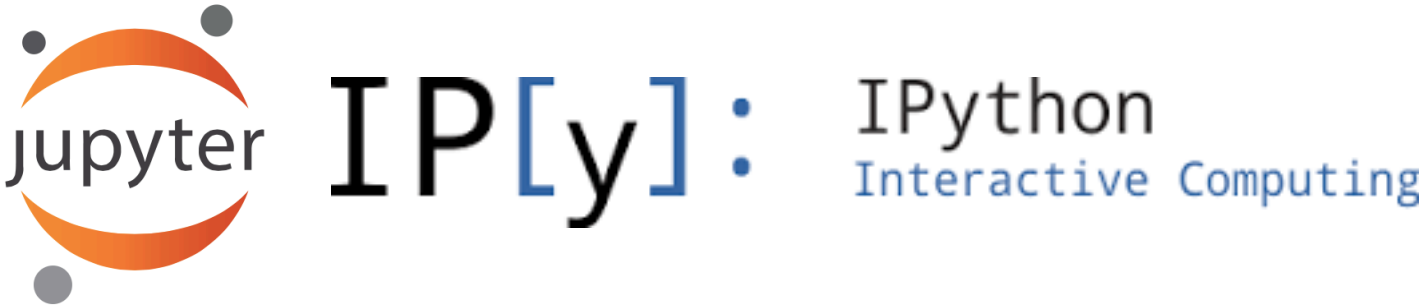


José Enrique Ruiz

Segundo Encuentro Python Granada

Centro de Enseñanzas Virtuales de la Universidad de Granada – CEVUG

26th June 2015

Digital Astronomy

Astronomy research lifecycle is entirely digital

- Observation proposals
- Data reduction pipelines

- Analysis of science ready data
- Catalogs of objects and data archives
- Publish process - ADS/arXiv
  - Materials and Methods
  - Results
  - Discussion

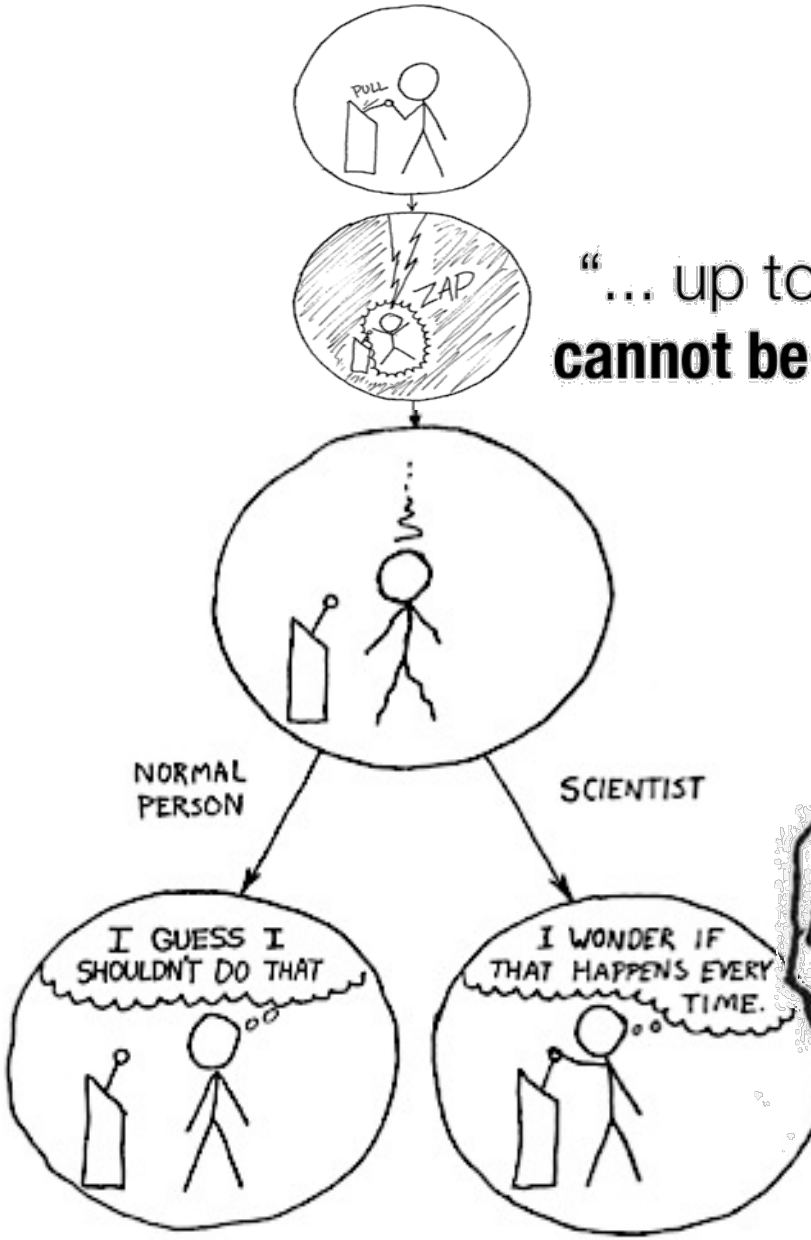
Reproducible research is still not possible in a digital world

A rich infrastructure of data is not efficiently used

A normalized preservation of methodology is needed

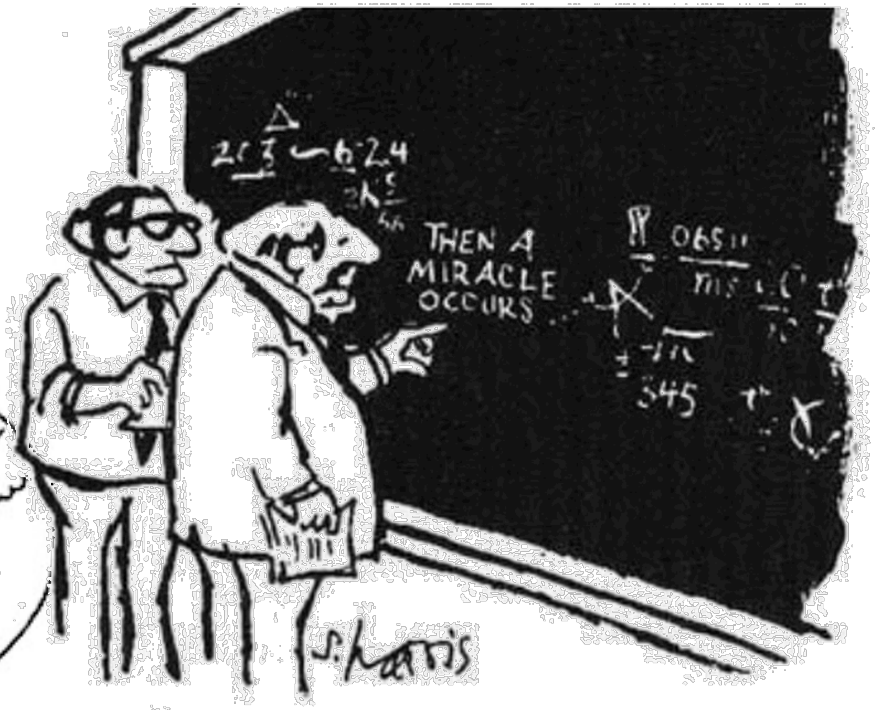
Tools

The Reproducibility Crisis



"... up to 70% of research from academic labs **cannot be reproduced**, representing an enormous waste of money and effort."

- Elizabeth Iorns, Science Exchange



"I think you should be more explicit here in step two."

http://xkcd.com/242/

The Reproducibility Crisis

Visibility, Efficiency and Reuse

Optimize return on investments made on big facilities

- Avoid duplication of efforts and reinvention
- How to discover and not duplicate ?
- How to re-use and not duplicate ?
- How to make use of best practices ?
- How to use the rich infrastructure of data ?
- **Intellectual contributions encoded in software**

More data in archives do not imply more knowledge

- Expose **complete scientific process**, not the story
- Allow easy **discovery** of methods and tools

More R's than Pirates !

