



A tool for datascience at scale.

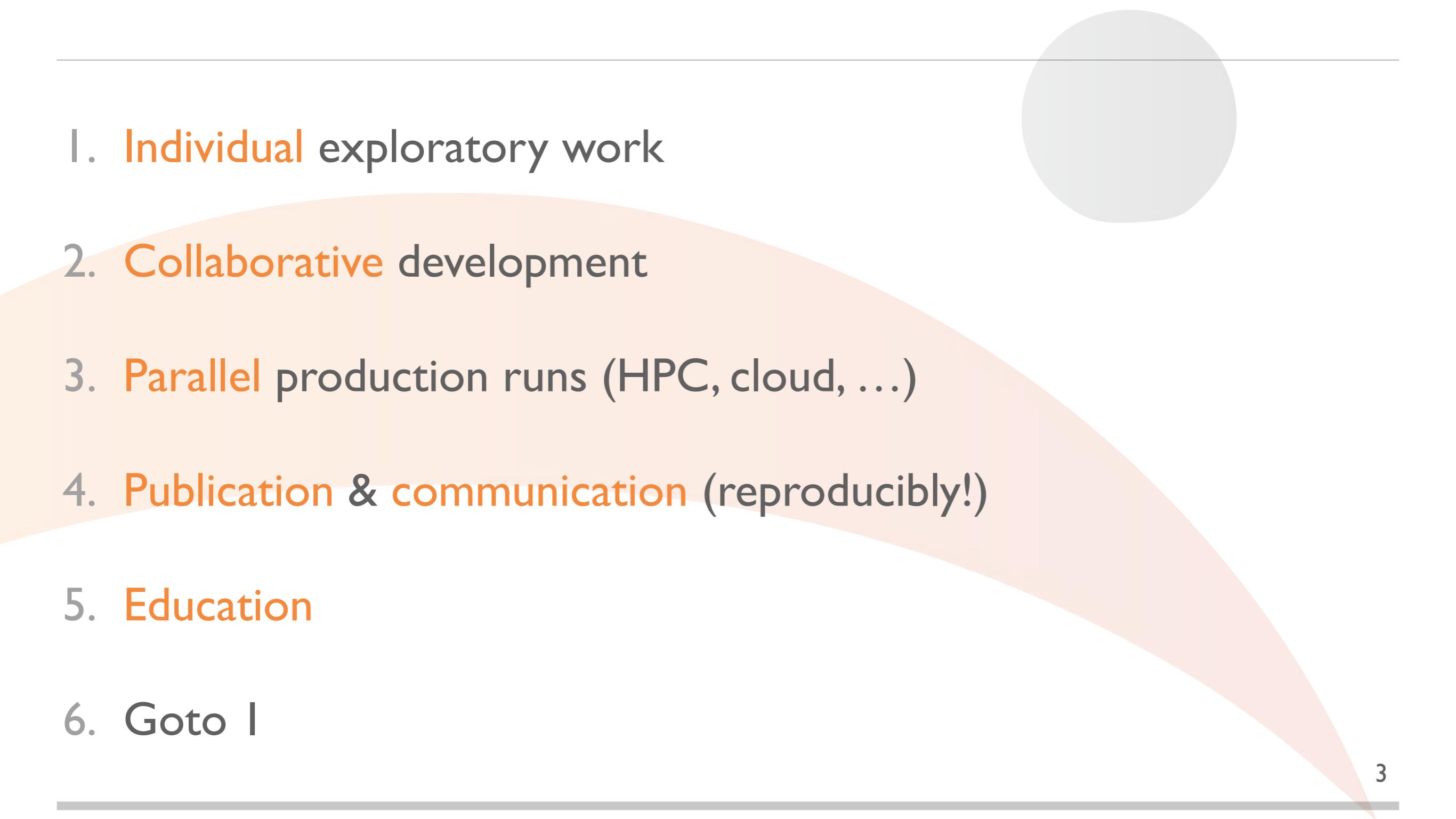
Matthias Bussonnier

(UC Berkeley mbussonnier@berkeley.edu)

Jupyter

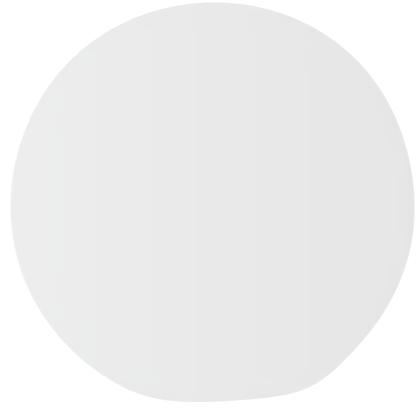
- A bit of History
- The Notebook Application
- The document format & Publication
- Multi-User & scaling
- The Ecosystem

The Lifecycle of a Scientific Idea

- 
1. Individual exploratory work
 2. Collaborative development
 3. Parallel production runs (HPC, cloud, ...)
 4. Publication & communication (reproducibly!)
 5. Education
 6. Goto 1

The Lifecycle of a Scientific Idea

- 
1. Individual exploratory work – Matlab command line
 2. Collaborative development – email scripts back and forth ?
 3. Parallel production runs (HPC, cloud, ...) – rewrite Fortran/MPI
 4. Publication & communication (reproducibly!) – Copy Paste in PPT
 5. Education – Specific tools
 6. Goto 1



Can we have a single tool that cover all the lifecycle of
a scientific idea, from data collection to publication ?

A bit of History



- Fernando Perez, 2001, CU Boulder (instead of writing a physics dissertation):
- Python can replace the collection of bash, perl, C/C++ Script. But the Python REPL can be better.

```
In [13]: run ~/scratch/error
reps: 5

ValueError                                     Traceback (most recent call last)
/home/fperez/scratch/error.py in <module>()
 70 if __name__ == '__main__':
 71     #explode()
-> 72     main()
 73     g2='another global'

/home/fperez/scratch/error.py in main()
 60     array_num = zeros(size,'d')
 61     for i in xrange(reps):
-> 62         RampNum(array_num, size, 0.0, 1.0)
 63         RNtime = time.clock()-t0
 64         print 'RampNum time:', RNtime

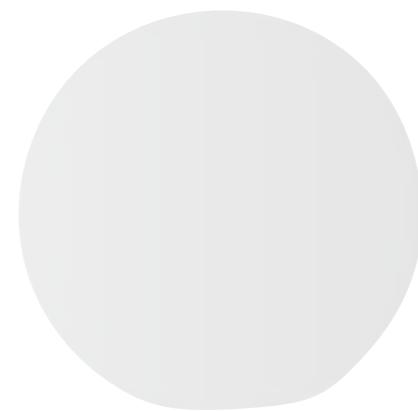
/home/fperez/scratch/error.py in RampNum(result, size, start, end)
 43     tmp = zeros(size+1)
 44     step = (end-start)/(size-1-tmp)
-> 45     result[:] = arange(size)*step + start
 46
 47 def main():

ValueError: shape mismatch: objects cannot be broadcast to a single shape

In [14]:
```

NOVEMBER 2001: "JUST AN AFTERNOON HACK"

- 259 Line Python script. (<https://gist.github.com/fperez/1579699>)
- sys.ps1 -> In [N].
- sys.displayhook -> Out[N], caches results.
- Plotting, Numeric, etc.



2014 (OPENHUB STATS)

- 19,279 commits
- 442 contributors
- Total Lines: 187,326
- Number of Languages : 7 (JS, CSS, HTML, ...)

