Data Science Tools for Analysis

Jim Pivarski

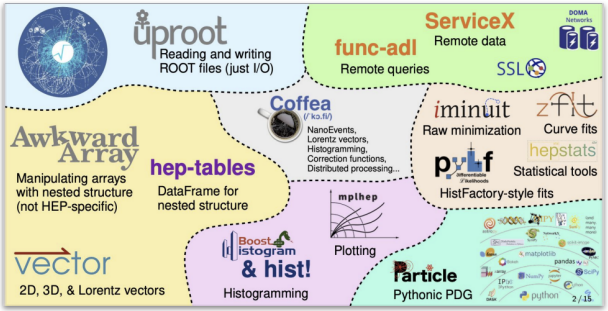
Princeton University – IRIS-HEP

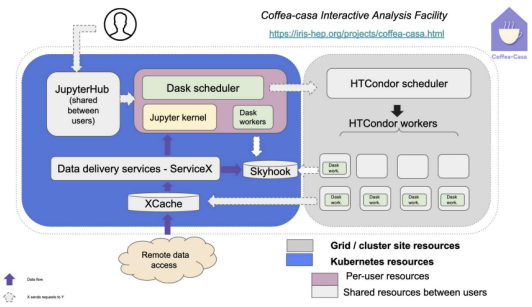
March 9, 2022

The idea behind the Analysis Grand Challenge

Analysis Tools

Analysis Facilities





The analysis tools developer community

Many people are developing tools. . .

The analysis tools developer community

Many people are developing tools. . .

- in IRIS-HEP and out (but mostly communicating on IRIS-HEP's Slack)

The analysis tools developer community

Many people are developing tools. . .

- in IRIS-HEP and out (but mostly communicating on IRIS-HEP's Slack)
- each focused on a single purpose, but fitting together into a larger ecosystem

The analysis tools developer community

Many people are developing tools...

- in IRIS-HEP and out (but mostly communicating on IRIS-HEP's Slack)
- each focused on a single purpose, but fitting together into a larger ecosystem
- using existing software from beyond HEP whenever possible

The analysis tools developer community

Many people are developing tools...

- in IRIS-HEP and out (but mostly communicating on IRIS-HEP's Slack)
- each focused on a single purpose, but fitting together into a larger ecosystem
- using existing software from beyond HEP whenever possible
- mostly in Python

Overview of these activities for the LHCC (arXiv:2202.02194)

Physics > Data Analysis, Statistics and Probability

[Submitted on 4 Feb 2022]

HL-LHC Computing Review Stage 2, Common Software Projects: Data Science Tools for Analysis

Jim Pivarski, Eduardo Rodrigues, Kevin Pedro, Oksana Shadura, Benjamin Krikler, Graeme A. Stewart

This paper was prepared by the HEP Software Foundation (HSF) PyHEP Working Group as input to the second phase of the LHCC review of High-Luminosity LHC (HL-LHC) computing, which took place in November, 2021. It describes the adoption of Python and data science tools in HEP, discusses the likelihood of future scenarios, and recommendations for action by the HEP community.

Comments: 25 pages, 7 figures; presented at this https URL (LHCC Review of HL-LHC Computing)
Subjects: **Data Analysis, Statistics and Probability (physics.data-an)**; High Energy Physics - Experiment (hep-ex)
Report number: FERMILAB-CONF-22-061-SCD
Cite as: arXiv:2202.02194 **[physics.data-an]**
(or arXiv:2202.02194v1 **[physics.data-an]** for this version)
https://doi.org/10.48550/arXiv.2202.02194

Submission history

From: Jim Pivarski [view email]
[v1] Fri, 4 Feb 2022 15:39:45 UTC (1,692 KB)

A bit different from an ordinary LHC project:

- The users are distributed (who are they? what do they want?)
- The developers are distributed (who's working on what?)

Gathered developers by lighting a beacon: Scikit-HEP

Search Scikit-HEP

Scikit-HEP on GitHub

Scikit-HEP project - welcome!

GitHub chat on gitter

The Scikit-HEP project is a community-driven and community-oriented project with the aim of providing Particle Physics at large with an ecosystem for data analysis in Python. Read more →

See our developer pages for information on developing Python packages!

Basics:

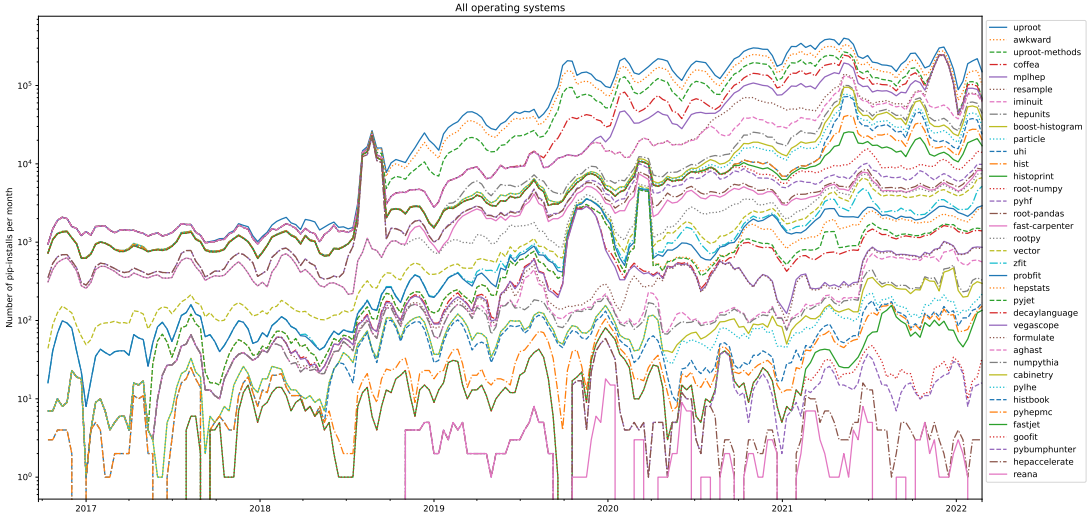
Awkward Array	Manipulate JSON-like data with NumPy-like idioms.
hepunits	Units and constants in the HEP system of units.
VECTOR	Manipulate Lorentz, 3D, and 2D vectors in NumPy, Numba, or Awkward.

