The Jupyter talk talk

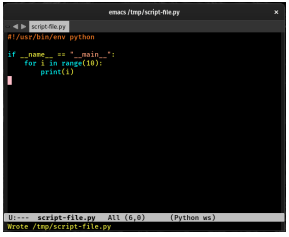
Jim Pivarski

Princeton University – IRIS-HEP

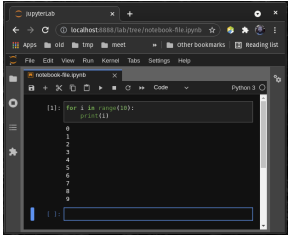
June 2, 2021

Three fundamental modes of computation

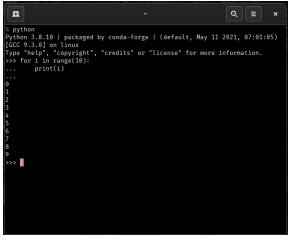
Batch



Notebook



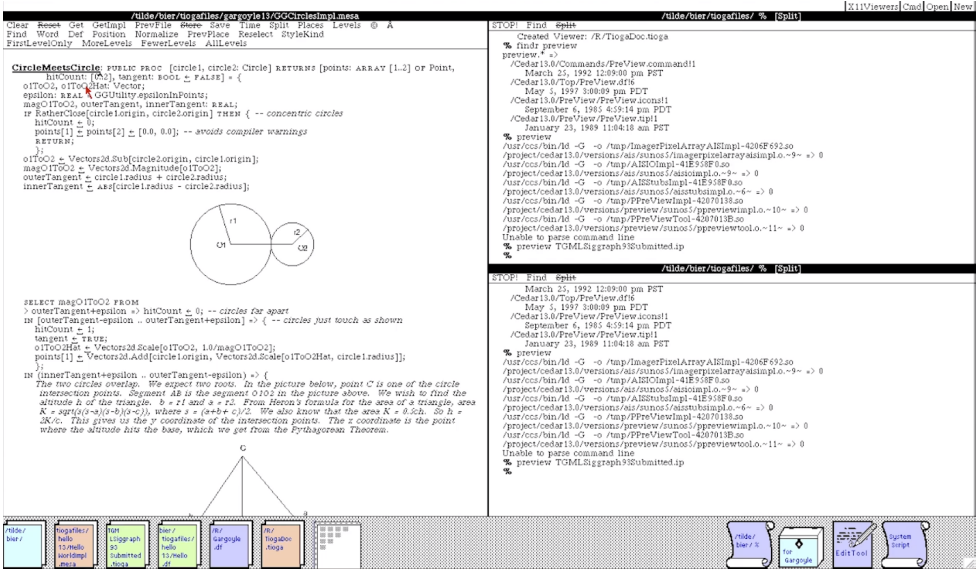
Terminal



Permanent

Ephemeral

“Interactive computational notebook” is not a new idea

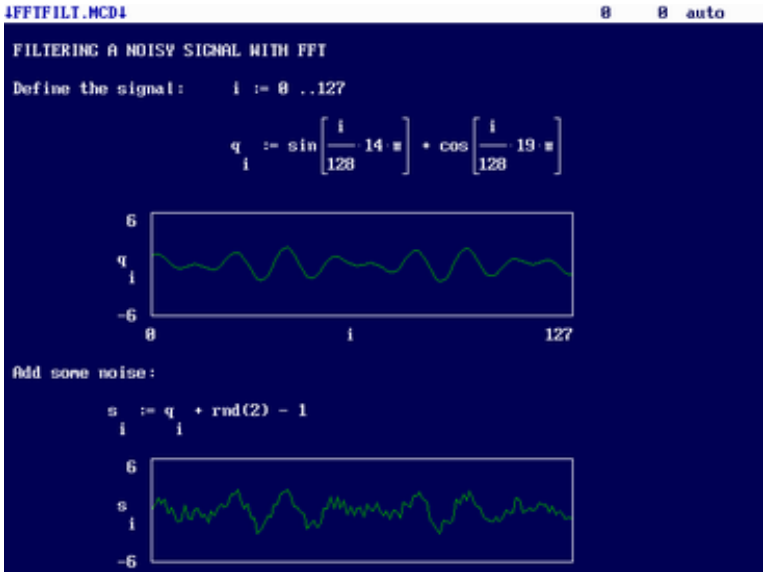


1982

Xerox’s Tioga editor for the Cedar language

(Xerox-PARC invented everything GUI-related)