Improve over the terminal

- ❖ The REPL as a network protocol

- ❖ Kernels

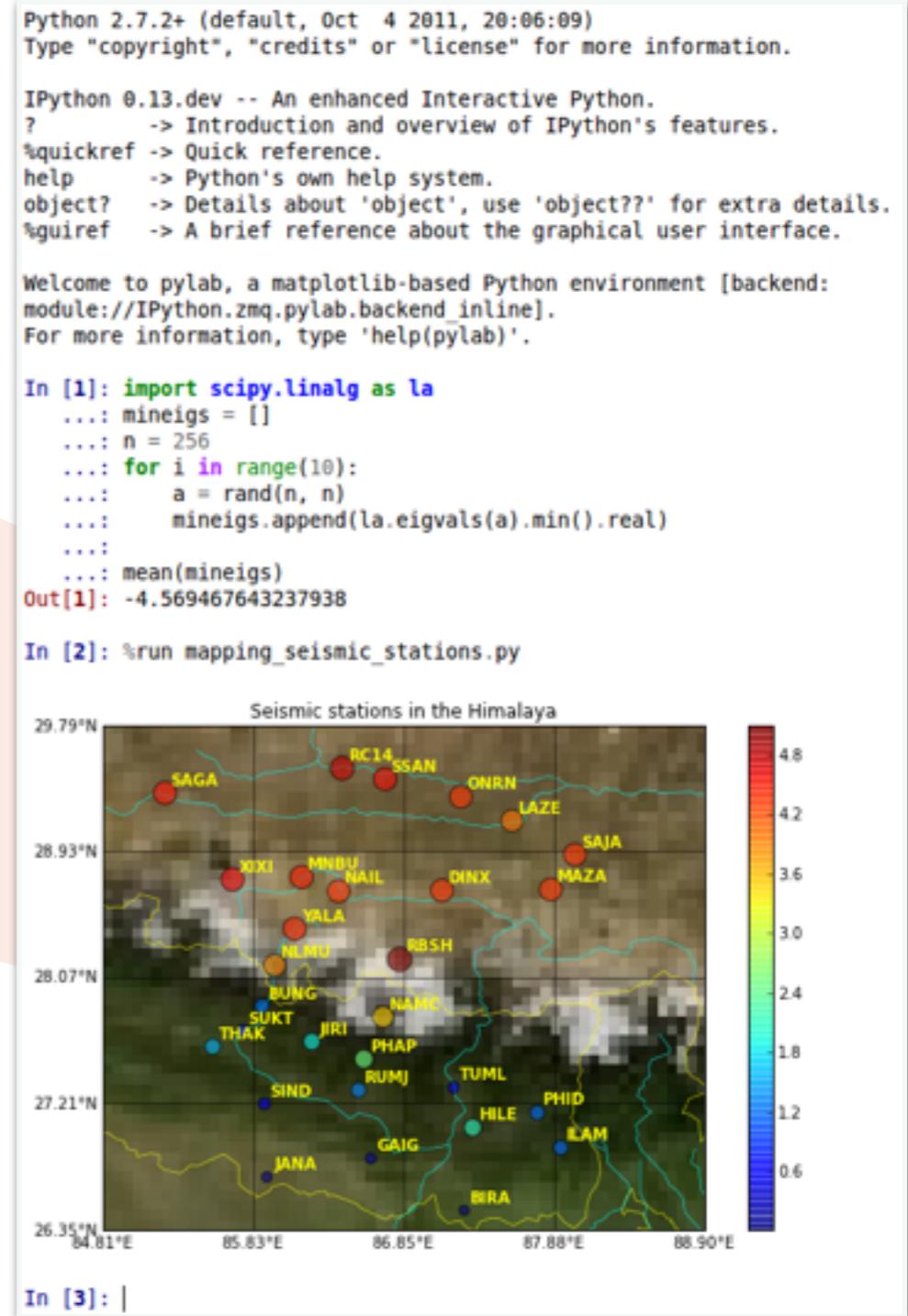
- ❖ execute code

- ❖ Clients

- ❖ Read input

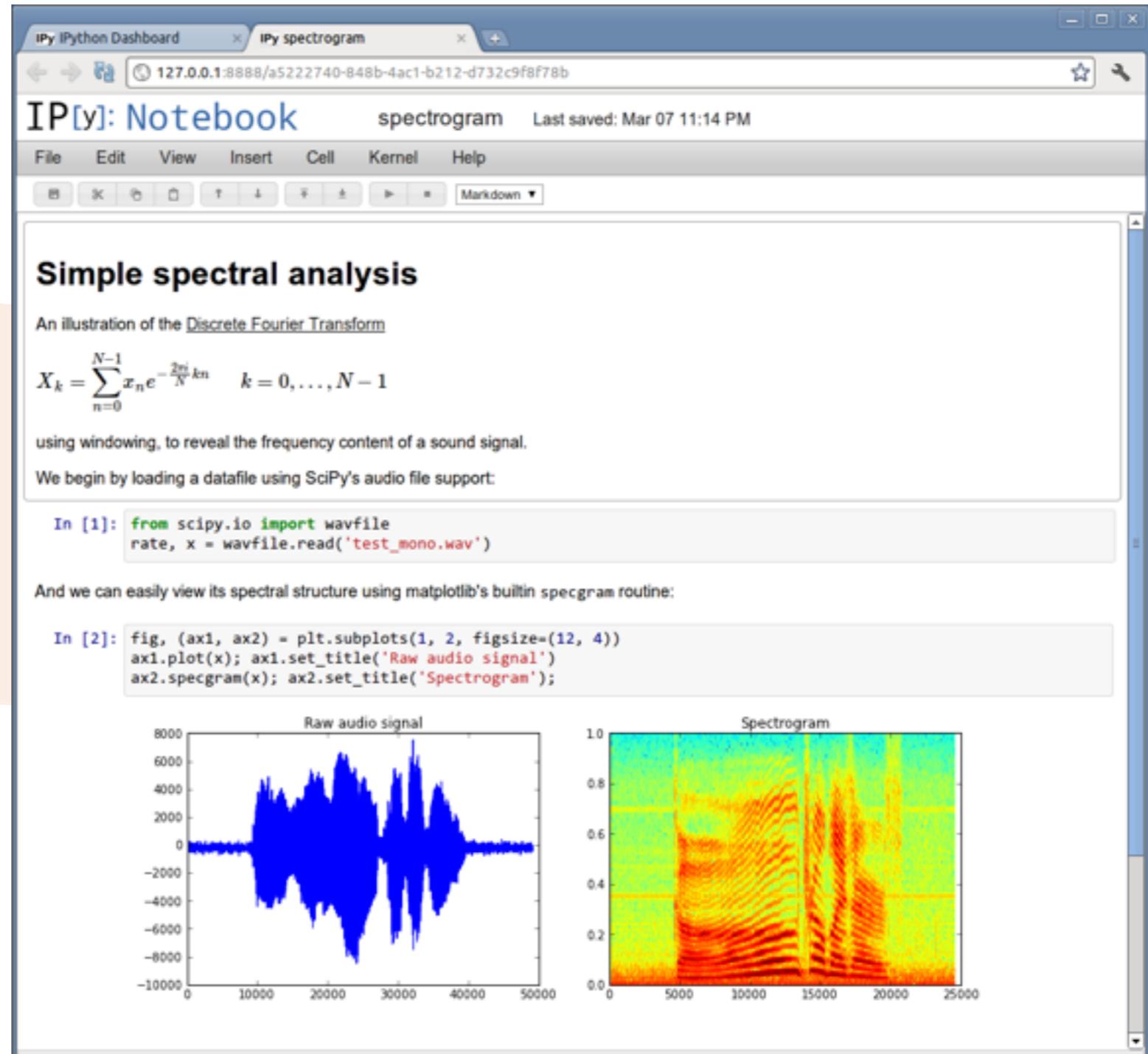
- ❖ Present output

Simple abstractions enable rich,
sophisticated clients



2011: The IPython Notebook

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



The Team

(people that spend a noticeable amount of time on the project, subjective of course)

- **Fernando Perez** (UC Berkeley LBL)
- **Brian Granger** (CalPoly)
 - Oberon Lopez (summer student)
 - Cameron Oelsen (summer student)
 - Simon Vurens (summer student)
 - Ryan Morshed (summer student)
- **Min Ragan-Kelley** (Simula)
- **Thomas Kluyver** (UK)
- **Matthias Bussonnier** (UC Berkeley)
- **Jon Frederic** (Cal Poly)
- Jess Hamrick (UC Berkeley)
- Kyle Kelley (Rackspace)
- Jason Grout (Bloomberg)
- Sylvain Corlay (Bloomberg)
- Kester Tong (Google)
- Nicholas Bollweg
- Will Whitney (MIT)
- Damián Avila (Continuum)
- Steven Silvester (Continuum)
- Chris Colbert (Continuum)
- David Willmer (Continuum)
- Peter Parente (IBM)
- Dan Gisolfi (IBM)
- Gino Bustelo (IBM)
- All 400+ GitHub contributors.

Funding



ALFRED P. SLOAN
FOUNDATION

THE LEONA M. AND HARRY B.
HELMSLEY
CHARITABLE TRUST

GORDON AND BETTY
MOORE
FOUNDATION

SIMONS FOUNDATION



CONTINUUM
ANALYTICS

POWERED BY
rackspace[®]
the open cloud company



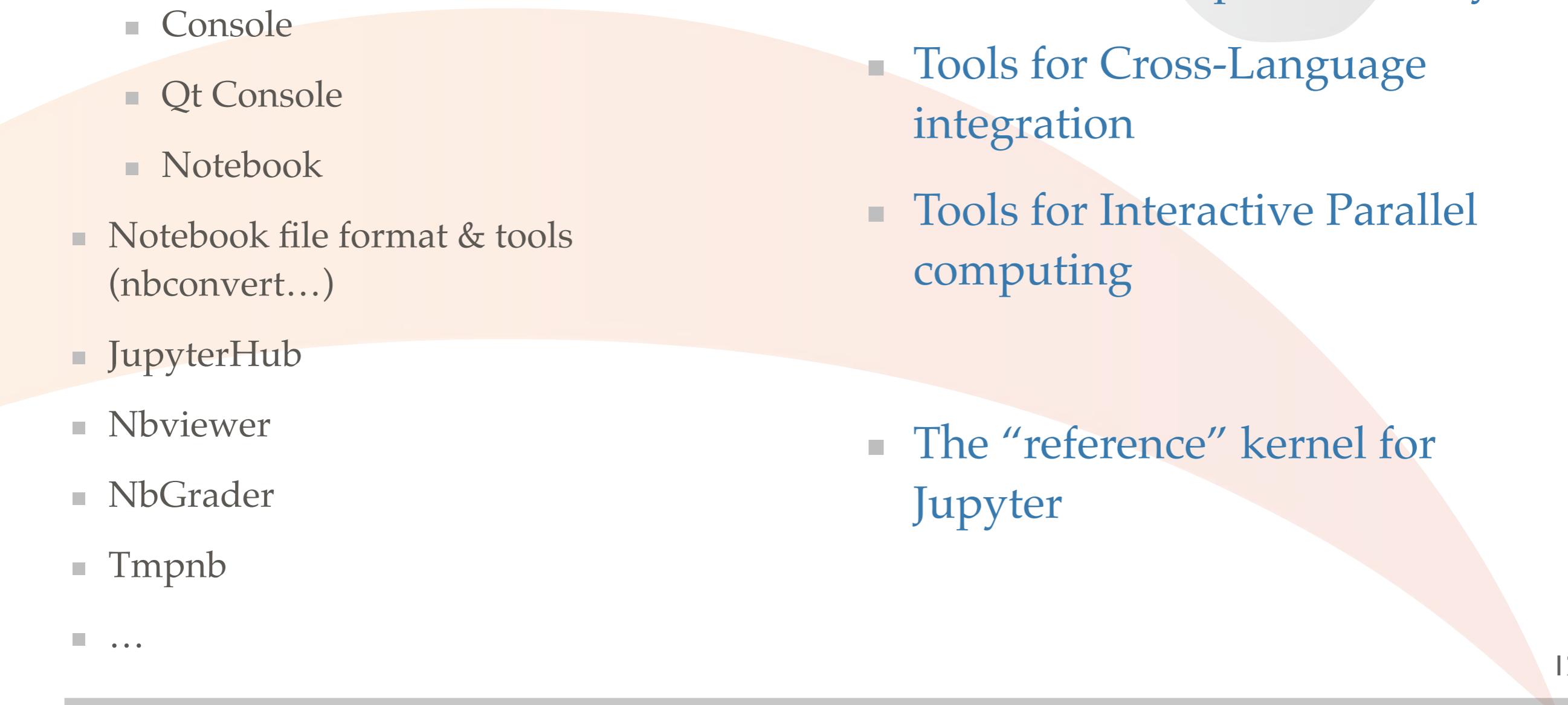
Microsoft

ENTHOUGHT
SCIENTIFIC COMPUTING SOLUTIONS

Google

Bloomberg

Jupyter vs IPython

- Network protocol for interactive computing
 - Clients for protocol
 - Console
 - Qt Console
 - Notebook
 - Notebook file format & tools (nbconvert...)
 - JupyterHub
 - Nbviewer
 - NbGrader
 - Tmpnb
 - ...
- 
- Interactive Python shell at the terminal
 - Kernel for this protocol in Python
 - Tools for Cross-Language integration
 - Tools for Interactive Parallel computing
 - The “reference” kernel for Jupyter

Why ?

- Don't reinvent the wheel: reimplement 1 piece, get the rest for free.
- You don't like the frontend, write a new one for Python get 50+ languages that work out of the box with it. (<https://github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages>)
- You don't like a language, write your own kernel, get all the IDEs, conversion tools.
- Etc..