Data manipulation and interoperability:

formulate	Easy conversions between different styles of expressions

- Home
- Packages
- Resources
- Who uses Scikit-HEP?
- Python Version Policy
- Developer information
- Code of conduct
- Getting in touch

Stacked download statistics for Scikit-HEP and related packages



Learning about user needs: very different from my expectations!

Big Data

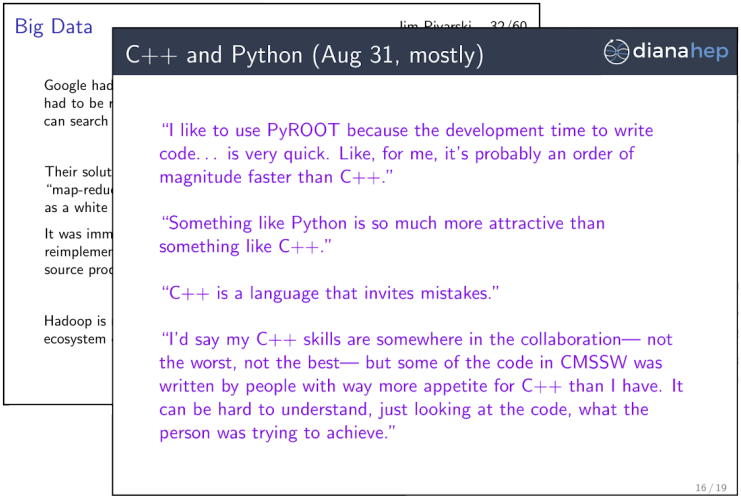
Google had an re-indexing problem: a set of webpages containing words had to be re-indexed as a set of words pointing to webpages, so that you can search for pages by keyword.

Their solution, called "map-reduce," was published as a white paper in 2004.

It was immediately reimplemented as an open source product, Apache Hadoop.

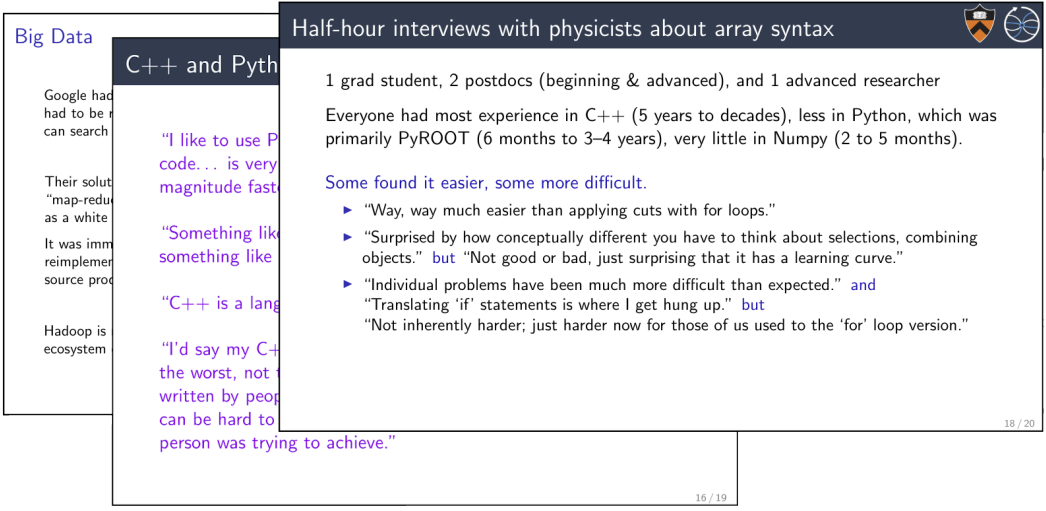
Hadoop is now almost synonymous with Big Data, and it has spawned an ecosystem of tools that interoperate with it, much like ROOT in HEP.

Learning about user needs: very different from my expectations!



Sought user input in focus groups, interviews, and surveys.

Learning about user needs: very different from my expectations!



Sought user input in focus groups, interviews, and surveys.

Learning about user needs: very different from my expectations!

Big Data

Google had
had to be
can search

Their solut
"map-redu
as a white

It was imm
reimplemer
source proc

Hadoop is
ecosystem

C++ and Pyth

"I like to use P
code... is very
magnitude fast

"Something lik
something like

"C++ is a lang

"I'd say my C+
the worst, not
written by peop
can be hard to
person was trying to achieve."

