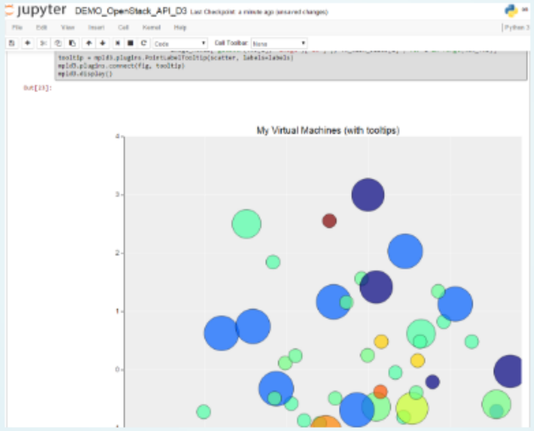
Jupyter for Everything Else

Michael Bright, EuroPython 2016 - Bilbao, 22 July.



mjbright *@mjbright* *mjbrightfr AT gmail* *mjbright.github.io*

About Me

Solution Architect working at Hewlett-Packard Enterprise, Grenoble, France

Working in the EMEA OpenNFV lab.

Interests:

- Docker, Docker, Docker, Jupyter, Python, OpenStack
- Run a local Python User Group in Grenoble
- From the UK, married to a Chilean, living in France for 24 years
- Argentinian Tango, Salsa, ...

TODO: add images ... Grenoble mountains, unicycle, Argentinian Tango, UK, NFV? SDN? Python UserGroup, Docker!!!, Jupyter, Linux, OStack

OUTLINE

- Introduction: From IPython to Jupyter
- The Jupyter Project & Ecosystem:
  - Kernels, Widgets, Extensions, Tools
  - Incubating projects
  - External: Thebe, Hosting, Binder ...
- Jupyter for Everything Else
  - Blog, Present, Web, Command-line, Status reports

IPYTHON - THE CONSOLE

"*an afternoon hack*" (Nov 2001) by Fernando Perez

A tool to help in the exploration process

- Individual exploration
- Collaborative work
- Parallel Production Runs
- Publication odf **reproducible** results
- Education
- Repeat

IP[y]: IPYTHON - THE CONSOLE

Initial 0.0.1 version Gist

- REPL in 259 lines
- Input / Output cells
- History
- Plotting

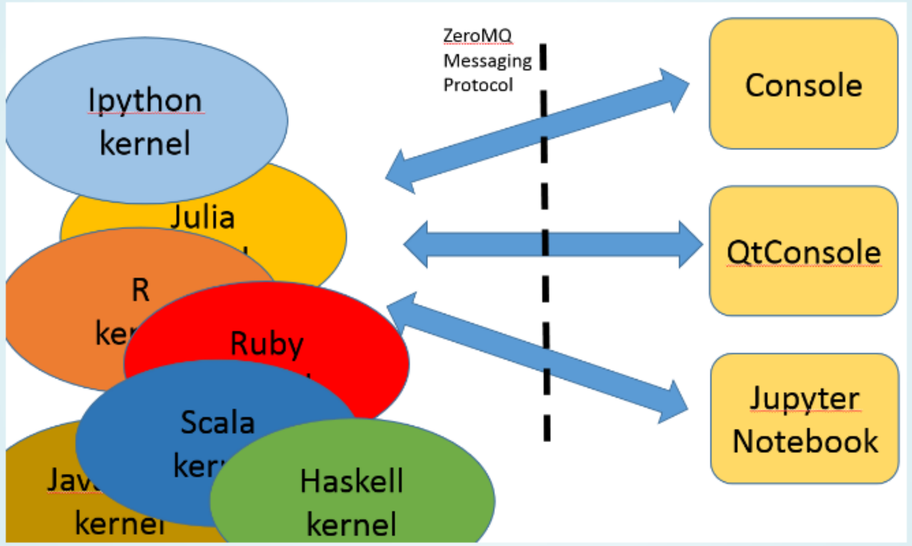
```
IPython 4.0.0 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's f
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' f

In [1]: def myfunc(msg):
   ...:     ''' prints msg '''
   ...:     print(msg)
   ...:

In [2]: myfunc("Hello World")
Hello World

In [3]: ?myfunc
Signature: myfunc(msg)
Docstring: prints msg
File:      d:\<ipython-input-1-e7b0ed71a6bd>
Type:      function
```