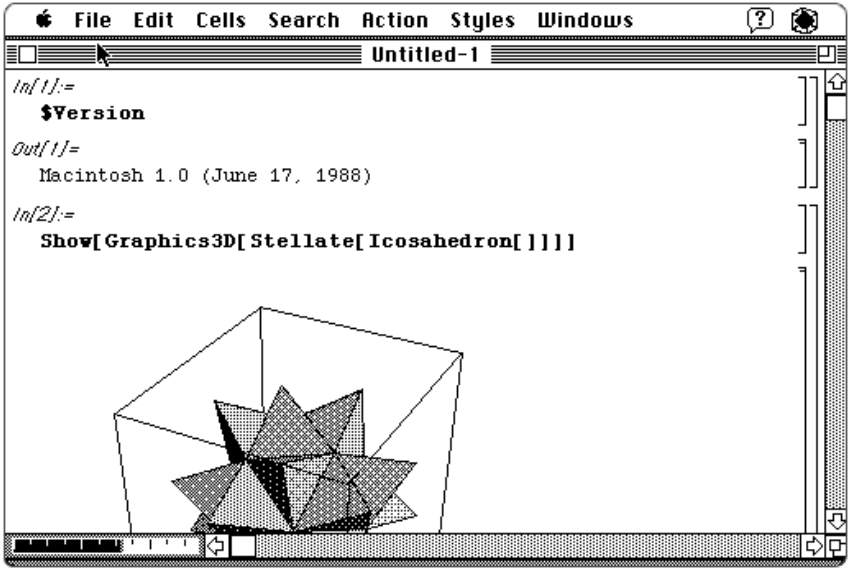
“Interactive computational notebook” is not a new idea



1986

Mathcad, first computational notebook to run on PCs

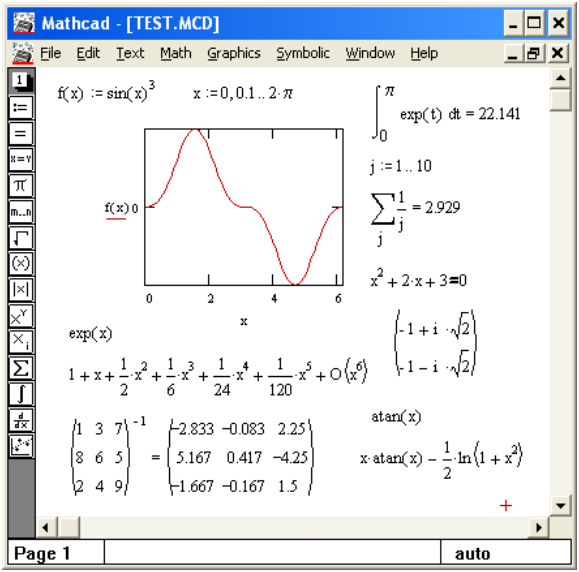
“Interactive computational notebook” is not a new idea