Prof. David de Roure

http://www.scilogs.com/eresearch/replacing-the-paper-the-twelve-rs-of-the-e-research-record/

http://www.scilogs.com/eresearch/more-rs-than-pirates/

Open Science is much more than Open Data

Repeatable

Tools and methodology available

I could repeat the experiment

Reproducible

Clear methodology and **all resources** available

I could reproduce the results

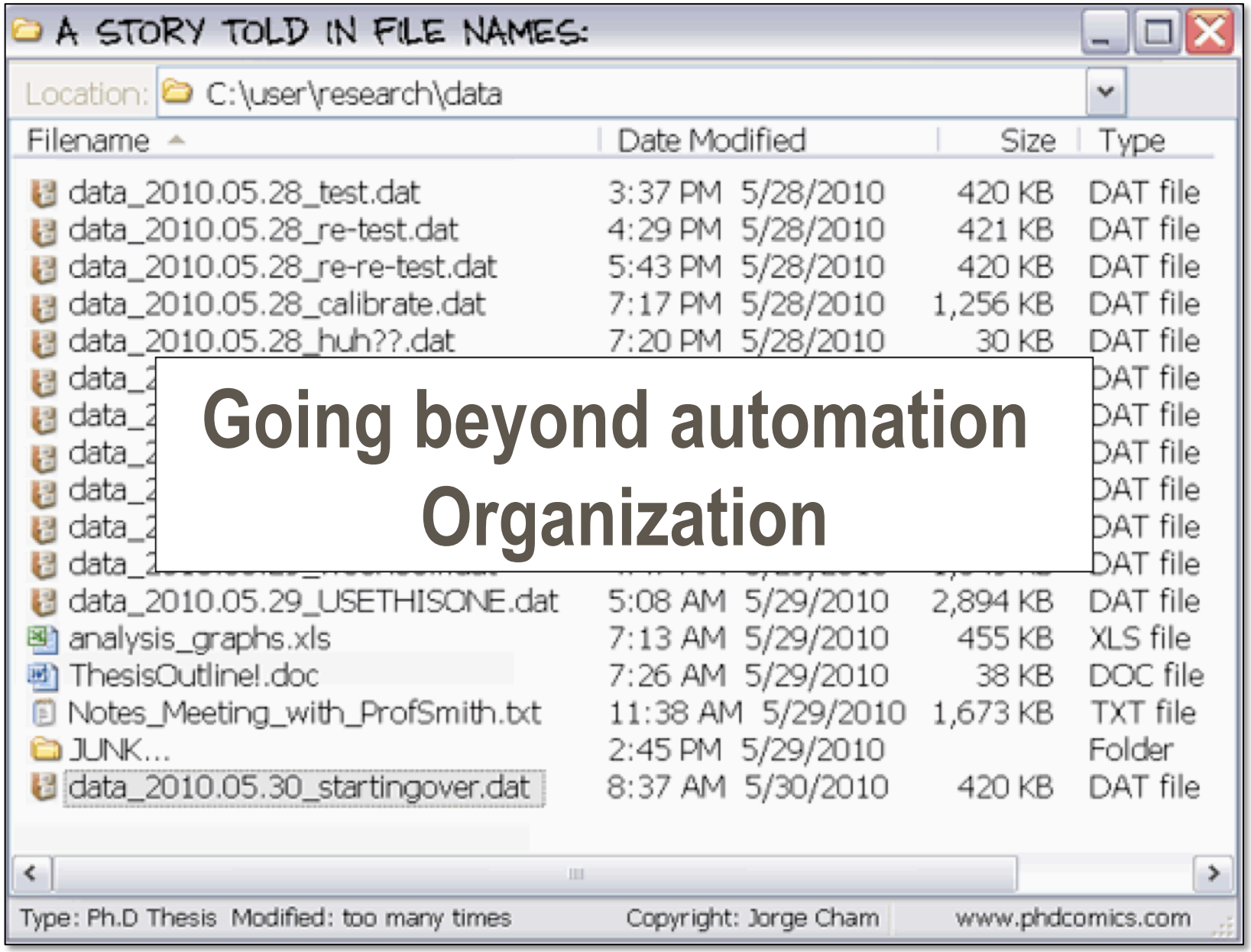
Reusable

I know how it could be useful for my needs

I could use all or some parts as it is

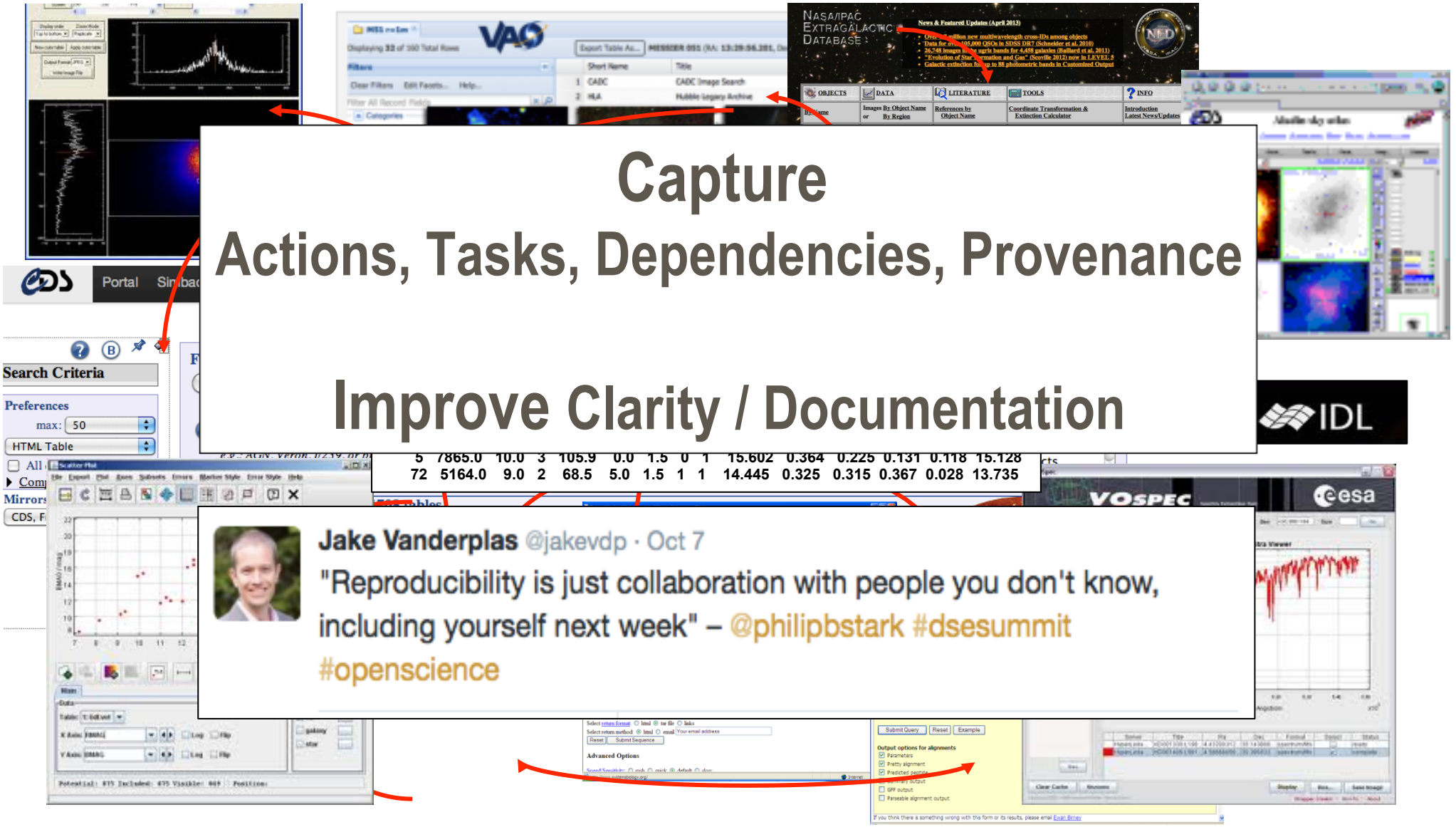
I could modify and adapt it even for other purposes