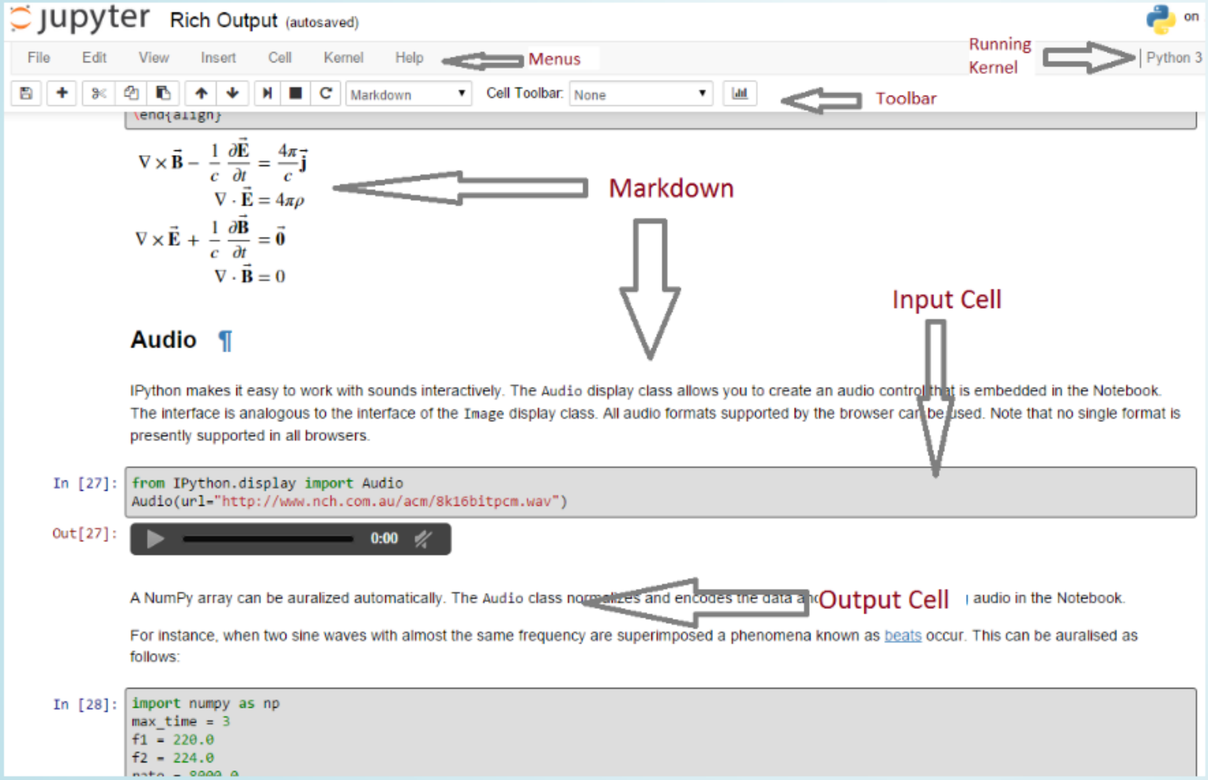
THE JUPYTER PROJECT



A notebook runs under one kernel

THE (JUPYTER) DASHBOARD & NOTEBOOK

Create & share documents of code, equations, visualizations and explanatory text as a **(reproducible) narrative**



JUPYTER: PUBLIC NOTEBOOKS

Jupyter notebooks are used in many

- scientific (physics, chemistry, biology, genomics, data analysis)
- and non-scientific (finance) domains

Site	URL	Info
nbviewer	https://nbviewer.org]	submit your url, browse by theme
github	https://github.com	> *200k notebooks* [*Announcement - May '15*]
IPython gallery	A-gallery-of-interesting-IPython-Notebooks	many notebooks organized by domain
Notebook Gallery	[**nb.binap.net** **[http://nb.bianp.net/]**	view submitted notebooks by '*most viewed*' or '*data*'