1987

Mathematica

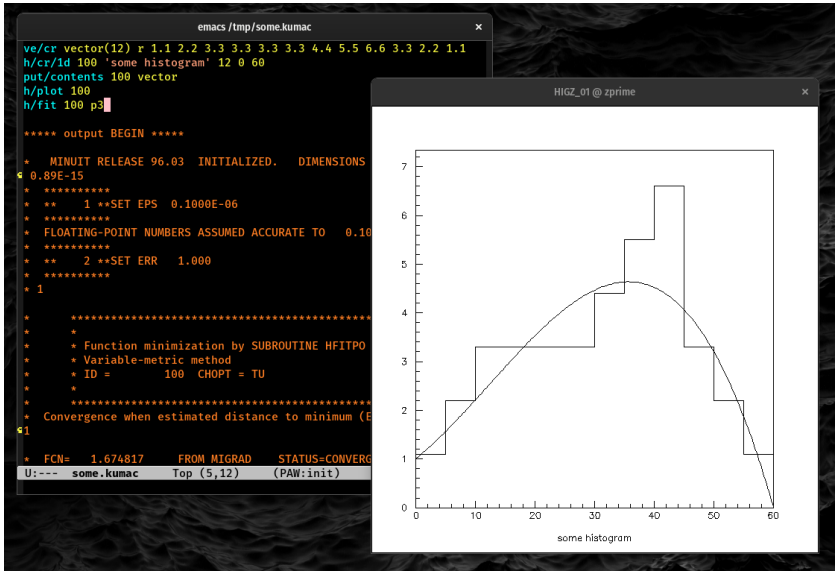
“Interactive computational notebook” is not a new idea



1992

Mathcad again (on Windows)

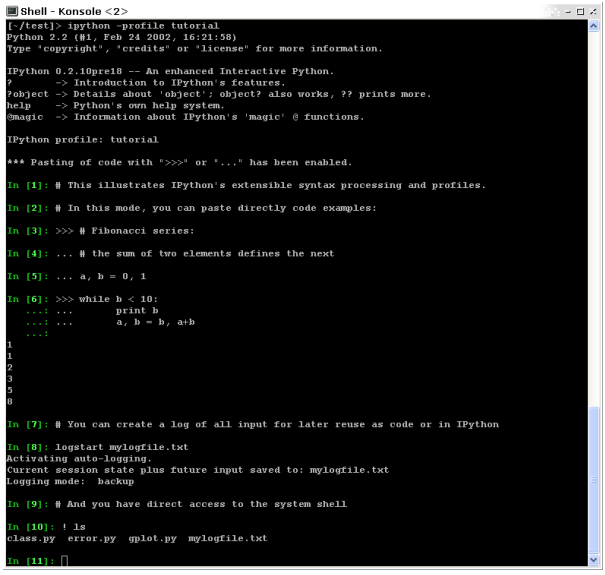
“Interactive computational notebook” is not a new idea



2000

PAW-mode for Emacs (number of users = 1)

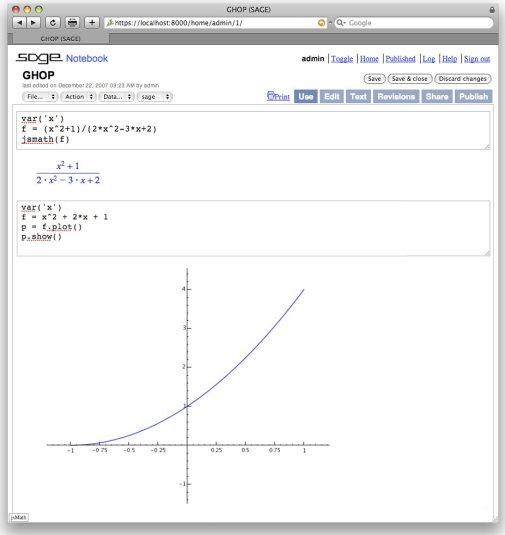
“Interactive computational notebook” is not a new idea