16 / 19

Half-hour interviews with physicists about array syntax

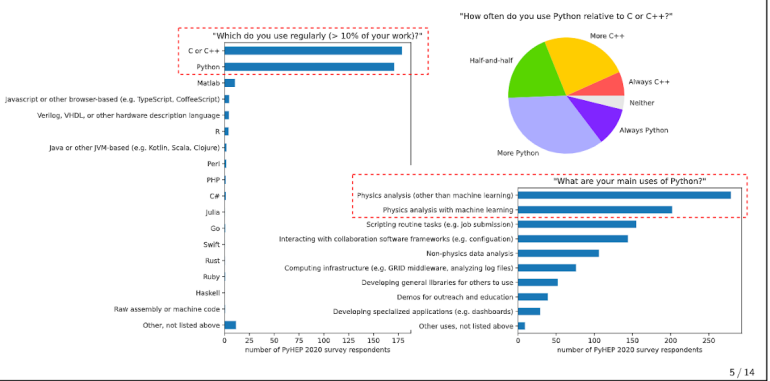
1 grad stu

Everyone
primarily

Some four

- "Way
- "Surp
object
- "Indiv
"Tran
"Not

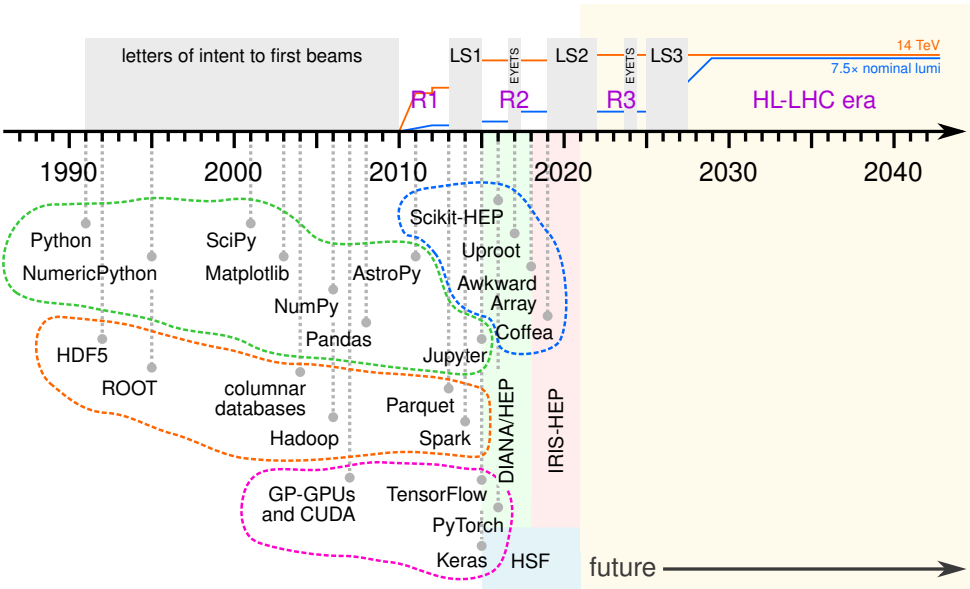
Consistent with survey results (PyHEP 2020 participants)



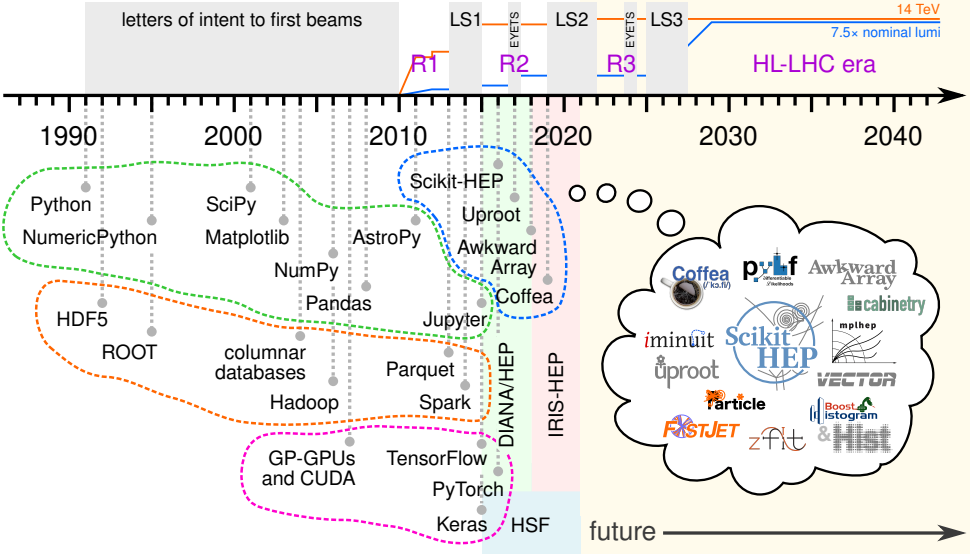
5 / 14

Sought user input in focus groups, interviews, and surveys.

What I learned: now is the time to make HEP "Pythonic"



What I learned: now is the time to make HEP "Pythonic"



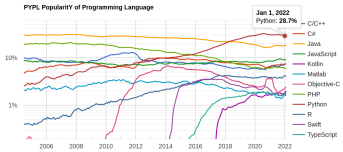
Why Python? Why now?