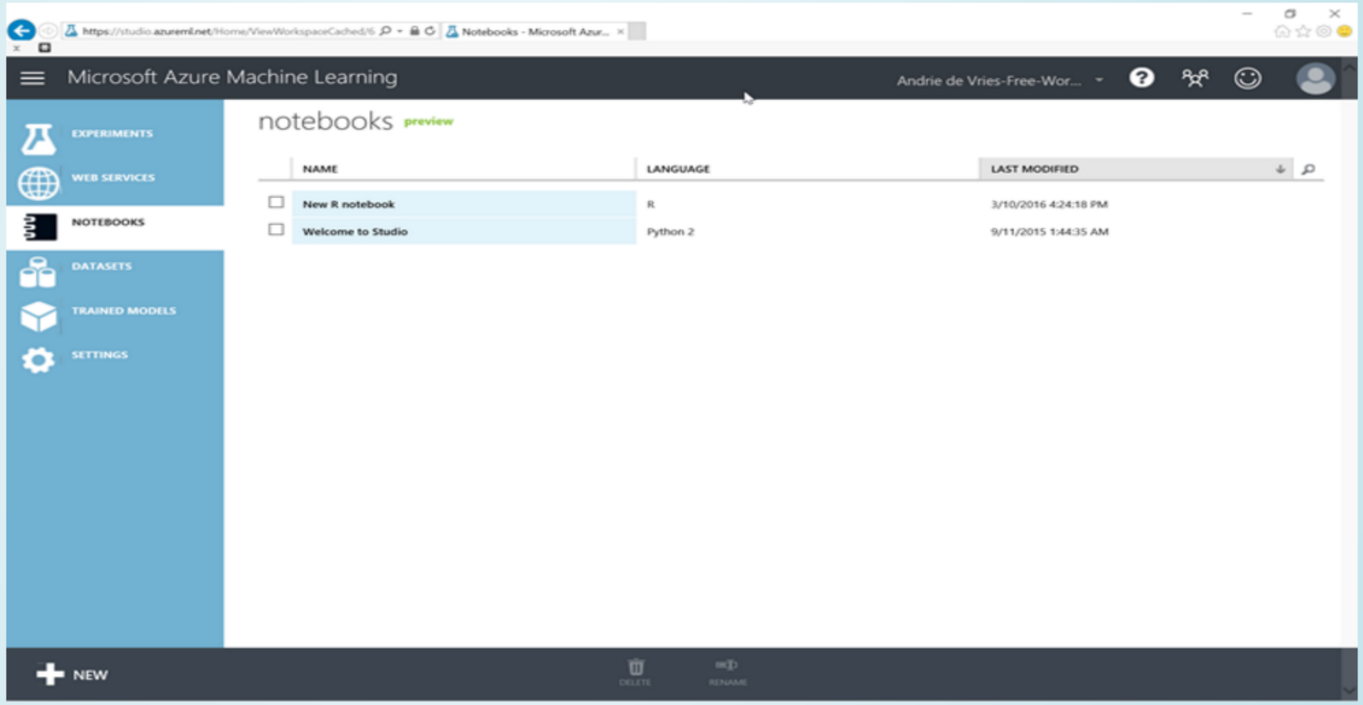
JUPYTER: RUNNING NOTEBOOKS

- Native OS Python distribution + Pip, or Anaconda
- JupyterHub, multi-user server
- Under Docker [e.g. *docker-stacks* images]
- Integrated into data science Cloud Hosting or plain IaaS:
  - Azure ML Studio
  - Google Cloud DataLab Beta
  - IBM Data Scientist Workbench
- Cloud hosted (ephemereal)
  - **tryjupyter.org** [uses *docker-demo* image]
  - **Binder** (https://mybinder.org), an example github repo

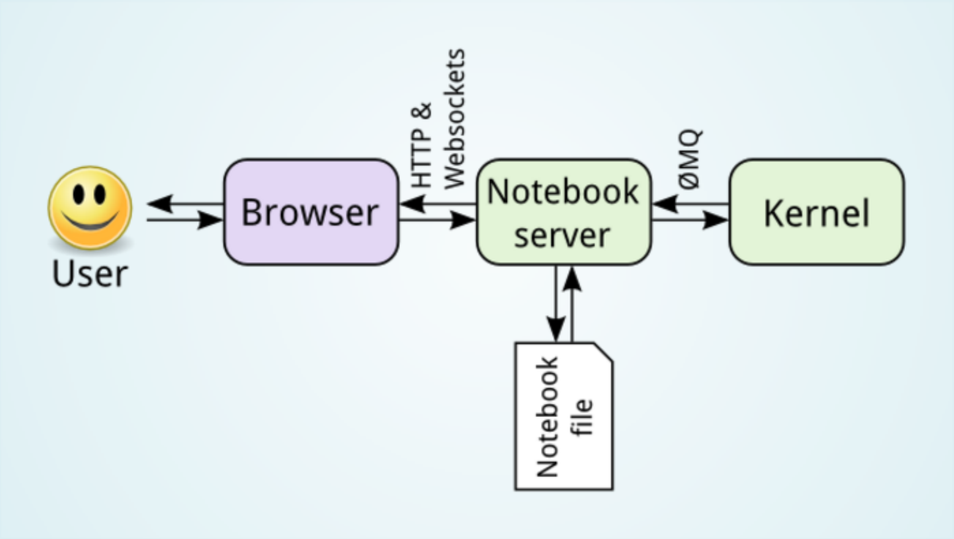
JUPYTER & AZURE ML STUDIO

Jupyter integration in Azure ML Studio

- R
- Python



KERNELS, WIDGETS & EXTENSIONS



~50 KERNELS

Kernels are execution environments - typically a language [Kernels Page]

IJulia	**IRKernel**	**IPython**	IRuby	IGo
IScilab	IMatlab	Hy	Clojure	**Bash**
Lua	Mochi	IErlang	**Spark**	**MetaKernel**
IVisual VP	Brainfuck	Brython	IOCaml	**MetaKernel_Bash**
IScala	IMathics	IAldor	Calico Pro	Calysto Prolog
IForth	**IPerl**	**IPerl6**	IPHP	IOctave
KDB+/Q Ker	ICryptol	**C++ (cling)**	**Xonsh**	IJavascript
cl-jupyter	IHaskell	IElixir	Java 9	Calysto LC
Redis	jove	Prolog	IFSharp	Calysto Scheme

WIDGETS

Widgets are eventful python objects with a representation in the browser.
[documentation]

Provided widgets include:

- IntSlider, FloatSlider, FloatProgress
- Buttons, Checkboxes, Radio buttons
  Dropdown menus

