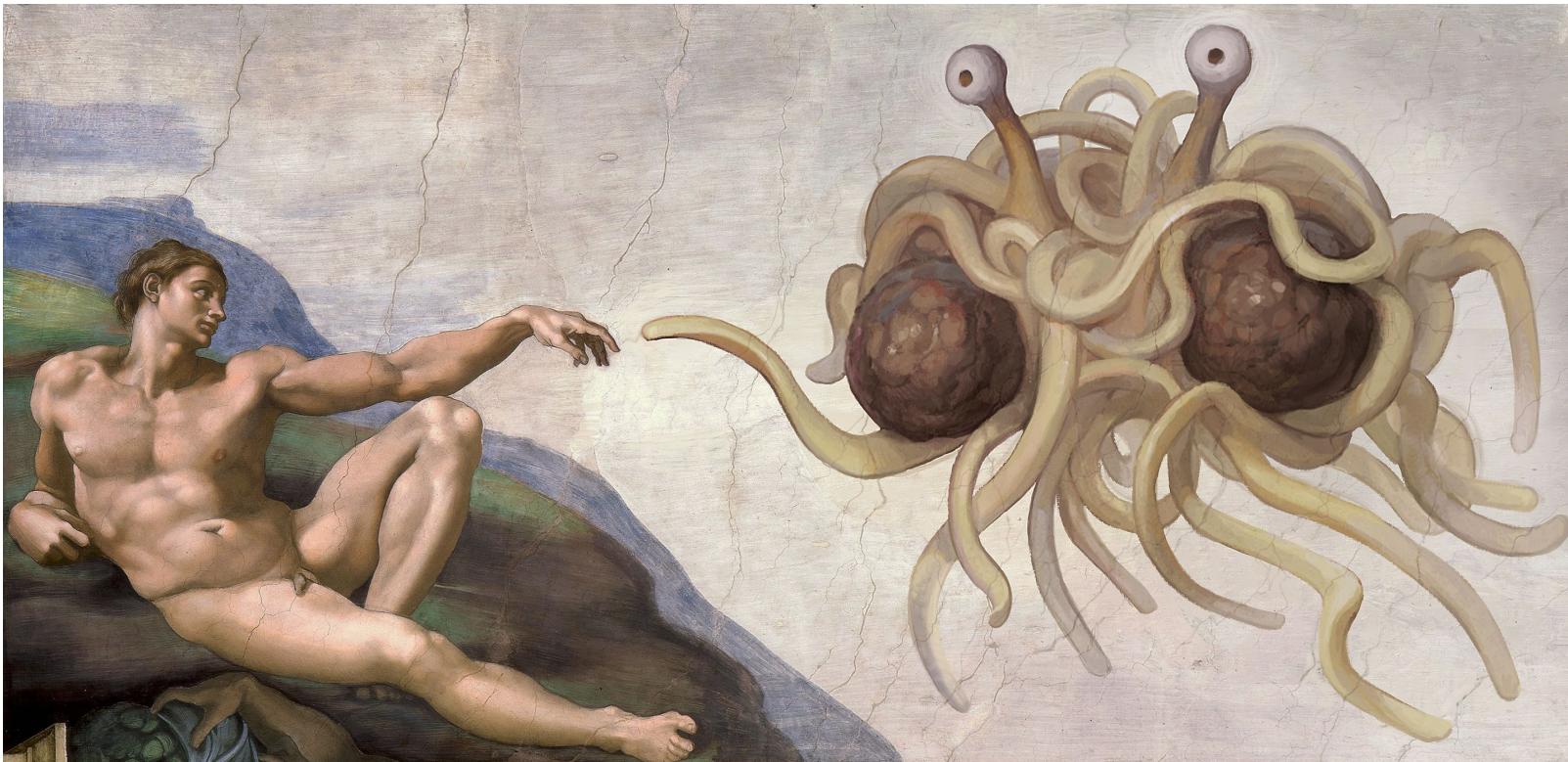


Interfacing with users



Three talks about our efforts

1. Interfacing with users ← (this talk)
(general discussion ~~on good and evil~~ on where things are going)
2. Community reference reconstruction as an entry point for users
(with live demo and description)
3. Community reference reconstruction for developers
(with in-depth details on backends, frontends and their connection)

Things to consider

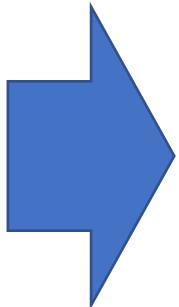
DISCLAIMER:

- Some things in this talk are **subjective** and **opinionated**. I want this talk to be a discussion. You don't agree – excellent, lets discuss it.
- **We, us, our** = HEP & NP developers in general (most of the time)

ASSUMPTIONS:

- Our software does physics
- Our software is technically OK
- Are users happy with our software?

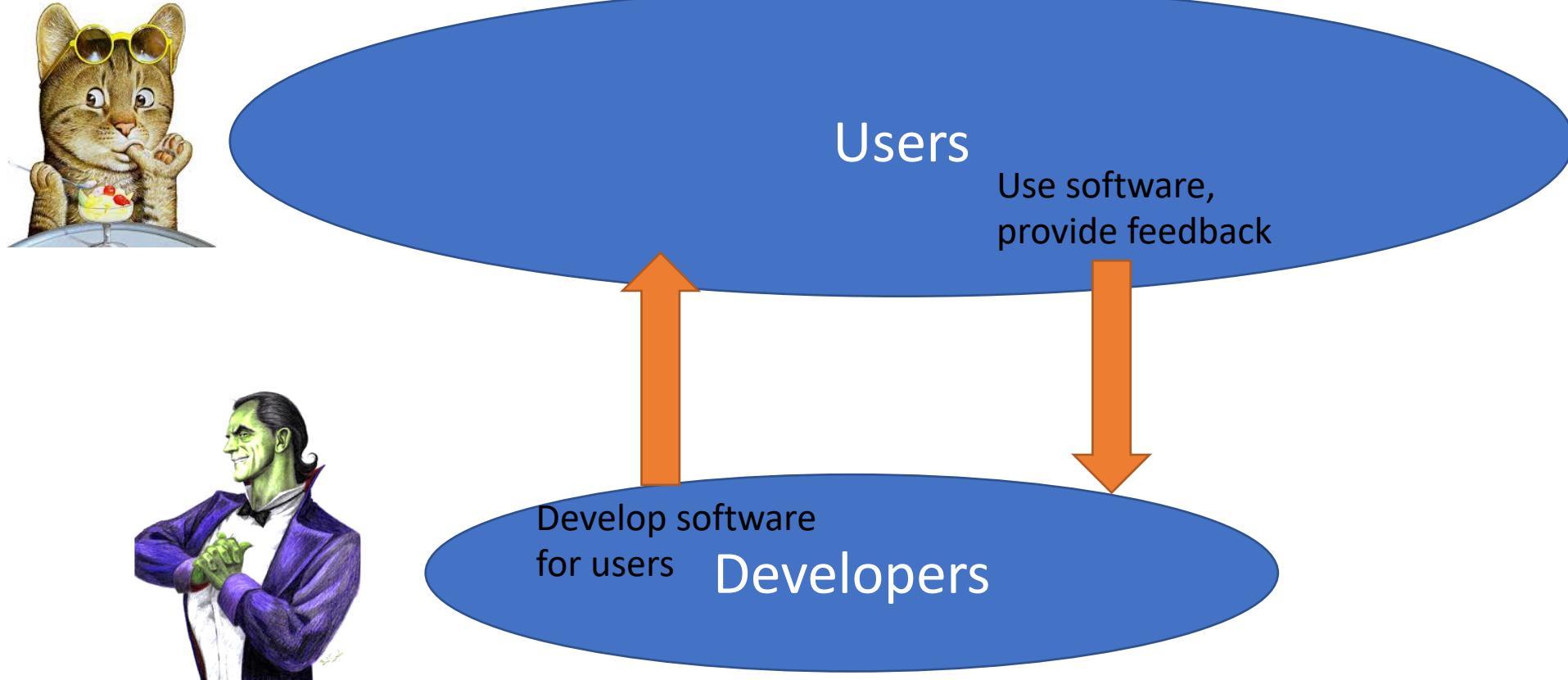
HOW 2019 impression



Users are not happy with
our (*HEP&NP*) software

(especially in analysis and
reconstruction part)

Usual view on users vs developers



My vision on users and developers

NO EXPERIENCE
Novice

HIGHLY
EXPERIENCED
Experts

- Users often are very experienced developers
- Developers are users
- Developers often are not very experienced



Lets start with ROOT

The screenshot shows the official website for the ROOT Data Analysis Framework. The header features a blue banner with the ROOT logo (a stylized 'R' inside a circle) and the text "ROOT Data Analysis Framework". A "Google Custom Search" bar is also present. Below the banner is a navigation menu with links: Download, Documentation, News, Support, About, Development, and Contribute. The main content area includes four large icons with labels: "Getting Started" (play button), "Reference Guide" (book), "Forum" (speech bubble), and "Gallery" (camera). To the left, a section titled "ROOT is ..." describes the software as a modular scientific toolkit for big data processing, statistical analysis, visualisation, and storage, primarily written in C++ with Python and R integration. It includes links to examples and browser usage. At the bottom left is a "Download ROOT" button with the logo, and at the bottom right is a "Read More ..." link. A large image on the right shows a computer monitor displaying ROOT's graphical user interface with various plots and data visualizations, including a "SCAN PYRAMIDS" watermark.

Google Custom Search

Download Documentation News Support About Development Contribute

Getting Started Reference Guide Forum Gallery

ROOT is ...

A modular scientific software toolkit. It provides all the functionalities needed to deal with big data processing, statistical analysis, visualisation and storage. It is mainly written in C++ but integrated with other languages such as Python and R.

