

IP[y]:
IPython



The Architecture of Jupyter

interactive data
exploration and
visualization across
languages

Fernando Pérez
(@fperez_org & fperez@lbl.gov)

LBL & UC Berkeley



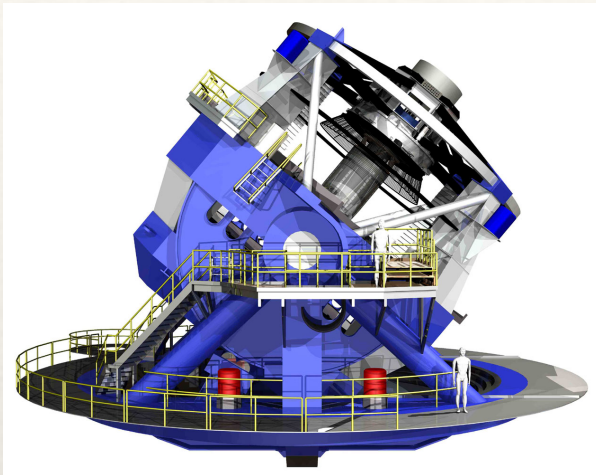
Mandatory COI disclosure



**“The purpose of computing is insight,
not numbers”**

–Hamming’62

Every research discipline is now awash in data



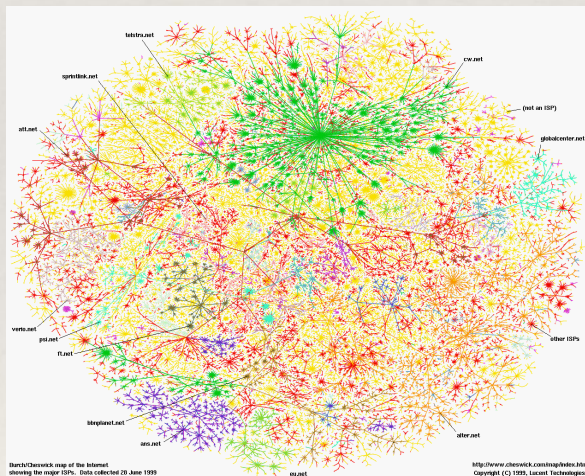
Astronomy: LSST



Physics: LHC



Personalized, data-driven
medicine



Sociology: The Web



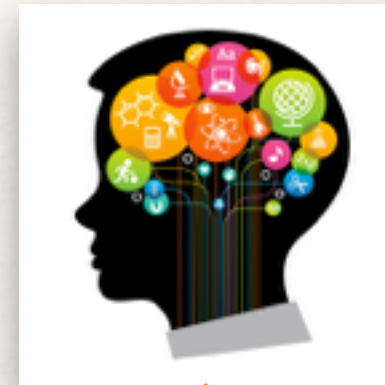
Biology: Sequencing



Economics: POS
terminals

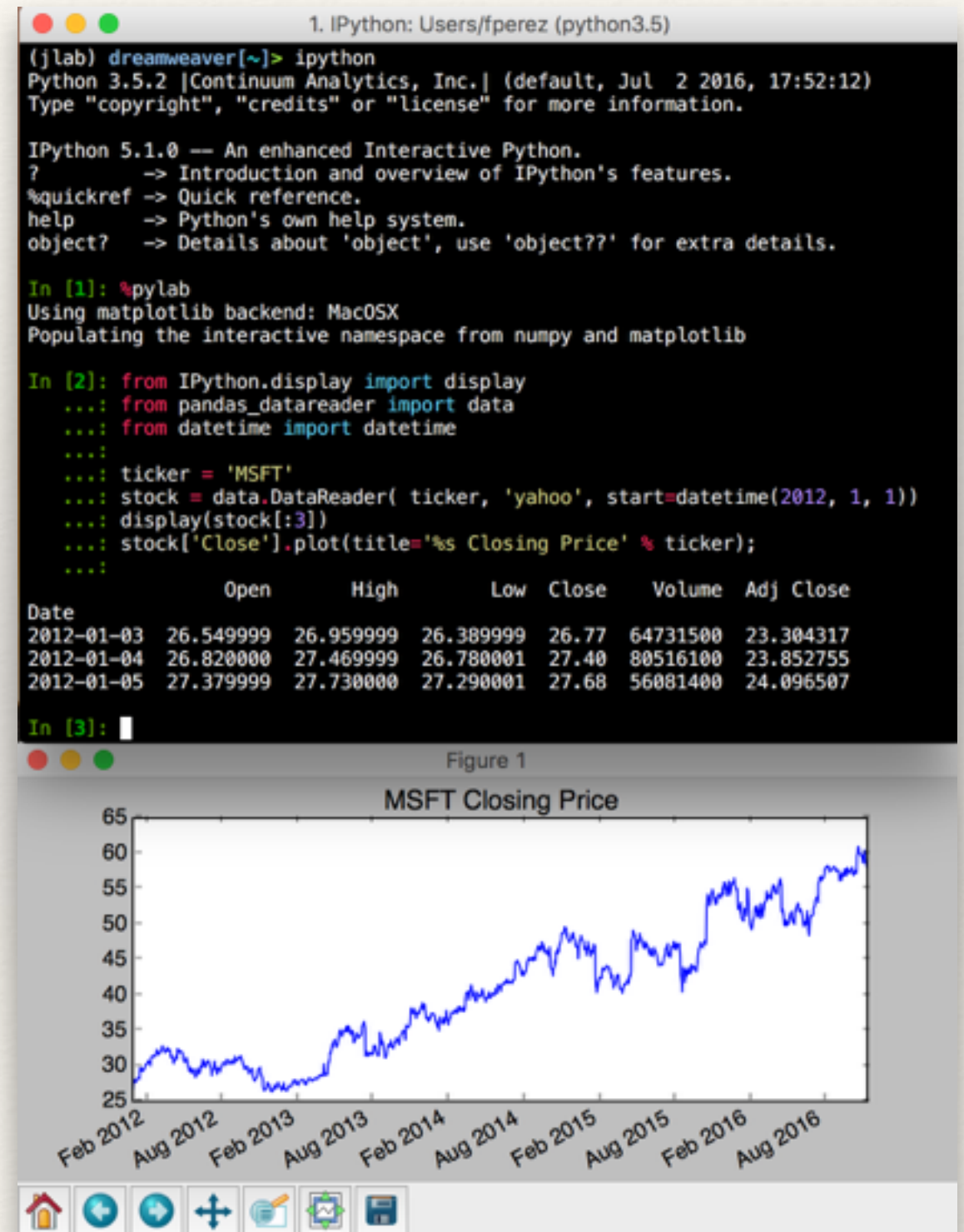


Neuroscience: EEG, fMRI



IPython: Interactive Python, 2001

- ❖ Object Introspection (TAB!)
- ❖ OS Integration
- ❖ Rich terminal client
- ❖ GUI support (plots, ...)
- ❖ %magic commands
- ❖ Embeddable



Team today: where *all the credit* goes



Plus ~ 500 more Open source contributors!