EXTENSIONS

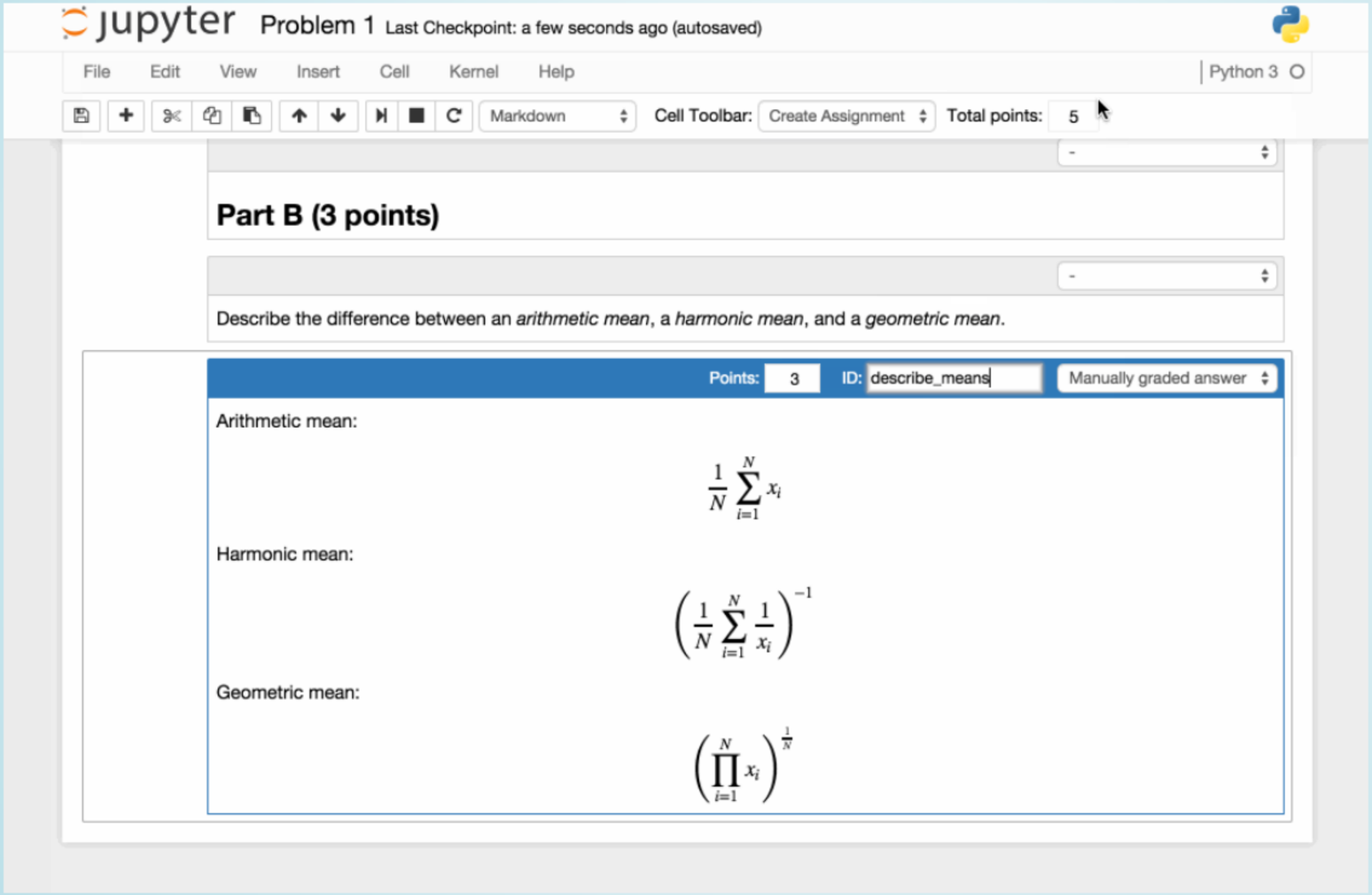
- Collection [github]: ipython-contrib/IPython-notebook-extensions
- Installed to http://localhost:8888/nbextensions/

Many extensions available, including:

- **RISE - these slides are running under Jupyter**
- nbgrader - creation/grading of classroom assignments

Generally installable via pip or from github repo

NBGRADER


jupyter Problem 1 Last Checkpoint: a few seconds ago (autosaved)
File Edit View Insert Cell Kernel Help
Python 3
Markdown
Cell Toolbar: Create Assignment
Total points: 5
Part B (3 points)
Describe the difference between an arithmetic mean, a harmonic mean, and a geometric mean.
Points: 3
ID: describe_means
Manually graded answer
Arithmetic mean:
$\frac{1}{N}\sum_{i=1}^{N} x_i$
Harmonic mean:
$\left(\frac{1}{N}\sum_{i=1}^{N} \frac{1}{x_i}\right)^{-1}$
Geometric mean:
$\left(\prod_{i=1}^{N} x_i\right)^{\frac{1}{N}}$