Start from [examples](#) or [try it in your browser!](#)

 [Download ROOT](#)

or [Read More ...](#)

of important cavities

Previous Pause Next

Our major elephant is the software room

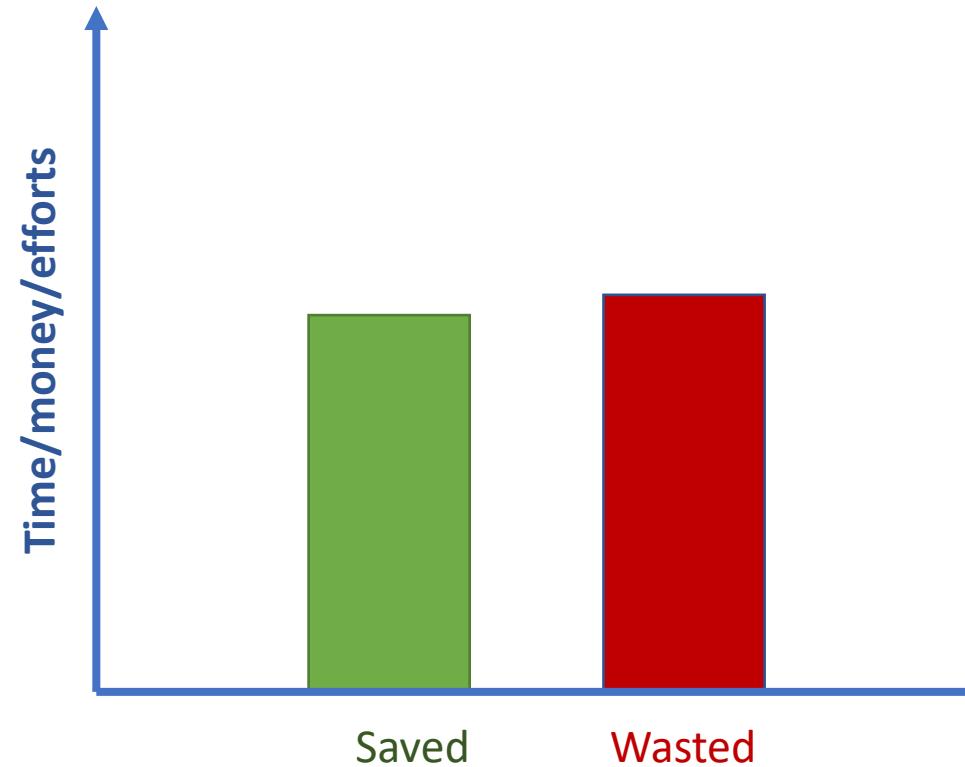
STUPID VS SOLID

- | | |
|---|---|
| <p>+++ • Singleton</p> <p>+++ • <i>Tight coupling</i></p> <p>+++ • Untestability</p> <p>--- • Premature Optimization</p> <p>+- • <i>Indescriptive Naming</i></p> <p>+ • <i>Duplication</i></p> | <p>-- • <i>Single responsibility</i></p> <p>- • <i>Open/closed</i></p> <p>+- • <i>Liskov substitution</i></p> <p>+- • <i>Interface segregation</i></p> <p>-- • <i>Dependency inversion</i></p> |
|---|---|



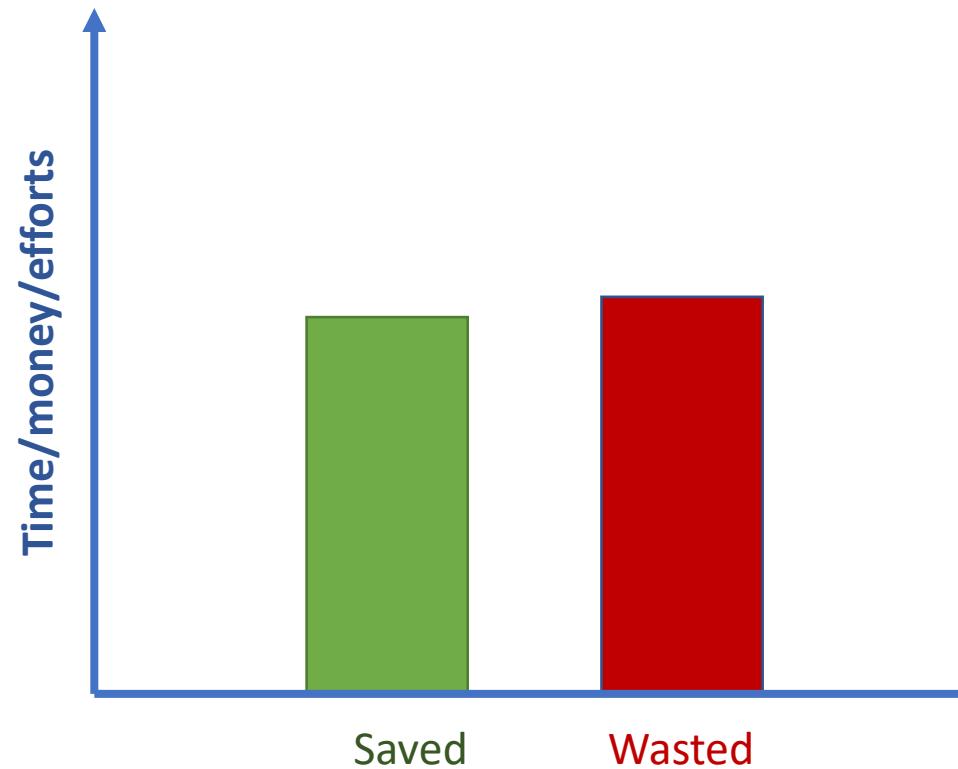
Unexperienced developers

Time & money saved on team of professional C++ developers



Time & money wasted by students, postdocs, professors and scientist on ROOT failures

Unexperienced developers



Unexperienced developers





ROOT 7 will save us!

- MUCH better API
- Professional team (love their users)
- Work being done on improving modularity

```
Fill a series of histograms
New interfaces
1. #include "ROOT/TAxis.h"
2. #include "ROOT/TFile.h"
3. #include "ROOT/TList.h"
4. #include "TreeReader.h"
5. #include "TreeReaderArray.h"
6. #include "Tree.h"
7. // Another function. Who knows what it does in a month from now.
8. void someOtherFunction();
9.
10. void fill(TTree* tree) {
11.     using namespace ROOT::Experimental;
12.     // Define the axes we will re-use when
13.     // filling the histograms. Much PI (GeV) > {0., 1., 10., 100.};
14.     TAxisConfig axispi("jet ET [GeV]", 10, 0, 1, 1000);
15.     TAxisConfig axiseta("jet ET versus tag value", 10, 0, 1, 1000);
16.     TAxisConfig axistag("tag value", 10, 0, 1, 1000);
17.
18.     // Create the histograms
19.     TH2F hmettag("muon ET versus tag value", axismet, axistag);
20.     TH2F hetaetag("jet ET versus tag value", axiseta, axistag);
21.
22.     // Set up reading from the TTree
23.     TreeReader reader(tree);
24.     TreeReaderArrayReader jetReader, tagReader, "jet.ET";
25.     TreeReaderArrayReader metReader, "jet.tag";
26.     TreeReaderArrayReader etaReader, "jet.eta";
27.     TreeReaderArrayReader tagReader, "tag.tag";
28.
29.     // Fill the histograms
30.     while (reader.Next()) {
31.         for (int ijet = 0; ijet < jetReader.size(); ++ijet) {
32.             hmettag.Fill(metReader[ijet], tagReader[ijet]);
33.             hetaetag.Fill(etaReader[ijet], tagReader[ijet]);
34.         }
35.     }
36.     someOtherFunction();
37.
38.     // Store the results. can create the file where really needed.
39.     TFileWriter file = TFile::Create("jetanalysis.root");
40.     file->Write("hMet_Jet", hmettag);
41.     file->Write("hEta_Jet", hetaetag);
42. }
```

ROOT 7 problem

It is for tomorrow while we need it **yesterday**



Can we live without CERN.ROOT?

NO

At least not today

Yes. We have trapped ourselves



- ROOT is dead. Long live the ROOT!