Funding and partnerships



**ALFRED P. SLOAN
FOUNDATION**



U.S. DEPARTMENT OF
ENERGY

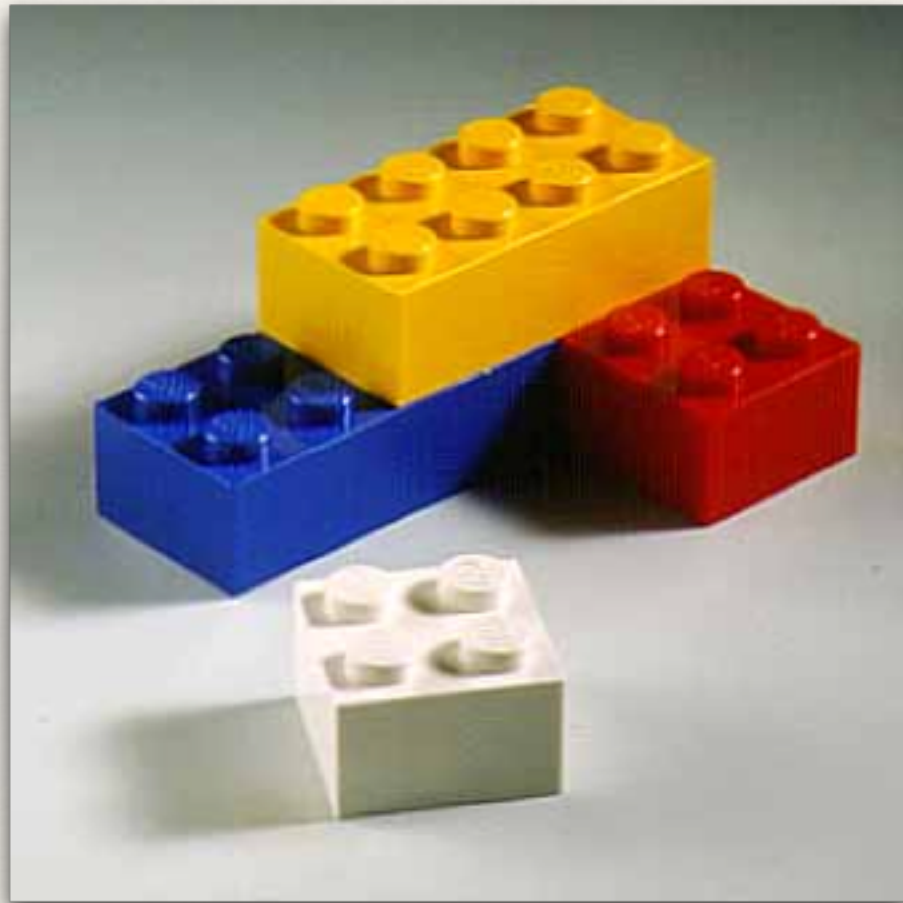
O'REILLY*



SIMONS FOUNDATION



Pragmatic abstractions: vocabulary



Interactivity as a protocol

- ❖ The REPL as a network protocol

- ❖ Kernels

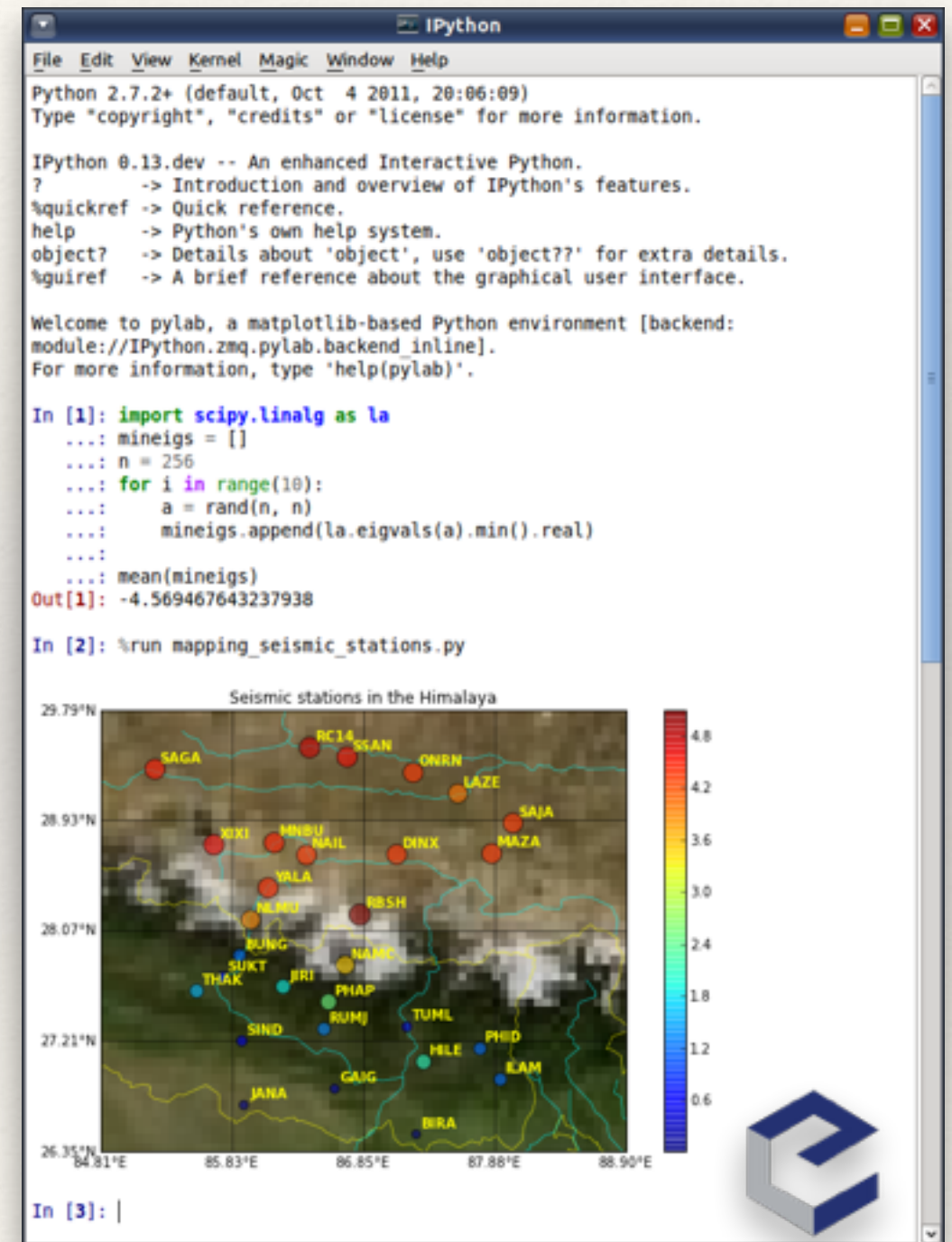
 - ❖ execute code

- ❖ Clients

 - ❖ Read input

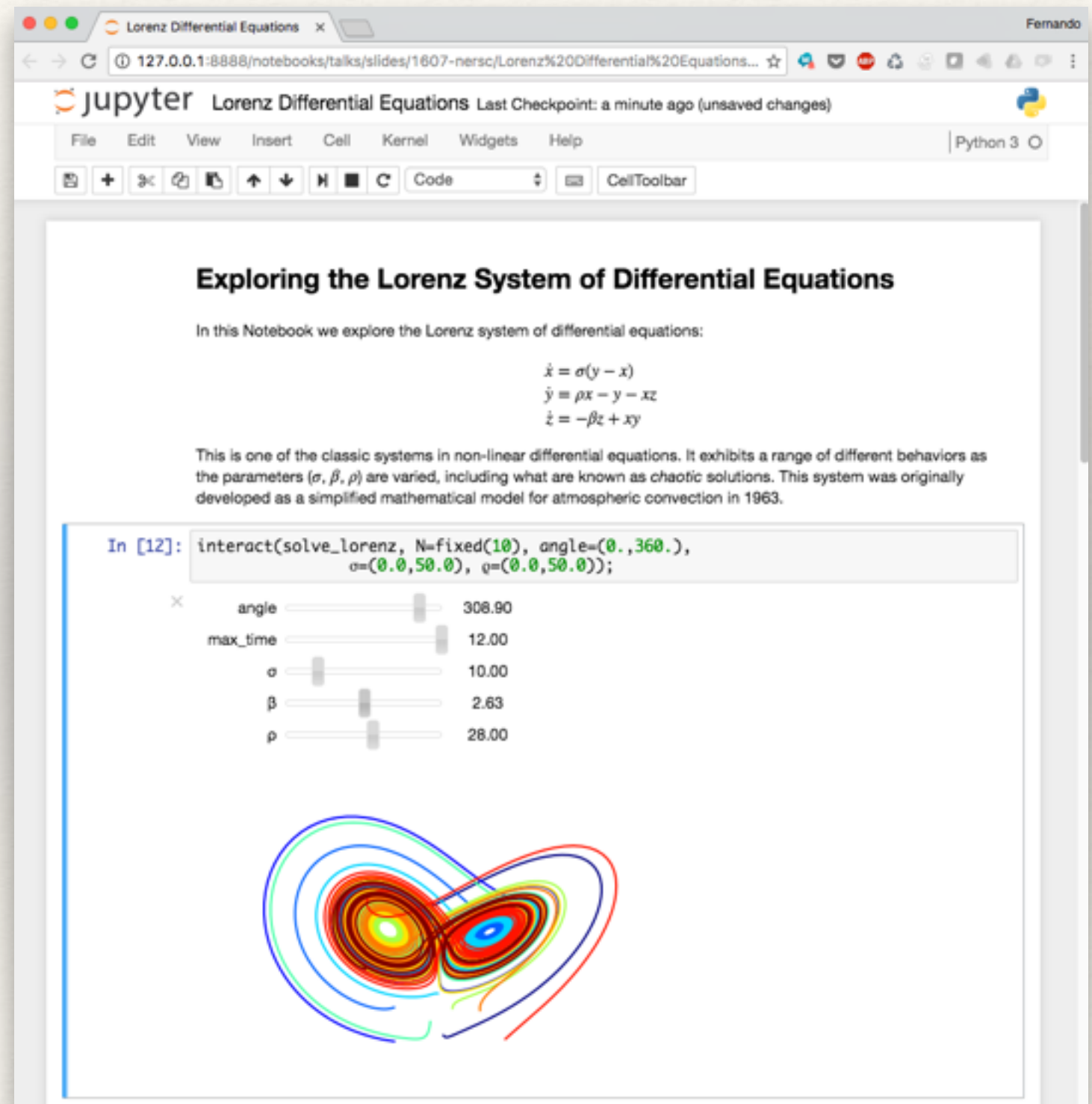
 - ❖ Present output

Simple abstractions enable rich,
sophisticated clients



The IPython/Jupyter Notebook

- ❖ Rich web client
- ❖ Text & math
- ❖ Code
- ❖ Results
- ❖ Share, reproduce.