Story Capture



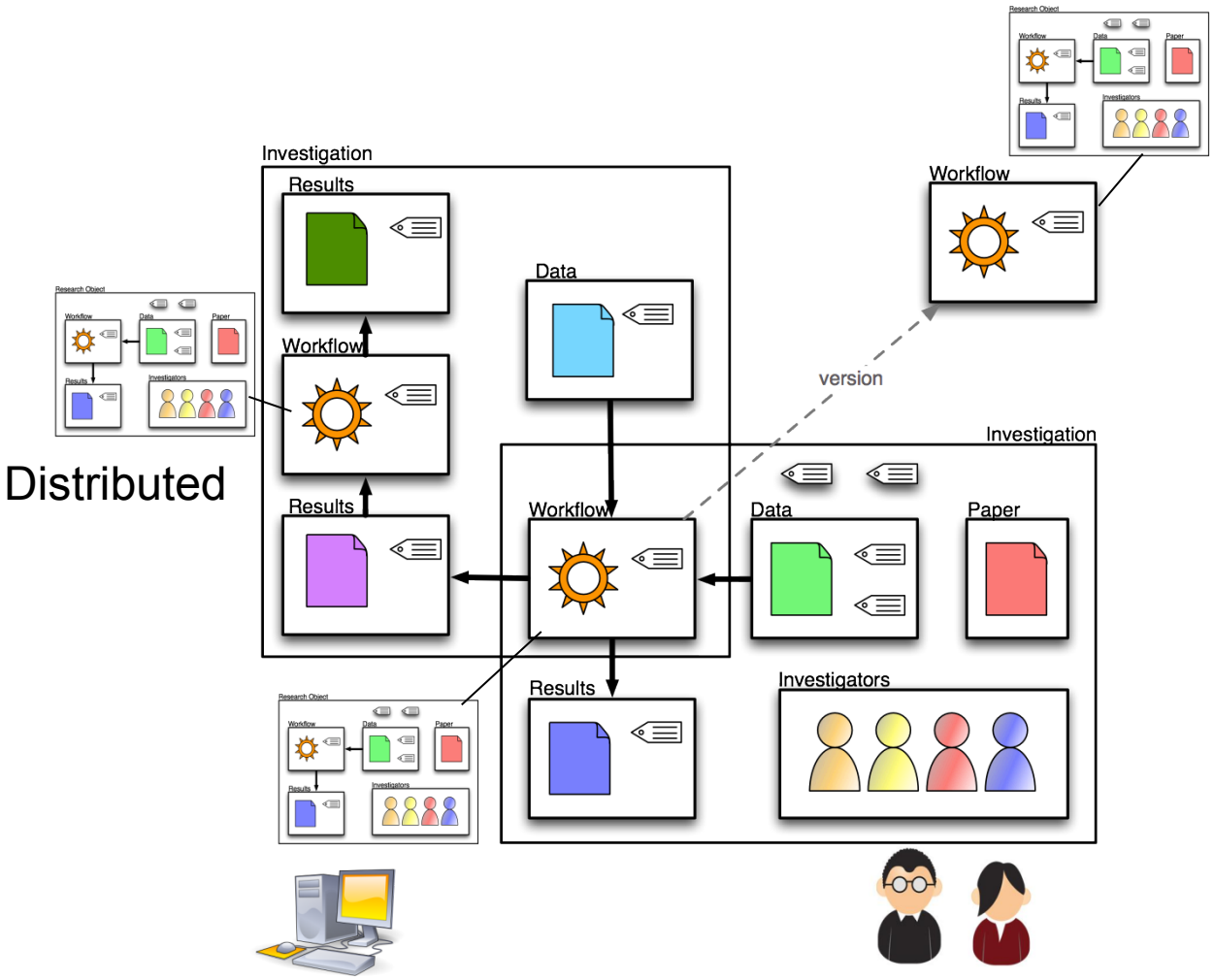
Story Capture

Story Capture



Research Objects

Expose experimental context in a structured way in order to be understood



researchobject

http://www.researchobject.org/

The Social Dimension

Visibility and Discovery

Pre-explosion Upper Limit on X-ray Emission from a Progenitor for SN 2014J

ATel #5798; *W. P. Maksym (U. Alabama), J. A. Irwin (U. Alabama), W. C. Keel (U. Alabama), D. Burke (CFA-SAO), K. Schawinski (ETH Zurich)*
on 23 Jan 2014; 20:50 UT
Credential Certification: Peter Maksym (peter.maksym@gmail.com)

Subjects: X-ray, Cataclysmic Variable, Supernovae

Referred to by ATel #: 5809, 5851

Tweet 46 Recommend 5

To identify or set limits on any possible accreting white dwarf progenitor to SN 20
#3792, ATel #5786), we examined available pre-explosion archival Chandra data
cover multiple observations of M82 dating back to 1999 Sep. 20.

Initial examination by D. Burke
(https://twitter.com/doug_burke/status/426065675497381888) over ~820 ks of m
and ACIS-S data with non-uniform coverage, and reveals no source at RA,Dec=09
+69:40:26.0 (CBET #3792). The source is also not visible in the longest single exp

To minimize effects of off-axis spreading of the Chandra PSF, we examined the ar
using only the sixteen ACIS-S and ACIS-I epochs with off-axis angle < 2 arcmin,
~500 ks in the range of 0.1-8 keV. Within r=0.5 arcsec (~1 Chandra pixel), we fin
with a 1-sigma upper limit of 5.7 counts using the NIRC2 localization by Tendulk
(ATel #5789) RA,Dec=9:55:42.217,+69:40:26.56. In addition, none of the sixteen
exposures has any evidence for X-ray flaring at the location of SN 2014J.

Bryan Gaensler @SciBry · Jan 23
A tweet has been cited as a reference in an astronomical publication! A world first for @doug_burke? #M82supernova astronomerstelegram.org/?read=5798
Expand Reply Retweeted Favorite More

Peter Maksym @StellarBones · Jan 23
@SciBry @doug_burke To be fair, ATel is pretty flexible as far as publications go :)
Expand Reply Retweet Favorite More

Bryan Gaensler @SciBry · Jan 23
@StellarBones @doug_burke but they are indexed on ADS. Good enough for me!
Expand Reply Retweet Favorite More

Douglas Burke @doug_burke · Jan 24
And to be fair to @adsabs , I'm sure they're all set up to accept references to tweets ;-) @SciBry @StellarBones
Expand Reply Retweet Favorite More

aaccomazzi @aaccomazzi · Jan 24
@doug_burke @adsabs @scibry @stellarbones Ugh... I knew this was bound to happen... Now what?
Hide conversation Reply Retweet Favorite More

Time has come to go beyond the PDF

IPy IPython Dashboard | IPy spectrogram

127.0.0.1:8888/a5222740-848b-4ac1-b212-d732c9f8f78b

IP[y]: Notebook spectrogram Last saved: Mar 07 11:14 PM

File Edit View Insert Cell Kernel Help

Markdown

Simple spectral analysis

IPython Notebooks

An illustration of the Discrete Fourier Transform

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \qquad k = ($$

All started as a browser-wrapper for iPython

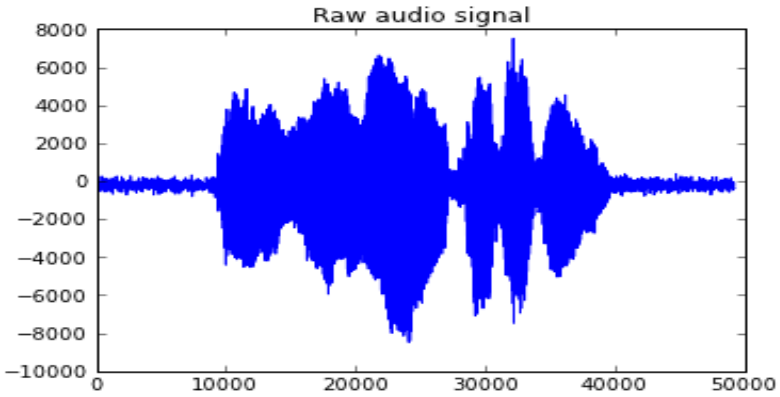
using windowing, to reveal the frequency content of a sound signal.