The Notebook

Demo

Try it on <https://try.jupyter.org>

Notebook app, also have a terminal, text editor, increasing number of plugins, and of course support 50 languages.

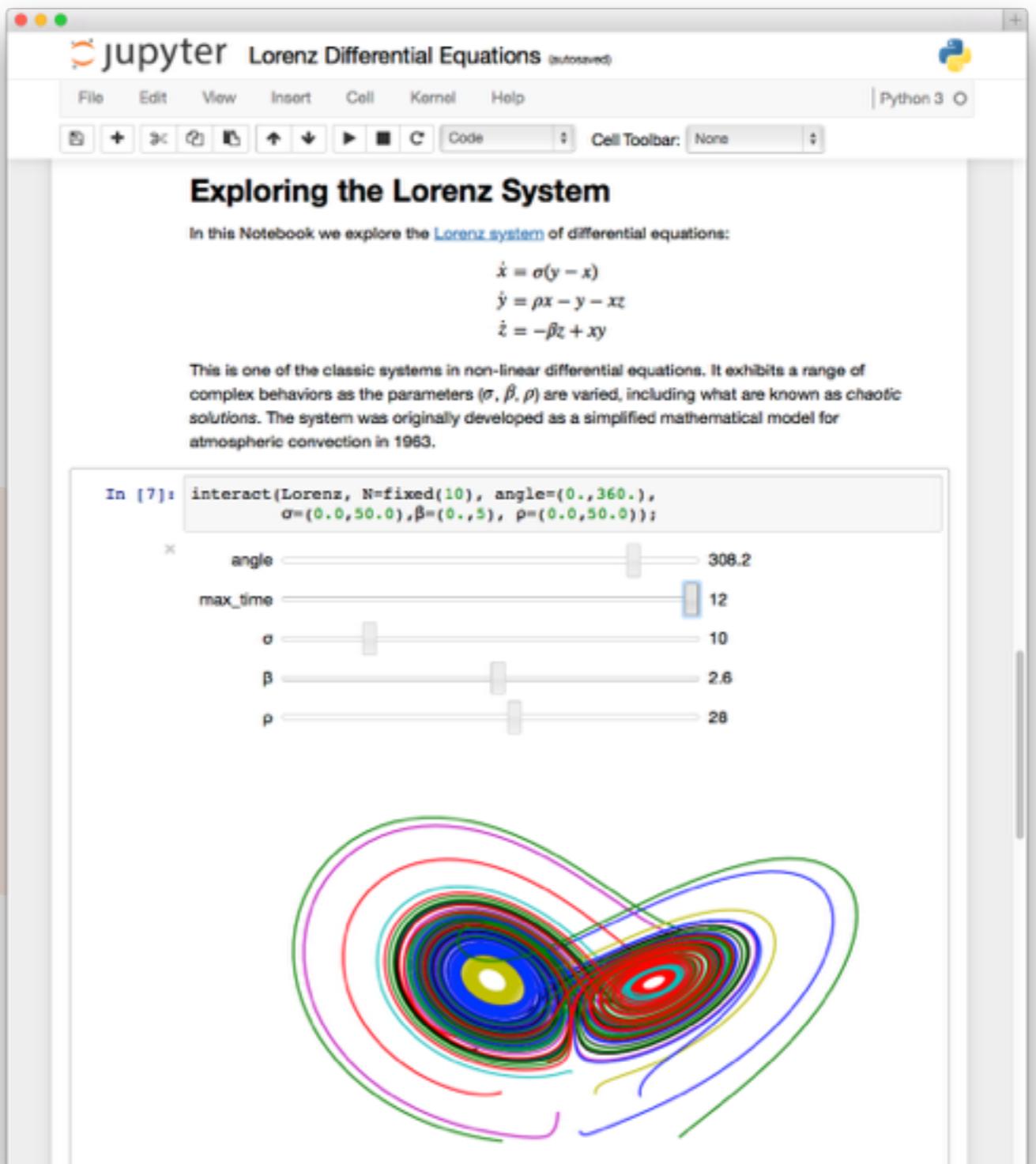
The notebook

Web Application, that allow code to produce web-rich representation (images, sound, video, math, ...)

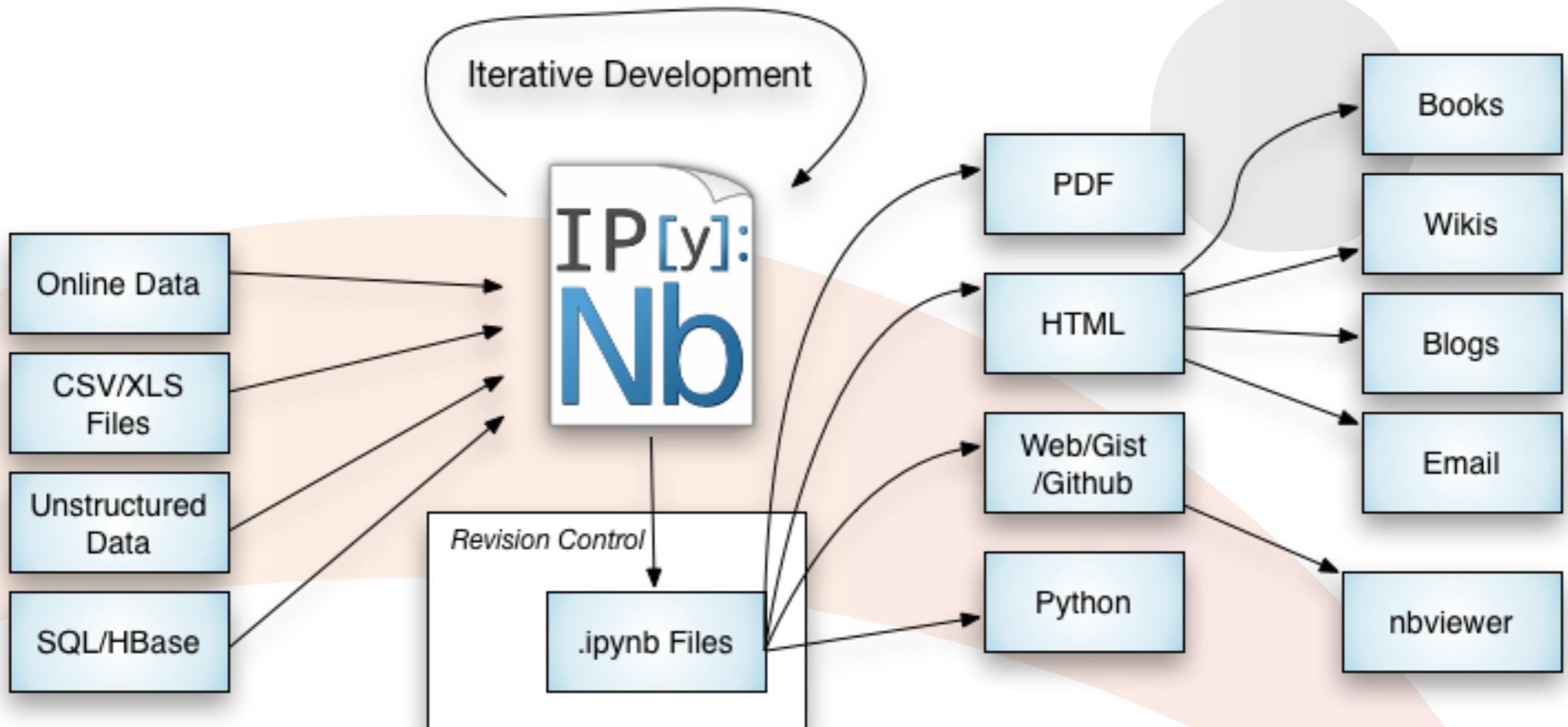
The Browser, Server, and kernel(s) can be on separate machines.

The default application to edit `ipynb` files.

`ipynb` file are JSON based files embedding input and output, so which can be read & converted without a running kernel.



The Notebook Fileformat (.ipynb)



NbViewer

- Zero-install reading of notebooks
- Just share a URL
- nbviewer.org

Under the hood: get raw URL and convert to HTML on the fly.

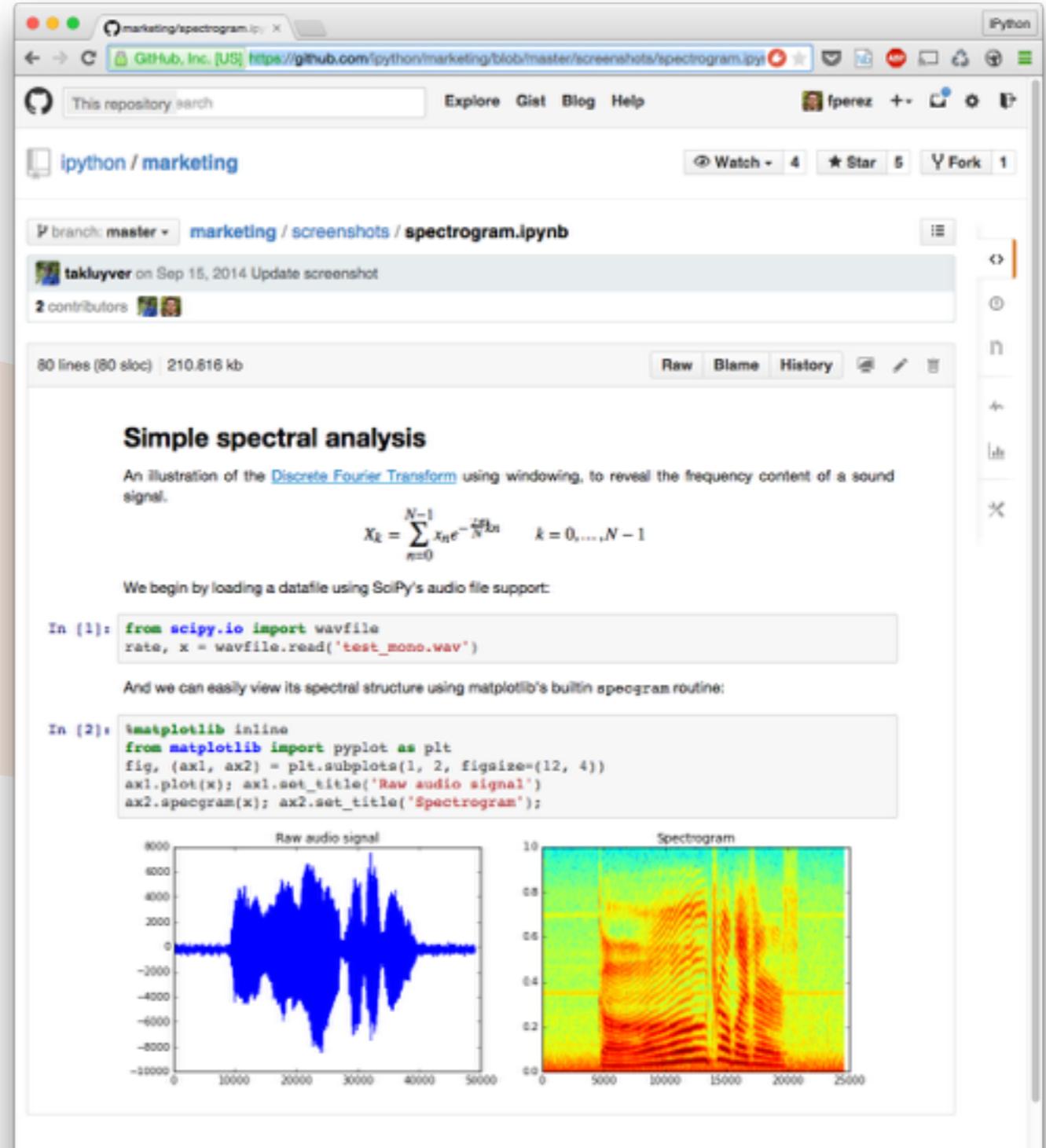
Sharing:

git push, or dropbox sync.