From IPython to Project Jupyter

IP[y]:
IPython



IPython ... Jupyter

- ❖ Interactive Python shell at the terminal
- ❖ Kernel for this protocol in Python
- ❖ Tools for Interactive Parallel computing

- ❖ Network protocol for interactive computing
- ❖ Clients for protocol
 - ❖ Console
 - ❖ Qt Console
 - ❖ Notebook
- ❖ Notebook file format & tools (nbconvert...)
- ❖ Nbviewer



Language Agnostic

Protocols: kernels & clients

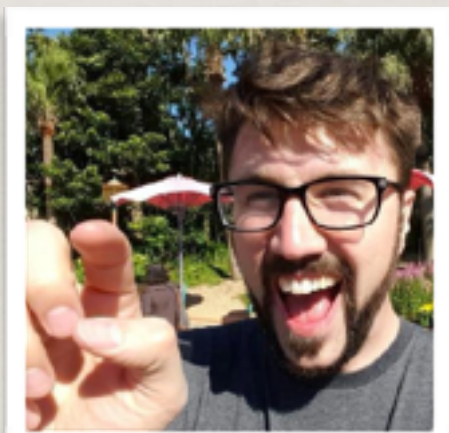
Jupyter Protocol is language agnostic



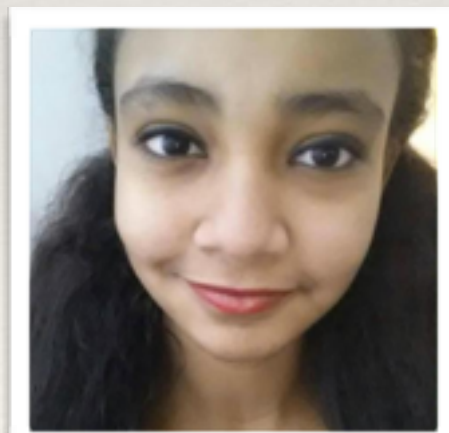
~75 different kernels: <https://github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages>

Alternate clients (nb): nteract

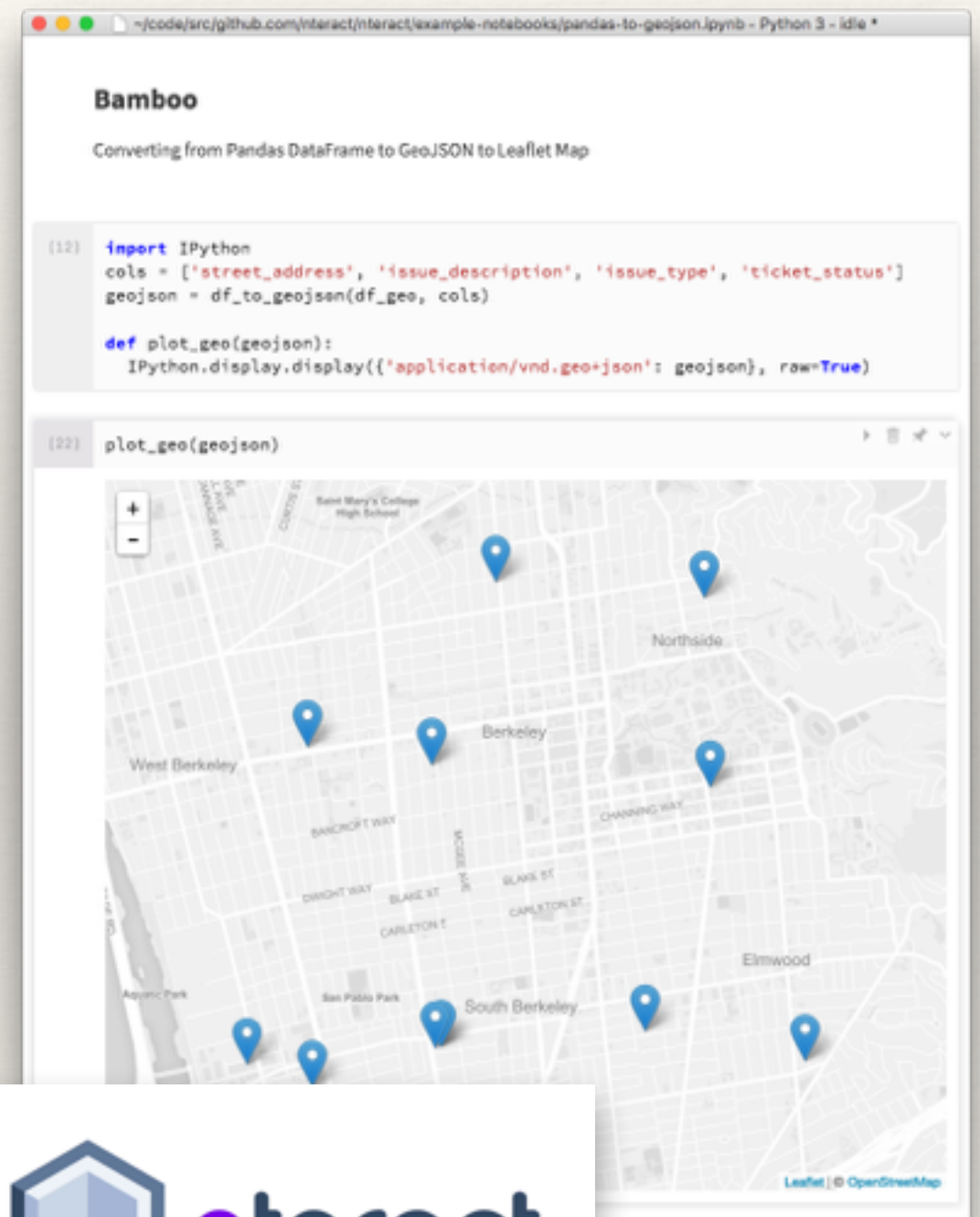
- ❖ Local desktop application
- ❖ Written in node.js (uses React)
- ❖ Uses:
 - ❖ Jupyter messaging protocols
 - ❖ Notebook file format.
- ❖ <https://github.com/nteract/nteract>