Num focus sponsored data science tools



Filter Projects

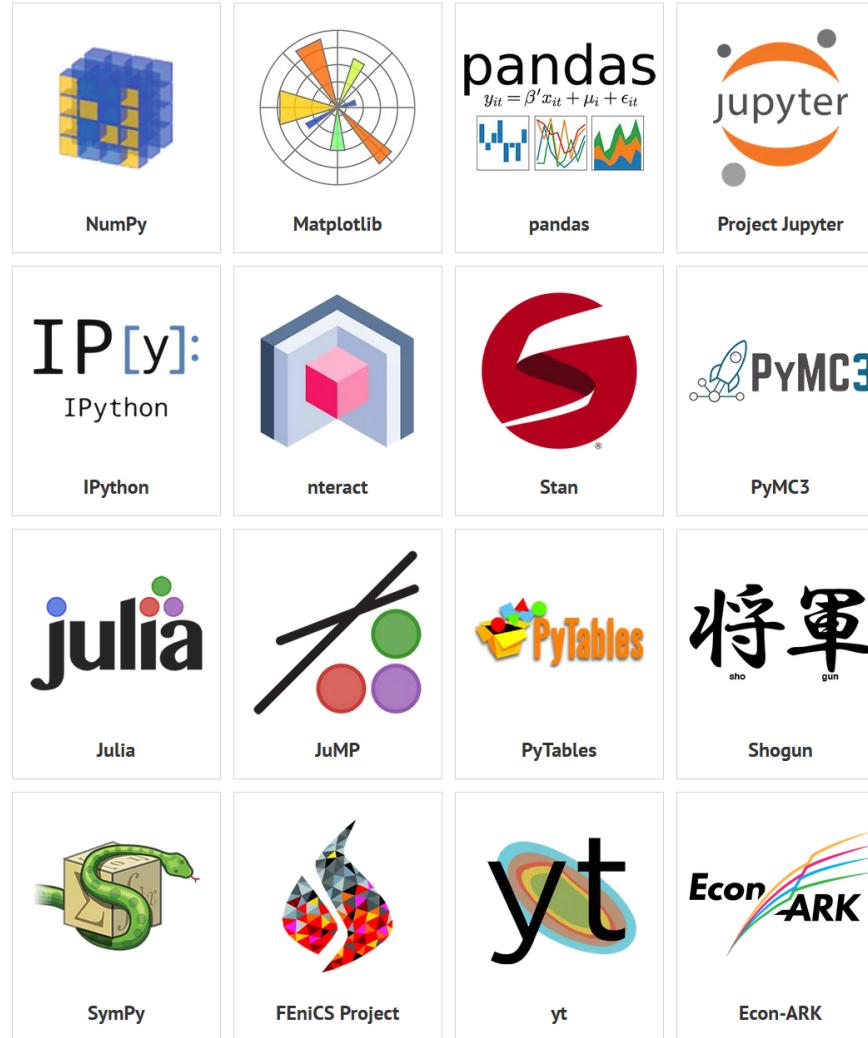
Show All Projects

Language

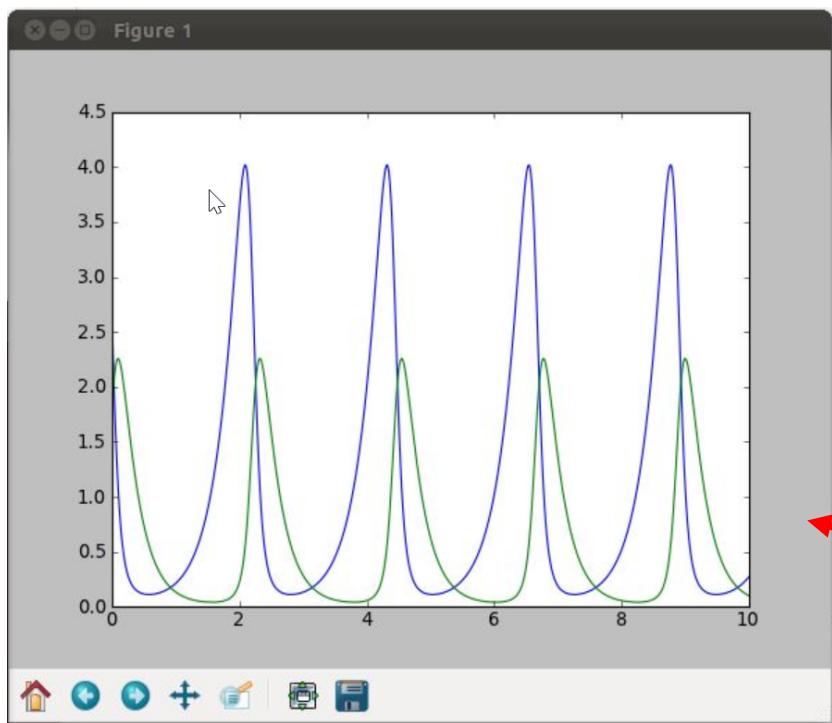
- Python
- R
- Julia
- JavaScript
- Other
- Multiple

Features

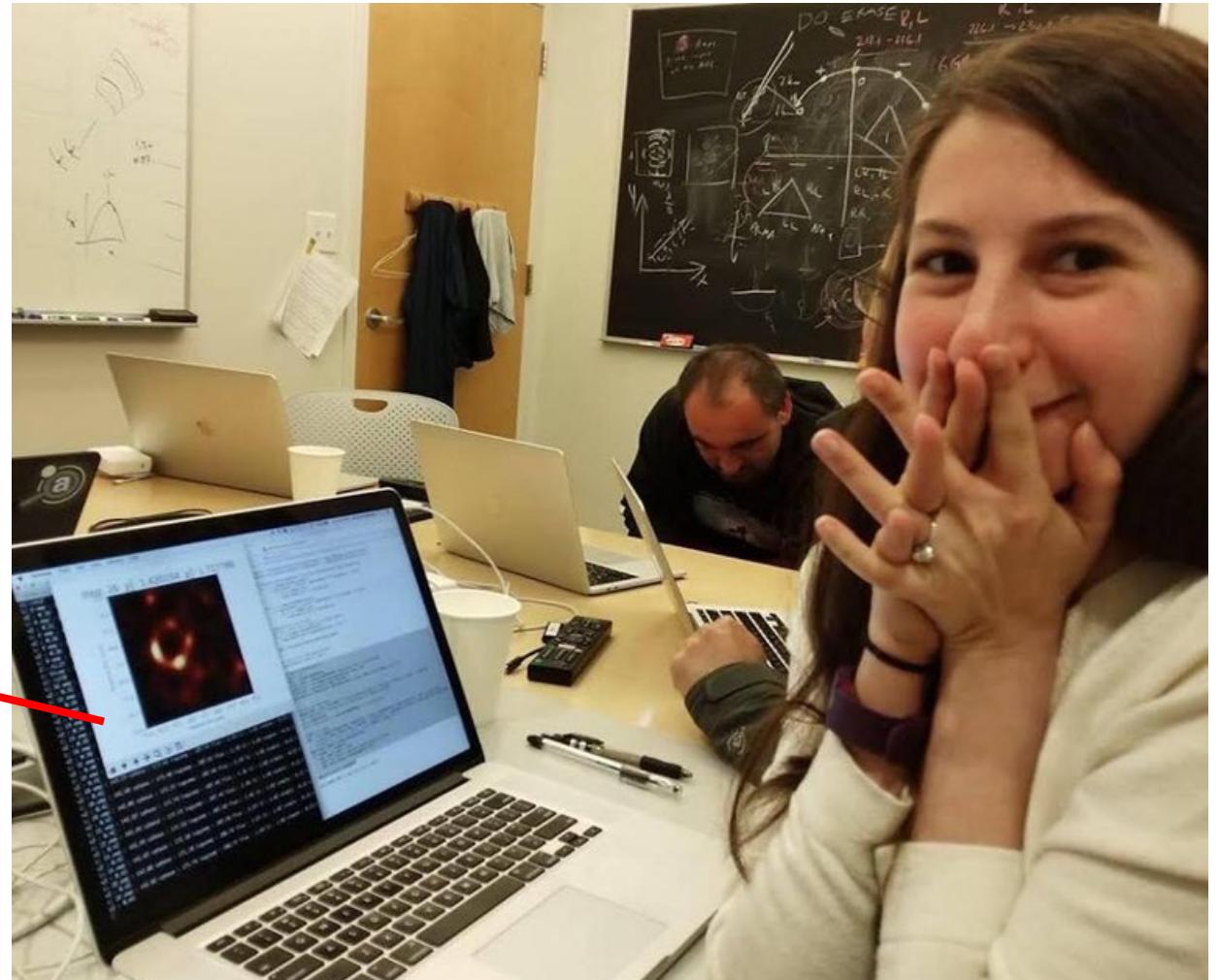
- Data Wrangling
- Modeling
- Visualization
- High Performance Computing
- Big Data
- Statistical Computing
- Numerical Computing
- Data Mining
- Text Processing
- Machine Learning
- Computing Language
- Subject Area Libraries
- Educational Outreach