Nbviewer on GitHub

- Since May GitHub renders Notebooks
- Powered by `nbconvert`, the library that deals with `*.ipynb` -> *
- Over 200,000 notebooks on GitHub



What content as notebook ?

The screenshot shows a web browser displaying a journal article from 'The ISME Journal'. The URL is www.nature.com/ismej/journal/v7/n3/full/ismej2012123a.html. The page includes a header with 'Login' and 'Cart' buttons, a search bar, and a navigation menu with links like 'Journal home', 'Archive', 'Commentaries', 'Full text', 'Journal home', 'Advance online publication', 'About AOP', 'Current issue', 'Archive', 'Focuses', 'Browse by subject', 'Press releases', 'Online submission', 'For authors', 'For referees', 'Contact editorial office', 'About the journal', 'Editors and Editorial Board', 'About the society', and 'For librarians'. The main content area displays the article's title, authors, abstract, and full text. The full text is presented in a sidebar with various links such as 'Download PDF', 'Send to a friend', 'View interactive PDF in ReadCube', 'Rights and permissions', 'Order Commercial Reprints', 'CrossRef lists 1 article citing this article', 'Data availability', 'References', 'Acknowledgements', 'Figures and Tables', 'Supplementary info', 'Export citation', 'Export references', and 'Papers by Ragan-Kelley'.

<http://www.nature.com/ismej/journal/v7/n3/full/ismej2012123a.html>

The screenshot shows an IPython Notebook interface. At the top, it says 'Instructions and supporting data for the QIIME/IPython/StarCluster demo at the 2012 NIH Cloud Computing the Microbiome workshop and our corresponding paper in the ISME Journal.' Below this, a message states: 'The analysis made use of the [IPython Notebook](#), [QIIME](#), [StarCluster](#), [PyCogent](#), and [PrimerProspector](#). All of these tools are pre-installed in the ami-9f69e1f6 public Amazon EC2 instance, which was used in this study.' A section titled 'Supporting Files' lists several files: 'NIH Cloud Demo (Complete)', 'NIH Cloud Demo (Fast)', 'Timing*', 'Variable Region Position B', 'Pearson v Robinson-Foulds', and 'V3 and V4 Regions Only'. Another section, 'This notebook is intended to calculate the positions of primers in an alignment, using functions from PrimerProspector.', contains a code cell starting with '# Code modified from PrimerProspector library slice_aligned_region.py (development version)'. The code defines variables for DNA codes, imports, and then uses a loop to define primer sequences for various samples (27f, 338r, 534r, 349f, 515f, 967f, 1492r, 1391f). It also defines a reference aligned file. The code is annotated with comments explaining its purpose and how to run it.

http://qiime.org/home_static/nih-cloud-apr2012

Papers with code as AMI/VMs

Blogs

Sign In | Register

Search ScientificAmerican.com

Subscribe News & Features Topics Blogs Videos & Podcasts Education Cit

SA en español

Blogs

Like 0 Tweet 2 g+ 3 in Share 3 reddit this!

SA Visual
Illustrating science since 1845

SA Visual Home · About · Contact

Visualizing 4-Dimensional Asteroids

By Jake VanderPlas | September 16, 2014

Multicolor plot

Let's put these all together. Rather than using two separate color scales to identify these asteroid groups, we can define a single two-dimensional color scale reflecting the asteroid chemistry and use these colors when plotting the same points in orbital space. The result is a plot very similar to the one that appeared in Parker et al., 2008, where this work was first reported:

The figure consists of two side-by-side scatter plots. The left plot has 'Near-IR Color (I - Z)' on the y-axis (ranging from -0.8 to 0.6) and 'Optical Color (a*)' on the x-axis (ranging from -0.3 to 0.4). The right plot has 'Sine of Orbital Inclination' on the y-axis (ranging from 0.00 to 0.30) and 'Semi-major Axis (AU)' on the x-axis (ranging from 2.0 to 3.2). Both plots use a color scale from blue to red. A dashed white rectangle highlights a central region in both plots. The right plot is further divided into 'Inner', 'Mid', and 'Outer' regions by vertical dashed lines at approximately 2.2 AU, 2.4 AU, and 2.8 AU.

nbviewer.ipython.org/github/jakevdp/SciAmBlogPost/blob/master/AsteroidVis.ipynb

Multicolor plot

Let's put these all together. Rather than using two separate color scales to identify these asteroid groups, we can define a single two-dimensional color scale reflecting the asteroid chemistry and use these colors when plotting the same points in orbital space. The result is a plot very similar to the one that appeared in Parker et al., 2008, where this work was first reported:

In [13]: `plot_multicolor()`

The figure is a scatter plot with 'Near-IR Color (I - Z)' on the y-axis (ranging from -0.8 to 0.6) and 'Optical Color (a*)' on the x-axis (ranging from -0.3 to 0.4). The plot is divided into three vertical regions labeled 'Inner', 'Mid', and 'Outer' by dashed white lines at approximately 2.2 AU, 2.4 AU, and 2.8 AU. The data points are colored according to their optical and near-infrared colors. A dashed white rectangle highlights a central region in the plot.

Jake van der Plas @ UW

[http://blogs.scientificamerican.com/
sa-visual/2014/09/16/visualizing-4-
dimensional-asteroids](http://blogs.scientificamerican.com/sa-visual/2014/09/16/visualizing-4-dimensional-asteroids)

Course, MOOCS

	Course	University	Instructor
0	Data Science and Visualization with Python	Santa Clara	Brian Granger
1	Python for Data Science	UC Berkeley	Josh Bloom
2	Introduction to Data Science	UC Berkeley	Michael Franklin
3	Working with Open Data	UC Berkeley	Raymond Yee
4	Introduction to Signal Processing	UC Berkeley	Miki Lustig
5	Data Science (CS 109)	Harvard University	Pfister and Blitzstein
6	Practical Data Science	NYU	Josh Attenberg
7	Scientific Computing (ASTR 599)	University of Washington	Jake Vanderplas
8	Computational Physics	Cal Poly	Jennifer Klay
9	Introduction to Programming	Alaskan High School	Eric Matthes
10	Aerodynamics-Hydrodynamics (MAE 6226)	George Washington University	Lorena Barba

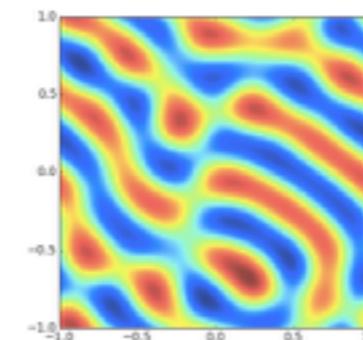
11	HyperPython: hyperbolic conservation laws	KAUST	David Ketcheson
12	Quantitative Economics	NYU	Sargent and Stachurski
13	Practical Numerical Methods with Python	4 separate universities + MOOC	Barba, et al.
14	Data Science: Algorithms	Columbia - Lede Program	Chris Wiggins
15	Data Science: Databases	Columbia - Lede Program	Chris Wiggins
16	Data Science: Foundations	Columbia - Lede Program	Chris Wiggins
17	Data Science: Platforms	Columbia - Lede Program	Chris Wiggins

Lorena A. Barba group



Announcing "Practical Numerical Methods with Python" MOOC

Posted on 07.26.2014



Pattern formation:
solution for a reaction-diffusion system like:

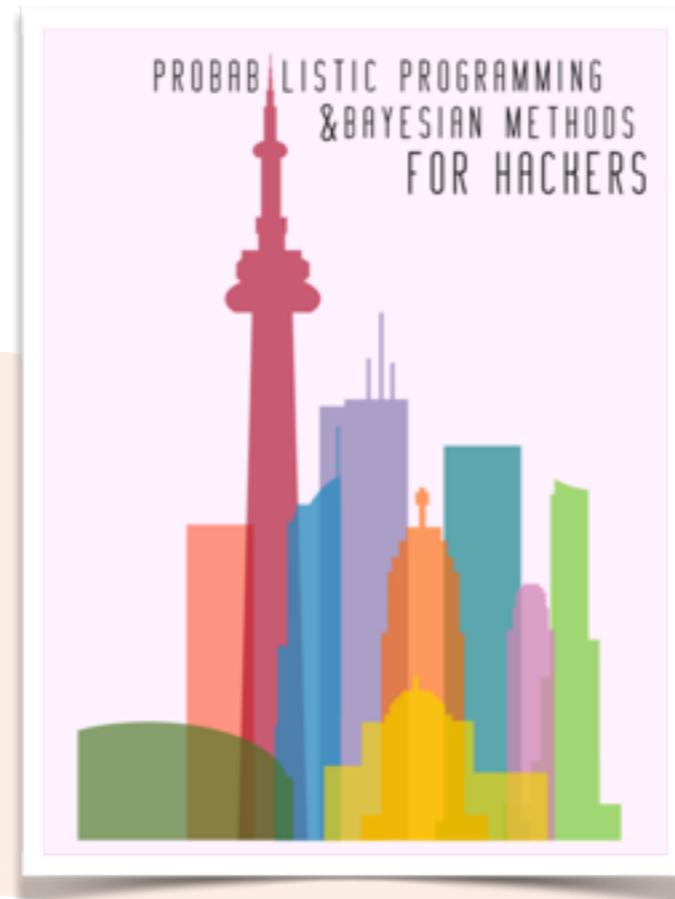
$$\begin{aligned} u_t &= \delta D_1 \nabla^2 u + f(u, v) \\ v_t &= \delta D_2 \nabla^2 v + g(u, v) \end{aligned}$$

An example of the types of problems we will learn to solve in this course, among others governed by differential equations.