Kyle Kelley



Safia Abdalla



Alternate clients (editor): hydrogen

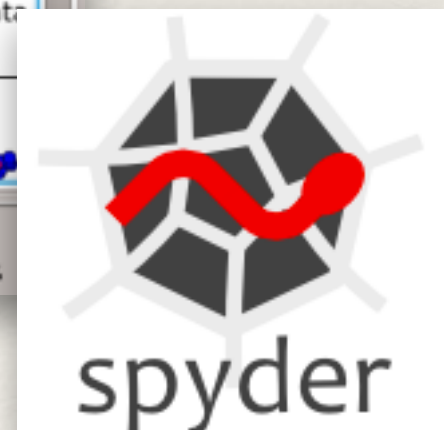
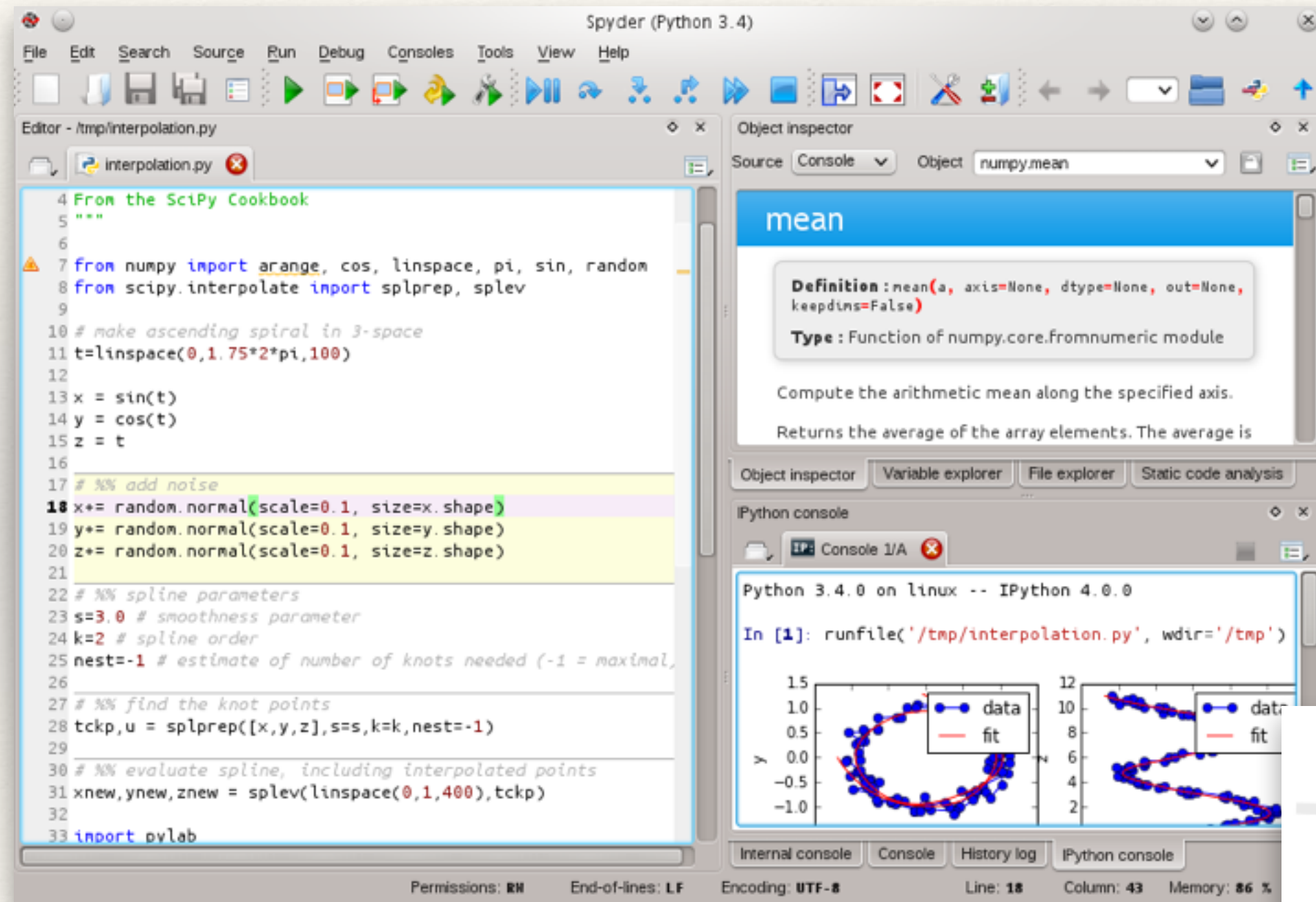


Lukas Geiger



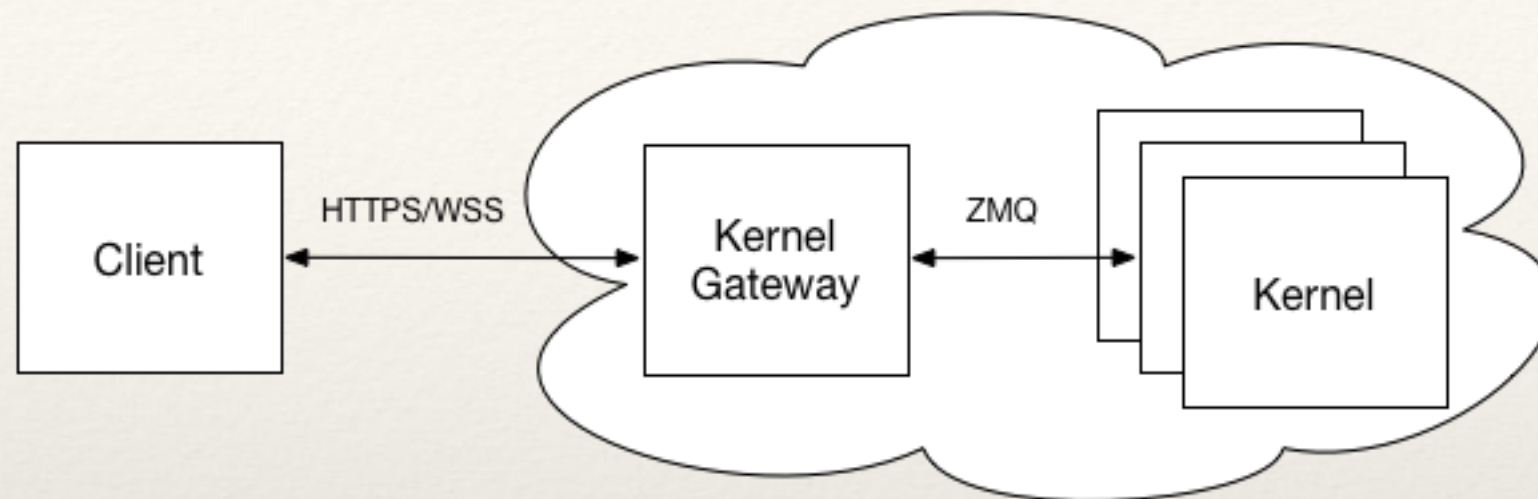
```
test.py
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from IPython.display import Latex, Markdown
4 import pandas as pd ✓
5
6 # One line outputs
7 print 'Hello World!'
8 print 'This is \x1b[00;38;5;033mHydrogen\x1b[0m:'
9
10 # An error will be thrown here
11 thiswillerror
12
13 # create x axis value
14 x = np.linspace(0, 20, 100)
15 x
16
17 # plot inline figure
18 plt.plot(x, np.sin(x))
19 plt.show()
20
21 # display a data frame
22 df = pd.DataFrame({'A': 1.,
                     'B': pd.Timestamp('20130102'),
                     'C': pd.Series(1, index=list(range(4)), dtype='float32'),
                     'D': np.array([3] * 4, dtype='int32'),
                     'E': pd.Categorical(["test", "train", "test", "train"]),
                     'F': 'foo'})
```