Data science tools

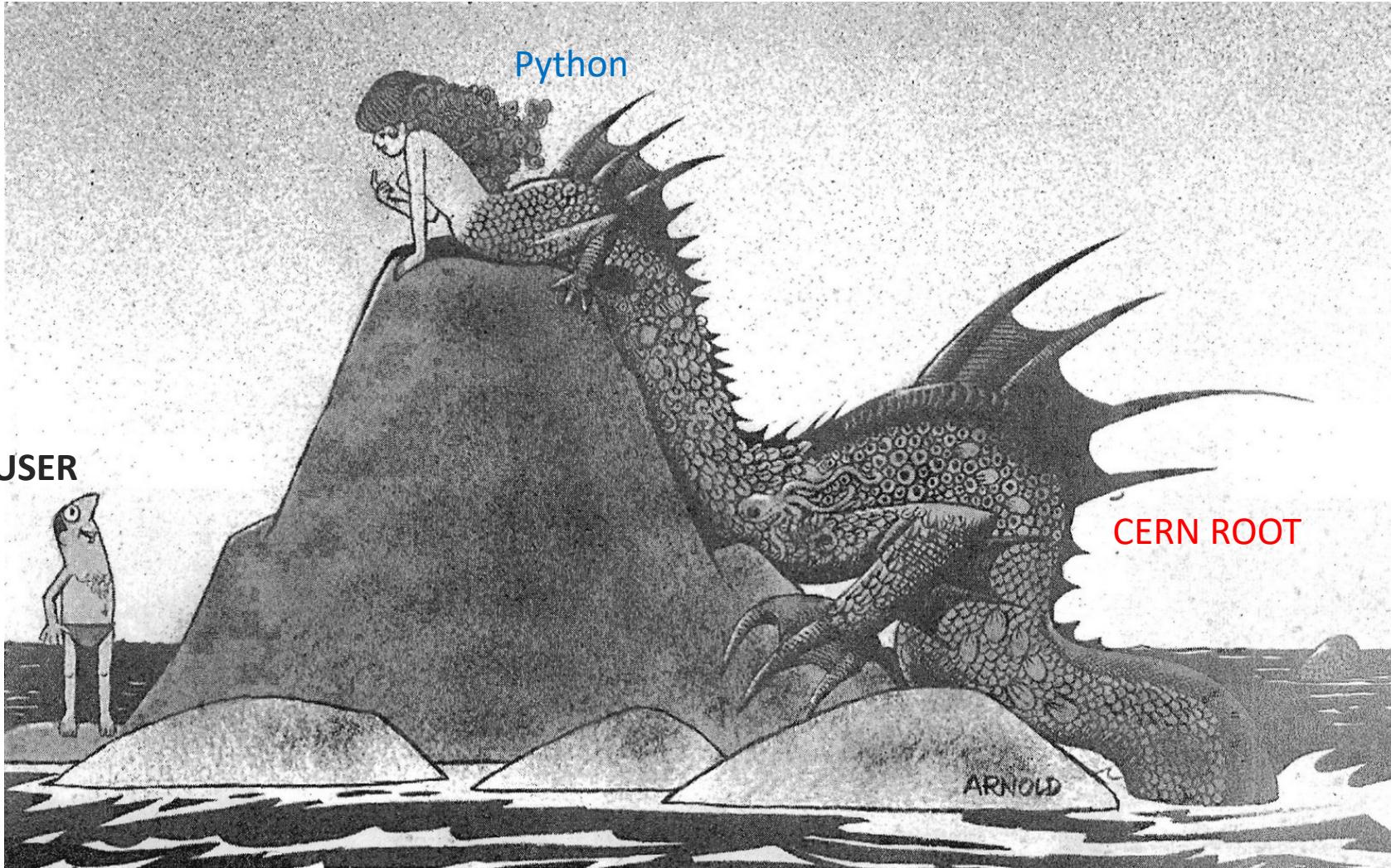


Python matplotlib GUI



First image of a blackhole in the galaxy M87

Root with a python interface



ROOT

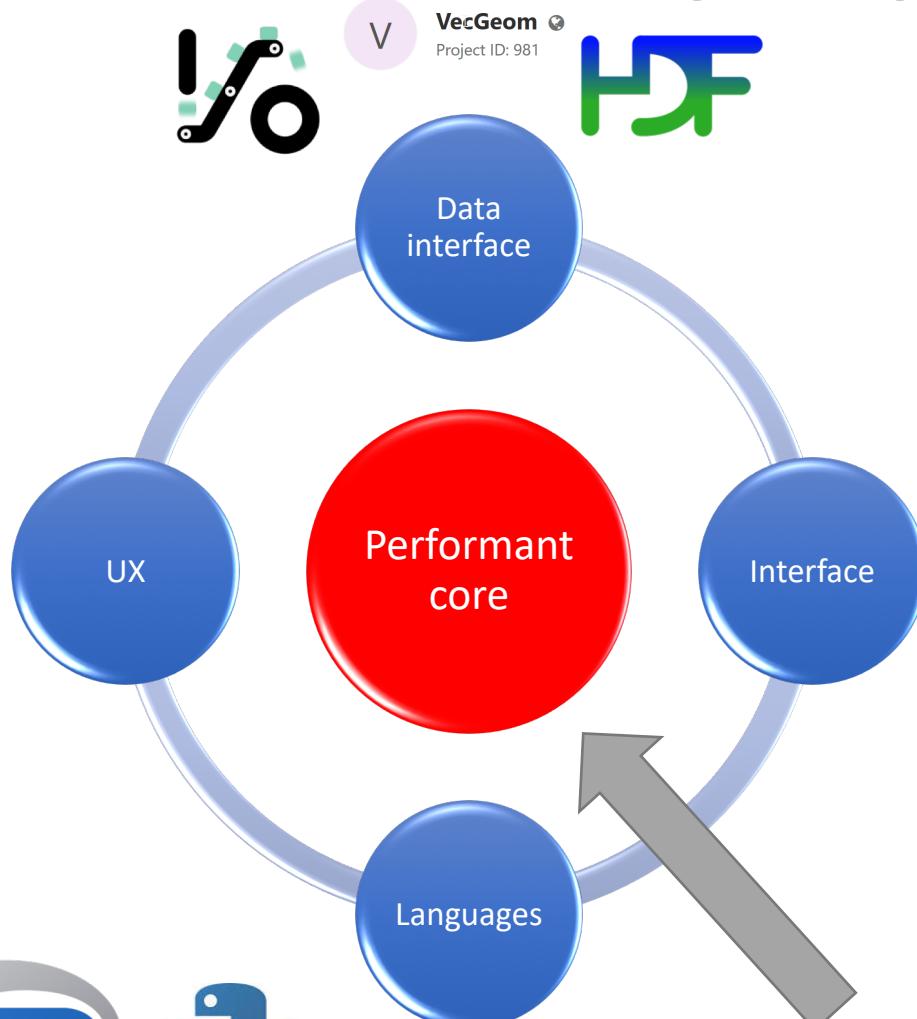
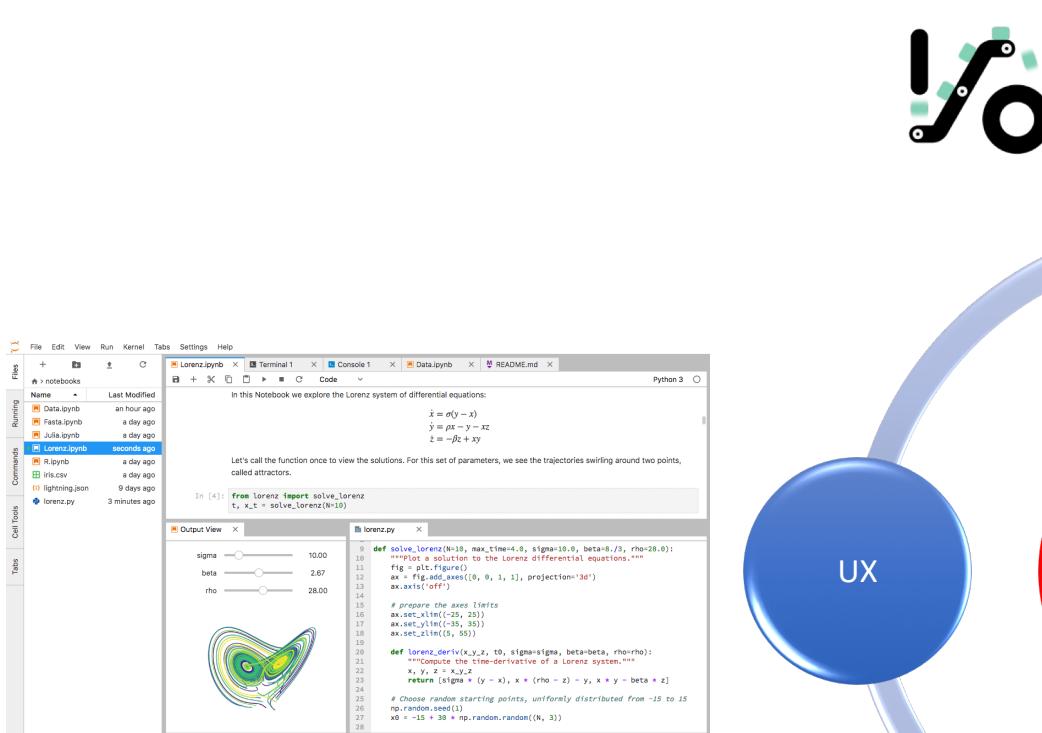
- Huge codebase of surrounding projects
- A lot of users invested time in it and doesn't want to invest time in other tools
- Active development

Modern Data tools

- Modern designed tools with far larger community than just HEP&NP
- Still many tools missing to solve our problems
- Some problems with interfaces with our software, data format, etc



Where are frameworks are going



Most probably C++



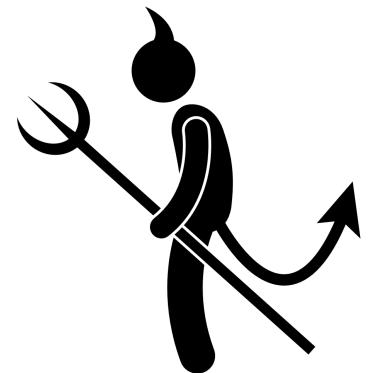
Complexity scaling explained

Some complex system



Will fail most of the times
because of
complexity scaling
problem

We try to make complex - simple



Complexity scaling

If we want to wrap/replace some complex system, the result is:

Bad

- Complex → Simple
- Moderate → More complex
- Simple → Complex



Good

- Complex → Simpler
- Moderate → Simpler
- Simple → Simple



How to handle complexity scaling

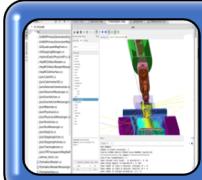
- Providing interfaces to internal complexity
(and better for everything)
(and even better – each step must be replicable)
- Low coupling (*helps*)
- Testability (*helps*)

Providing tools for users

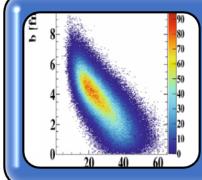
Docker containers



Event generator



Simulation



Reconstruction

Establishing things right

- CI
- CD
- Documentation

We provide just tools



Splunk.com | Documentation | Splunkbase | Answers | Wiki | Blogs

splunk>docs

Product version: 4.2.2 (latest release) Select manual: User Manual

[Topics](#) | [pdf version](#) [Documentation](#) | [Discussion](#) | [View source](#) | [History](#)

This documentation applies to the following versions of Splunk: 4.2, 4.2.1, 4.2.2

Contents

Access the data inputs configuration page
Can't find the data you know is in Splunk?

Add data to your indexes

As you read in the "About data and indexes" topic, Splunk can index logs, configuration and code, and performance data from all your applications, servers and network devices via Splunk Web.

Access the data inputs configuration page

If you have the appropriate permissions, you can view and manage all of the data in your inputs configuration page. To access this page:

- Click the Manager link on the upper right hand corner of the screen. This link shows you are currently in.
- From the list of Splunk system configuration pages, click Data inputs. The data input type of data and a count of the existing inputs for each type.

To add new data from files and directories, via TCP or UDP, or using a script, click the Add button.

For more specifics about data inputs and how to add them, see "What Splunk can index".

Can't find the data you know is in Splunk?

When you add an input to Splunk, that input gets added relative to the app you're in. If you ship with Splunk, write input data to a specific index (in the case of *Nix and Windows data that you're certain is in Splunk, be sure that you're searching the right index).

If you add an input, **Splunk adds that input to a copy of inputs.conf that belongs to the input**. This means that if you navigated to Splunk Manager, directly from the Launcher (\$SPLUNK_HOME/etc/apps/launcher/local/inputs.conf).

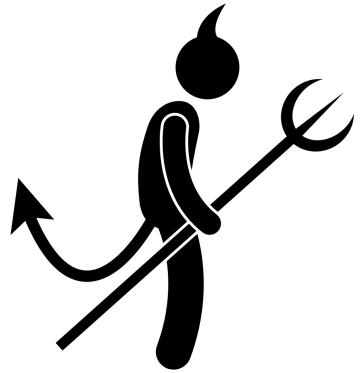
[Start](#) [Prev:](#)

Categories: V.4.2beta | V.4.2 | V.4.2.1 | V.4.3 | V.4.2.2

Was this documentation topic useful?