Python is currently leading every "most popular programming language" index

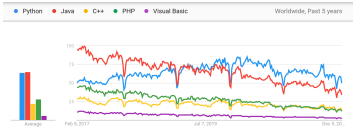
Tiobe

Jan 2022	Jan 2021	Change	Programming Language	Ratings	Change
1	3	^	Python	13.59%	+1.66%
2	1	v	C	12.44%	-4.94%
3	2	v	Java	10.95%	-1.30%
4	4		C++	8.29%	+0.73%
5	5		C#	5.88%	+1.73%
6	6		Visual Basic	4.74%	+0.90%
7	7		JavaScript	2.09%	-0.11%
8	11	^	Assembly language	1.85%	+0.21%

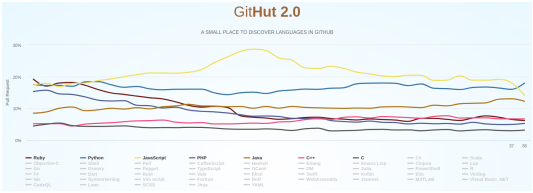
PYPL



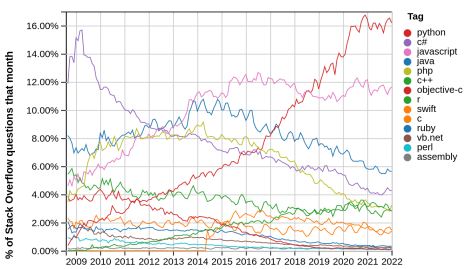
Google Trends



GitHut

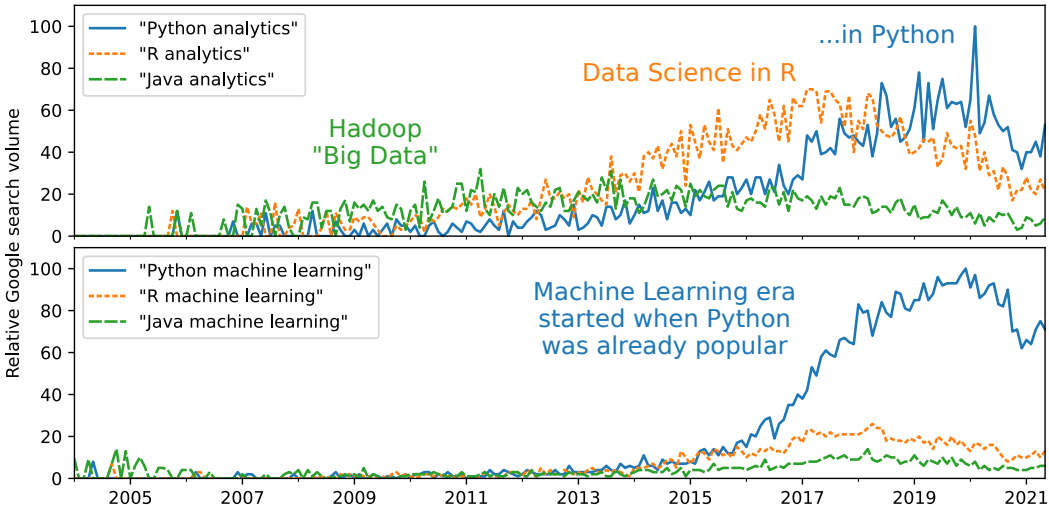


StackOverflow



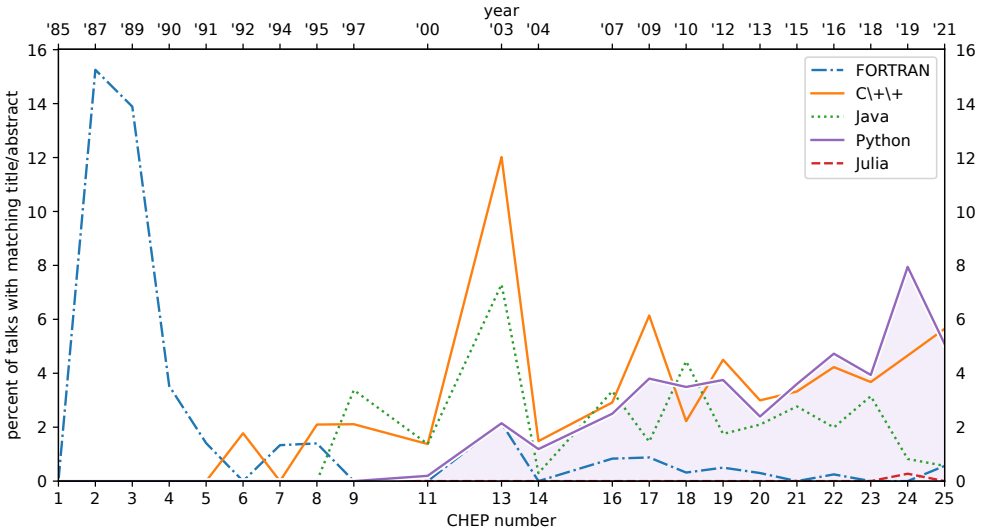
Why Python? Why now?