Alternate clients (IDE): spyder



<https://github.com/spyder-ide/spyder>

Jupyter Kernel Gateway

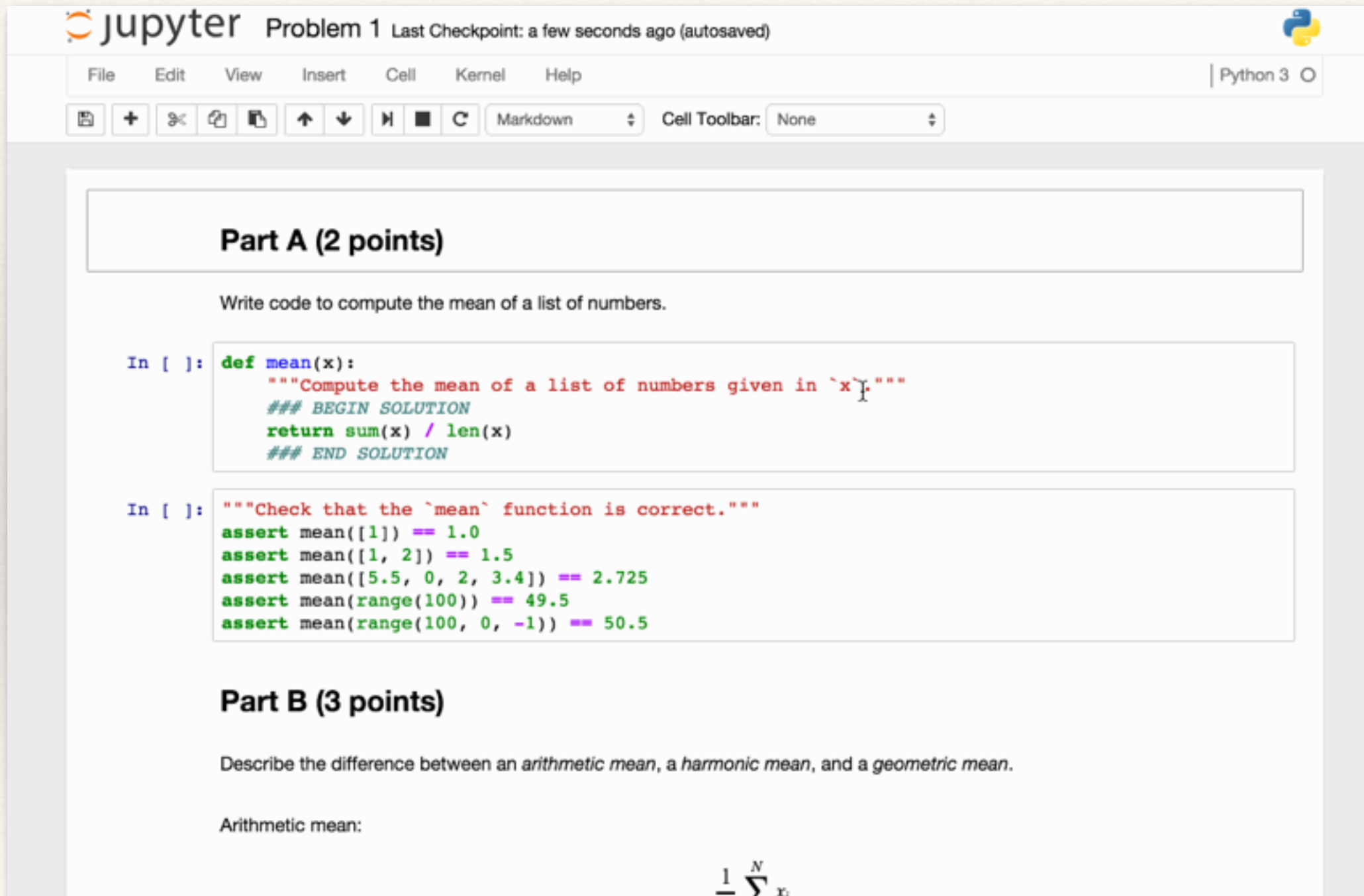


- ❖ Web server for spawning and communicating with kernels over HTTP / WebSocket
- ❖ Defaults to letting web clients talk the Jupyter protocol
- ❖ Extensible with other modes / personalities

Slides / credit: Peter Parente, [@parente](#)

Notebooks: structured files
(metadata!)

nbgrader: notebook homework



The screenshot shows a Jupyter Notebook interface. At the top, the title bar says "jupyter Problem 1 Last Checkpoint: a few seconds ago (autosaved)". Below the title bar is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar is a "Python 3" button. Below the menu bar is a toolbar with various icons for file operations, cell navigation, and execution. The main content area is divided into two sections: "Part A (2 points)" and "Part B (3 points)".

Part A (2 points)

Write code to compute the mean of a list of numbers.

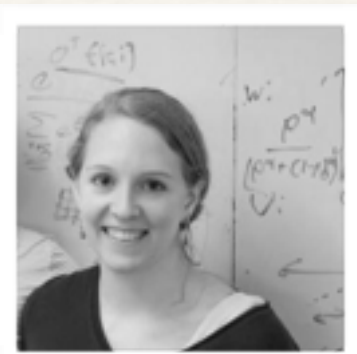
```
In [ ]: def mean(x):  
        """Compute the mean of a list of numbers given in `x`."""  
        ### BEGIN SOLUTION  
        return sum(x) / len(x)  
        ### END SOLUTION
```

```
In [ ]: """Check that the `mean` function is correct."""  
assert mean([1]) == 1.0  
assert mean([1, 2]) == 1.5  
assert mean([5.5, 0, 2, 3.4]) == 2.725  
assert mean(range(100)) == 49.5  
assert mean(range(100, 0, -1)) == 50.5
```

Part B (3 points)

Describe the difference between an *arithmetic mean*, a *harmonic mean*, and a *geometric mean*.

Arithmetic mean:

$$\frac{1}{N} \sum_{i=1}^N x_i$$


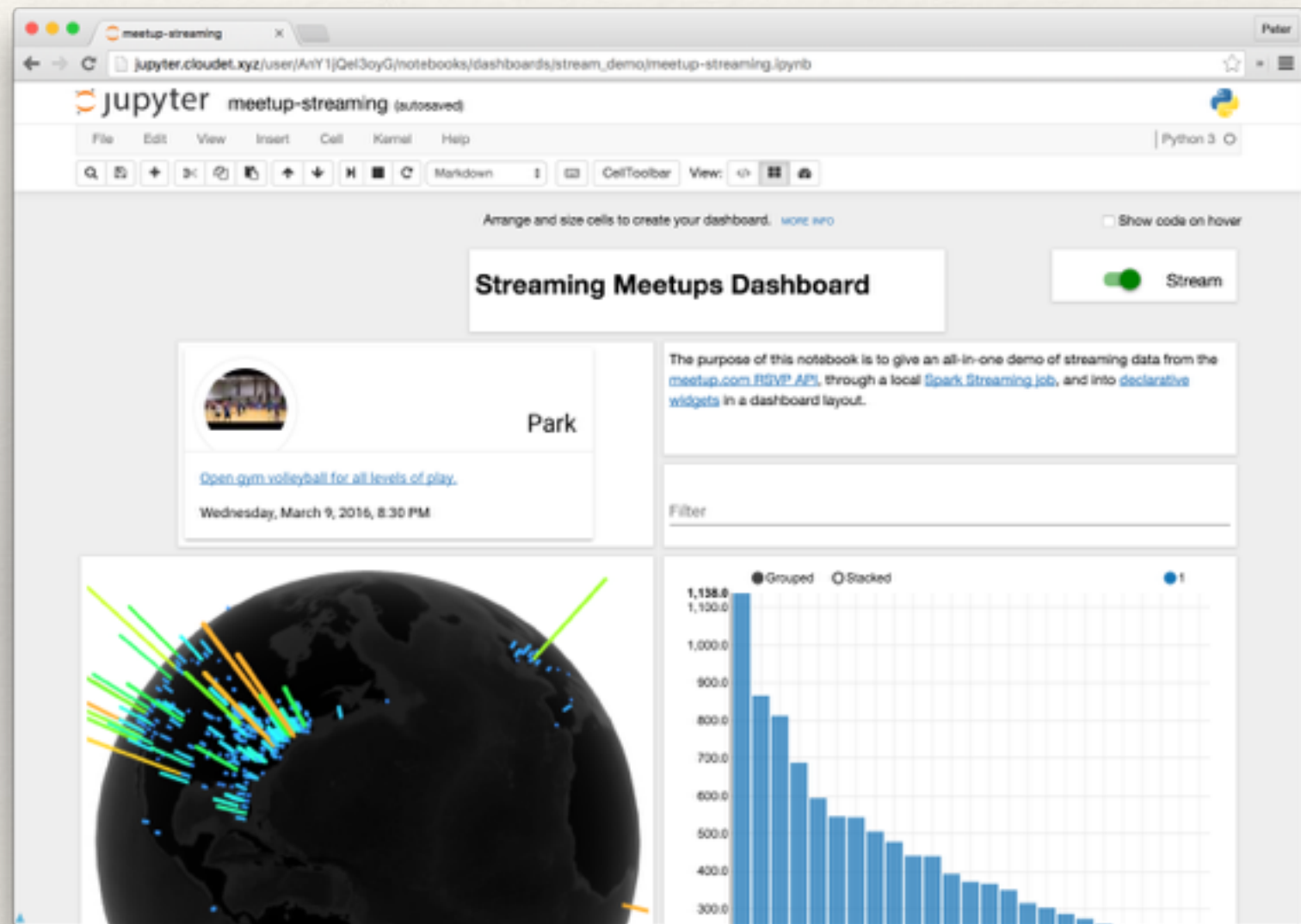
J. Hamrick
UC Berkeley



B. Granger
Cal Poly

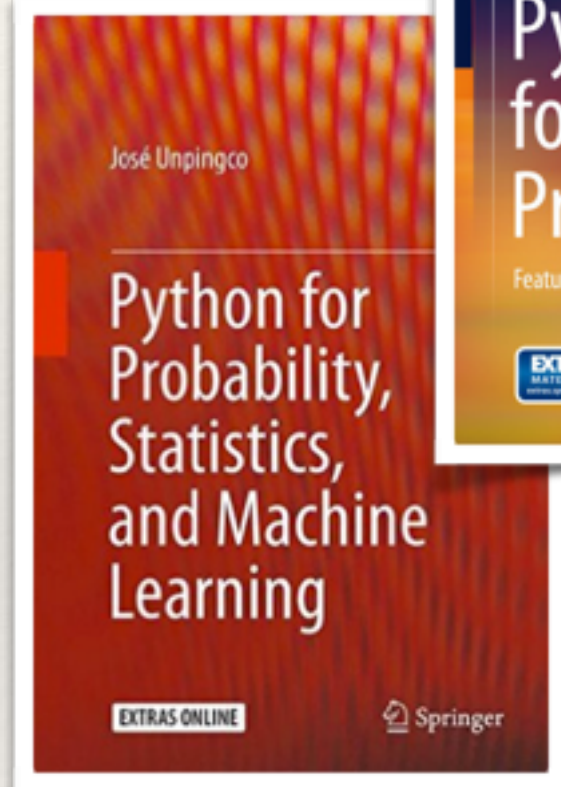
Notebooks as dashboards

- ❖ Same file format
- ❖ Metadata based
- ❖ Live dashboard with Jupyter kernel
- ❖ Web view with hidden details (code, setup, etc)