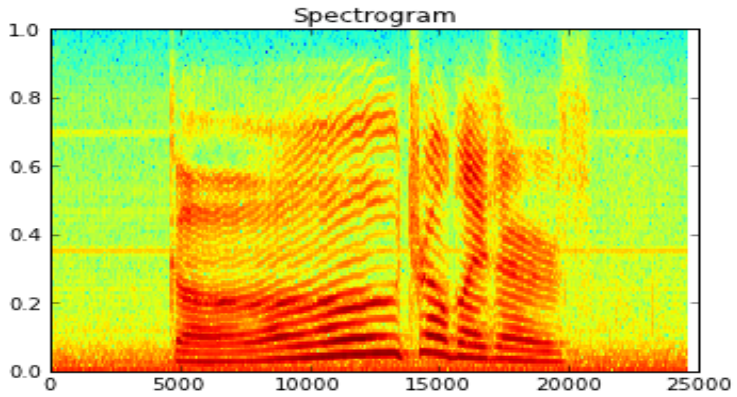
We begin by loading a datafile using SciPy's audio file support:

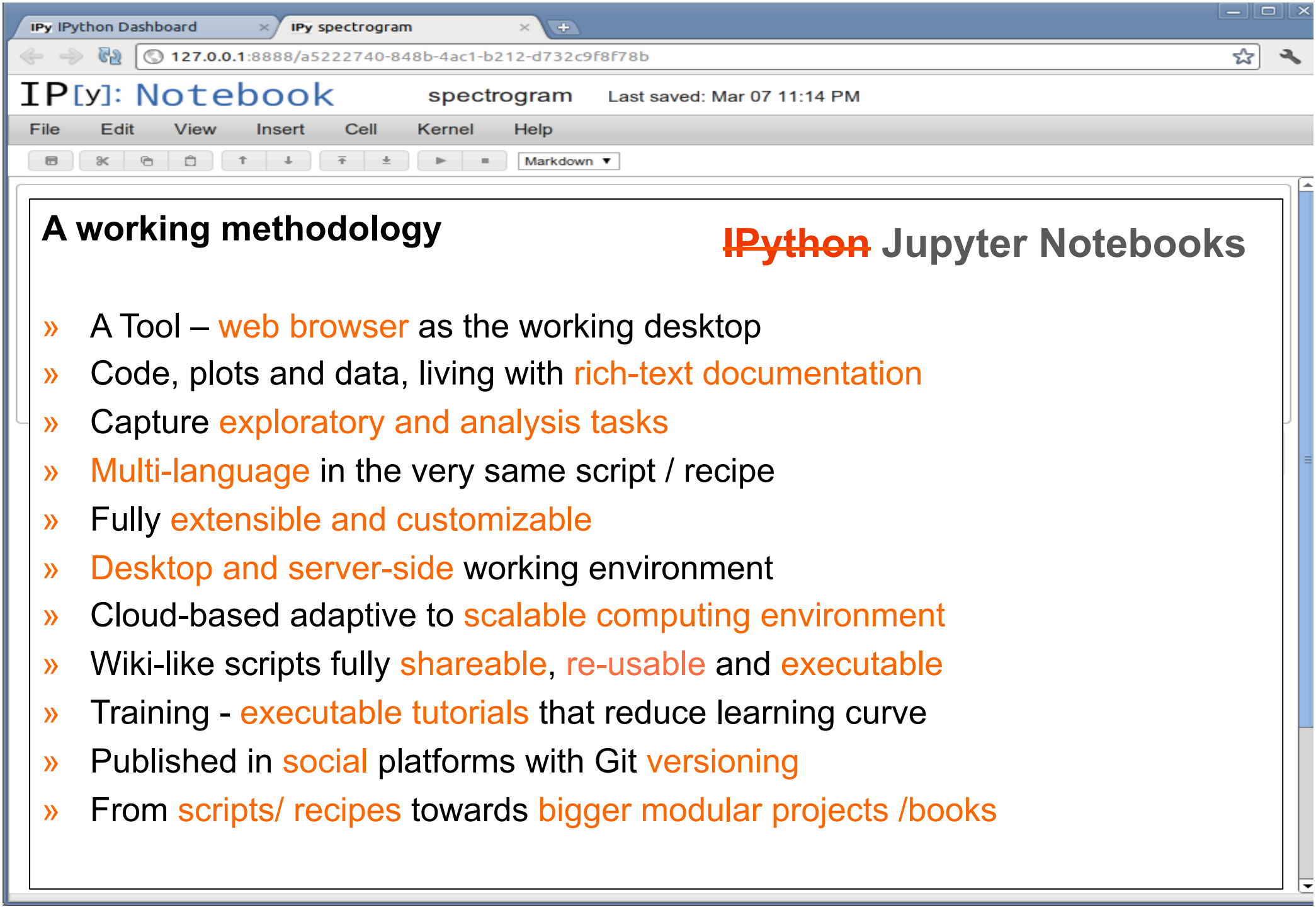
In [1]:
```
from scipy.io import wavfile
rate, x = wavfile.read('test_mono.wav')
```

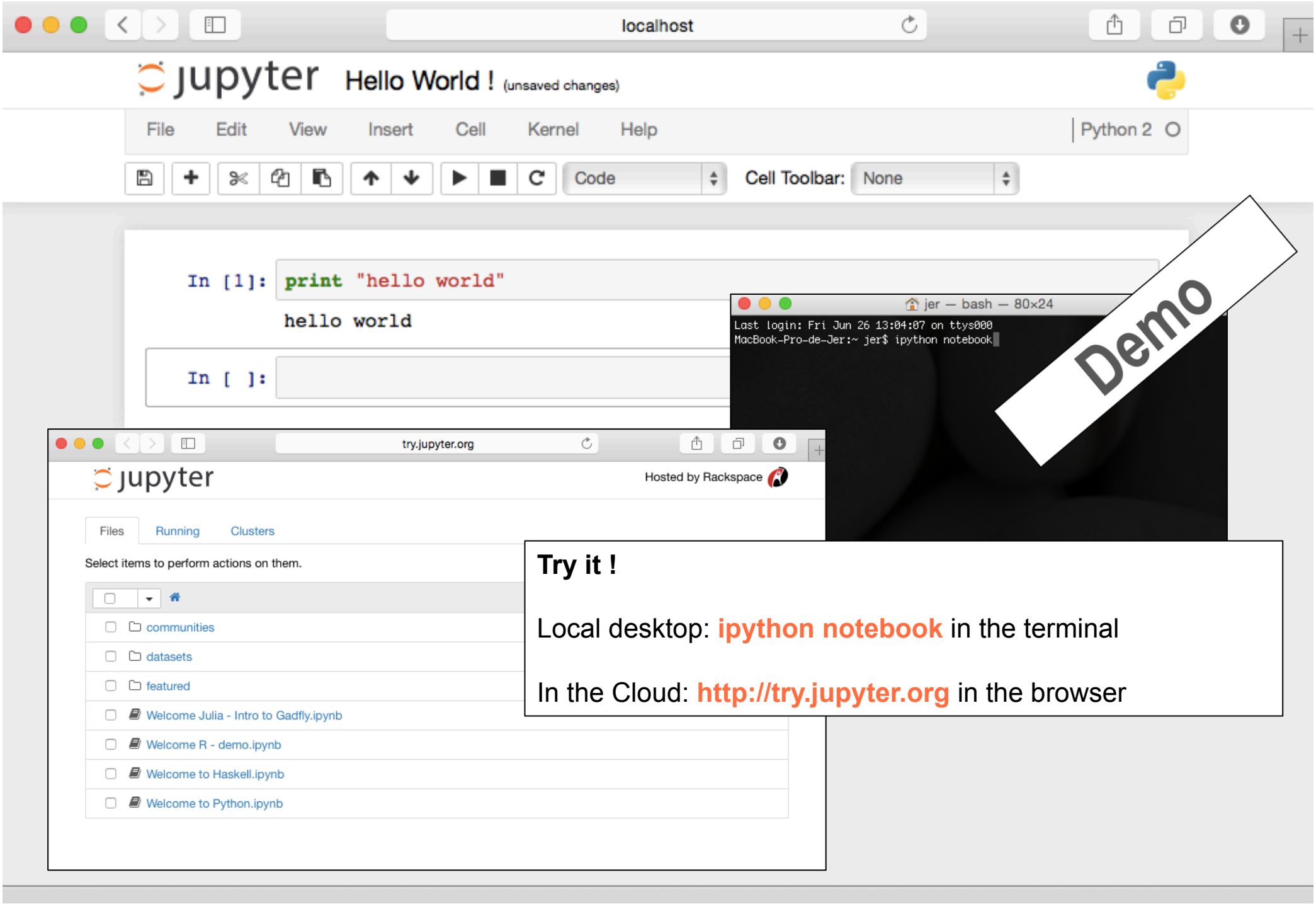
And we can easily view its spectral structure using matplotlib's builtin `specgram` routine:

In [2]:
```
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))
ax1.plot(x); ax1.set_title('Raw audio signal')
ax2.specgram(x); ax2.set_title('Spectrogram');
```