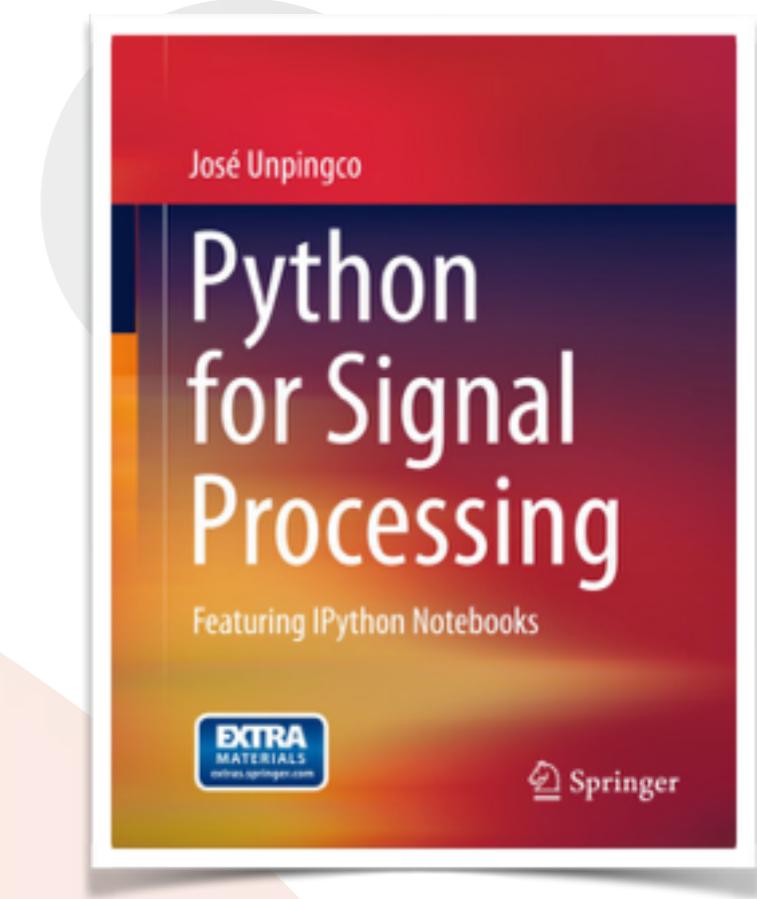
Books



By [Matthew Russell](#)



By [Cameron Davidson-Pilon](#)



By [José Unpingco](#)

You can download and execute the books locally.

Check The Gallery

A gallery of interesting IPython Notebooks

Fernando Perez edited this page 8 days ago · 229 revisions

This page is a curated collection of IPython notebooks that are notable for some reason. Feel free to add new content here, but please try to only include links to notebooks that include interesting visual or technical content; this should not simply be a dump of a Google search on every ipynb file out there.

Important contribution instructions: If you add new content, please ensure that for any notebook you link to, the link is to the rendered version using nbviewer, rather than the raw file. Simply paste the notebook URL in the nbviewer box and copy the resulting URL of the rendered version. This will make it much easier for visitors to be able to immediately access the new content.

Note that Matt Davis has conveniently written a set of bookmarklets and extensions to make it a one-click affair to load a Notebook URL into your browser of choice, directly opening into nbviewer.

Table of Contents

1. Entire books or other large collections of notebooks on a topic
 - Introductory Tutorials
 - Programming and Computer Science
 - Statistics, Machine Learning and Data Science
 - Mathematics, Physics, Chemistry, Biology
 - Earth Science and Geo-Spatial data
 - Linguistics and Text Mining
 - Signal Processing
2. Scientific computing and data analysis with the SciPy Stack
 - General topics in scientific computing
 - Social data
 - Psychology and Neuroscience
 - Machine Learning
 - Physics, Chemistry and Biology
 - Economics
 - Earth science and geo-spatial data

Reproducible academic publications

This section contains academic papers that have been published in the peer-reviewed literature or pre-print sites such as the ArXiv that include one or more notebooks that enable (even if only partially) readers to reproduce the results of the publication. If you include a publication here, please link to the journal article as well as providing the nbviewer notebook link (and any other relevant resources associated with the paper).

1. Reply to 'Influence of cosmic ray variability on the monsoon rainfall and temperature': a false-positive in the field of solar-terrestrial research by Benjamin Laken, 2015. Reviewed article will appear in JASTP. The IPython notebook reproduces the full analysis and figures exactly as they appear in the article, and is available on Github: link via figshare.
2. The probability of improvement in Fisher's geometric model: a probabilistic approach, by Yoav Ram and Lilach Hadany. (Theoretical Population Biology, 2014). An IPython notebook, allowing figure reproduction, was deposited as a supplementary file.
3. Stress-induced mutagenesis and complex adaptation, by Yoav Ram and Lilach Hadany (Proceedings B, 2014). An IPython notebook, allowing figures reproduction, was deposited as a supplementary file.
4. Automatic segmentation of odor maps in the mouse olfactory bulb using regularized non-negative matrix factorization, by J. Seelert et al. (NeuroImage 2014, Open Access). The notebook allows to reproduce most figures from the paper and provides a deeper look at the data. The full code repository is also available.
5. Multi-tiered genomic analysis of head and neck cancer ties TP53 mutation to 3p loss, by A. Gross et al. (Nature Genetics 2014). The full collection of notebooks to replicate the results.
6. powerlaw: a Python package for analysis of heavy-tailed distributions, by J. Alstott et al.. Notebook of examples in manuscript, ArXiv link and project repository.
7. Collaborative cloud-enabled tools allow rapid, reproducible biological insights, by B. Ragan-Kelley et al.. The main notebook, the full collection of related notebooks and the companion site with the Amazon AMI information for reproducing the full paper.
8. A Reference-Free Algorithm for Computational Normalization of Shotgun Sequencing Data, by C.T. Brown et al.. Full notebook, ArXiv link and project repository.
9. The kinematics of the Local Group in a cosmological context by J.E. Forero-Romero et al.. The Full notebook and also all the data in a github repo.

<https://github.com/ipython/ipython/wiki/A-gallery-of-interesting-IPython-Notebooks>

Replicating, simpler for readers

The screenshot shows a news article from the journal 'nature' titled 'Interactive notebooks: Sharing the code'. The article discusses the use of IPython notebooks for data analysis. It includes a photograph of two Brontosaurus fossils, a sidebar with a 'Top story' about the taxonomic status of the Brontosaurus, and a list of recent articles at the bottom.

nature International weekly journal of science

Home | News & Comment | Research | Careers & Jobs | Current Issue | Archive | Audio & Video | For Authors

Archive > Volume 515 > Issue 7525 > Toolbox > Article

NATURE | TOOLBOX

Interactive notebooks: Sharing the code

The free IPython notebook makes data analysis easier to record, understand and reproduce.

Helen Shen

05 November 2014

PDF Rights & Permissions

Illustrations by The Project Twins

Top story

Beloved *Brontosaurus* makes a comeback

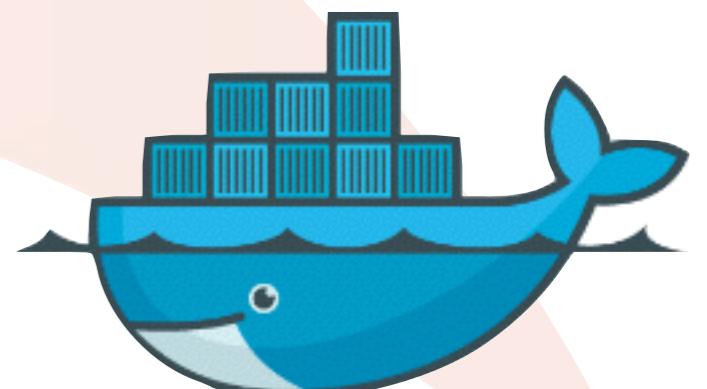
Jurassic giant's taxonomic status is restored.

Recent Read Comments Emailed

1. History: Women at the edge of science
Nature | 08 April 2015
2. Scientific instrumentation: The aided eye
Nature | 08 April 2015
3. Books in brief
Nature | 08 April 2015
4. Antibody shows promise as

What if you didn't
had to install anything ?

Docker Container
Just for you.



<http://www.nature.com/news/interactive-notebooks-sharing-the-code-1.16261>

Replicating, simpler for authors

I. Fixed set of notebooks/dependencies

1. **Tmpnb**, (<https://lambdaops.com/ipythonjupyter-tmpnb-debuts/>)

2. **CodeNeuro**, (<http://codeneuro.org/>)

2. Build on demand

1. **Everware**, (<https://github.com/everware>)

2. **Binder**, (MyBinder.org)