2002

iPython: cell-based Python in the terminal

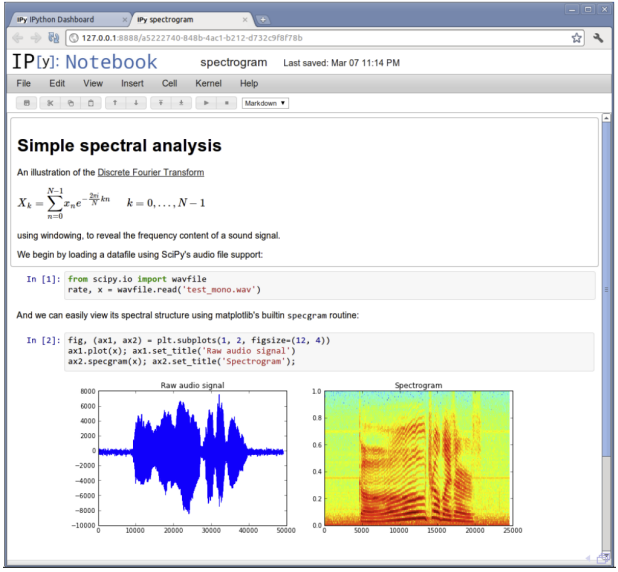
“Interactive computational notebook” is not a new idea



2005

SAGE: cell-based Python in a web browser

“Interactive computational notebook” is not a new idea

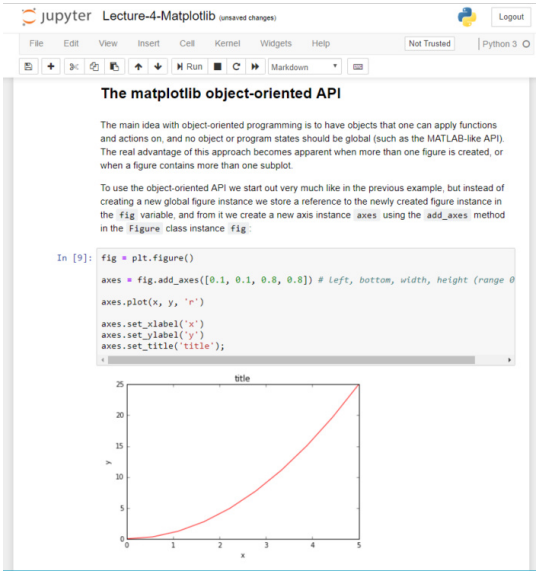


2012

iPython-notebook: cell-based Python in a web browser

But popular this time!