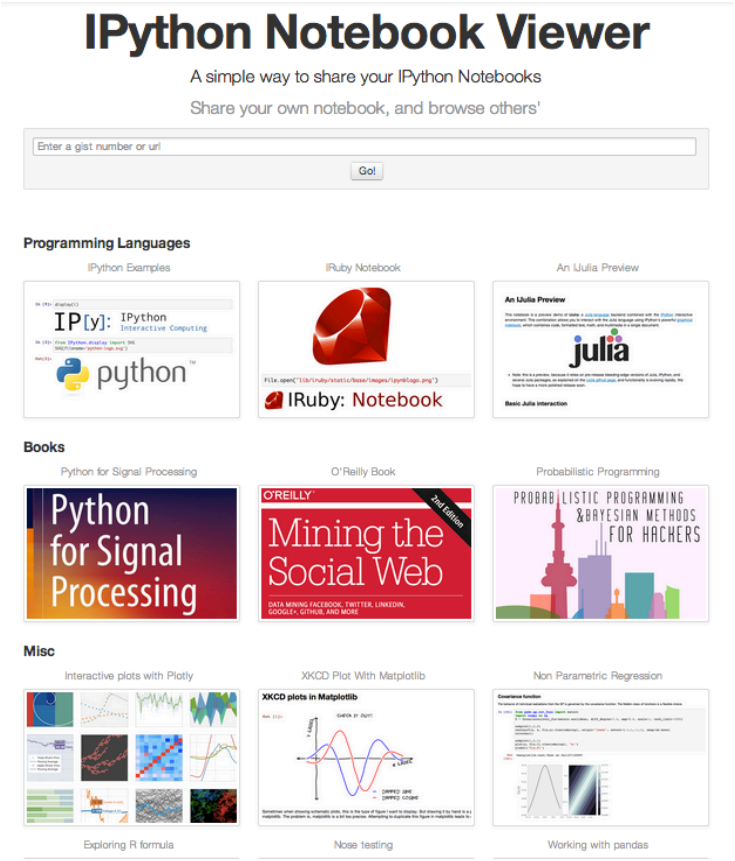



IPy IPython Dashboard
IPy spectrogram
127.0.0.1:8888/a5222740-848b-4ac1-b212-d732c9f8f78b
IP[y]: Notebook
spectrogram
Last saved: Mar 07 11:14 PM
File Edit View Insert Cell Kernel Help
Markdown
A working methodology
~~IPython~~ Jupyter Notebooks
» A Tool – web browser as the working desktop
» Code, plots and data, living with rich-text documentation
» Capture exploratory and analysis tasks
» Multi-language in the very same script / recipe
» Fully extensible and customizable
» Desktop and server-side working environment
» Cloud-based adaptive to scalable computing environment
» Wiki-like scripts fully shareable, re-usable and executable
» Training - executable tutorials that reduce learning curve
» Published in social platforms with Git versioning
» From scripts/ recipes towards bigger modular projects /books


localhost
jupyter Hello World ! (unsaved changes)
File Edit View Insert Cell Kernel Help
Python 2
Code
Cell Toolbar: None
In [1]: print "hello world"
hello world
In [ ]:
jer — bash — 80×24
Last login: Fri Jun 26 13:04:07 on ttys000
MacBook-Pro-de-Jer:~ jer$ ipython notebook
Demo
try.jupyter.org
jupyter
Hosted by Rackspace
Files Running Clusters
Select items to perform actions on them.
communities
datasets
featured
Welcome Julia - Intro to Gadfly.ipynb
Welcome R - demo.ipynb
Welcome to Haskell.ipynb
Welcome to Python.ipynb
Try it !
Local desktop: ipython notebook in the terminal
In the Cloud: http://try.jupyter.org in the browser

The Library

http://lmgtfy.com/?q=ipython+notebook+examples#



Cloud Solutions

SageMathCloud™ collaborative computational mathematics

Wakari.io

Web-based Python Data Analysis

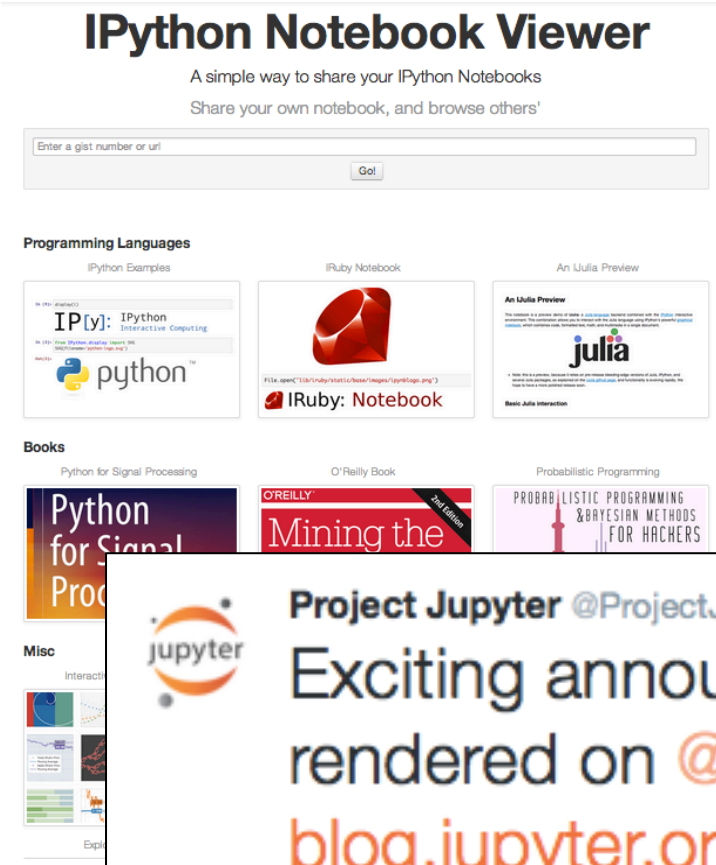
Visualization of **static** notebooks previously published as **gists** **shareable** via URL link, may be **downloaded** and executed locally

http://nbviewer.ipython.org

github SOCIAL CODING

The Library

http://lmgtfy.com/?q=ipython+notebook+examples#



Cloud Solutions

SageMathCloud™ collaborative computational mathematics

Wakari.io

Visu
shar
local

http://nbviewer.ipython.org

thub
SOCIAL CODING

IPython/Jupyter kernels:

1. IJulia
2. IHaskell
3. IFSharp
4. IRuby
5. IGo
6. IScala
7. IMathics
8. IAldor
9. Calico Project - kernels implemented in Mono, including Java, Iro
   Logo, BASIC, and many others
10. LuaJIT/Torch
11. Lua Kernel
12. Simple example kernel
13. IRKernel (for the R language)
14. IErlang
15. IOCaml
16. IForth
17. IPerl
18. IPerl6
19. IPHP
20. IOctave
21. IScilab
23. ICSharp
24. Bash
25. Clojure Kernel
26. Hy Kernel
27. Redis Kernel
28. jove, a kernel for io.js
29. IJavascript
30. Calysto Scheme
31. Calysto Processing
32. idl_kernel
33. Mochi Kernel
34. Lua (used in Splash)
35. Spark Kernel
36. Skulpt Python Kernel
37. MetaKernel Bash
38. MetaKernel Python
39. Brython Kernel
40. IVisual VPython Kernel
41. Brainfuck Kernel (IBrainfuck)
42. KDB+/Q Kernel (IKdbQ)
43. ICryptol
44. C++ (cling)

https://github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages

Magic functions

https://ipython.org/ipython-doc/dev/interactive/magics.html
http://nbviewer.ipython.org/github/ipython/ipython/blob/1.x/examples/notebooks/Cell%20Magics.ipynb

%cd

Change the current working directory.

%install_ext

Download and install an extension from a URL, e.g.:

```
%install_ext https://bitbucket.org/birkenfeld/ipython-physics/raw/d1310a2ab15d/physics.py
```

%load

Load code into the current frontend.

%matplotlib

```
In [1]: %matplotlib inline
```

%run

Run the named file inside IPython as a program.

%save

Save a set of lines or a macro to a given filename.

Magic functions

https://ipython.org/ipython-doc/dev/interactive/magics.html
http://nbviewer.ipython.org/github/ipython/ipython/blob/1.x/examples/notebooks/Cell%20Magics.ipynb

%%html

Render the cell as a block of HTML

%%javascript

Run the cell block of Javascript code

%%latex

Render the cell as a block of latex

%%perl

%%perl script magic

Run cells with perl in a subprocess.

This is a shortcut for `%%script perl`

%%script

Run a cell via a shell command

%%bash

%%bash script magic

Run cells with bash in a subprocess.

This is a shortcut for `%%script bash`

%%writefile

```
%writefile [-a] filename
```

Write the contents of the cell to a file.