THE ECOSYSTEM & FUTURE PROJECTS

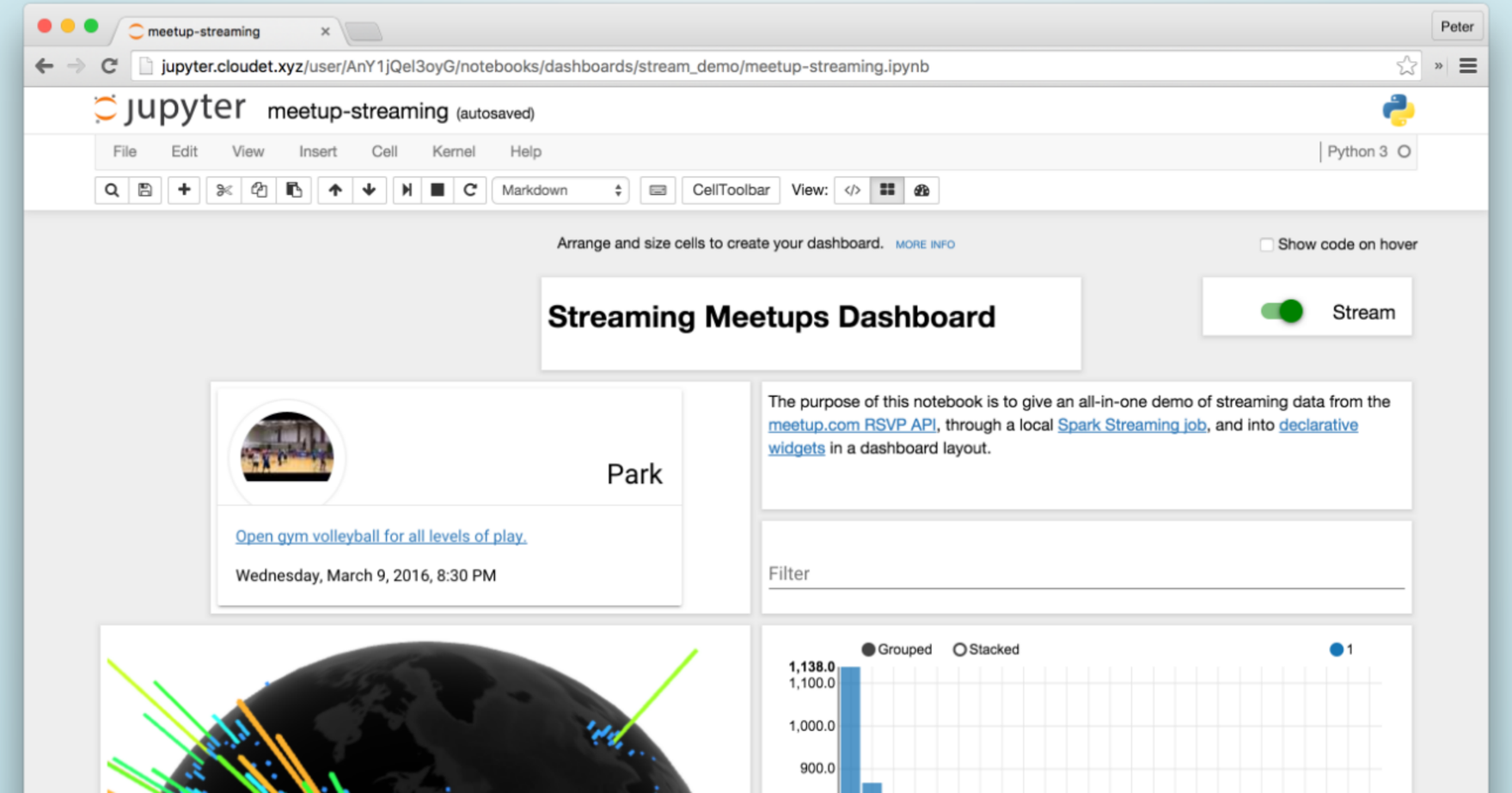
JUPYTER INCUBATOR PROJECTS

(https://github.com/jupyter-incubator) proposals

sparkmagic	Jupyter magics and kernels for working with remote Spark clusters
declarativewidgets	Declare Widgets in HTML
dashboards	Create Dashboards from Notebooks
contentmanagement	Extensions for search, notebook modules/cookbooks, ToC, bundlers vid
kernel_gateway	Support different protocols to Jupyter server, e.g. non-nb web clients, u-services, cluster

