org:

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

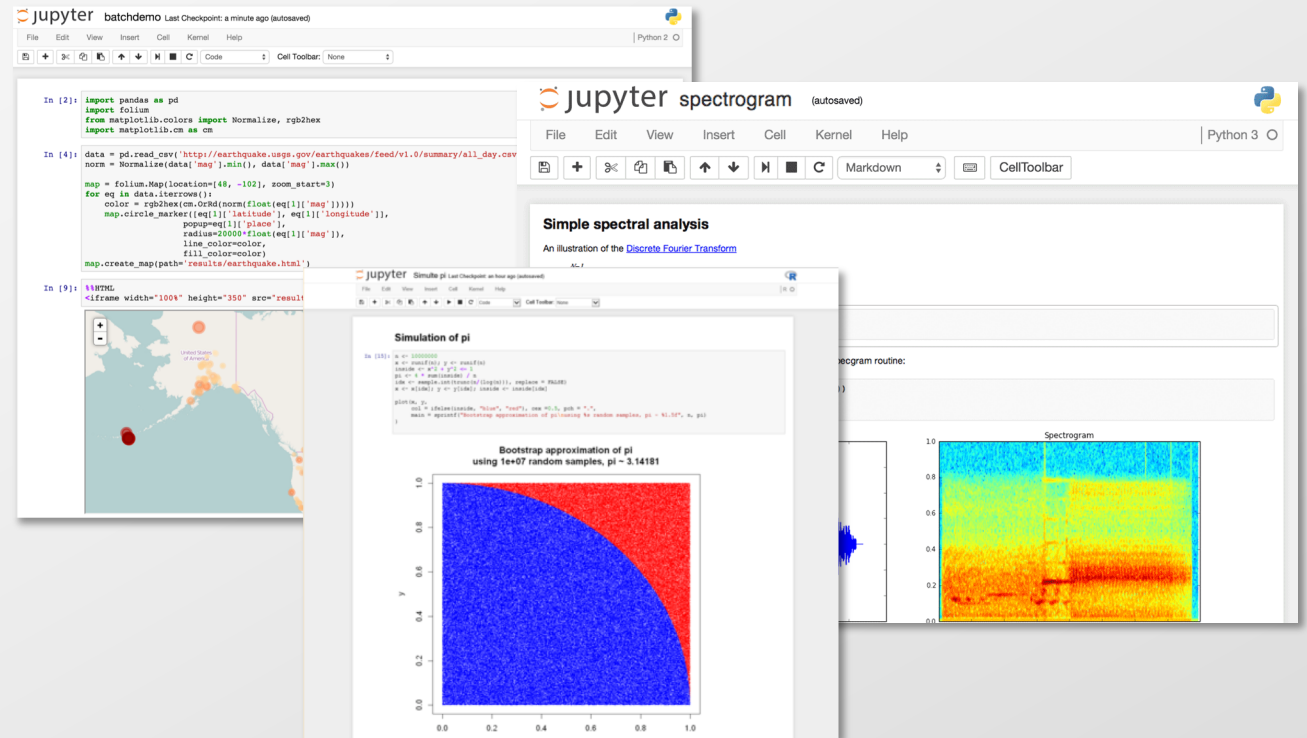
## Key Advantages

- Collaboration
- Easy access to resources
- Building blocks
- Reproducibility

## Examples

- LIGO Gravitational Wave Data
- Satellite Imagery Analysis
- 12 Steps to Navier-Stokes
- Computer Vision
- Machine Learning

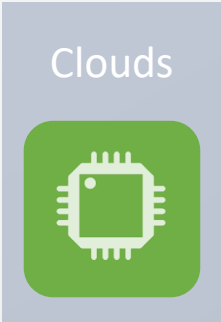
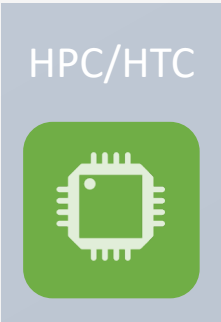
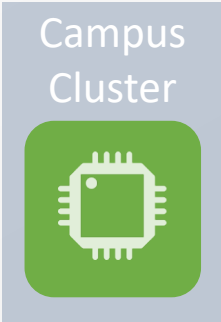
<https://unidata.github.io/online-python-training/introduction.html>