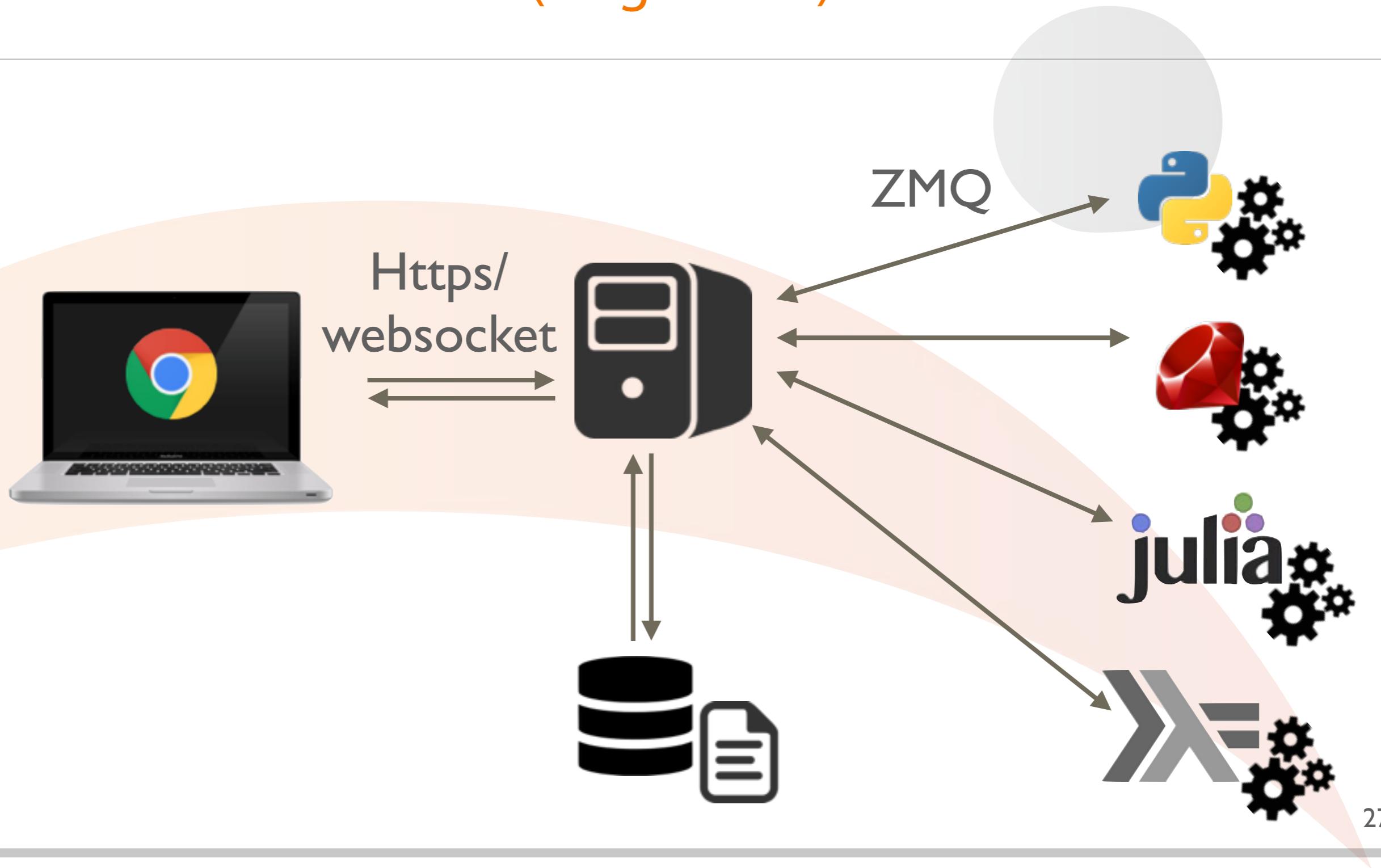


<https://github.com/binder-project>

by Jeremy Freeman

(Demo mybinder.org)

The networking architecture (single user)



MULTI-USER

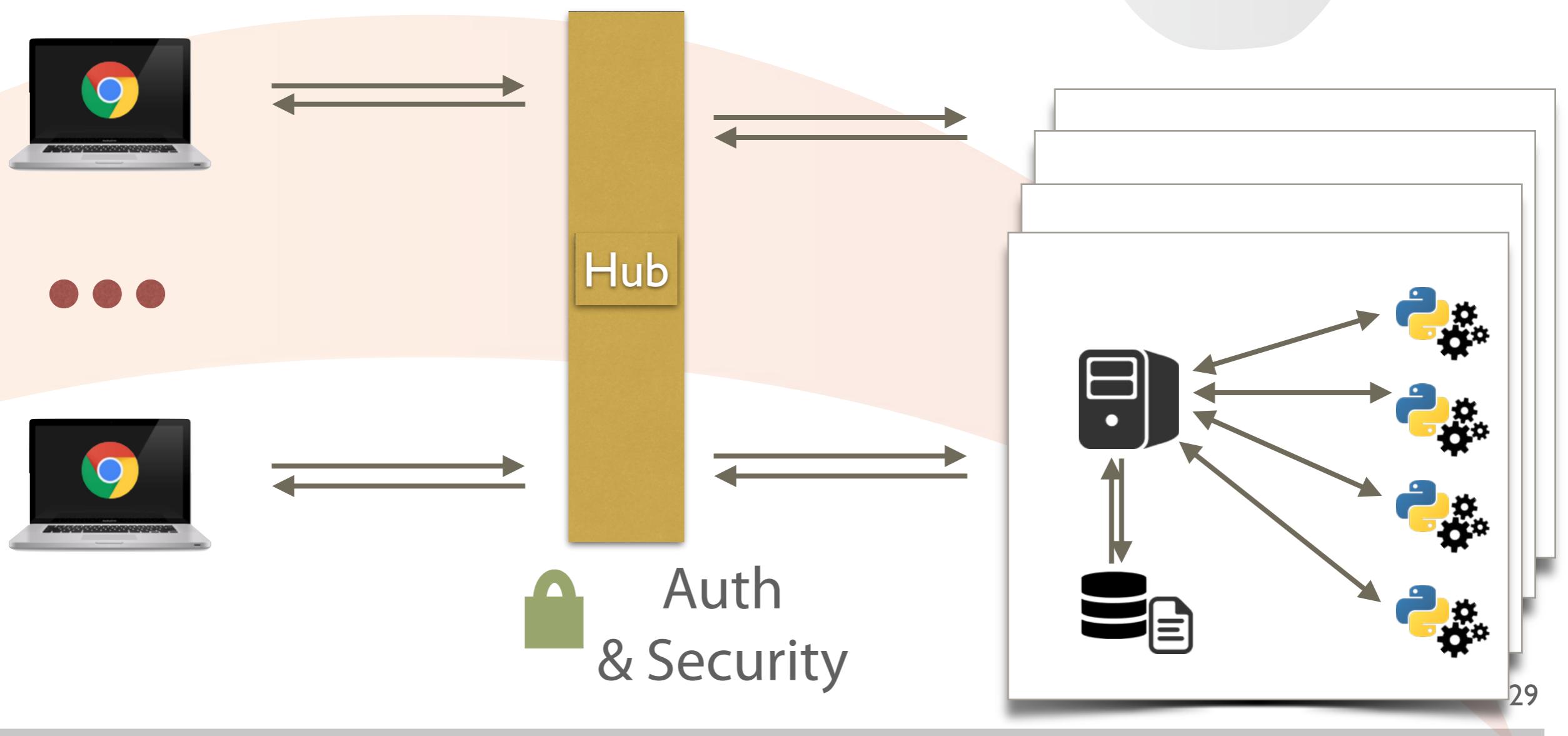
Jupyter Notebook is Single-User by design.

Multi-User enable through JupyterHub

- Allow Better scalability
- resources monitoring/user
- Per-user configuration/version of IPython
- Better integration with existing infrastructure



Https/websocket proxy





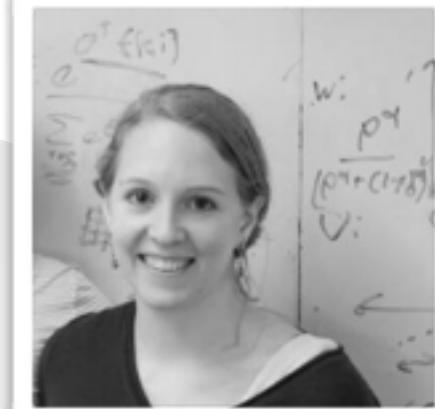
Everything* is a plugin

- Auth:
 - Unix PAM (default), OAuth, LDAP... (ie, not yet another thing to manage)
- Spawner – Start a single user server for each user.
 - Localhost, Rackspace, EC2, Docker
- Meant for sysadmin,
 - Deployments relatively involved
 - Recent software stack (Node.js/Python3)

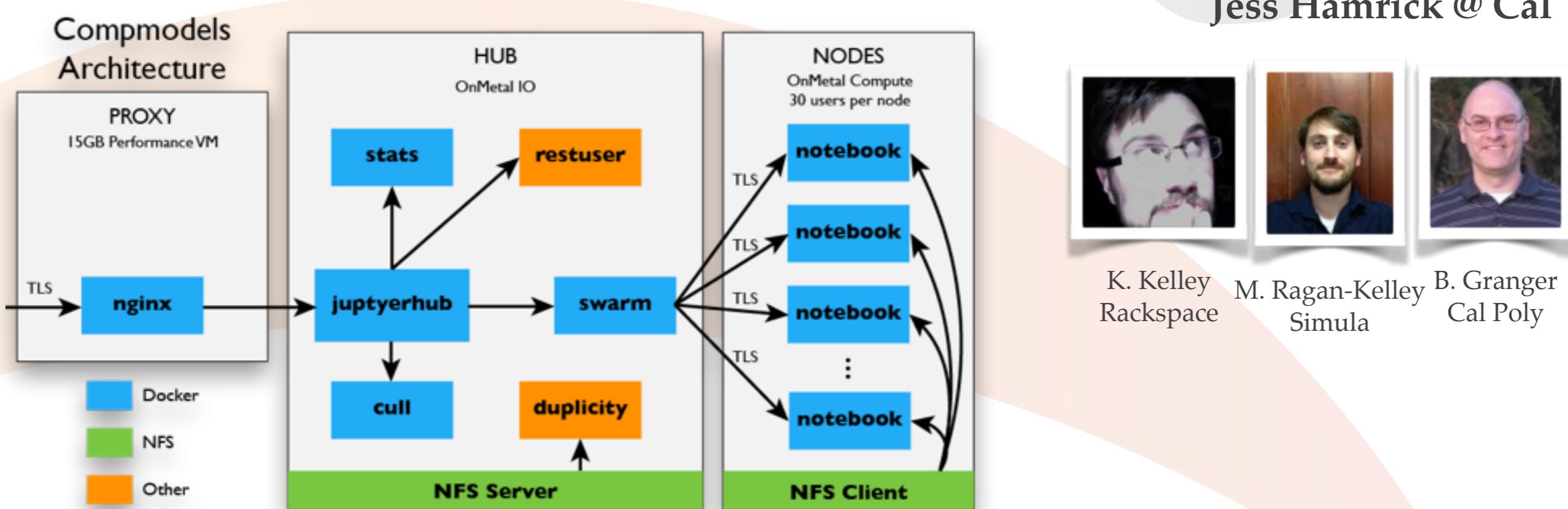
Quick Demo

JupyterHub in education

- ❖ Computationally intensive course, ~220 students
- ❖ Fully hosted environment, zero-install
- ❖ Integration with autograding.



Jess Hamrick @ Cal



<https://developer.rackspace.com/blog/deploying-jupyterhub-for-education>

Deploy at larger scale this fall at UC Berkeley

- Data Science 101
- Everyone with CalNet account.