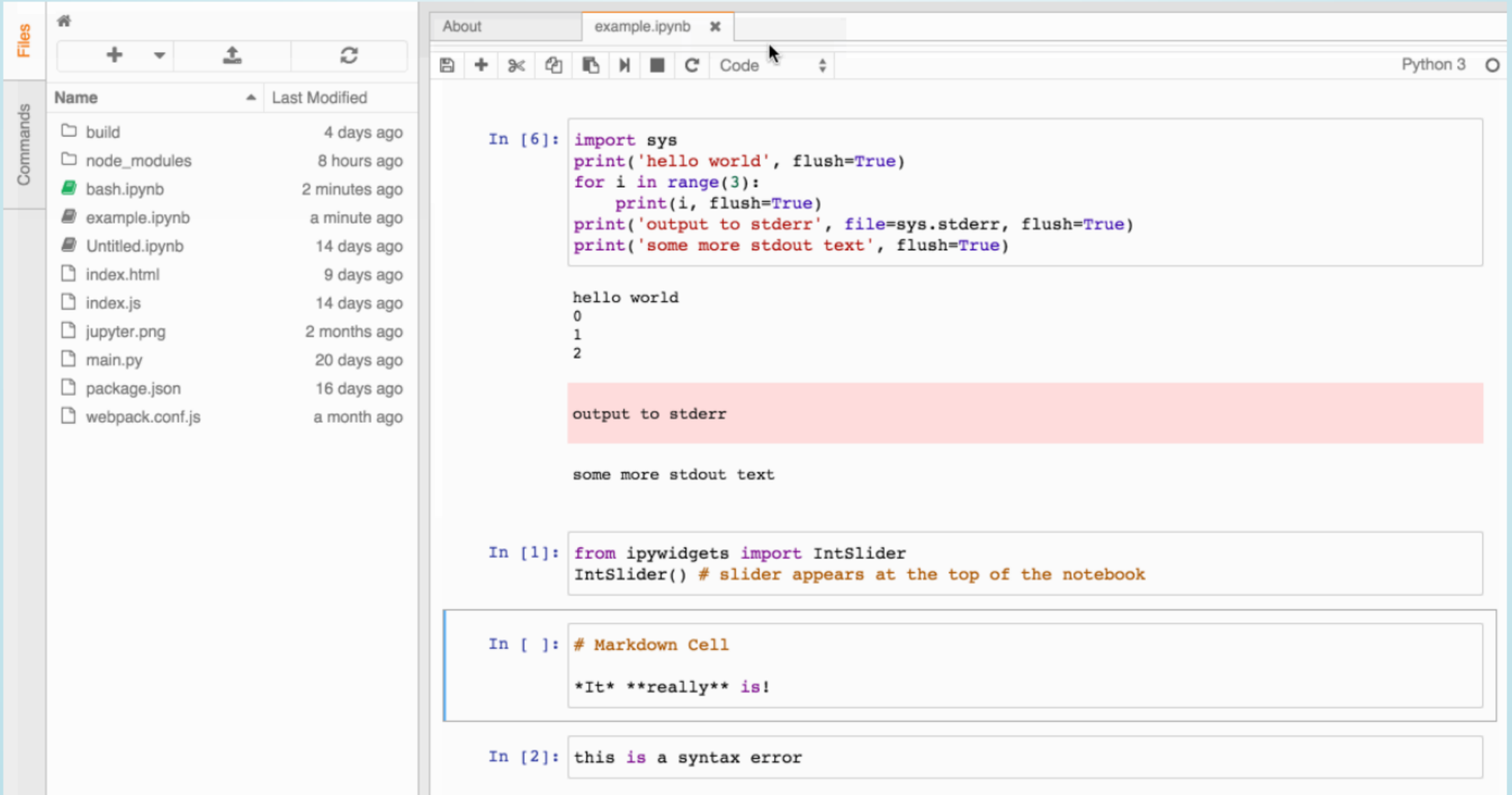
INCUBATOR: JUPYTER DASHBOARDS

(https://github.com/jupyter-incubator/dashboards) - alternative layouts



JUPYTERLAB

(https://github.com/jupyter/jupyterlab/) - the future interface



EXTERNAL JUPYTER PROJECTS

There are many external projects such as *Beaker*, *Hydrogen (ATOM)*, *EIN (Emacs)*, *Rodeo*, *SageMathCloud* integrating Jupyter.

Publishers are also turning to Jupyter for books, blogs, reports, theses sometimes with live code examples.

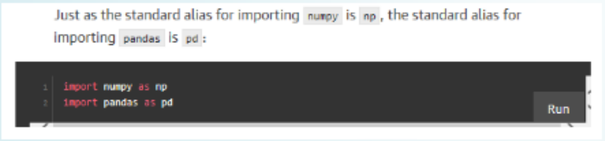
- e.g. **Thebe** (O'Reilly)
- Nature, Scientific American Magazines is a simplified notebook interface

Educators

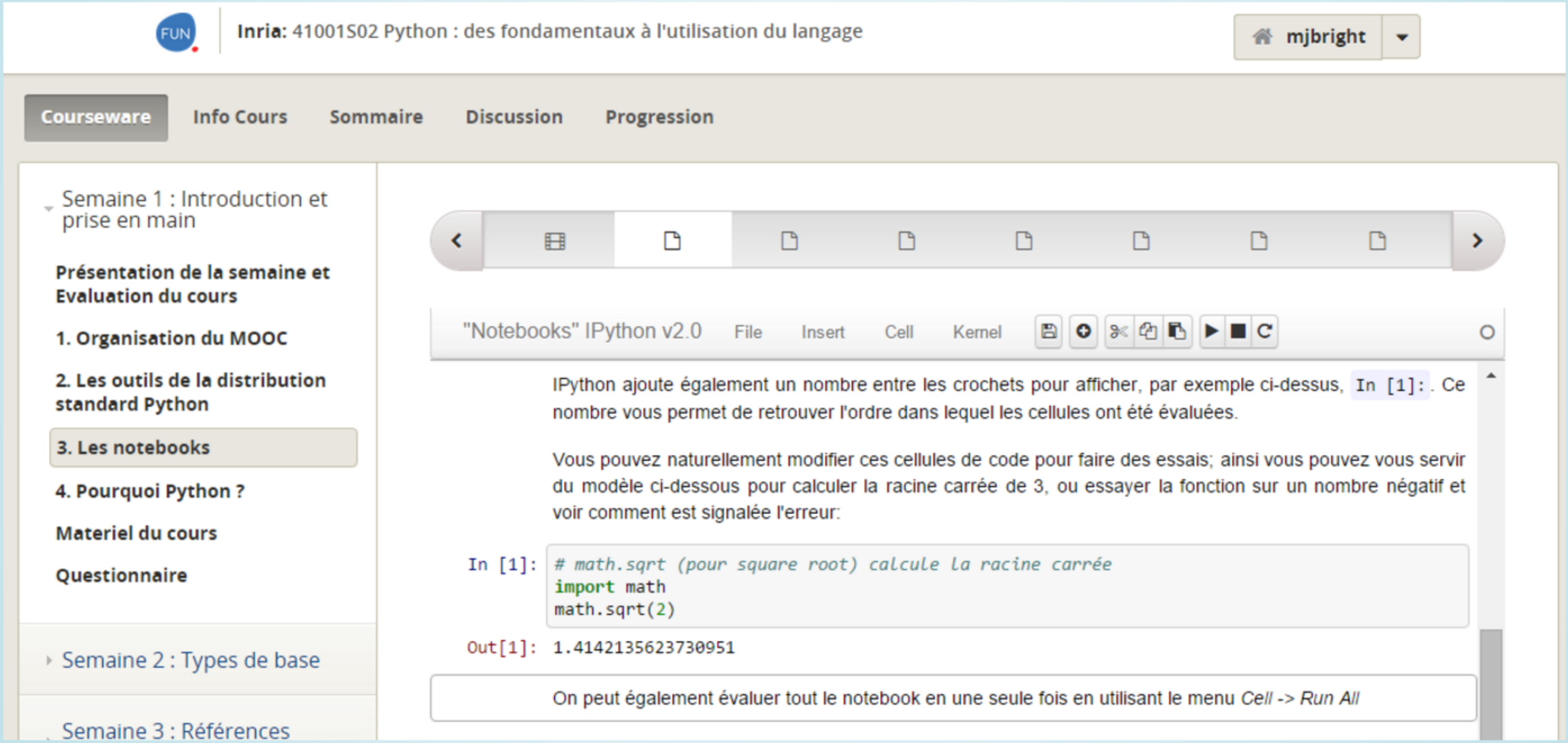
- tutorials, assignments, presentations, documenting
- MOOCs - online education:
  - notebook-based (**Edx/Apache Spark**)