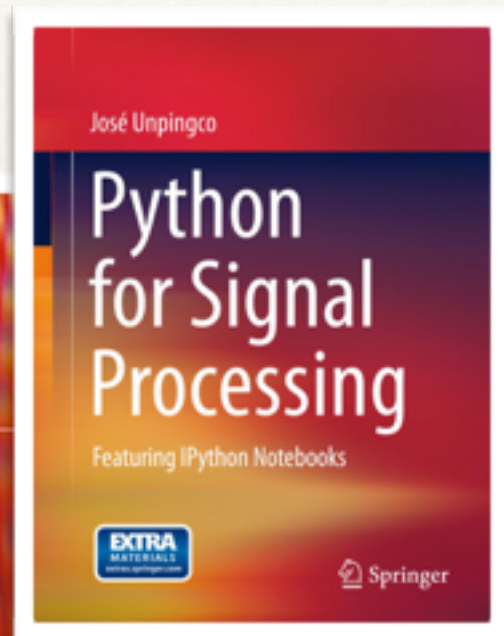


<https://github.com/jupyter-incubator/dashboards>

“Executable books”

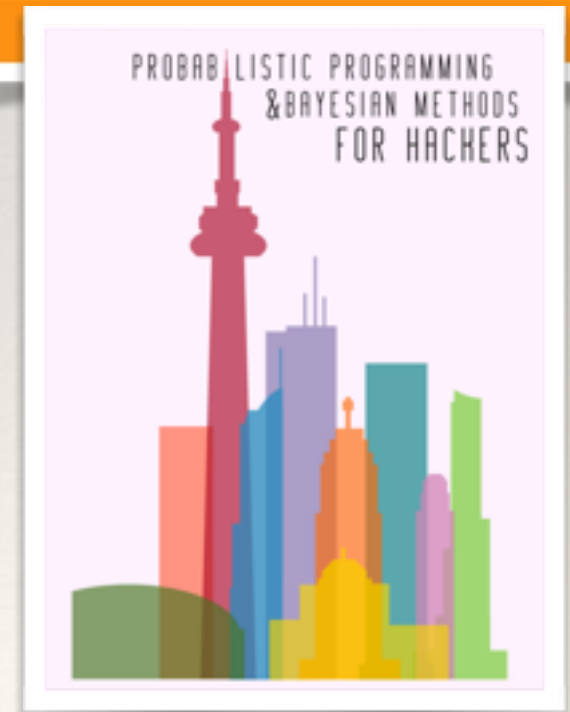
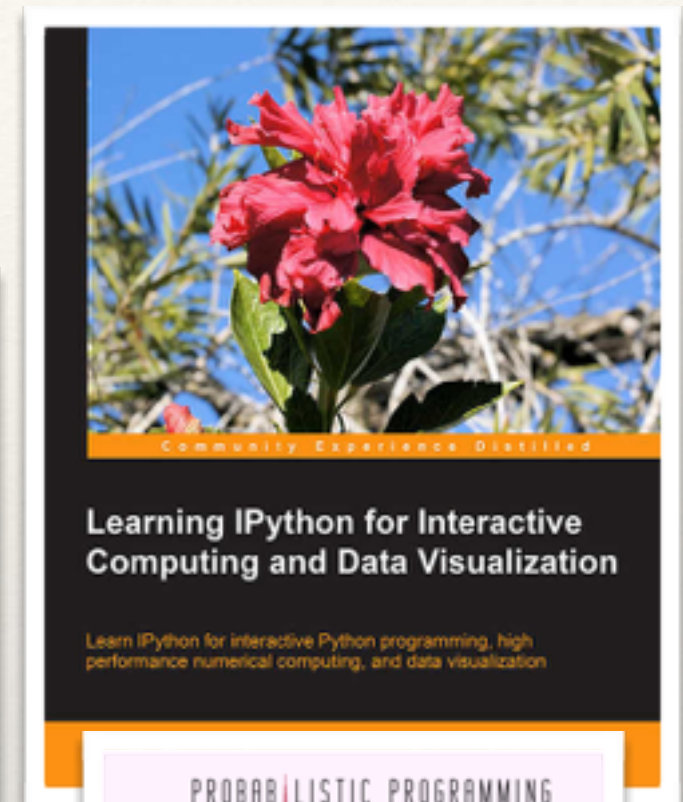
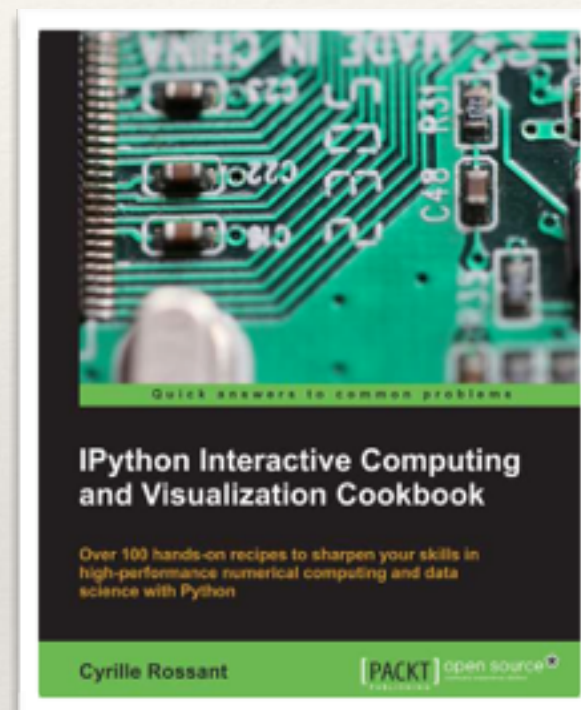


By Jose Unpingco



By Matthew Russell

By Cyrille Rossant



By Cameron Davidson-Pilon

“The Notebook”: reusable web application

mybinder.org



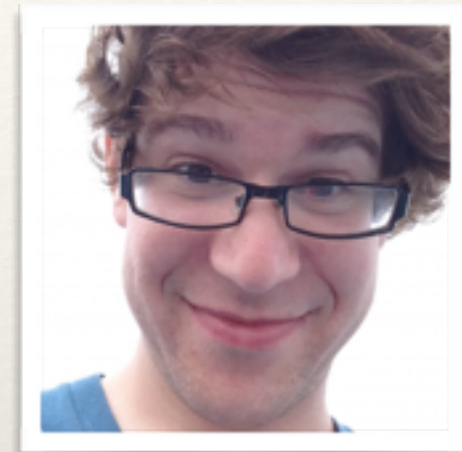
Turn a GitHub repo into a
collection of interactive
notebooks

Have a repository full of Jupyter notebooks? With Binder, you can add a badge that opens those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

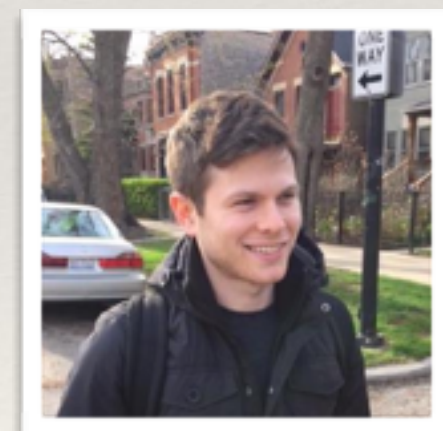
100% free and open source. Browse examples. Read the FAQ.

Build a repository

submit



[github.com / freeman-lab](https://github.com/freeman-lab)



[github.com / andrewosh](https://github.com/andrewosh)

Andrew Osheroff's SciPy'16 talk:

<https://www.youtube.com/watch?v=OK6M4w7LYIc>

Oriole: executable, video-narrated tutorials

Oriole Online Tutorials

Learn alongside smart people solving hard problems

Oriole is a unique new medium that blends code, data, text, and video into a narrated learning experience with executable content.

Led by some of the most brilliant minds in technology, each lesson is an easily digestible and engaging thought-by-thought tour of the instructor's approach to the problem in both narrative and executable code. No set-up or installation is necessary; Oriole Online Tutorials require nothing more than an internet connection and a laptop. You can write and run code within the environment. Make a mistake? Change it, and try again.

Oriole combines the expert insight and hands-on learning of in-person or online courses with the on-demand, at-your-own desk, back-up-and-run-it-again convenience of video training. You learn by doing, on your own schedule, and at your own pace.