# Running Pegasus workflows with Jupyter



WAN LAN



The screenshot shows a Jupyter notebook titled "Pegasus-Tutorial-Split". The top part displays a Directed Acyclic Graph (DAG) with nodes like "clean\_up\_local\_level\_4.0", "stage\_out\_local\_local\_1.1", "register\_local\_1.1", and "cleanup\_split\_0\_local". Below the DAG, there is a text block explaining the `status()` method and its arguments: `loop` and `delay`. A code cell shows the command `instance.status(loop=True, delay=5)` and its output: `Progress: 100.0% (Success) (Completed: 17, Queued: 0, Running: 0, Failed: 0)`. A final text block explains how to use the `outputs()` command.

```

File for submitting this DAG to Condor: split-0.dag.condor.sub
Log of DAGMan debugging messages: split-0.dag.dagman.out
Log of Condor library output: split-0.dag.lib.out
Log of Condor library error messages: split-0.dag.lib.err
Log of the life of condor_dagman itself: split-0.dag.dagman.log

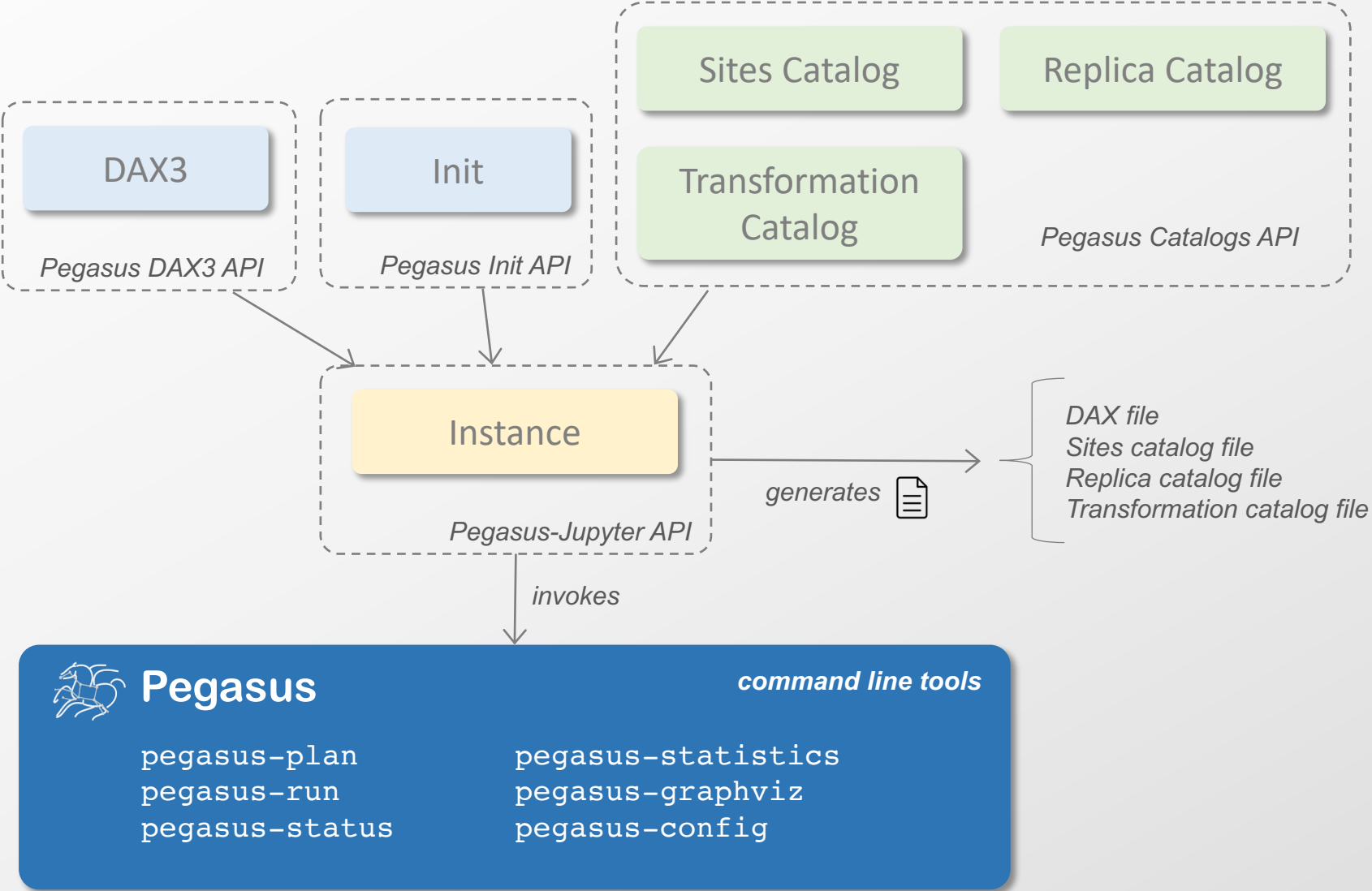
-----
Your database is compatible with Pegasus version: 4.7.0
Submitting to condor split-0.dag.condor.sub
Submitting job(s).
1 job(s) submitted to cluster 1068.

Your workflow has been started and is running in the base directory:
/Users/silva/Downloads/split-submit-host-2017-03-27T10:17:45/submit/silva/pegasus/split/run0002

*** To monitor the workflow you can run ***

pegasus-status -l /Users/silva/Downloads/split-submit-host-2017-03-27T10:17:45/submit/silva/pegasus/split/run0002
    
```