OREILLY BLOG ARTICLE - USING THEBE

This blog post containers modifiable, runnable code cells with a **RUN** button as shown below



MOOCS, E.G. F.U.N., GWU, EDX/SPARK...



JUPYTER FOR EVERYTHING ELSE

- Use of web technologies: mix-in HTML, CSS, js, SVG ...
- Use of bash kernel for command-line work
- Supplement command-line tools with graphics
- Create interactive presentations (thanks *RISE* extension !)
- Publish "live blog posts"
- Creating status reports from notebooks using nbconvert

EVERYTHING ELSE: WEB TECHNOLOGIES

- HTML/JavaScript/css experimentation
  - Many HTML, CSS, JS capabilities if you **proceed with care**
  - d3.js animations if
    - Need more interactivity
    - Prototyping a D3 project
    - Reusing existing D3 e.g. from http://bl.orcks.org
- SVG
- Example notebook

Select Theme: mysky

Everything Else: web technologies

- HTML/JavaScript/css experimentation
  - Many HTML, CSS, JS capabilities if you **proceed with care**
  - d3.js animations if
    - Need more interactivity
    - Prototyping a D3 project
    - Reusing existing D3 e.g. from http://bl.orcks.org
- SVG
- Example notebook

Select Theme: mybloodredsky

Everything Else: command-line

Two bash kernels are available for Jupyter bash_kernel and calysto/metakernel_bash

Calysto Metakernels

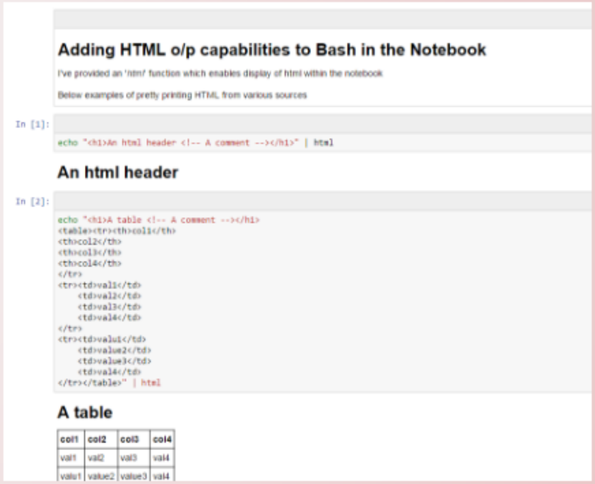
- easy to fix for Windows/Cygwin
- Provides magics
- Family of metakernels
- Under development

Everything Else: command-line

Why?

Inspired by the notebook as an educational tool
I wanted to use it for command-line tasks

- **Docker demos / labs in Jupyter**
- Why not Bash as 1st-class citizen with magics, graphics ?
- **Example notebook**
- Write command-line tutorials, cheat sheets
  in an easy to maintain "live notebook" (runnable) format.



Adding HTML o/p capabilities to Bash in the Notebook

I've provided an 'html' function which enables display of html within the notebook

Below examples of pretty printing HTML from various sources

In [1]:

```
echo "<h1>An html header <!-- A comment --></h1>" | html
```

An html header

In [2]:

```
echo "<h1>A table <!-- A comment --></h1>
<table><tr><th>col1</th>
<th>col2</th>
<th>col3</th>
<th>col4</th>
</tr>
<tr><td>val1</td>
    <td>val2</td>
    <td>val3</td>
    <td>val4</td>
</tr>
<tr><td>value1</td>
    <td>value2</td>
    <td>value3</td>
    <td>val4</td>
</tr></table>" | html
```

A table

col1	col2	col3	col4
val1	val2	val3	val4
value1	value2	value3	val4

Everything Else: Binder "live notebooks"

Everything Else: "live notebooks" as runnable tutorials

We can create live tutorials online on Binder.