New Jupyter in Education Mailing List:

<https://groups.google.com/forum/#!forum/jupyter-education>

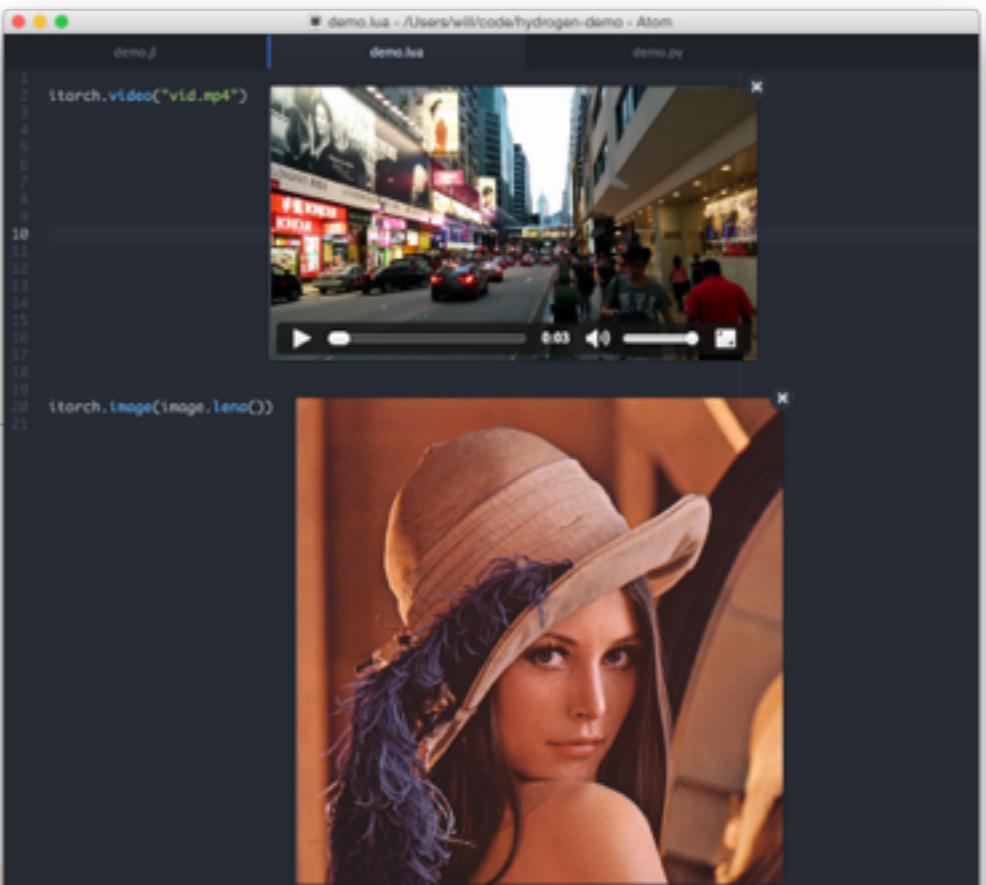


Ecosystem

Non-Notebook projects

IDE/Frontends:

Atom Hydrogen



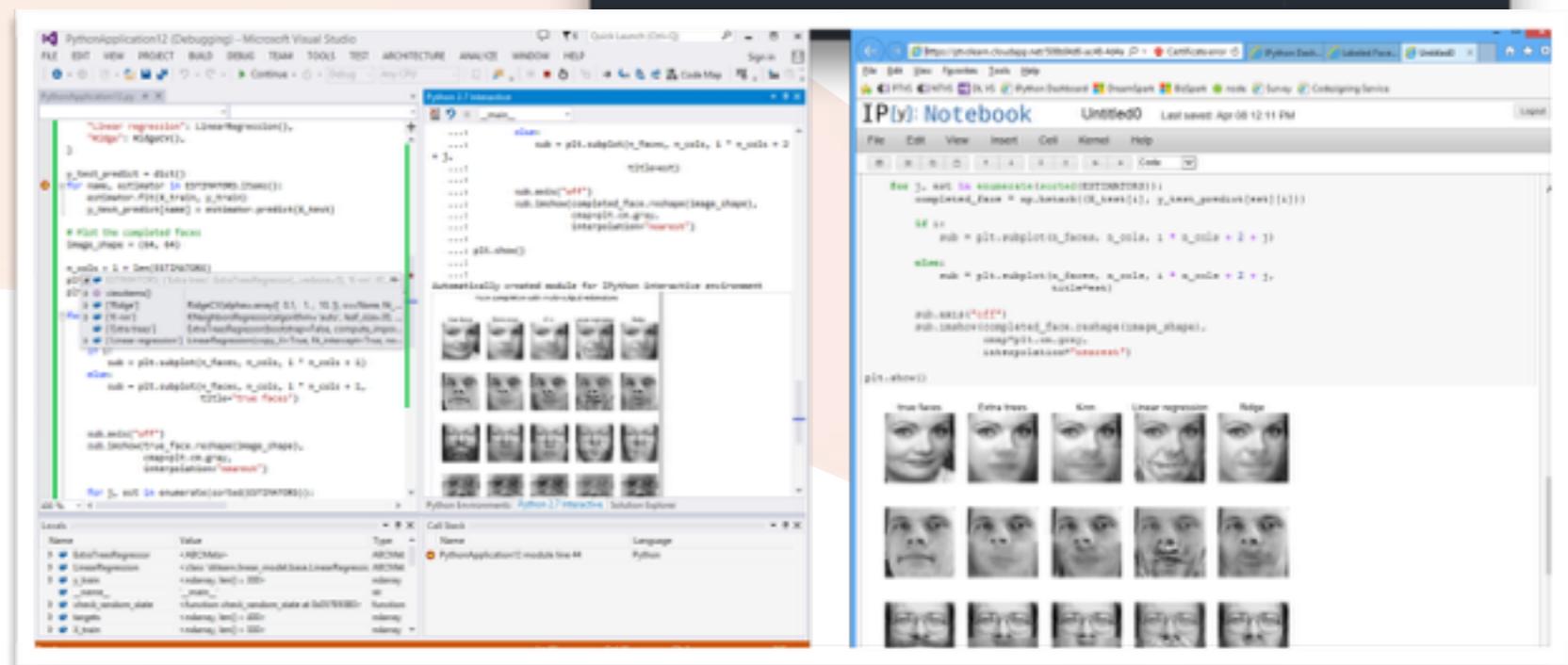
EIN

VIM IPython

Rodeo

PyCharm

Microsoft Visual Studio



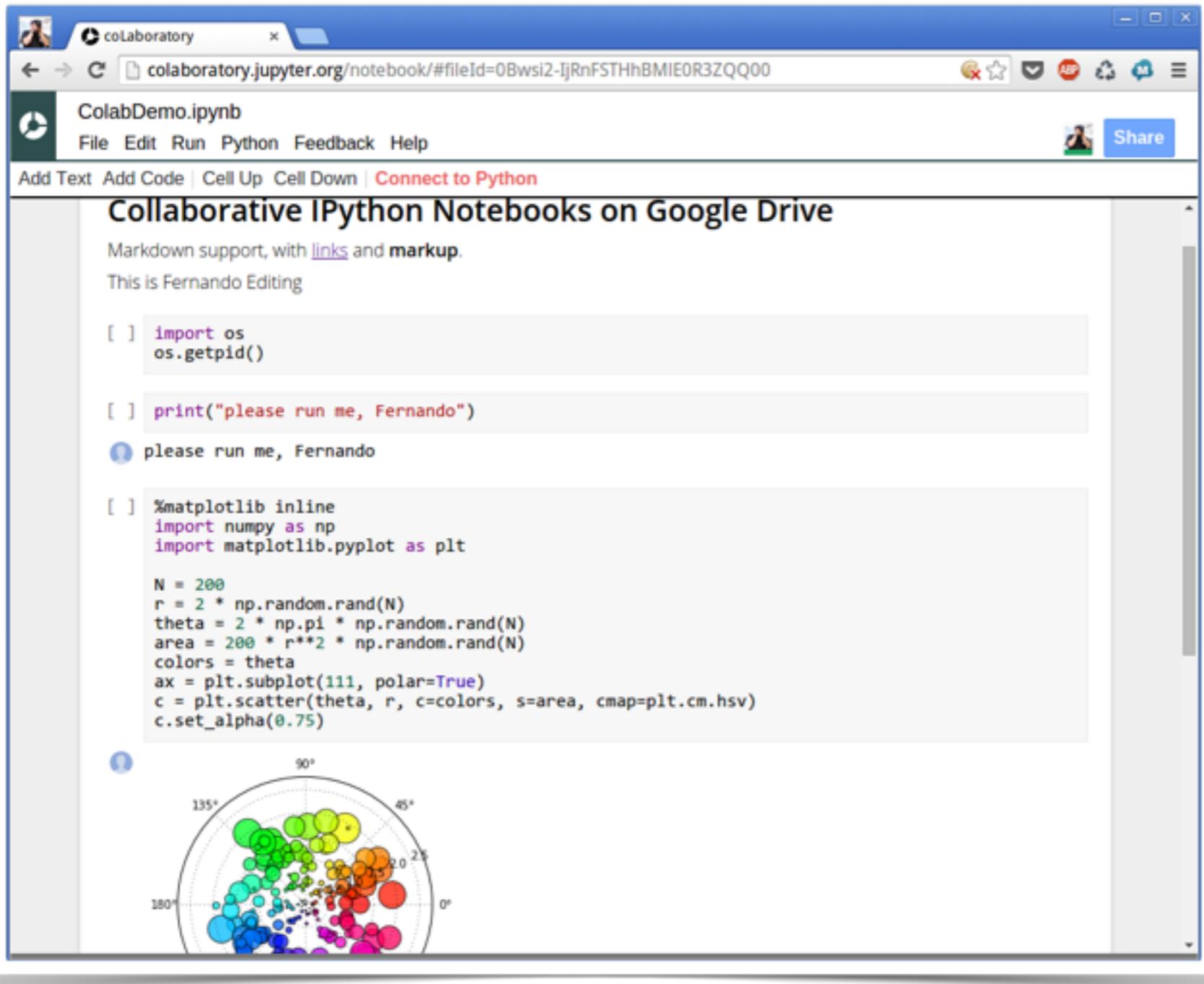
Non-Notebook projects

- RISE (interactive slideshow)
- runipy (notebooks are report templates)
- ipymd (store notebook as markdown)
- NbGrader (grade assignments)
- Jupyter-Drive (store notebooks on G-drive)
- pgcontents (store notebooks on PostGres)
- urth (declarative widget, + dashboard from notebook)



Google CoLaboratory

Kayur Patel, Kester Tong, Mark Sanders, Corinna Cortes @ Google
Matt Turk @ NCSA / UIUC



The screenshot shows a browser window for 'coLaboratory' at 'colaboratory.jupyter.org'. The title bar says 'ColabDemo.ipynb'. The page header reads 'Collaborative IPython Notebooks on Google Drive'. It mentions 'Markdown support, with [links](#) and [markup](#)'. Below this, it says 'This is Fernando Editing'. The code cell contains:

```
[ ] import os  
os.getpid()  
  
[ ] print("please run me, Fernando")  
please run me, Fernando  
  
[ ] %matplotlib inline  
import numpy as np  
import matplotlib.pyplot as plt  
  
N = 200  
r = 2 * np.random.rand(N)  
theta = 2 * np.pi * np.random.rand(N)  
area = 200 * r**2 * np.random.rand(N)  
colors = theta  
ax = plt.subplot(111, polar=True)  
c = plt.scatter(theta, r, c=colors, s=area, cmap=plt.cm.hsv)  
c.set_alpha(0.75)
```

Below the code, there is a polar scatter plot with radial axes labeled from 0° to 90° and angular axes labeled 0°, 45°, 90°, 135°, and 180°. The plot shows a distribution of colored points.