Not just tools but workflow





Good workflow

vs

Good tool



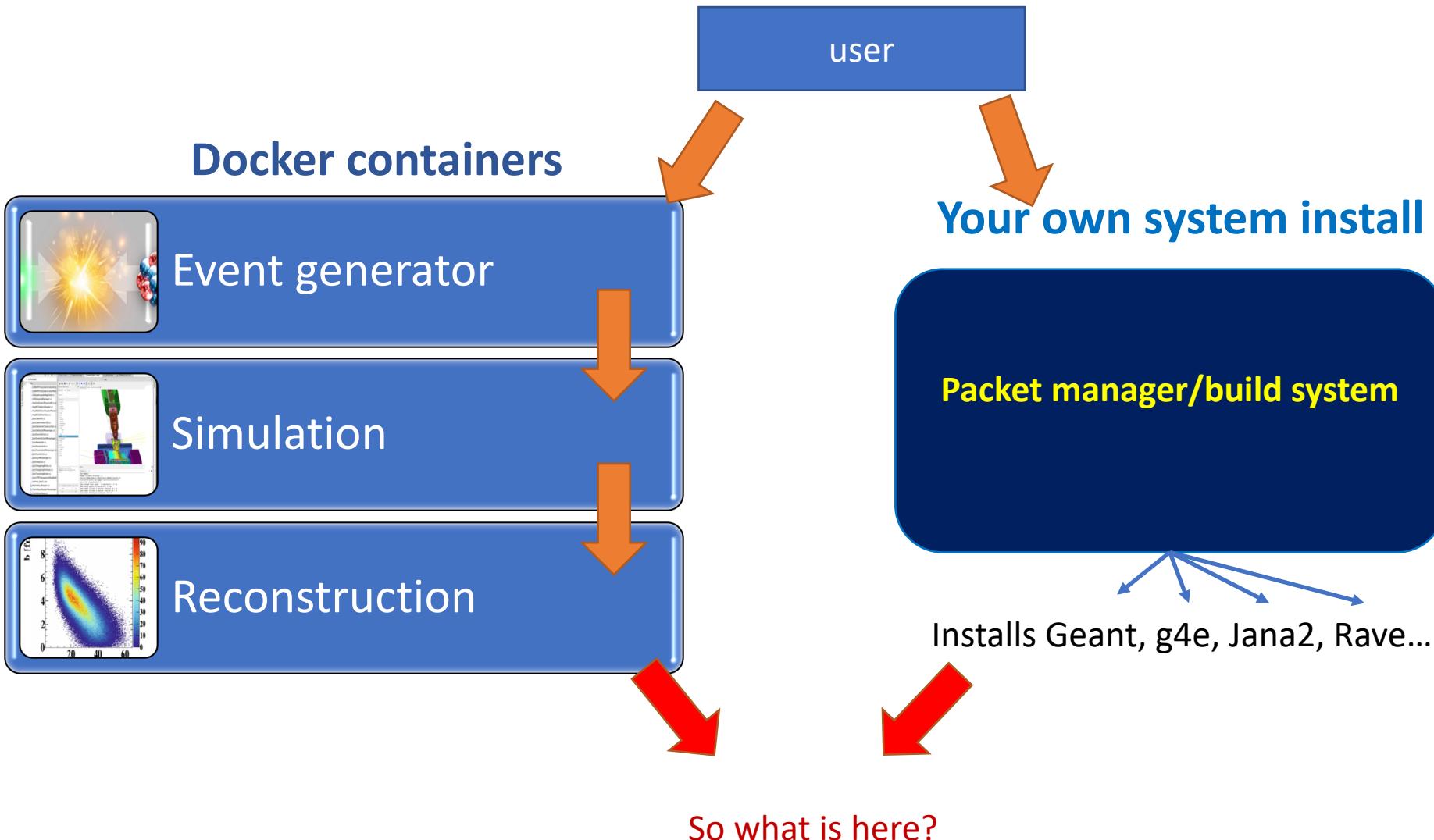
Even for code and libs:

Github -> Tutorial -> Selling you workflow of your future work

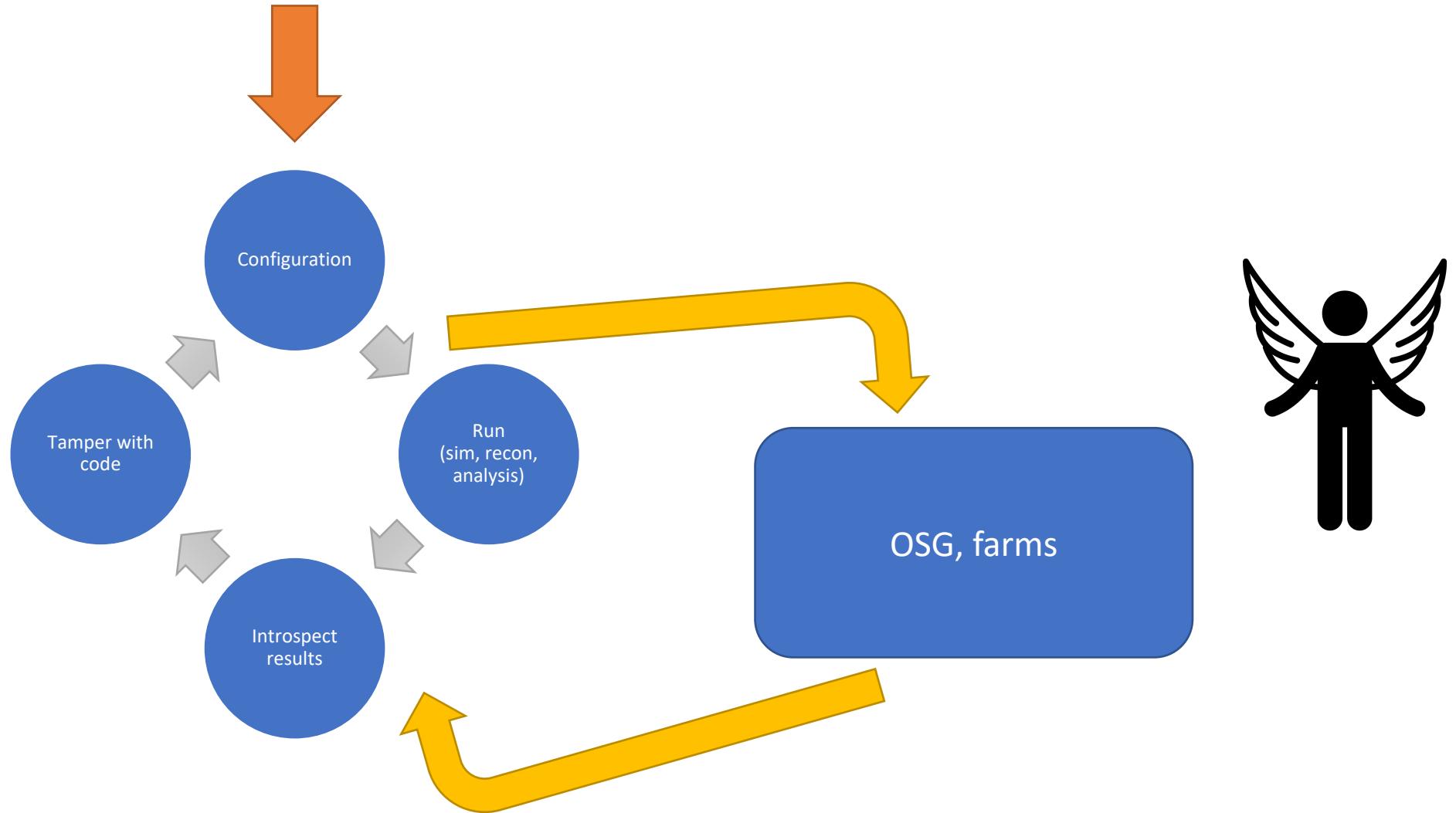
Workflow thinking everywhere

- CI → **Test workflows**
- CD → **Provide workflows**
- Documentation → **Teach workflows**