Especially in data analysis (below: coincidence of search terms in Google Trends)



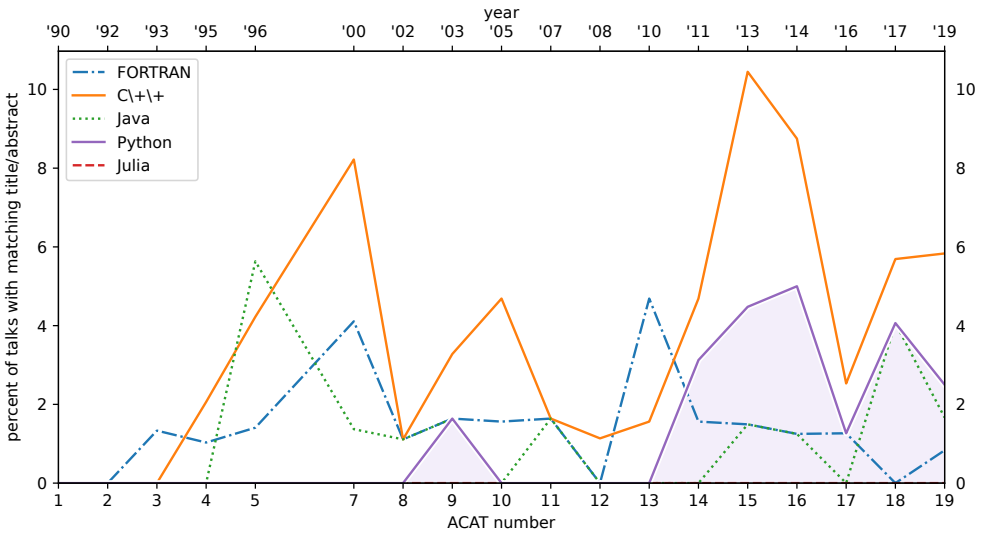
Python has also been relevant in HEP *for 20 years*

Programming languages mentioned in CHEP



Python has also been relevant in HEP *for 20 years*

Programming languages mentioned in CHEP and ACAT (this one is not as clear)



Emerging Standard ? Python as "Software Glue"

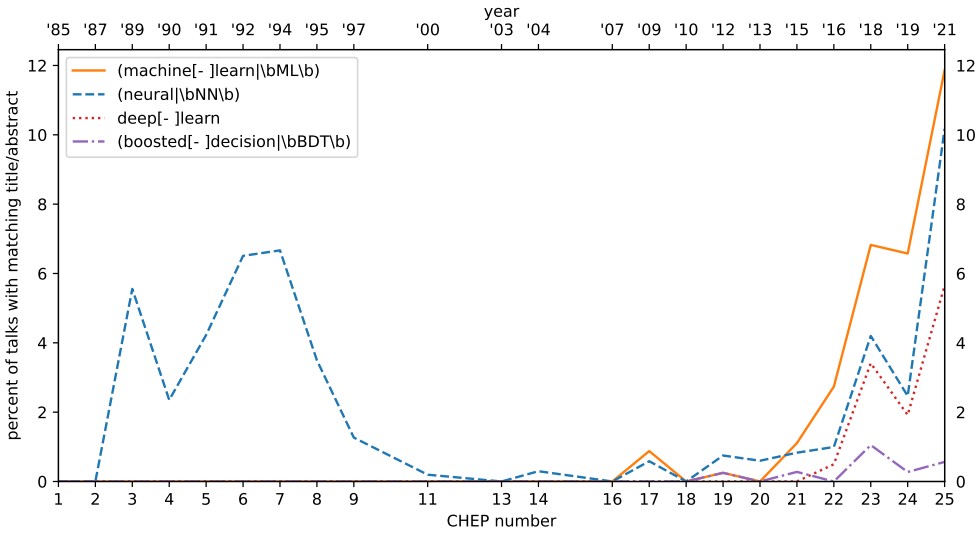
- Clear trend towards Python
  - Used by: ATLAS (Athena),CMS, D0, LHCb (Gaudi), SND,...
  - Used by: Lizard/Anaphe, HippoDraw, JAS (Jython)...
  - Architecturally, scripting is "just another service"
  - ROOT is the exception to the "Python rule"
    - CINT interpreter plays a central role
    - Developers and users seem happy
- Python is popular with developers...
  - Rapid prototyping; gluing together code
  - (Almost) auto-generation of wrappers (SWIG)
- ...but acceptance by users not yet proven
  - Another language to learn, syntax,...

"Summary of Track 2: Data Analysis and Visualis. Lucas Taylor, Northeastern U. CHEP 01, Beijing, 3-7 S.

Note: PyROOT introduced in 2004 (v4.00/04).

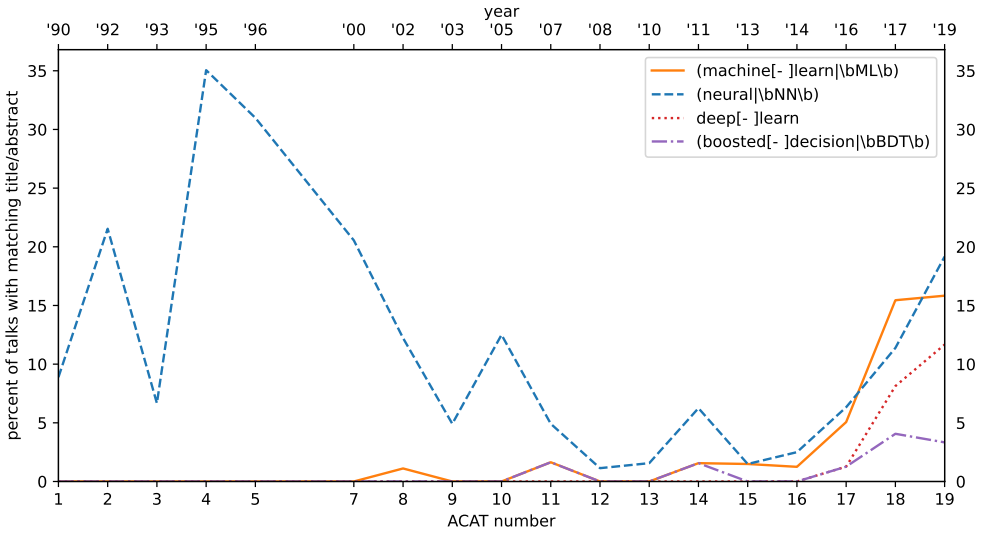
Much longer than the resurgence of interest in machine learning