In Oriole, we get the complete integration of video synchronized with the flow of the text, as well as the ability to execute the code: **this is probably as close as we can get to learning side-by-side with Peter himself.**

Fernando Perez, creator of IPython, which evolved into Project Jupyter.

Regex Golf - O'Reilly Media

Arbitrary Lists

What Next?

Summary

Thanks!

PETER-REGEX GOLF
SO I WROTE A PROGRAM THAT PLAYS REGEX GOLF WITH ARBITRARY LISTS...
UNOH!

We have now fulfilled panel two of the strip. Let's try another example, separating the top ten best-selling drugs from the top 10 cities to visit:

```
1 drugs = words('lipitor nexium plavix advair abilify so  
2 cities = words('paris trinidad capetown riga zurich s  
3  
4 report(drugs, cities)
```

Characters: 15, Parts: 6, Competitive ratio: 5.3,
Winners: 10, Losers: 10

'o.\$(x|l|r|q|b|en'

Run Again



Paco Nathan



Taylor Martin




Andrew Odewahn



oreilly.com/learning/regex-golf-with-peter-norvig

Microsoft, IBM, Google, Continuum...



Microsoft

Microsoft Azure Notebooks

Overview Libraries FAQ/Support What's New Sign In

jupyter

Notebooks hosted on Microsoft Azure

Go to my Notebook Server

Show me some samples

WHAT IS JUPYTER?

- Interactive Notebooks for Data Science and Technical Computing
- Browser-based REPL with Markdown and inline interactive graphics
- Support for Python 2, Python 3 and R

ABOUT THIS SERVICE

- This notebook service is provided by the Azure Data Group
- Your notebooks are stored in Azure and linked to your Microsoft account
- Enjoy some free cycles on us

Machine Learning Blog


Introducing Jupyter Notebooks in Azure ML Studio

Related Links

- Microsoft Azure ML
- Microsoft Data Platform Insider Blog
- Microsoft Big Data Solutions
- Data Science Dojo
- Azure Big Data Blog

Tags

- ADF
- asa
- Azure ML
- Azure Stream Analytics
- Cortana Analytics Suite
- Customers Data Science Machine Learning Partners
- Python r Webinar



IBM

Data Scientist Workbench

Prepare data. Analyze data. Get answers.

Prepare data effortlessly.

Explore Data.

Find and explore large data sets with ease.

Clean and Transform Data

Easily clean messy data and transform formats.

Reconcile and Match Data

Link and extend your datasets with web services.

Analyze data interactively.

Powerful Notebook Environment

Use IPython/Jupyter notebooks to combine code execution, text, plots and rich media.

Extend

Install Python and R libraries. Install others as needed.

Create and Share

Share what others have done and

Google

Google Cloud Platform

Why Google Products Solutions Launcher Pricing Customers Documentation Support Partners Free Trial Contact Sales

CLOUD DATALAB^{BETA}

An easy-to-use interactive tool for large-scale data exploration, analysis, and visualization.

TRY IT FREE

Powerful Data Exploration

Cloud Datalab is a powerful interactive tool created to explore, analyze and visualize data with a single click on Google Cloud Platform. It runs on Google App Engine and orchestrates multiple services automatically so you can focus on exploring your data.

Google Cloud Datalab

Notebooks Sessions

Notebook Folder Upload Repository

/ datalab / intro

- Introduction to Notebooks.ipynb
- Introduction to Python.ipynb
- Working with Datalab.ipynb

CONTINUUM ANALYTICS

Gallery About Pricing Anaconda Help Download Anaconda Sign In

Interactive Stock Prices Downsampling

Hover Over Points

My Gist Activity

Data Visualization in Python

Scientific Programming in Python

Texas Unemployment Choropleth

JupyterHub: multiuser support



Jupyter for Organizations

JupyterHub is a multiuser version of the notebook designed for centralized deployments in companies, university classrooms and research labs.



Pluggable authentication

Manage users and authentication with PAM, OAuth or integrate with your own directory service system. Collaborate with others through the Linux permission model.



Centralized deployment

Deploy the Jupyter Notebook to all users in your organization on centralized servers on- or off-site.



Container friendly

Use Docker containers to scale your deployment and isolate user processes using a growing ecosystem of prebuilt Docker containers.



Code meets data

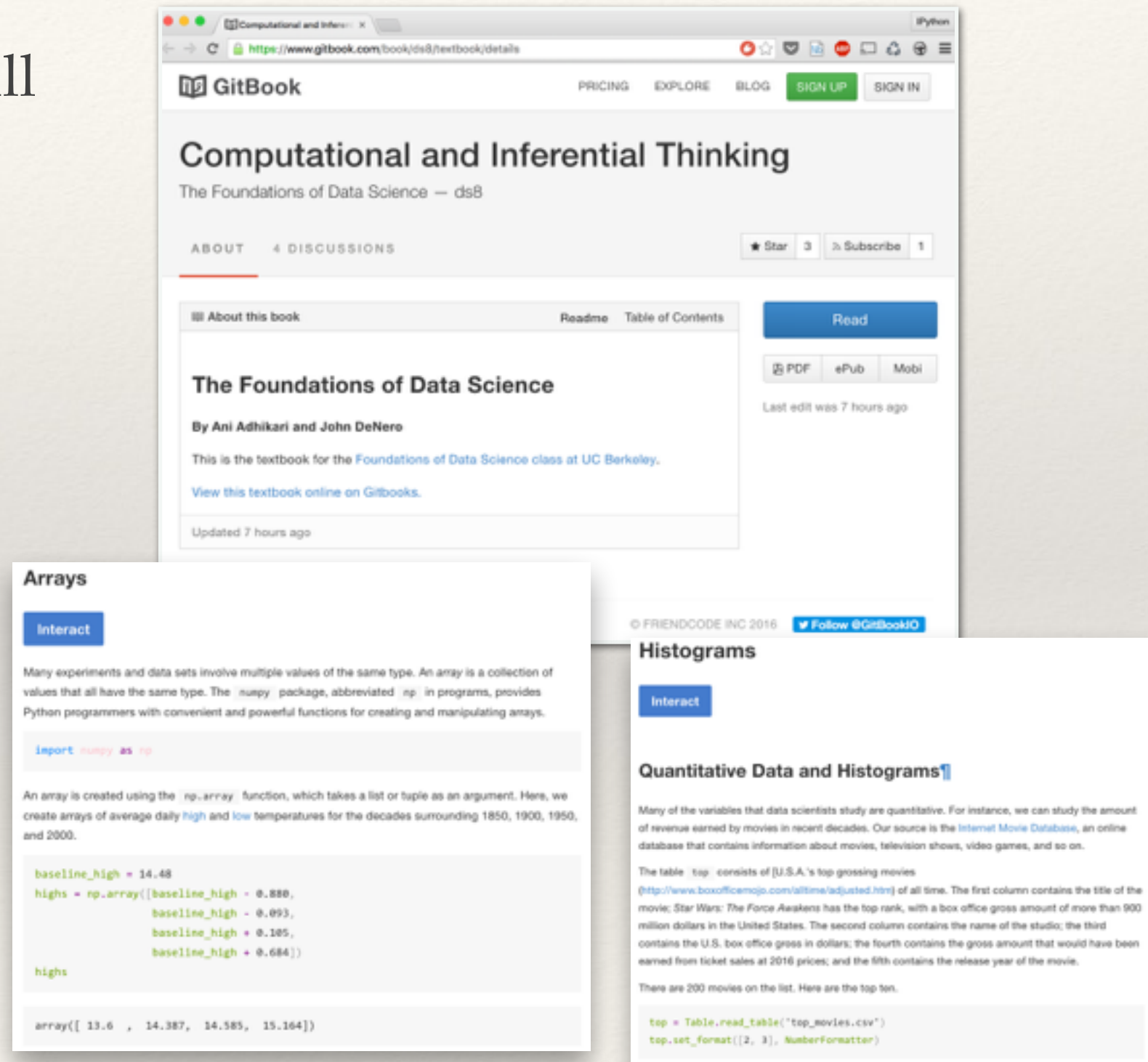
Deploy the Notebook next to your data to provide unified software management and data access within your organization.

Berkeley's *Foundations of Data Science*

- ❖ New curriculum aimed at all freshmen at UC Berkeley
- ❖ Interactive textbook is Jupyter Notebooks
- ❖ Course deployment is JupyterHub
 - ❖ Off Jess Hamrick's work



data.berkeley.edu, data8.org



Computational and Inferential Thinking
The Foundations of Data Science — ds8

ABOUT 4 DISCUSSIONS

Star 3 Subscribe 1

Readme Table of Contents

The Foundations of Data Science

By Ani Adhikari and John DeNero

This is the textbook for the [Foundations of Data Science class at UC Berkeley](#).