Extensions

Customizing a Notebook Server

https://github.com/ipython/ipython/wiki/Extensions-Index

CSV Magic

Tools for quickly importing and exporting data from CSV files.

```
%install_ext https://raw.githubusercontent.com/FrankSalad/ipython-csvmagic/master
```

fortran magic

```
%install_ext https://raw.github.com/mgaitan/fortran_magic/master/fortranmagic.py
```

Rmagic

Magic command interface for interactive work with R in ipython.

IDL

Provides magics for embedding IDL and GDL code using pIDLy.

Matlab

The Python MATLAB bridge enables calling of MATLAB code and functions from an IPython session and adds a `%%matlab` cell magic, which allows embedding matlab code in IPython notebooks.

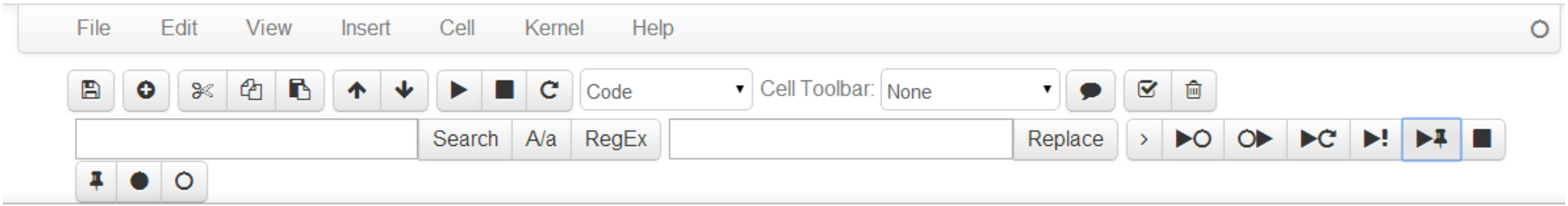
Extensions

Customizing a Notebook Server

https://github.com/ipython-contrib/IPython-notebook-extensions

Javascript extensions for added UI functionality

Name	Description
usability	Additional functionality for the notebook
publishing	Getting your notebooks out in the wild
styling	Styling schemes for different looks of the notebook
slidemode	Slideshow creation
testing	Extensions in a early stage



Customizing a Notebook Server

Startup Files

If you want some code to be run at the beginning of every IPython session, the easiest way is to add Python (.py) or IPython (.ipy) scripts to your `profile_default/startup/` directory. Files here will be executed as soon as the IPython shell is constructed, before any other code or scripts you have specified. The files will be run in order of

- **Access to common global functions and variables**
- **Functions as wrappers to access CLI server-side software**
- **Integrated access to server-side databases**

- **Customized execution framework for a research group**
- **Internal library of executable recipes**
- **Access restricted content**

Modular Complex Notebooks

IP[y]: Notebook Masterlink Last Checkpoint: Nov 20 10:33 (autosaved)

File Edit View Insert Cell Kernel Help

Heading 1 Cell Toolbar: None

Masterlink Notebook

This notebook links to Linked Notebook and **EXECUTES IT DINAMICALLY !!**
Support for this functionality is provided in special python script:

- */mnt/USERS/amiga/.ipython/profile_nbserver/startup/run-pynb.py* loaded at IPython Notebook Server startup.

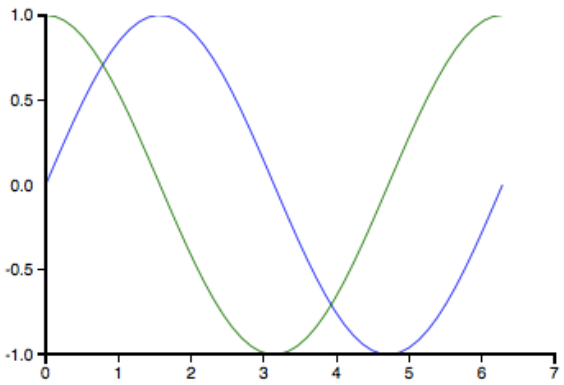
Tim Bernes Lee is widely known for things as simple and spectacular like this one.

Execute Linked Notebook

In [1]:
```
execute_notebook("LinkedNotebook.ipynb")
```

Now we do our analysis/plotting staff with the variables provided by Linked Notebook

In [2]:
```
pylab.plot(x,s,x,c)
pylab.show()
```

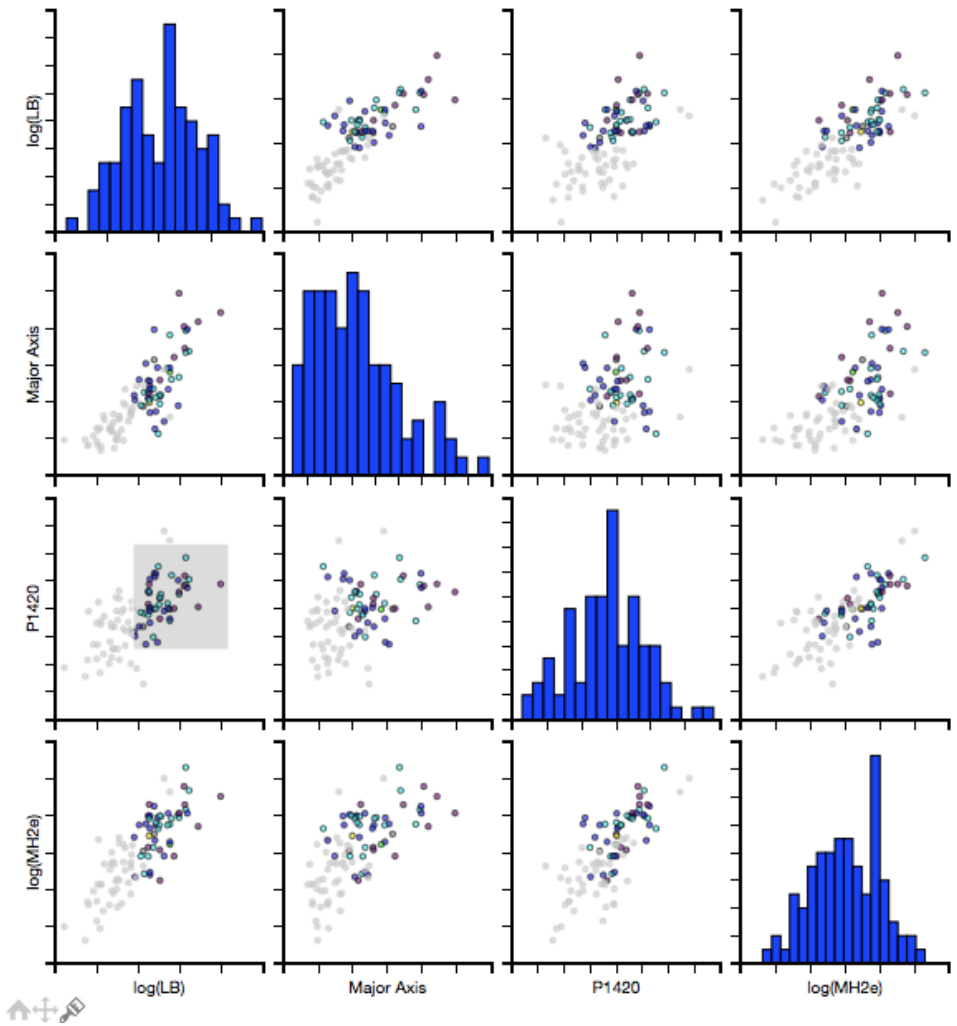


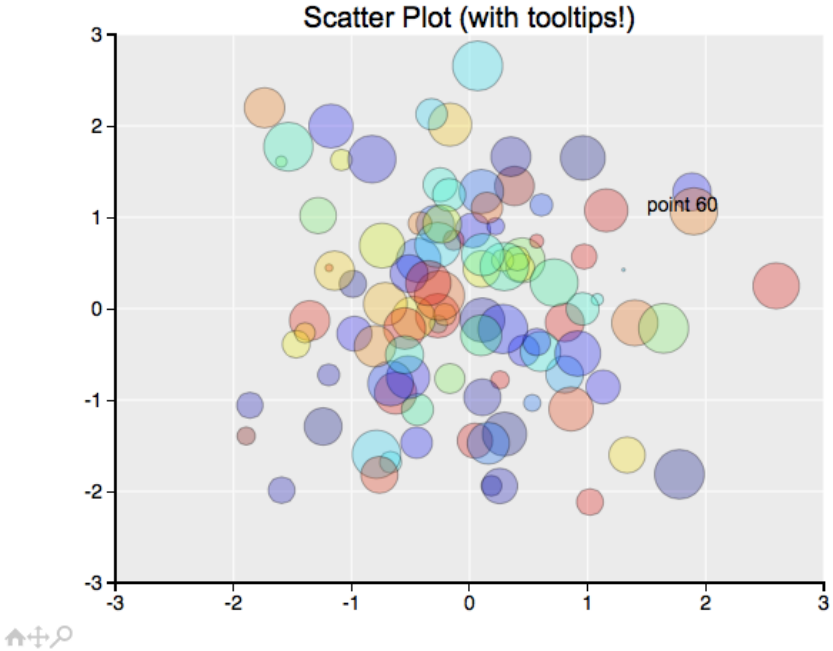
You can edit and save the Linked Notebook, and then re-execute the command above to produce other result. **Try it !**