Machine learning terms mentioned in CHEP



Much longer than the resurgence of interest in machine learning

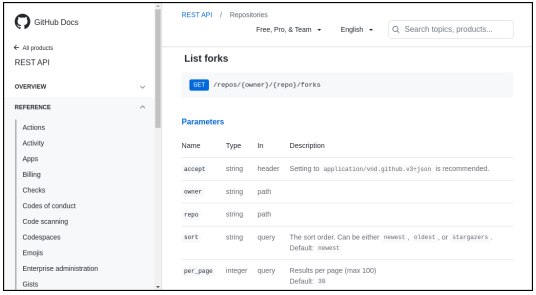
Machine learning terms mentioned in CHEP and ACAT



How is Python being used by physicists?

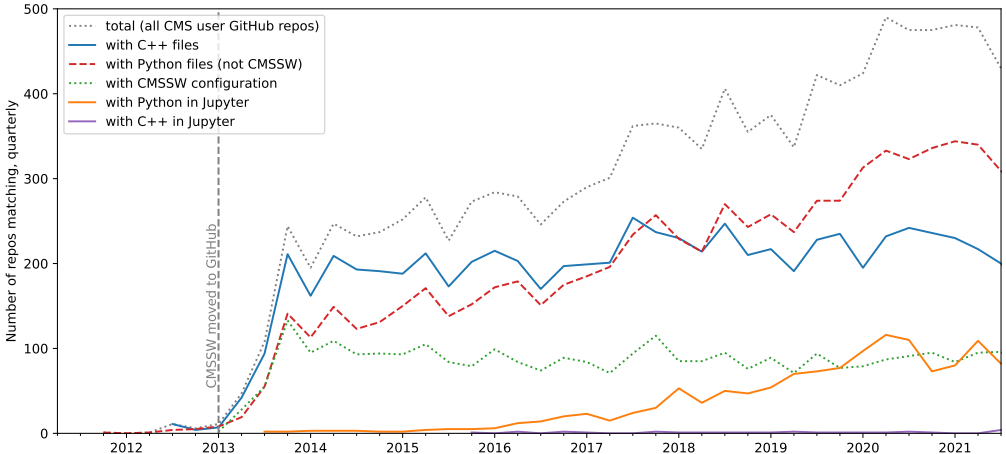
Analyze code in 11 635 GitHub repos written by 2 172 physicists:

1. Ask GitHub which users forked CMSSW and call them "CMS physicists." (CMSSW has been on GitHub for a long enough time to see trends.)
2. Clone all of the physicists' repos (the ones that are not forks of something else).
3. Search the code of these repos and count matches.
4. Take care to exclude CMSSW configuration files, which are also Python.



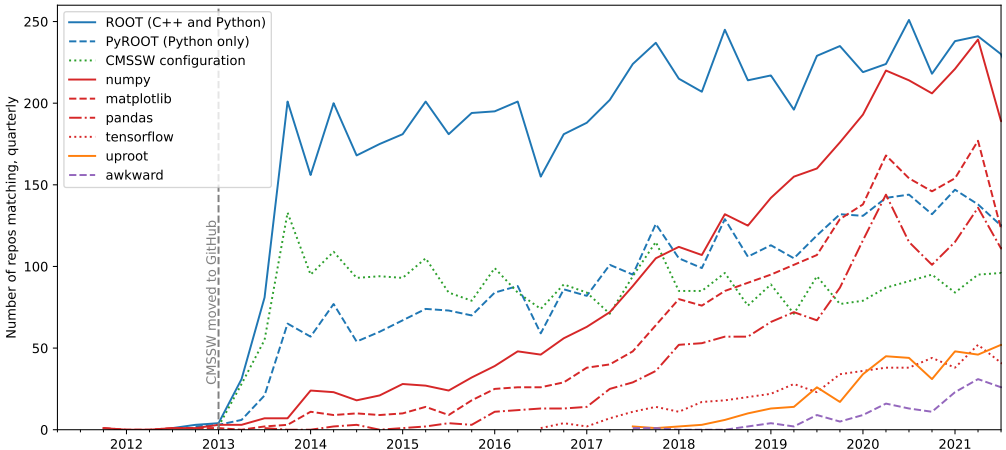
How is Python being used by physicists?

Number of non-fork GitHub repos created by CMS physicists (users who forked CMSSW)



How is Python being used by physicists?

Same sample, now counting regex matches for `import XYZ`, `from XYZ import`, etc.



Conclusions

- Python has been slowly growing in HEP for 20 years (unlike the "big bang" when C++ replaced Fortran).

Conclusions

- Python has been slowly growing in HEP for 20 years (unlike the "big bang" when C++ replaced Fortran).
- But physicists started using the SciPy ecosystem (NumPy, Matplotlib, Pandas, etc.) about 5 years ago.