“Interactive computational notebook” is not a new idea

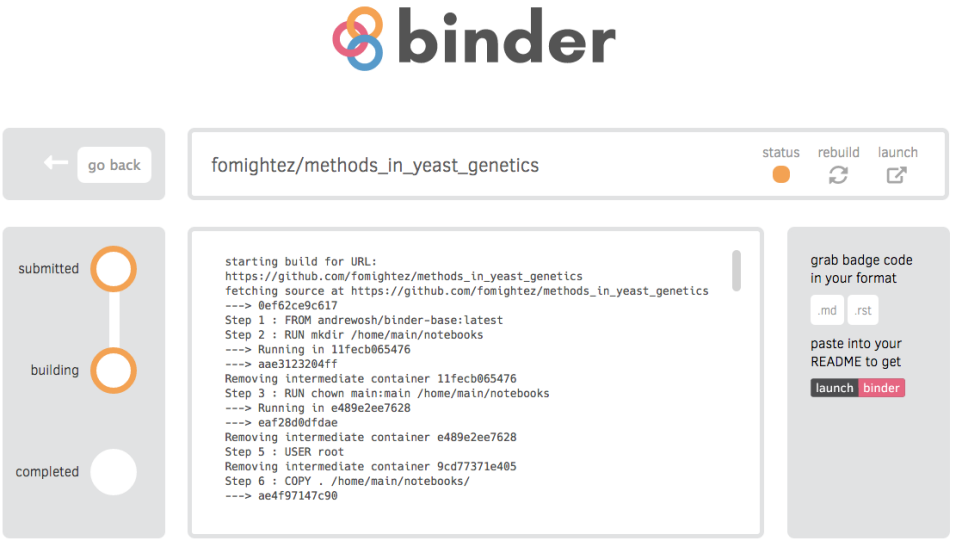


2015

Jupyter: rebranding to emphasize multi-language support

Julia + **Py**thon + **R**

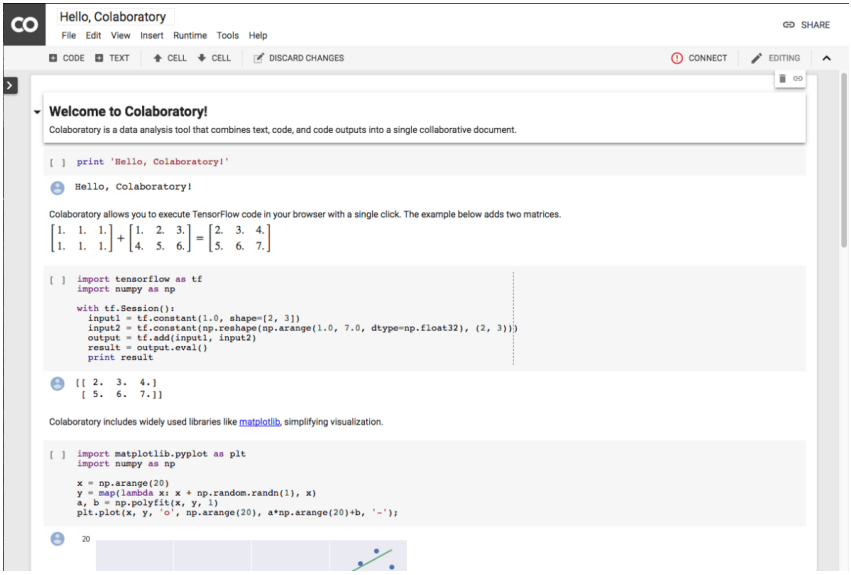
“Interactive computational notebook” is not a new idea



2016

Binder:
Jupyter
notebooks
on the cloud

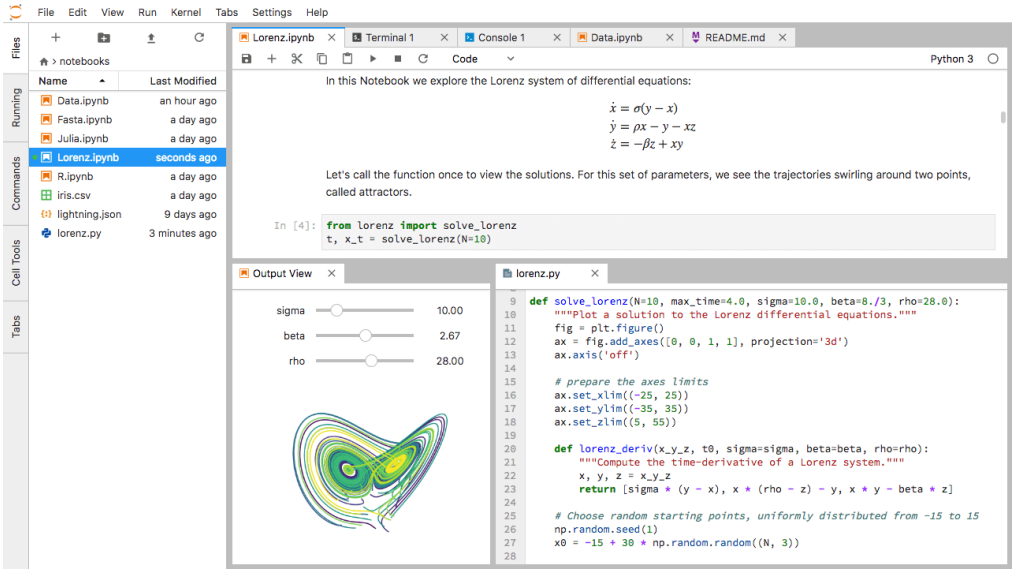
“Interactive computational notebook” is not a new idea