# Pegasus – Jupyter Integration Overview



# Pegasus-Jupyter Python API

```
from Pegasus.jupyter.instance import *
```

*importing the API*

```
instance = Instance(dax)
```

*creating an instance  
of the DAX*

```
instance.run(site='condorpool')
```

*running a workflow*

```
# Create an abstract dag
```

```
dax = ADAG("split")
```

```
# the split job that splits the webpage into smaller chunks
```

```
split = Job("split")
```

```
split.addArguments("-l", "100", "-a", "1", webpage, "part.")
```

```
split.uses(webpage, link=Link.INPUT)
```

```
# associate the label with the job. All jobs with same label
```

```
# are run with PMC when doing job clustering
```

```
split.addProfile( Profile("pegasus", "label", "p1"))
```

```
dax.addJob(split)
```

*using the Pegasus DAX3 API  
to write a workflow*



```
instance.status(loop=True, delay=5)
```

*monitoring a workflow execution*

```
Progress: 100.0% (Success) (Completed: 17, Queued: 0, Running: 0, Failed: 0)
```

Available since:

 Pegasus 4.8

 Pegasus

# Additional capabilities...

```
wf_image_exe = instance.view(abstract=False)

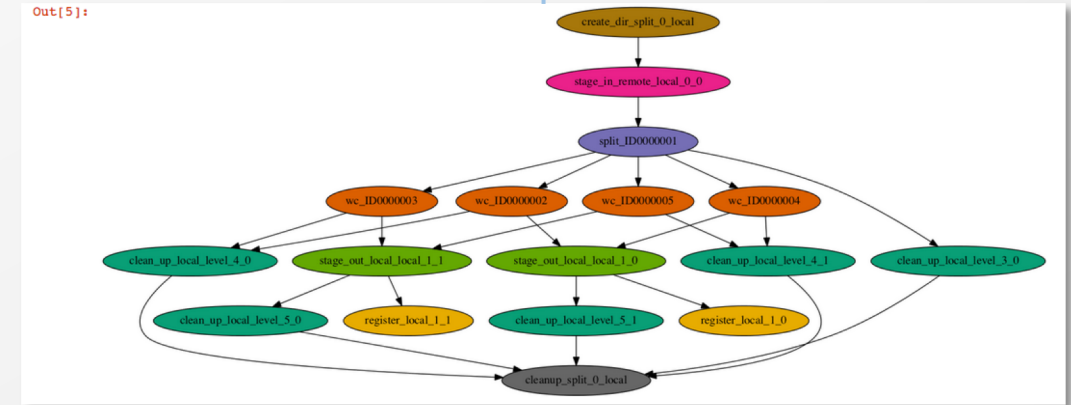
# IPython package for visualizing images
from IPython.display import Image
Image(wf_image_exe)
```