[View this textbook online on Gitbooks.](#)

Updated 7 hours ago

Read

PDF ePub Mobi

Last edit was 7 hours ago

© FRIENDCODE INC 2016 Follow @GitBookIO

Arrays

Interact

Many experiments and data sets involve multiple values of the same type. An array is a collection of values that all have the same type. The `numpy` package, abbreviated `np` in programs, provides Python programmers with convenient and powerful functions for creating and manipulating arrays.

```
import numpy as np
```

An array is created using the `np.array` function, which takes a list or tuple as an argument. Here, we create arrays of average daily `high` and `low` temperatures for the decades surrounding 1850, 1900, 1950, and 2000.

```
baseline_high = 14.48
highs = np.array([baseline_high - 0.880,
                  baseline_high - 0.093,
                  baseline_high - 0.105,
                  baseline_high - 0.684])
highs
```

```
array([ 13.6 , 14.387, 14.585, 15.164])
```

Histograms

Interact

Quantitative Data and Histograms¹

Many of the variables that data scientists study are quantitative. For instance, we can study the amount of revenue earned by movies in recent decades. Our source is the [Internet Movie Database](#), an online database that contains information about movies, television shows, video games, and so on.

The table `top` consists of [U.S.A.'s top grossing movies (<http://www.boxofficemojo.com/alltime/adjusted.htm>)] of all time. The first column contains the title of the movie; *Star Wars: The Force Awakens* has the top rank, with a box office gross amount of more than 900 million dollars in the United States. The second column contains the name of the studio; the third contains the U.S. box office gross in dollars; the fourth contains the gross amount that would have been earned from ticket sales at 2016 prices; and the fifth contains the release year of the movie.

There are 200 movies on the list. Here are the top ten.

```
top = Table.read_table('top_movies.csv')
top.set_format([2, 3], NumberFormatter)
```

Data Science: *Connector Courses*



Data Science,
Demography, &
Immigration



Children in the
Developing World



Data Science
For Smart Cities



Data and Ethics



Social Networks



Computational
Structures in
Data Science

BERKELEY DATA SCIENCE EDUCATION PROGRAM

Fall 2016 Connector Course Offerings

Designed to Complement

Data 8: Foundations of Data Science

data.berkeley.edu



Social Data
Revolution



Making Sense of
Cultural Data



Genomics and Data
Science



Data Science for
Cognitive
Neuroscience



Data Science and
the Mind



Probability and
Math Stats in
Data Science

JupyterHub: interactive HPC



Shreyas Cholia



Cori @ NERSC: Department of Energy
Supercomputing Center (LBNL)



Rollin Thomas

JupyterLab: the notebook,
evolved...

The “Notebook”?

The image displays a collage of Jupyter Notebook and JupyterLab interfaces, illustrating the concept of a "Notebook".

Top Left: Jupyter Notebook Interface

- Browser: `127.0.0.1:8888/notebooks/jupyter/notebooks/SierpinskiTriangle.ipynb`
- Page Title: `jupyter SierpinskiTriangle (unsaved changes)`
- Content: A notebook titled "SierpinskiTriangle" showing a fractal plot.

Bottom Left: Terminal Window

```
PID USER
86365 fperez
1 root
49 root
50 root
53 root
54 root
59 root
60 _applee
61 root
67 root
68 root
70 root
76 root
81 root
82 root
83 root
85 root
86 root
87 root
89 root
91 _locati
93 _displa
94 root
95 root
101 _ednsre
103 root
104 root
105 root
106 root
```

Bottom Center: Jupyter Notebook Interface (Code Editor)

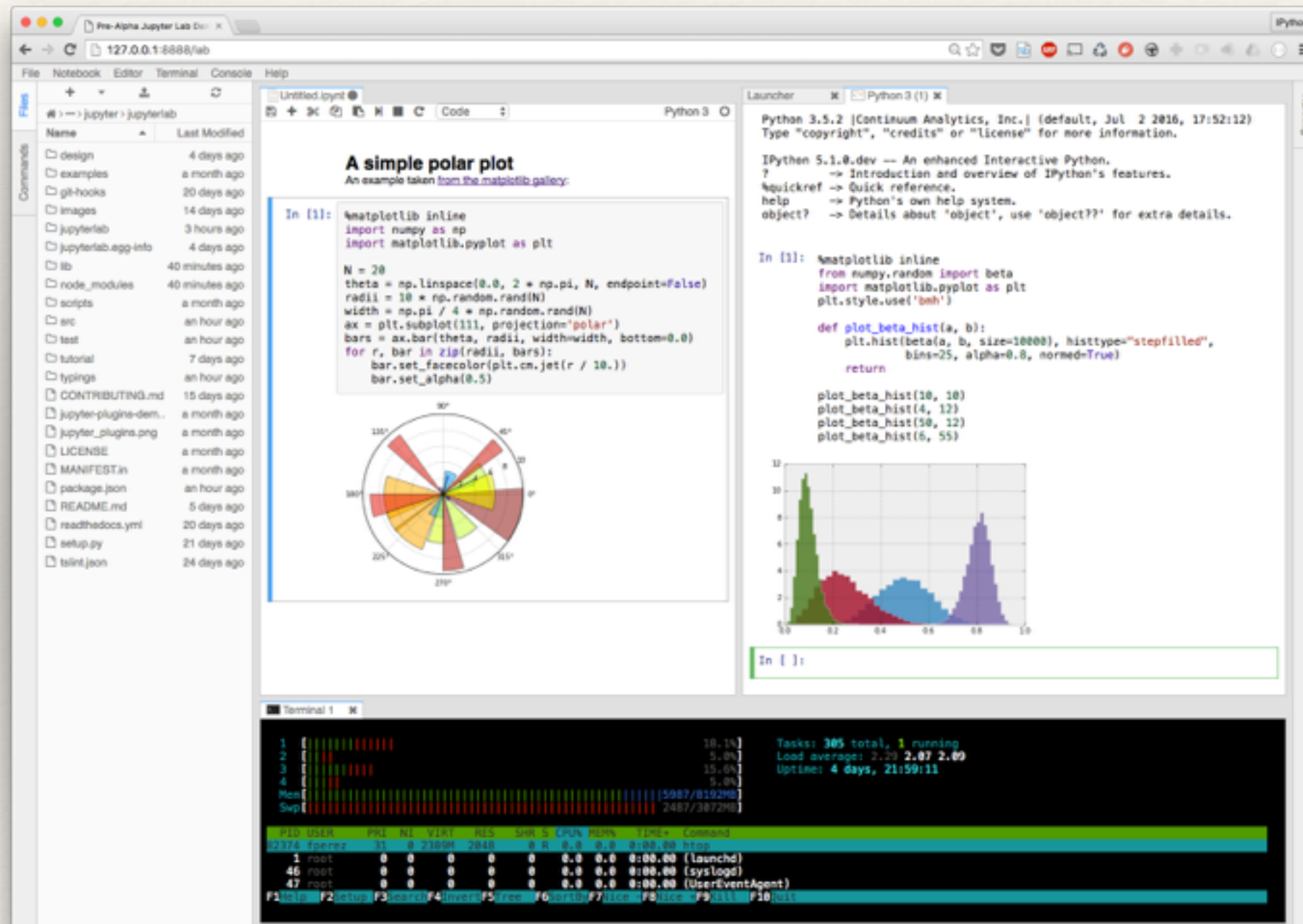
- Browser: `127.0.0.1:8888/edit/scratch/datehist.py`
- Page Title: `jupyter datehist.py`
- Content: A code editor showing Python code for `datehist.py`.

Bottom Right: JupyterLab Interface (Running Tab)

- Browser: `127.0.0.1:8888/tree/scratch#ipyclusters`
- Page Title: `jupyter`
- Content: A table showing IPython parallel computing clusters.

profile	status	# of engines	action
default	running	4	Stop
default-old	stopped		Start
foo	stopped		Start
julia	stopped		Start
mpi	stopped		Start
nbserver	stopped		Start
profile_mpi	stopped		Start
v24	stopped		Start

JupyterLab: unifying these ideas



Jason Grout
(here)

+Brian, Steven, Darian,
Sylvain, Carol, Cameron,
Farica, Paul, Reese, Kyle,
Chris, Ian, Matthias, ...

Bloomberg





Live Demo!

Demo credits / thank you:
Brian Granger (Cal Poly SLO)
Jason Grout (Bloomberg)

Thank You