Currently being merge
into Jupyter itself.

O'Reilly: authoring and delivering executable books

Atlas, ipymd and Thebe

A screenshot of a web browser displaying an article titled "Data visualization with Seaborn". The article is from the "DATA TOOLS" section of the O'ReILLY website. It features a red header with the O'REILLY logo and navigation links for Ideas, Learning, Events, and Shop. The main content area has a white background with a large heading "Data visualization with Seaborn" in red. Below the heading is a snippet of text: "Seaborn provides an API on top of matplotlib, which uses sane plot and color defaults and simple functions for common statistical plot types." At the bottom of the snippet is the author's name, "By Jake VanderPlas, May 7, 2015".

A screenshot of a web-based code editor window titled "Data visualization with Seaborn". The URL in the address bar is <https://beta.oreilly.com/learning/data-visualization-with-seaborn>. The editor displays a Python code snippet:

```
with sns.axes_style('white'):
    sns.jointplot("x", "y", data, kind='hex')
```

 A "run" button is visible at the bottom right of the editor. The status bar at the bottom indicates the code is "idle".

A screenshot of a web-based code editor window titled "Data visualization with Seaborn". The URL in the address bar is <https://beta.oreilly.com/learning/data-visualization-with-seaborn>. The editor displays a Python code snippet:

```
with sns.axes_style('white'):
    sns.jointplot("x", "y", data, kind='hex')
```

 Below the code is a scatter plot with hexagonal bins. The plot shows a positive correlation between the variables "x" and "y". A text annotation above the plot states: "There are other parameters which can be passed to `jointplot`: for example, we can use a hexagonally-based histogram instead:". The status bar at the bottom indicates the code is "idle".

beta.oreilly.com

The Future

Project Jupyter gets \$6M to expand collaborative data science software

Fernando

cs.lbl.gov/news-media/news/2015/enhancing-open-source-software-for-scientific-computing-and-data-sci...

BERKELEY LAB COMPUTING SCIENCES
LAWRENCE BERKELEY NATIONAL LABORATORY

U.S. DEPARTMENT OF ENERGY

A-Z INDEX | PHONE BOOK | CAREERS | SHARE | FOLLOW

Home About News & Media Seminars Careers Awards Safety search...

Home » News & Media » News » Project Jupyter gets \$6M to expand collaborative data science software

NEWS & MEDIA

News CS In the News InTheLoop

Facebook Google Twitter

Project Jupyter gets \$6M to expand collaborative data science software

JULY 7, 2015

Tags: CRD

PALO ALTO, Calif. July 7, 2015 — Three foundations pledged \$6M over the next three years to Project Jupyter, an open-source software project that supports scientific computing and data science across a wide range of programming languages via a large, public, open and inclusive community.

Fernando Perez of University of California, Berkeley and [Lawrence Berkeley National Laboratory \(Berkeley Lab's\) Computational Research Division](#) and Brian Granger of California Polytechnic University, San Luis Obispo will lead the project at their institutions. Perez and Granger's efforts with Project Jupyter are the result of their work developing IPython, a popular user interface for interactive computing across multiple programming languages.

With this award from the Leona M. and Harry B. Helmsley Charitable Trust, Alfred P. Sloan Foundation, and Gordon and Betty Moore Foundation, these researchers will expand and improve the capabilities of the Jupyter Notebook, a web-based platform that allows scientists, researchers and educators to combine live code, equations, narrative text and rich media into a single, interactive document.



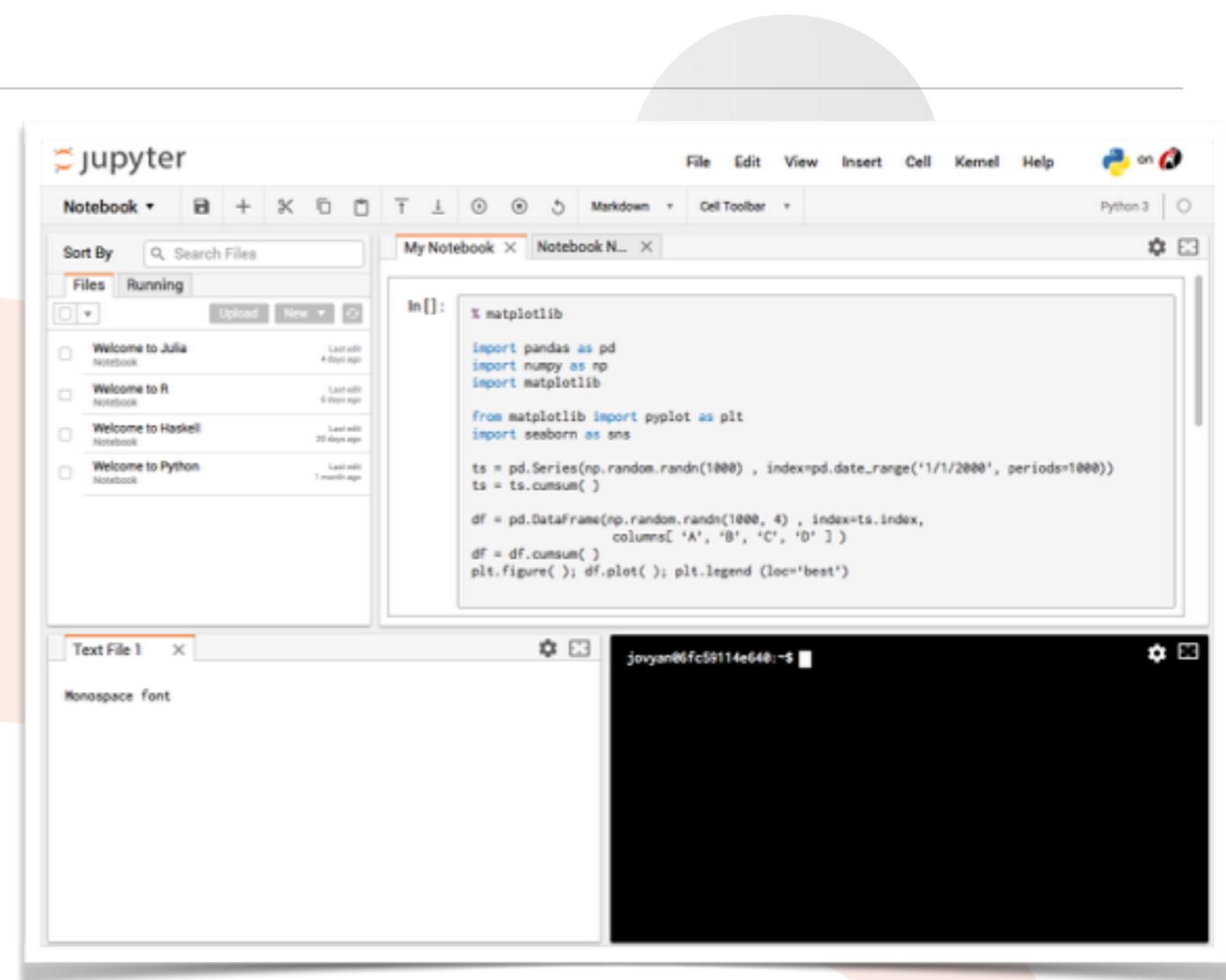
Fernando Perez and Brian Granger discuss the architecture of Project Jupyter, as its scope expands to reach data science applications in over 40 programming languages.
Photo credit: Adriana Restrepo

Future work

- Interactive Computing
 - Notebooks as interactive applications
 - Modular, reusable UI/UX
 - Software engineering with notebooks
- Computational Narratives
 - nbconvert
 - Element filtering
 - Documentation
- Collaboration
 - Real time collaboration
 - JupyterHub
- Sustainability
 - People
 - Events

Future work

- Component and tiled-layout are oft requested feature.
- Collaboration with Continuum Analytics.
- Plan on adding panels for Text editor, output, variable inspectors, debuggers, ...
- Discussion with Microsoft PTVS team for “debugger protocol”





The screenshot shows a web browser window with the URL opendreamkit.org in the address bar. The page title is "OpenDreamKit - Open Digital Research Environment Toolkit". The main content area features a large heading: "OpenDreamKit: Open Digital Research Environment Toolkit for the Advancement of Mathematics". Below the heading is a paragraph of text describing the project as a Horizon 2020 European Research Infrastructure. To the left of the main content is a sidebar with a blue background. It features the European Union flag icon and the text "OpenDreamKit". Below this, it says "A project funded by the Horizon 2020 European Research Infrastructures Work Programme." A navigation menu includes links for "Home", "News", "About", "Job openings", "Follow us" (with icons for RSS, Twitter, and GitHub), "Edit this page", and "Currently v0.2.0". At the bottom of the sidebar is the copyright notice "© 2015. All rights reserved."

OpenDreamKit: Open Digital Research Environment Toolkit for the Advancement of Mathematics

OpenDreamKit is a [Horizon 2020 European Research Infrastructure](#) project that will run for four years, starting from September 2015. It will provide substantial funding to the open source computational mathematics ecosystem, and in particular popular tools such as [LinBox](#), [MPIR](#), [SageMath](#), [GAP](#), [Pari/GP](#), [LMFDB](#), [Singular](#), [MathHub](#), and the [IPython/Jupyter](#) interactive computing environment.

From this ecosystem, OpenDreamKit will deliver a flexible toolkit enabling research groups to set up [Virtual Research Environments](#), customised to meet the varied needs of research projects in pure mathematics and applications, and supporting the full research life-cycle from exploration, through proof and publication, to archival and sharing of data and code.

Altogether the project involves about 50 people spread over 15 sites in Europe, with a total budget of about 7.6 million euros. The largest portion of that will be devoted to employing an average of 11 researchers and developers working full time on the project. Additionally, the participants will contribute the equivalent of six other people working full time.

[Read more...](#)

Hiring

- At UC Berkeley
 - Two new postdocs
 - Project manager
 - Web developer, tech writer (short contracts)
 - One administrative assistant.
- At Cal Poly
 - Three software engineers (one already hired)
 - One designer
 - One administrative assistant.

Thanks

Time for questions ?