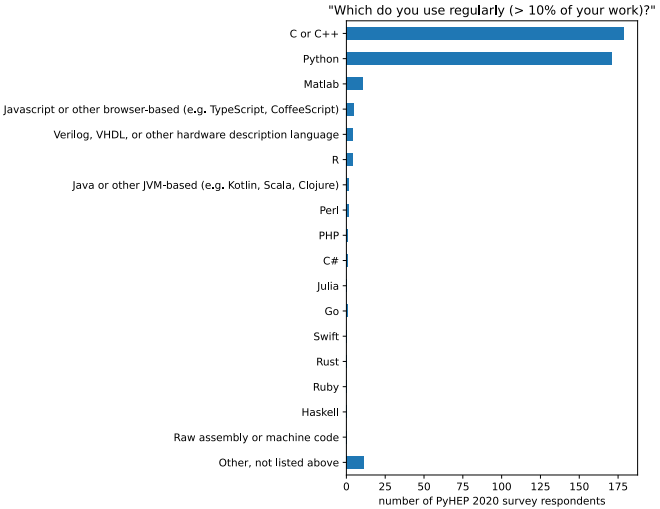
Conclusions

- Python has been slowly growing in HEP for 20 years (unlike the "big bang" when C++ replaced Fortran).
- But physicists started using the SciPy ecosystem (NumPy, Matplotlib, Pandas, etc.) about 5 years ago.
- Scikit-HEP also started 5 years ago, just in time to serve this need.

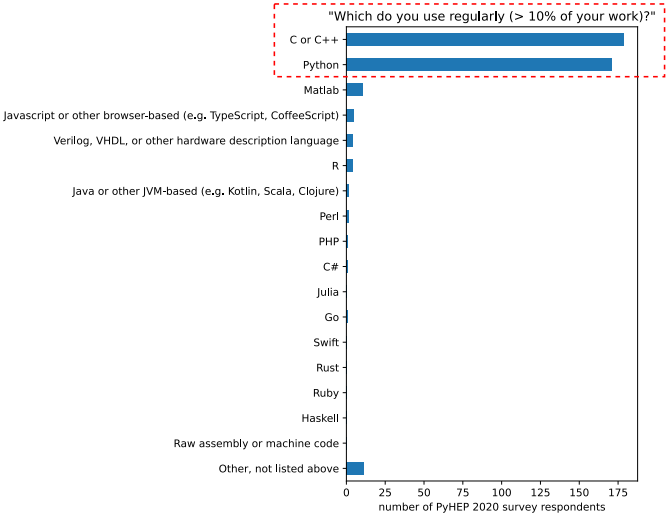
Conclusions

- Python has been slowly growing in HEP for 20 years (unlike the "big bang" when C++ replaced Fortran).
- But physicists started using the SciPy ecosystem (NumPy, Matplotlib, Pandas, etc.) about 5 years ago.
- Scikit-HEP also started 5 years ago, just in time to serve this need.
- C++, ROOT, and collaboration software configuations have been *steady*, while Python and the SciPy ecosystem have been *rising*.

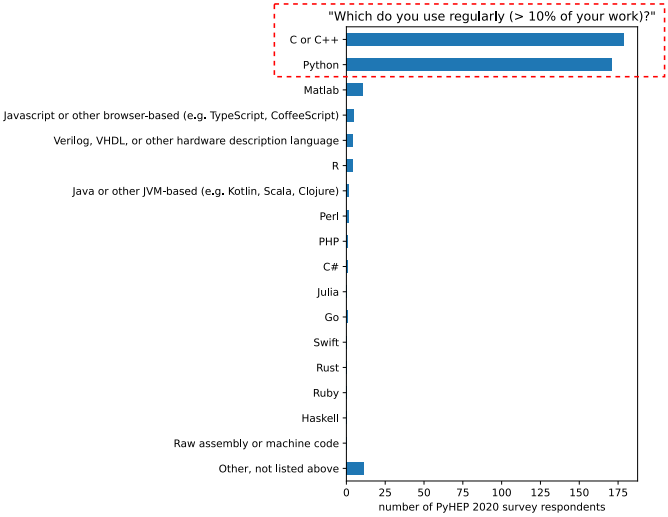
PyHEP 2020 survey respondents ($N = 406$): same picture