WOW !!

```
import io
from IPython.nbformat import current

def execute_notebook(nbfile):

    with io.open(nbfile) as f:
        nb = current.read(f, 'json')

    ip = get_ipython()

    for cell in nb.worksheets[0].cells:
        if cell.cell_type != 'code':
            continue
        ip.run_cell(cell.input)
```

Interactive Plotting Exploration





Seamless work with databases

catherinedevlin / ipython-sql

Watch 19 | Star 159 | Fork 29

branch: master ▾ ipython-sql / examples / writers.ipynb

Catherine Devlin on Mar 29, 2013 Allow multiple SQL per cell. Because why not?

0 contributors

112 lines (112 sloc) | 2.305 kB | Raw | Blame | History

```
In [3]: %load_ext sql

In [4]: %sql sqlite://
Out[4]: 'Connected: None@None'

In [5]: %%sql
        CREATE TABLE writer (first_name, last_name, year_of_death);
        INSERT INTO writer VALUES ('William', 'Shakespeare', 1616);
        INSERT INTO writer VALUES ('Bertold', 'Brecht', 1956);
Out[5]: []

In [6]: %sql select * from writer
```

Out[6]:

first_name	last_name	year_of_death
William	Shakespeare	1616
Bertold	Brecht	1956

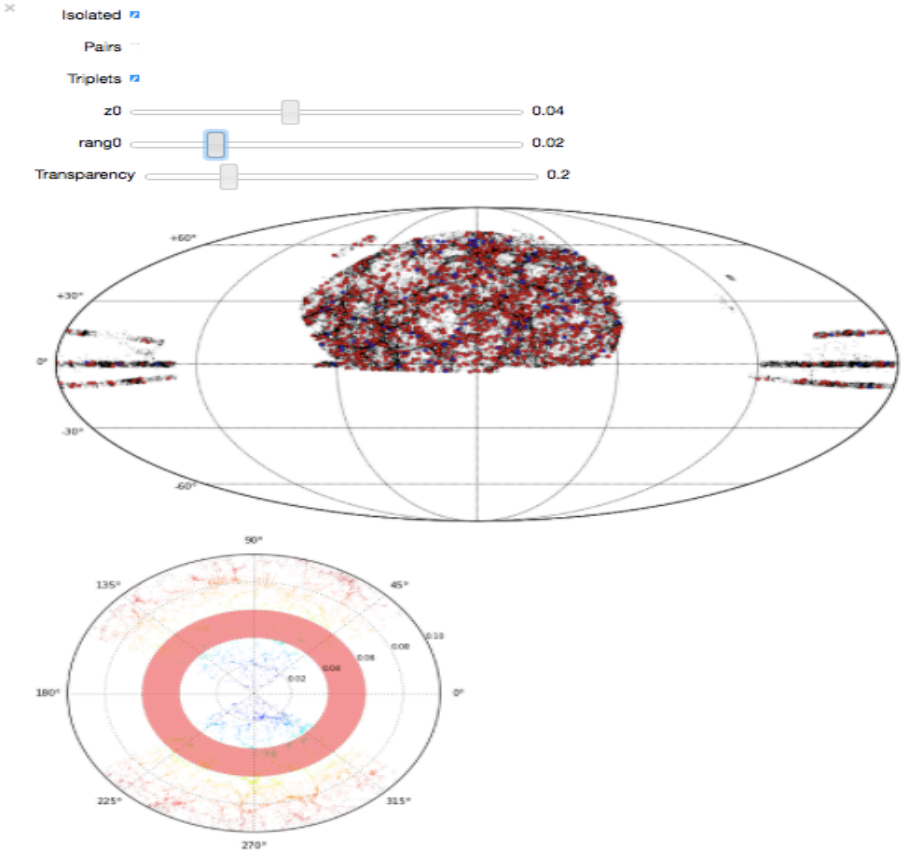
```
In [ ]:
```

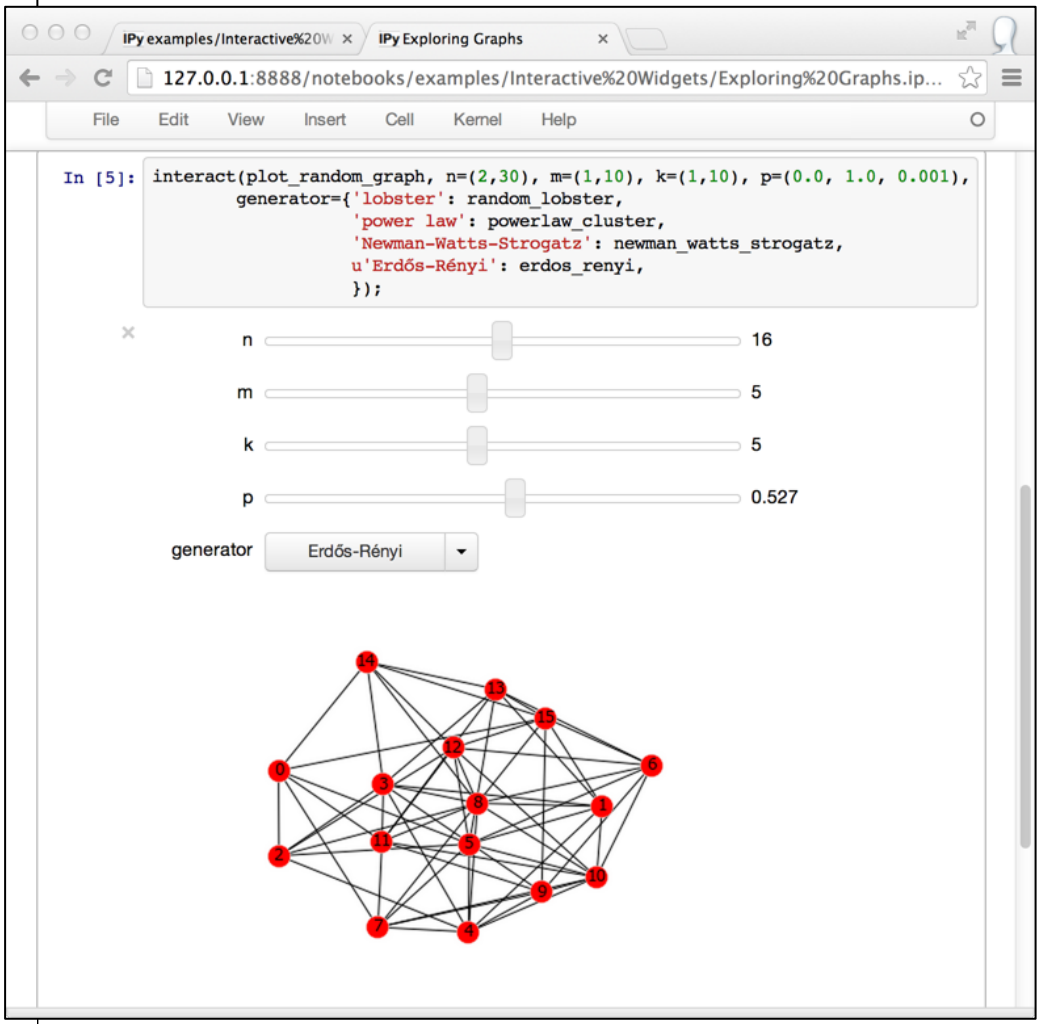
Widgets

LSSGALPY - Mollweide

Python tool for the interactive visualization of the large-scale environment around galaxies on the 3D space.