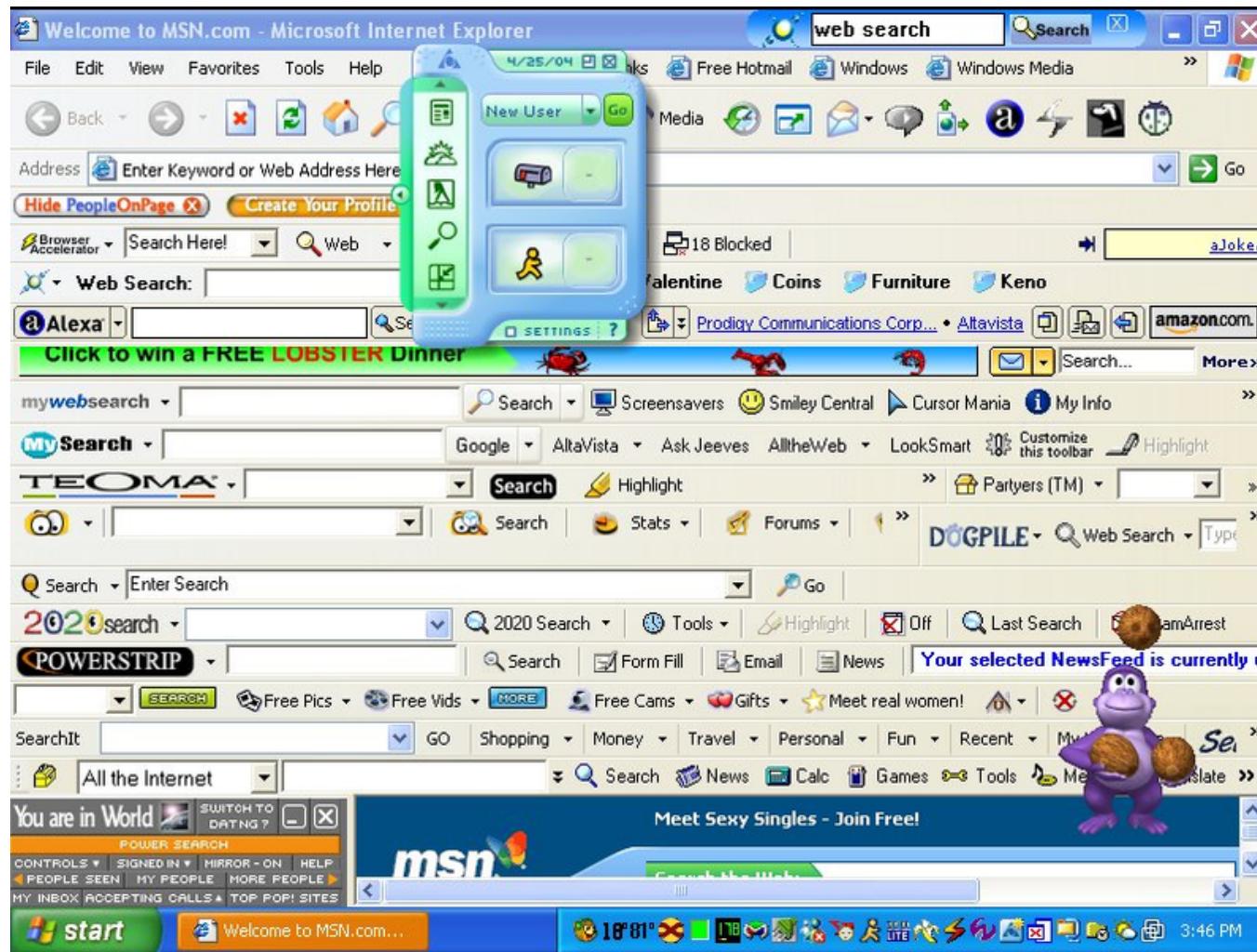
Thinking in workflows



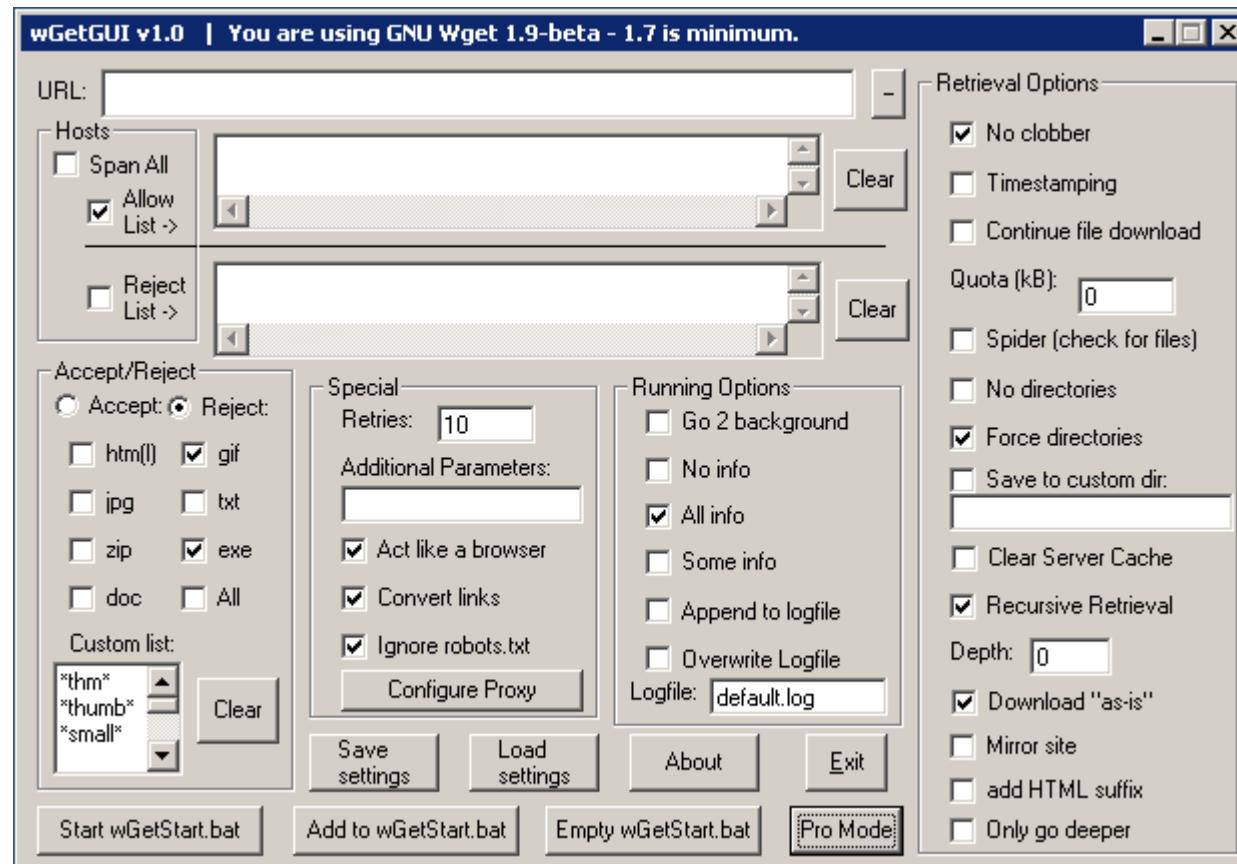
Can we identify entry point workflows



Lets talk about GUIs



Horrors of GUI

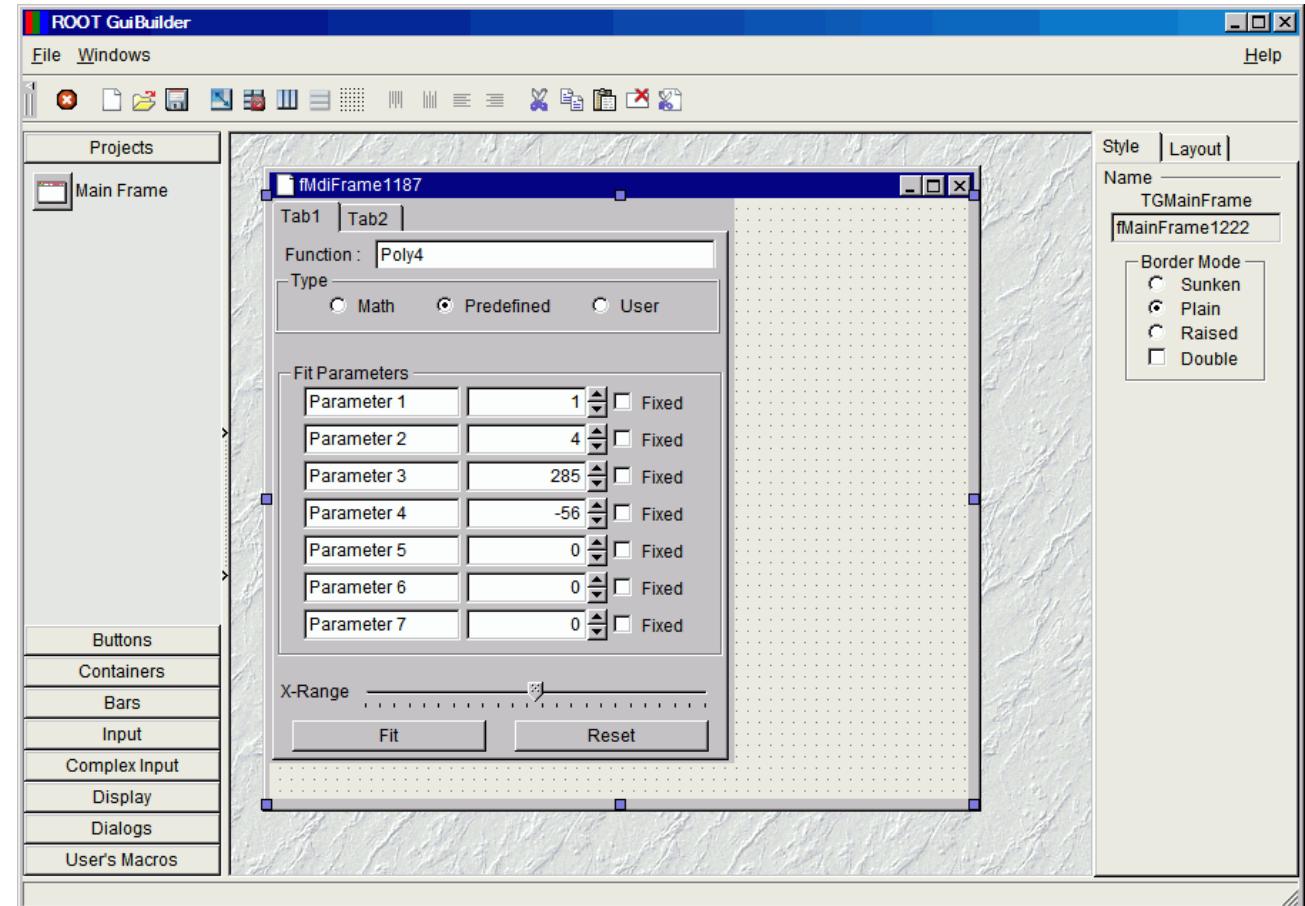


Complexity scaling

If we want to wrap/replace some complex system, the result is:

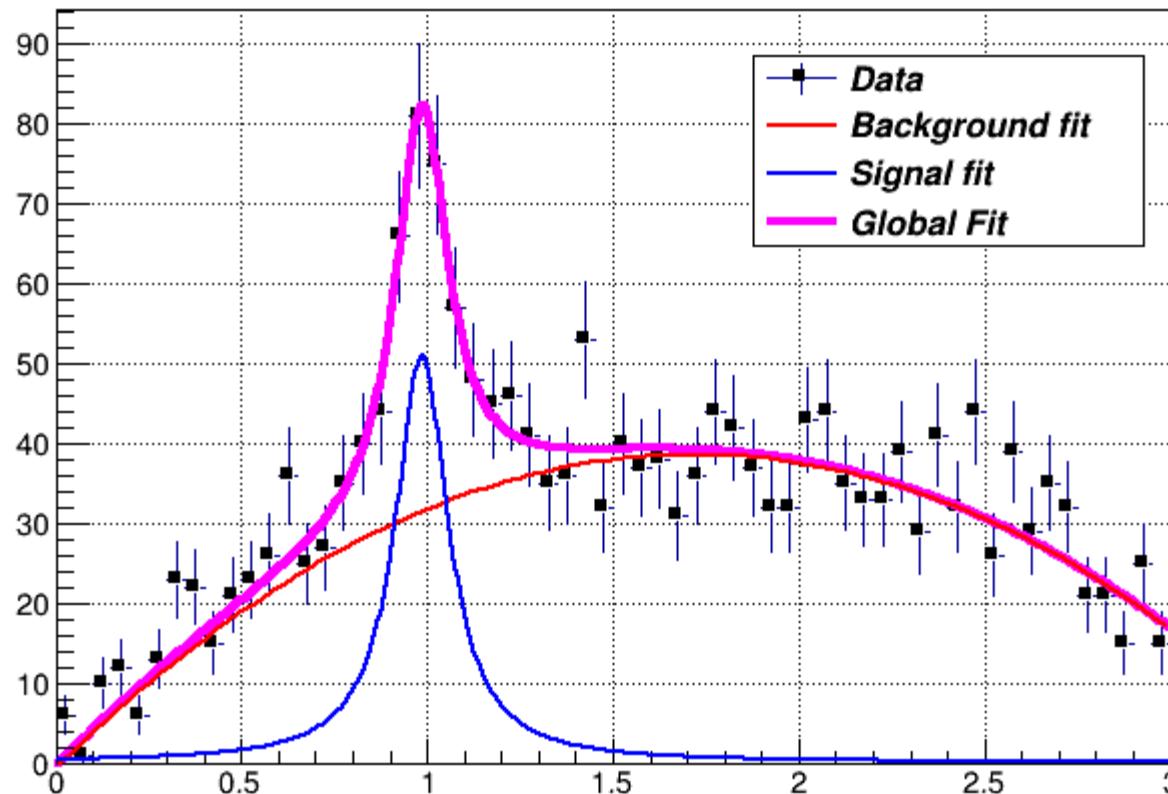
Bad

- Complex → Simple
- Simple → Complex

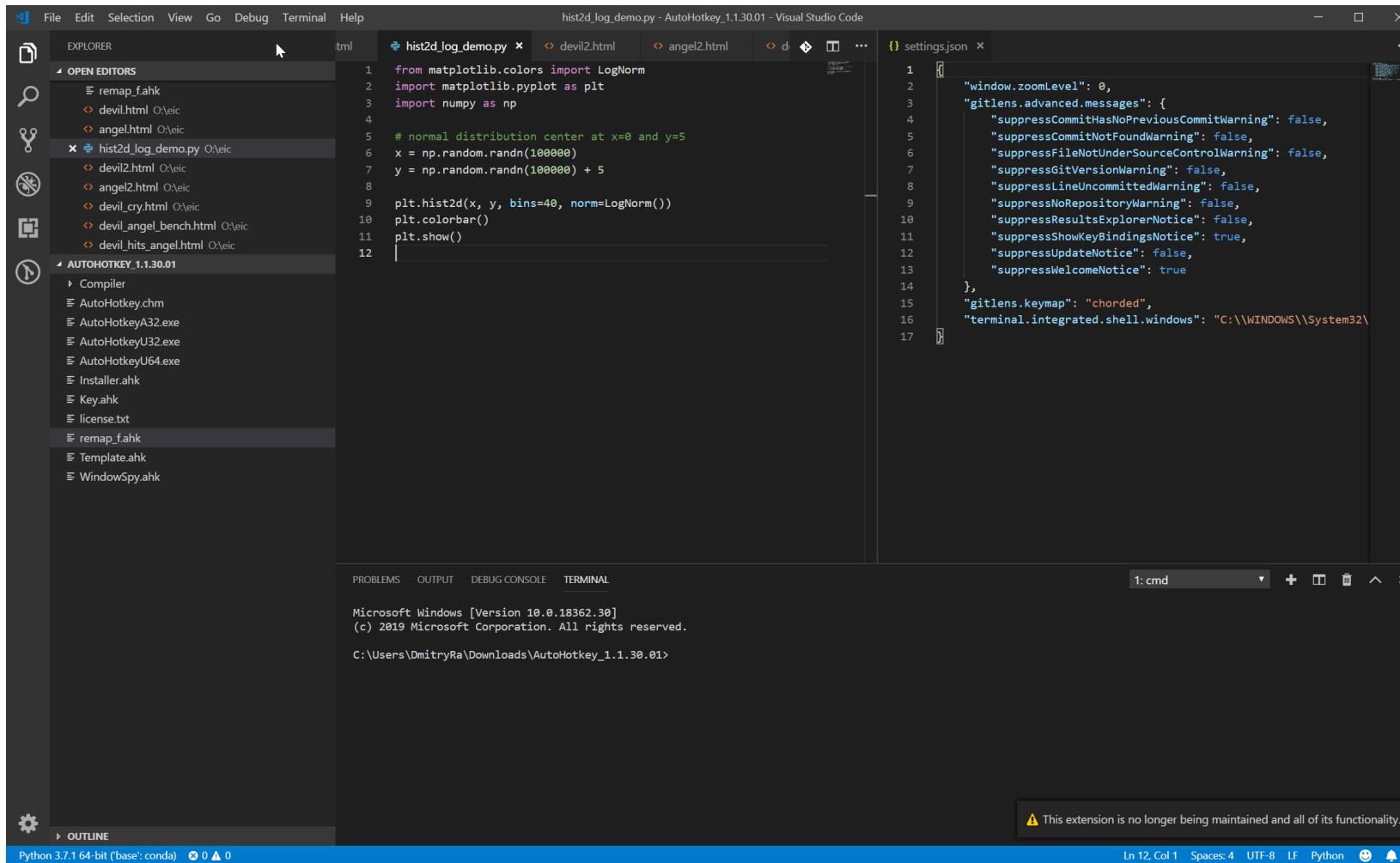


But you use and love GUI, right?

Lorentzian Peak on Quadratic Background



GUI for CODE related work (success story)



Web GUI vs Native GUI



hist2d_log_demo.py

```
1 from matplotlib.colors import LogNorm
2 import matplotlib.pyplot as plt
3 import numpy as np
4
5 # normal distribution center at x=0 and y=5
6 x = np.random.randn(100000)
7 y = np.random.randn(100000) + 5
8
9 plt.hist2d(x, y, bins=40, norm=LogNorm())
10 plt.colorbar()
11 plt.show()
```

settings.json

```
1 {
2     "window.zoomLevel": 0,
3     "gitlens.advancedSettings": {
4         "suppressCommitTheStashWarning": false,
5         "suppressCommitNotFoundWarning": false,
6         "suppressFileOutlineSourceControlWarning": false,
7         "suppressGitVersionWarning": false,
8         "suppressIncomittedWarning": false,
9         "suppressNoRepositoryWarning": false,
10        "suppressShowRebindingNotice": false,
11        "suppressUpdateNotice": false,
12        "suppressWelcomeNotice": true
13    },
14    "gitlens.keymap": "chorded",
15    "terminal.integrated.shell.windows": "C:\Windows\System32\cmd"
16}
17
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Microsoft Windows [Version 10.0.18362.300]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\DmitryRa\Downloads\Autohotkey_1.1.30.01>

OUTLINE

Python 3.7.4 64-bit ('base': croot) 0 0 0

In 12.0s 1 Spaced 4 UTT 0 If Python

More like WEB gui

ROOT is with us HERE!

JSROOT

Examples

[Examples, APIs](#)

Use:

[5.6.4 \(latest\)](#)
[dev \(expert\)](#)
[5.5.2 \(old\)](#)

Download:

[5.6.4, dev, all](#)

Read:

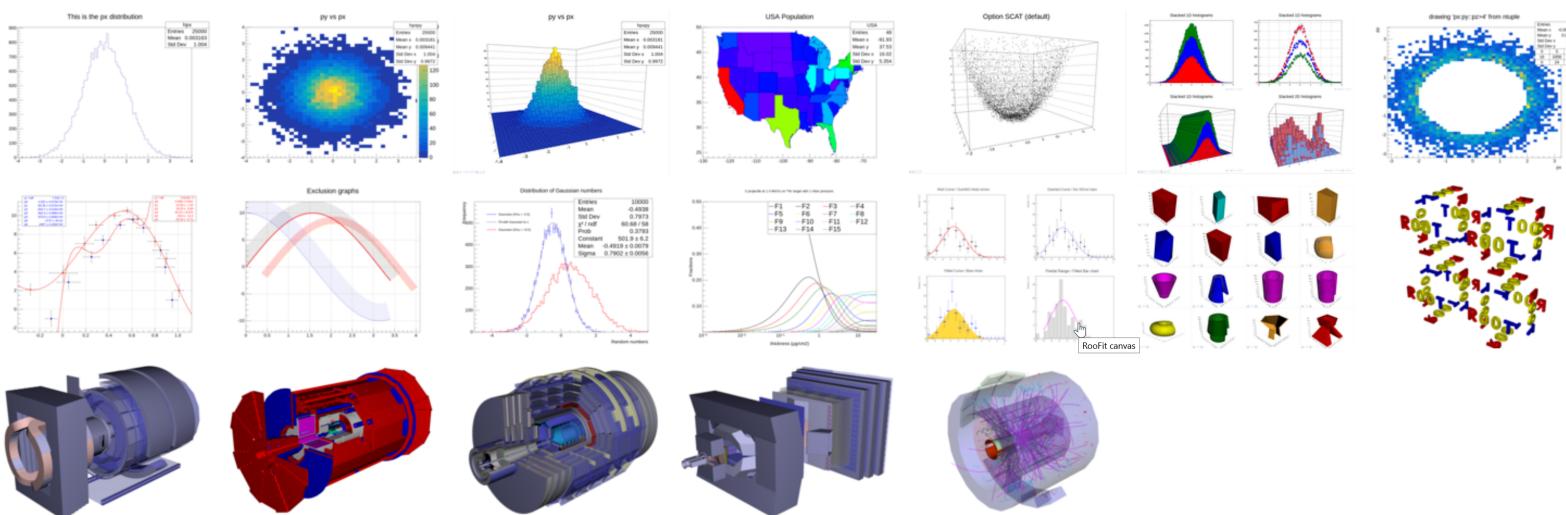
User guide
Reference API
Change log
THtppServer

Visit:

github.com
root.cern.js/
jsroot.gsi.de

Contact:

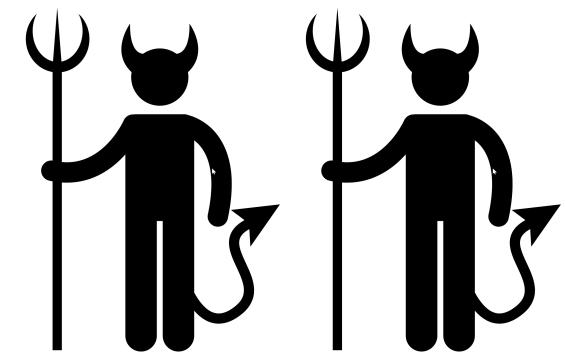
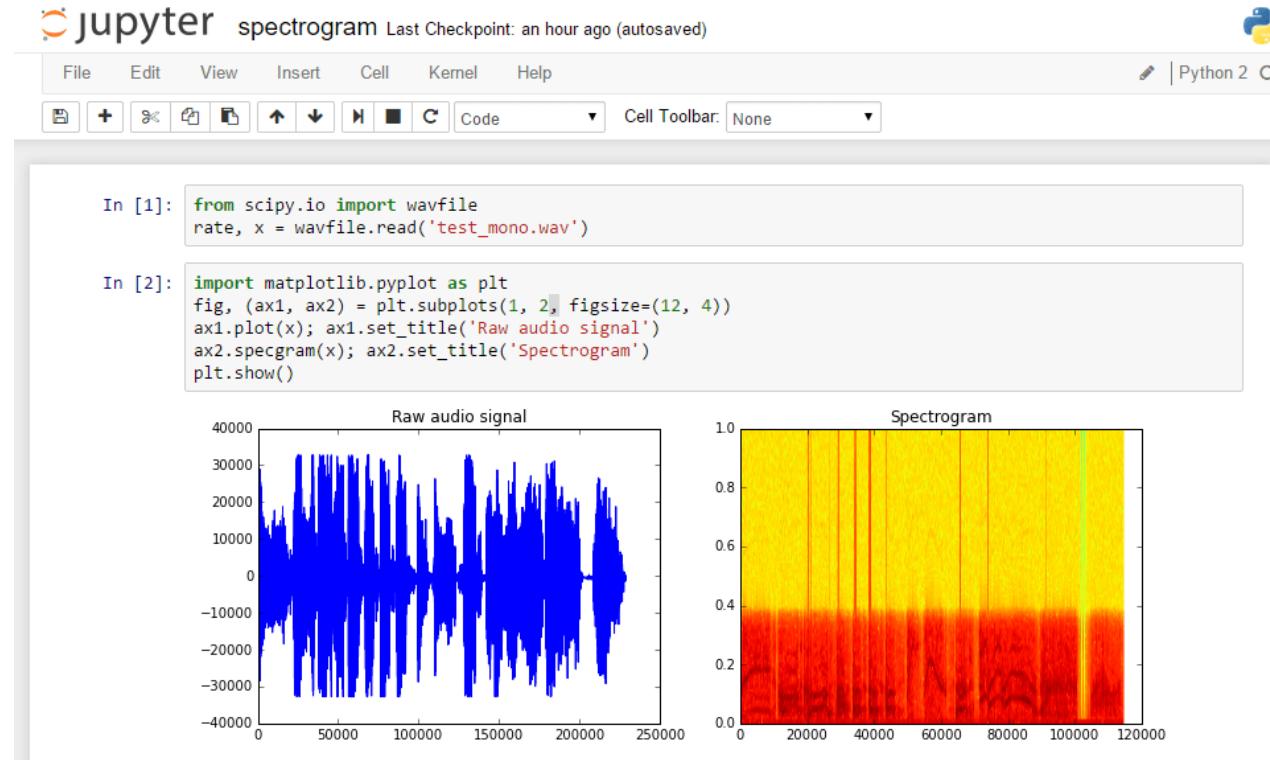
[S.Linev](#)
[B.Bellenot](#)



Applications



Jupyter notebooks



Jupyter lab

- Modular + plugins
- One place to tamper with scripts and see the output
- Better interacts with system
- Can be used as SaaS



The screenshot shows the Jupyter Lab interface. On the left, the file browser lists notebooks like Data.ipynb, Fasta.ipynb, Julia.ipynb, and Lorenz.ipynb. In the center, a notebook cell displays the Lorenz system of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

Below the equations, a note says: "Let's call the function once to view the solutions. For this set of parameters, we see the trajectories swirling around two points, called attractors." A code cell in the notebook contains:

```
from lorenz import solve_lorenz
t, x_t = solve_lorenz(N=10)
```

On the right, an "Output View" panel shows a 3D plot of the Lorenz attractor with sliders for sigma (10.00), beta (2.67), and rho (28.00). The code for generating the plot is visible in the "lorenz.py" file:

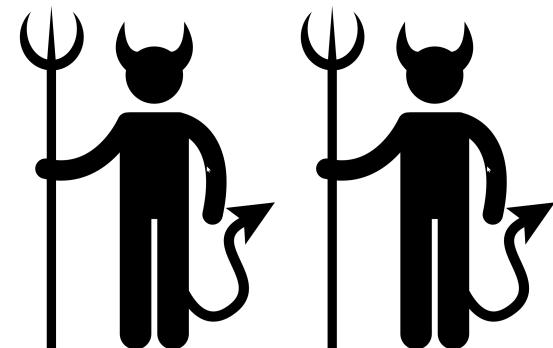
```
def solve_lorenz(N=10, max_time=4.0, sigma=10.0, beta=8./3, rho=28.0):
    """Plot a solution to the Lorenz differential equations."""
    fig = plt.figure()
    ax = fig.add_axes([0, 0, 1, 1], projection='3d')
    ax.axis('off')

    # prepare the axes limits
    ax.set_xlim((-25, 25))
    ax.set_ylim((-35, 35))
    ax.set_zlim((5, 55))

    def lorenz_deriv(x_y_z, t0, sigma=sigma, beta=beta, rho=rho):
        """Compute the time-derivative of a Lorenz system."""
        x, y, z = x_y_z
        return [sigma * (y - x), x * (rho - z) - y, x * y - beta * z]

    # Choose random starting points, uniformly distributed from -15 to 15
    np.random.seed(1)
    x0 = -15 + 30 * np.random(N, 3)
```

- Still in development
- Some workflows are there
- Have many limitations compared to a native GUI



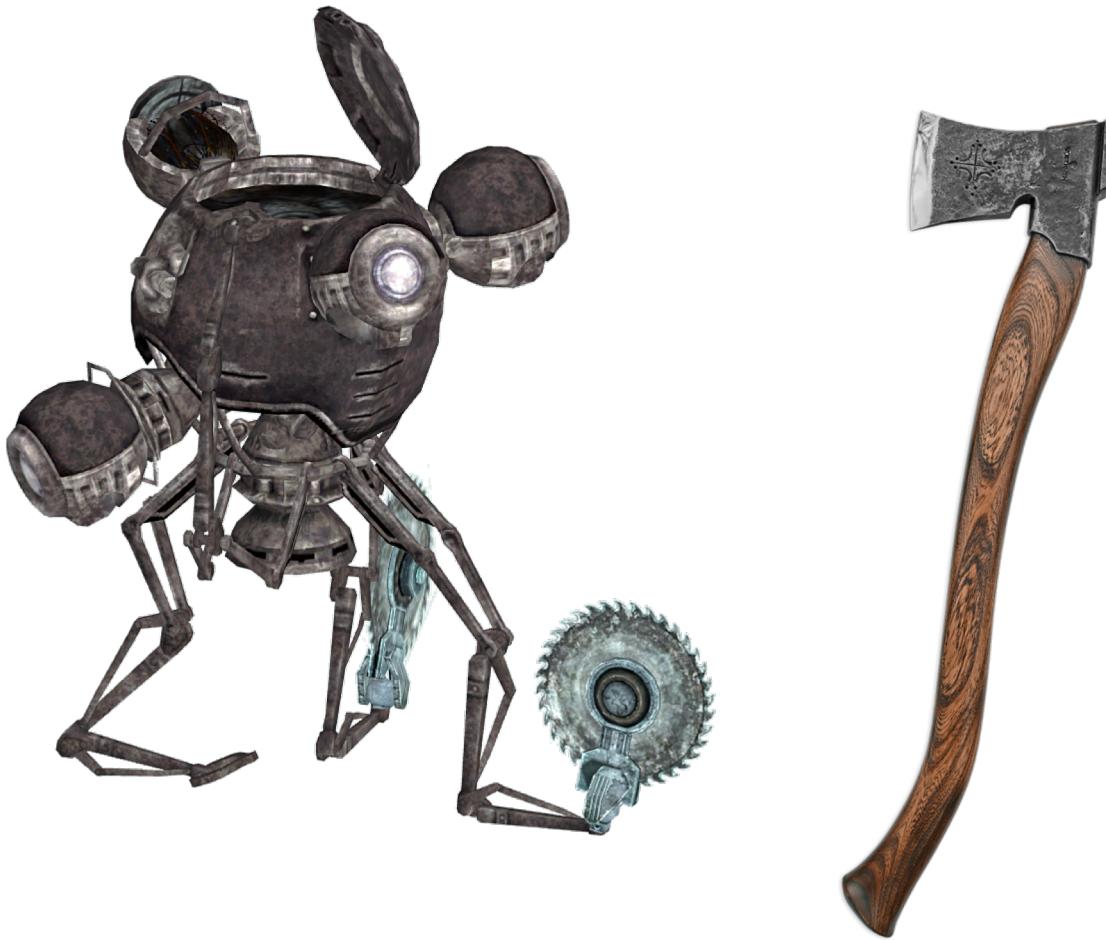
Conclusions

A heated discussion should conclude this talk!



*We will present our thoughts on users entry point in terms of
working prototype
in the next talk*

Backup slides



C++ everywhere is a strange choice

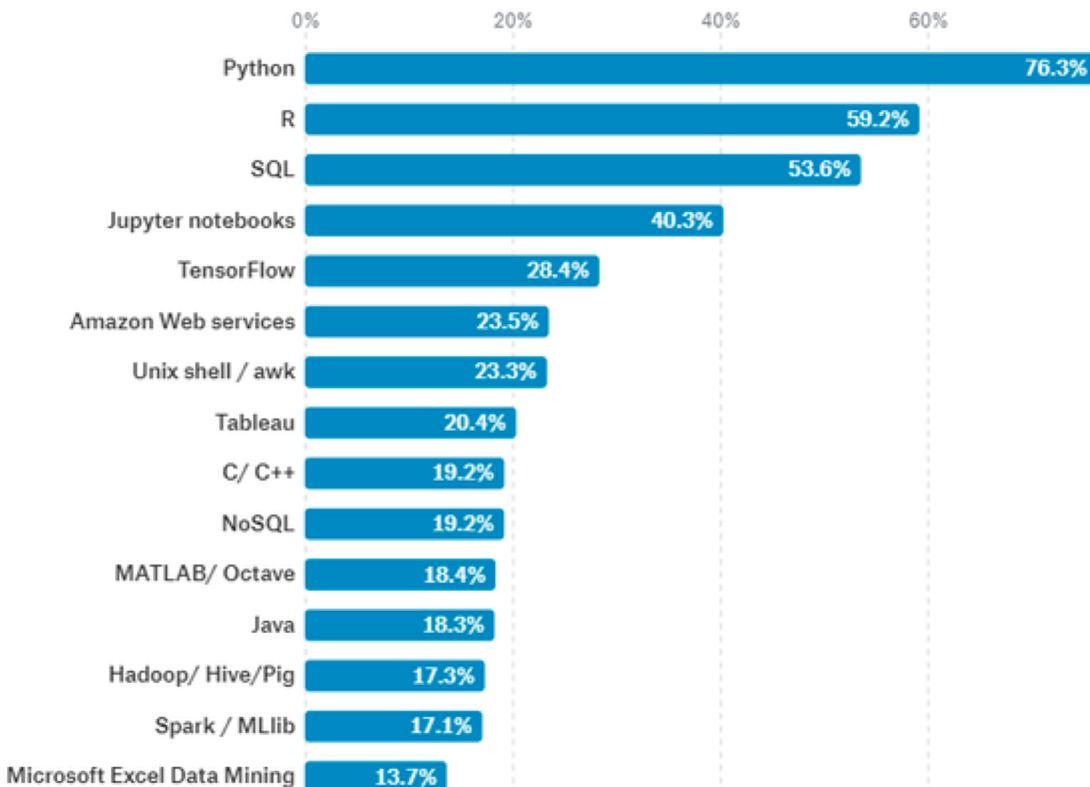


Mario Alemi

Physicist

... So I started using ROOT. I was eager to learn C++. I took courses, read Bjarne Stroustrup's book. And then asked myself –**did ROOT people *first* get a good dose of LSD (lysergic acid) and *after* decided to use C++ as scripting language?**

Data science languages



7,955 responses

Only displaying the top 15 answers. There are 38 answers not shown.

CERN root-7

- Example (and what could go wrong)

<https://root.cern.ch/root-7>