The notebook server can be launched by clicking on

the binder icon in a github repo

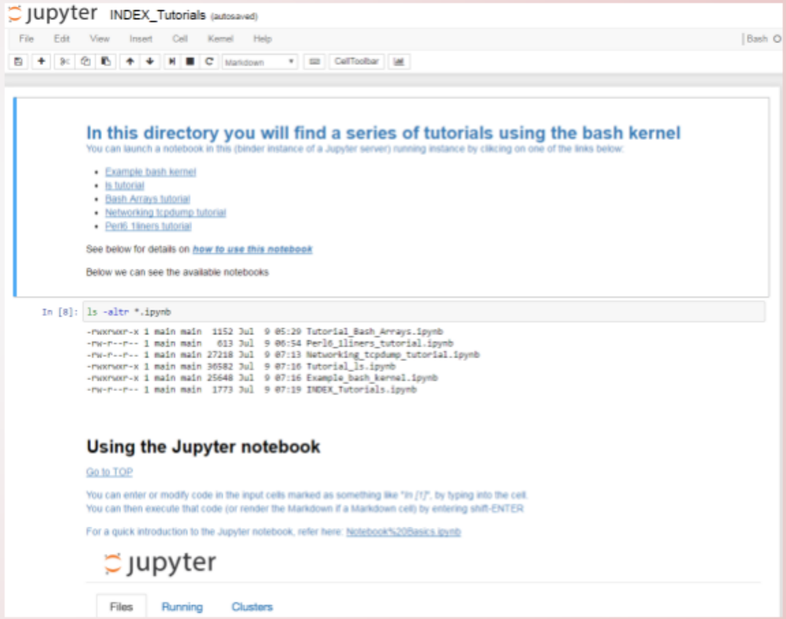
launch binder

I wrote a Blog post with link to "**live notebook**" of bash tutorials

Everything Else: Binder command-line "live notebooks"

In that github **repo** I created an **INDEX notebook**

other notebooks in the same repo



Everything Else: Slideshows

This **slideshow** is made using Jupyter with the RISE extensions

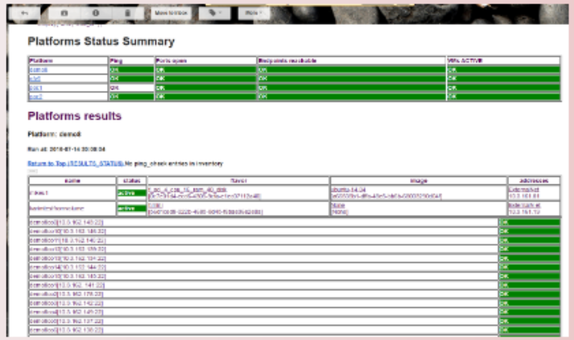
RISE adds special "Slide Type" menu options to each cell to specify one of

- Slide
- Sub-Slide
- Fragment
- Notes
- Skip

Everything Else: Automated e-mail status reports

Using nbconvert we can execute notebooks from the command-line

- Automatically run the notebook under cron
- **nbconvert** notebook result to html
- send html report as e-mail via cron
- Capabilities created incrementally in notebook,
  code migrated to modules to reduce notebook (report) code

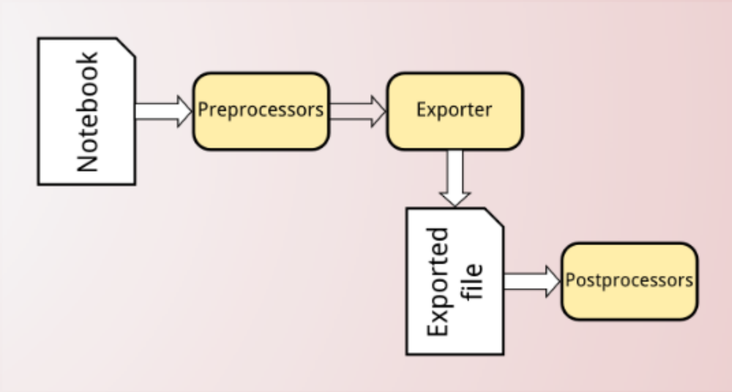


Everything Else: Automated e-mail status reports (nbconvert)

We can use nbconvert to automatically run the notebook under cron and send the results by e-mail

```
nbconvert --execute --template
basic --to html Monitoring.ipynb
```

- --execute: execute the notebook
- --template: specify the o/p template
- --to: specify o/p format
- Input notebook



Everything Else: Coming up ...

- Experiment with JupyterHub, nbgrader, Binder
  - I'd like to reimplement some labs as a set of graded assignments
- More Metakernel_Bash experiments
- Make pull requests to Metakernel_bash
- Propose this stuff outside of the Python community
- Xonsh_kernel
  - Take advantage of new Python / unix-like shell
- CLing C++ interpreter kernel

Questions ?

mjbright @mjbright mjbrightfr AT gmail mjbright.github.io

The End ...

References: IPython / Jupyter Books

Learning IPython for Interactive Computing & Data Visualization

Cyrille Rossant

Introductory usage

IPython Interactive Computing & Visualization Cookbook

Cyrille Rossant

Advanced usage

The Jupyter GitBook

Extension writing

Documentation on ReadTheDocs

Extension writing