This code contains the visualisation tools developed for the A&A Article Catalogues of isolated galaxies, isolated pairs, and isolated triplets in the local Universe by M. Argudo-Fernández, S. Verley, G. Bergond, S. Duarte Puertas, E. Ramos Carmona, J. Sabater, M. Fernández-Lorenzo, D. Espada, J. Sulentic, J. E. Ruiz, and S. Leon.





https://ipython.org/ipython-doc/dev/whatsnew/version2.0.html#interactive-widgets
https://github.com/margudo/LSSGALPY

Time has come to go beyond the PDF

The Executable Paper

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

IP[y]: Notebook Nature (autosaved)

File Edit View Insert Cell Kernel Help

Markdown Cell Toolbar: None

nature

Introduction

Welcome! You have just launched a live example of an IPython Notebook. The notebook is an environment that lets you combine live code, narrative text, mathematics, plots and rich n documents provide a complete reproducible record of a computation and its results and can be example, email, web-hosting services such as GitHub, Dropbox, and nbviewer).

You can edit anything in this temporary demonstration notebook, including the text you are readi 'Expand' icon in the lower right corner of the frame around this notebook.

This notebook showcases some of IPython's capabilities for researchers.

This demonstration is hosted by Rackspace and is running on its bare metal offering, OnMetal. T through Rackspace's developer+ page.

Basic Python code and plotting

The box below (known as a code cell) contains the Python code to plot $y = x^2$ over the range [(by # explain what the code does.

To run the code:

1. Click on the cell to select it.
2. Press SHIFT+ENTER on your keyboard or press the play button (▶) in the toolbar above.

nature International weekly journal of science

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

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

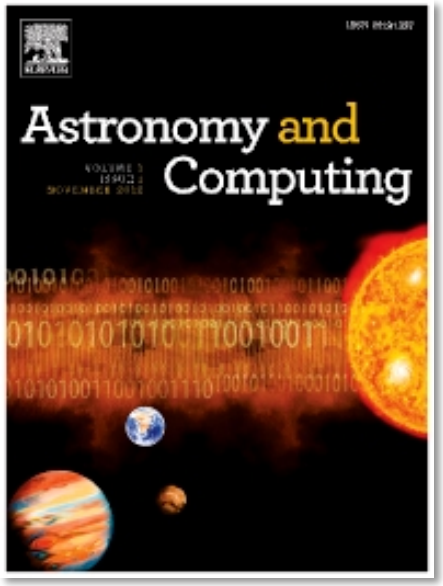
05 November 2014

PDF Rights & Permissions

Illustrations by The Project Twins

ELSEVIER article of the future

Publishing Research Objects



Graphical abstract

Source code repositories

The journal strongly encourages authors to make source code available where appropriate, especially in the case

Video data

Elsevier accepts video material and animation sequences to support and enhance your scientific research. Authors who have video or animation files that they wish to submit with their article are strongly encouraged to include links to these within the body of the article. This can be done in the same way as a figure or table by referring to the video or animation content and noting in the body text where it should be placed. All submitted files should be properly labeled so that they directly relate to the video file's content. In order to ensure that your video or animation material is directly usable, please provide the files in one of our recommended file formats with a preferred maximum size of 50 MB. Video and animation files supplied will be published online in the electronic version of your article in Elsevier Web products, including ScienceDirect: http://www.sciencedirect.com. Please supply 'stills' with your files: you can choose any frame from the video or animation or make a separate image. These will be used instead of standard icons and will personalize the link to your video data. For more detailed instructions please visit our video

video and animation cannot be

electronic and the print version for the

AudioSlides

MATLAB FIG files

***NEW* Inline supplementary computer code**

Elsevier now offers you the possibility to place supplementary computer code, data snippets, algorithms and other machine readable structures at the right place in your online article in reusable .txt format. This will allow readers to easily view this material in the appropriate context, and to directly copy it to the clipboard or download the original source file for testing or re-use. If you would like to have reusable "computer code" inserted into the body of your online article please indicate in your manuscript where they should be placed and number them in order of appearance, e.g. "Insert Inline Supplementary Computer Code 1 here". To support discoverability and reusability please submit these items in *.txt format and make sure to include a descriptive title and caption that references the characteristics and the appropriate environment of this material , e.g. 'An algorithm for filtering text files in R'. For more information please visit http://www.elsevier.com/ism.

ELSEVIER article of the future

Digital Science - Towards the Executable Paper
Publishing Research Objects

http://youtu.be/1dXkmgkYuEg

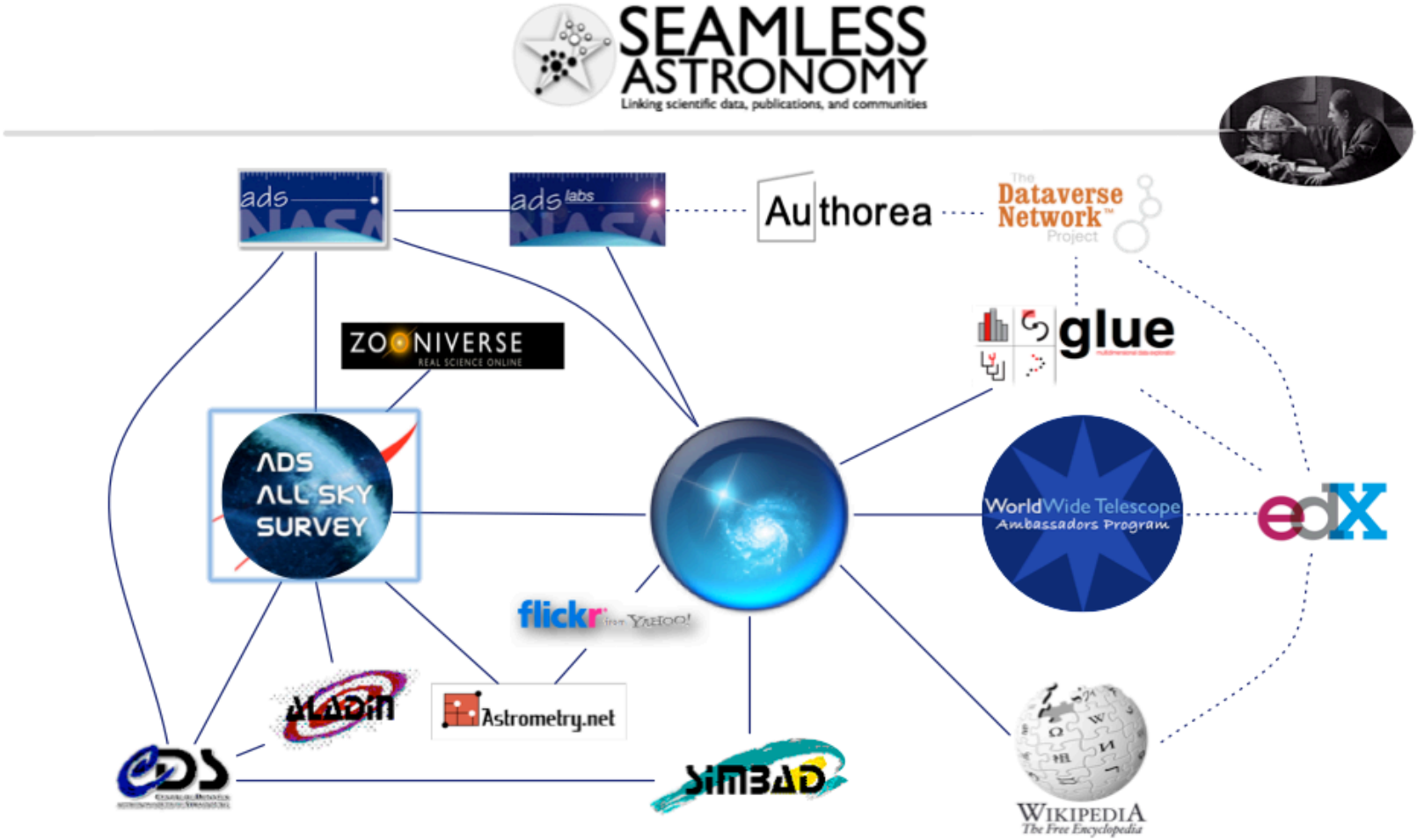
http://youtu.be/FQ5FaeHcdwo

Write science together.

Authorea

Authorea is the collaborative platform for research. Write and manage your technical documents in one place.

Collaborative Writing and Versioning



http://youtu.be/ObJG3i3jjPg

Supported by Microsoft Research