2018

Google Colab: Jupyterish notebooks on the cloud

“Interactive computational notebook” is not a new idea

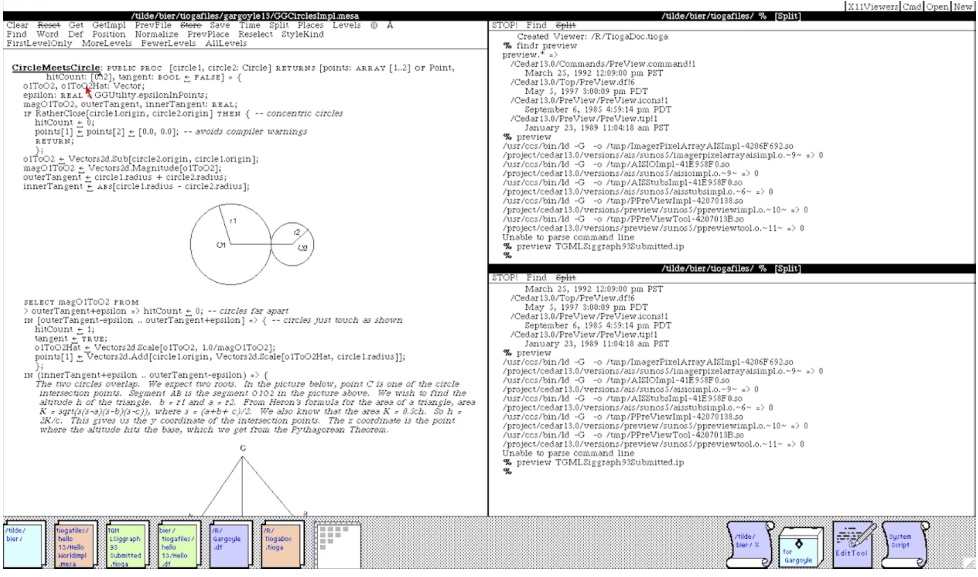


2018

JupyterLab: fully featured editor environment

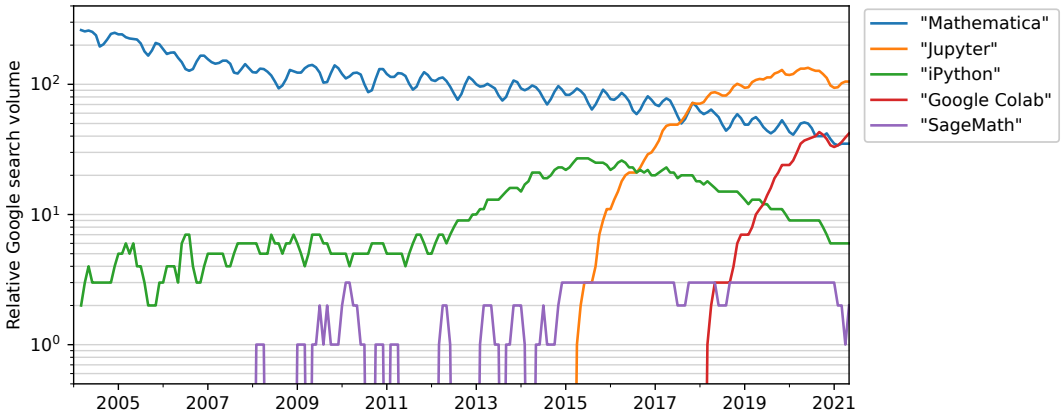
Now it’s starting to look like Tioga. . .

“Interactive computational notebook” is not a new idea



a bit. . .

Jupyter is now mainstream



Reasons for using notebooks

- Quick prototyping
- Iterative investigation, such as data analysis
- Literate programming: interweaving a program and text
- Publishing a widget that other people can run
- Giving a talk/teaching a class that others join *in real time*

Reasons for using notebooks

- Quick prototyping
- Iterative investigation, such as data analysis
- Literate programming: interweaving a program and text
- Publishing a widget that other people can run
- Giving a talk/teaching a class that others join *in real time*

what this talk is about

The problems

1. “It’s too small!”
2. “You’re scrolling too fast!”
3. “Too much code in the cells!”
4. “It doesn’t work for me!”