*visualizing the workflow*

```
instance.statistics()
```

*collect statistics*

**Workflow Wall Time: 47 min, 23 secs**



```
# creating a site catalog. A local site is automatically created
sites_catalog = SitesCatalog()

# adding a site with some profile characteristics
sites_catalog.add_site('condorpool', Arch.X86_64, OSType.LINUX)
sites_catalog.add_profile('condorpool', Namespace.ENV, 'JAVA_HOME', '/usr/local/jre')

dax.set_sites_catalog(sites_catalog)
```

*create catalogs: site, replica, and transformation*

Available since:

 Pegasus 4.8

 Pegasus

# Requirements

## Pegasus submit node

Python 2.7 or higher (Jupyter requires version 2.7+)

Java 1.8 or higher

Pegasus 4.8.0 or higher

<https://pegasus.isi.edu/downloads/>

Jupyter

<http://jupyter.org/install.html>

## JupyterHub

Due to the strict requirement of Python 3 for running the multi-user hub, our API requires the Python `future` package in order to be compatible with Python 3.

Python Future package:

<https://pypi.python.org/pypi/future>



# References

## Documentation

<https://pegasus.isi.edu/documentation/jupyter.php>

## API Reference

Instance: <https://pegasus.isi.edu/documentation/python/instance.html>

Catalogs:

[https://pegasus.isi.edu/documentation/python/sites\\_catalog.html](https://pegasus.isi.edu/documentation/python/sites_catalog.html)

[https://pegasus.isi.edu/documentation/python/replica\\_catalog.html](https://pegasus.isi.edu/documentation/python/replica_catalog.html)

[https://pegasus.isi.edu/documentation/python/transformation\\_catalog.html](https://pegasus.isi.edu/documentation/python/transformation_catalog.html)

## Example Tutorial Notebook

Distributed with Pegasus releases (since 4.8)

Also available in the Pegasus Tutorial VM (<https://pegasus.isi.edu/downloads/>)

Instructions

<https://pegasus.isi.edu/documentation/jupyter-example.php>





# Pegasus est. 2001

Automate, recover, and debug scientific computations.

## Get Started

**Pegasus Website**

<http://pegasus.isi.edu>

**Users Mailing List**

[pegasus-users@isi.edu](mailto:pegasus-users@isi.edu)

**Support**

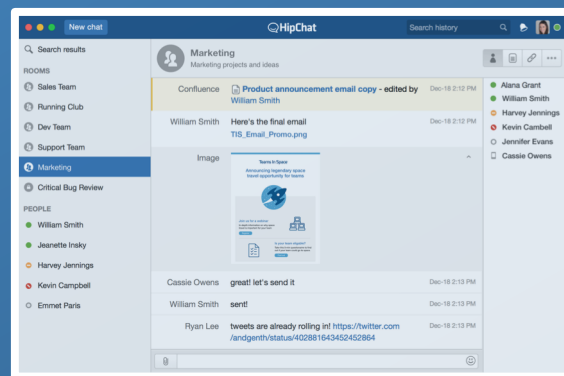
[pegasus-support@isi.edu](mailto:pegasus-support@isi.edu)

### Pegasus Online Office Hours

<https://pegasus.isi.edu/blog/online-pegasus-office-hours/>

*Bi-monthly basis on second Friday of the month, where we address user questions and also apprise the community of new developments*

### HipChat





# Developing Pegasus Workflows via Jupyter Notebooks

## Thank You

---

## Questions?

Rafael Ferreira da Silva, Ph.D.  
rafsilva@isi.edu

USC Viterbi  
School of Engineering  
Information Sciences Institute

## Meet our team



Ewa Deelman



Karan Vahi



Mats Rynge



Rajiv Mayani



Rafael Ferreira da Silva