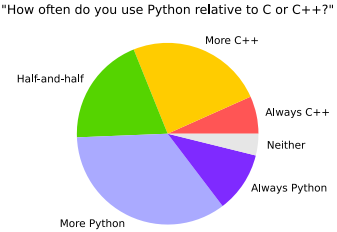


PyHEP 2020 survey respondents ($N = 406$): same picture

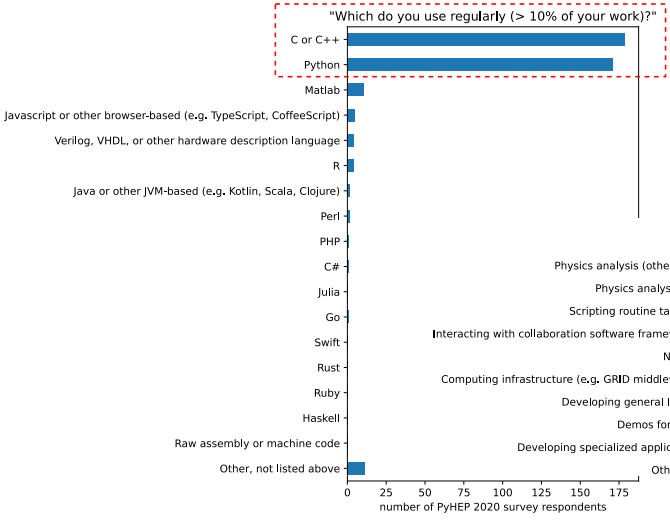


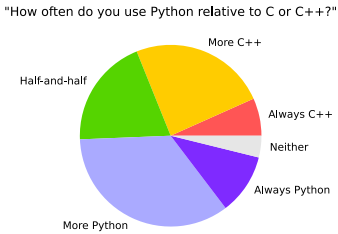
PyHEP 2020 survey respondents ($N = 406$): same picture

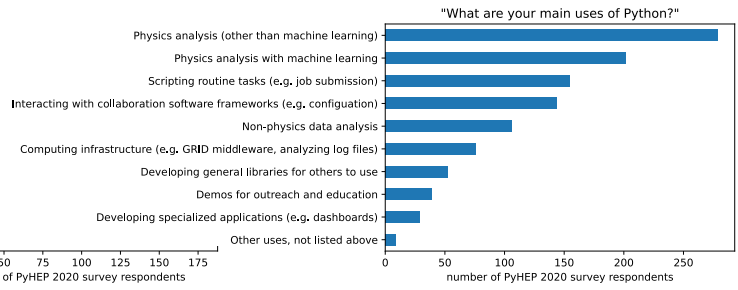




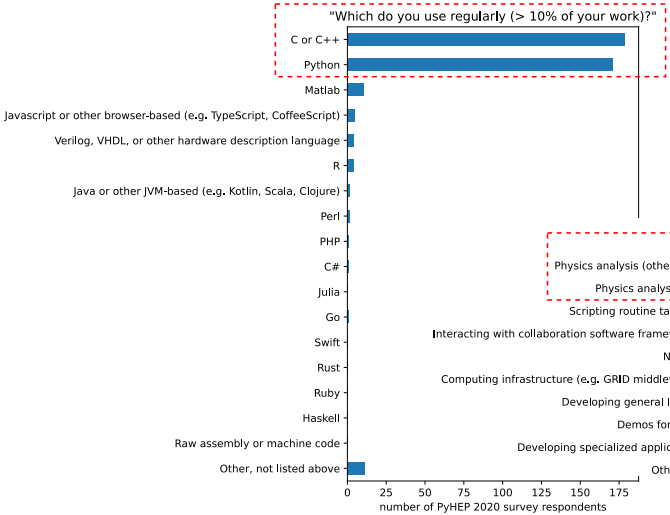
PyHEP 2020 survey respondents ($N = 406$): same picture

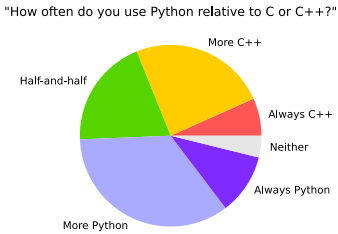


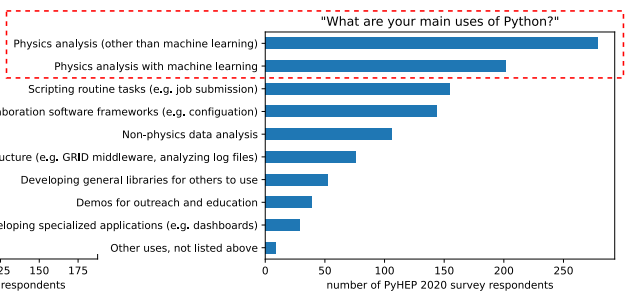




PyHEP 2020 survey respondents ($N = 406$): same picture

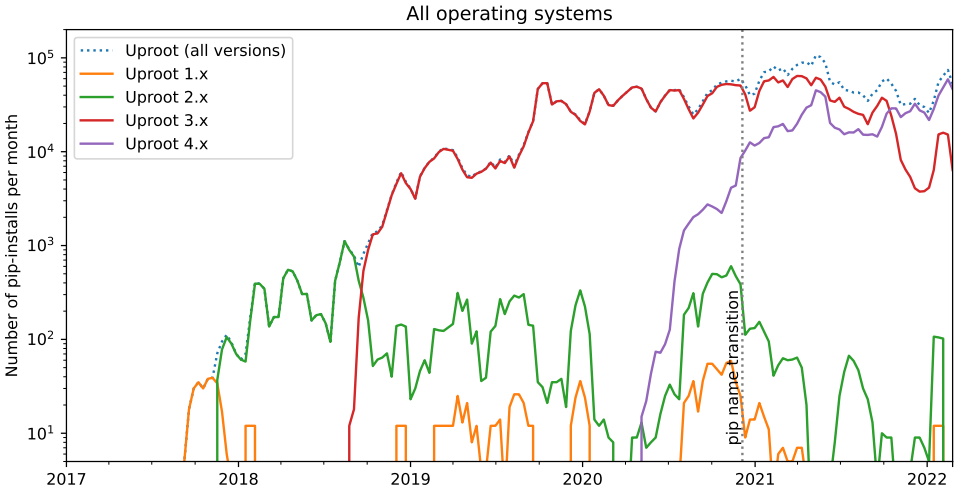






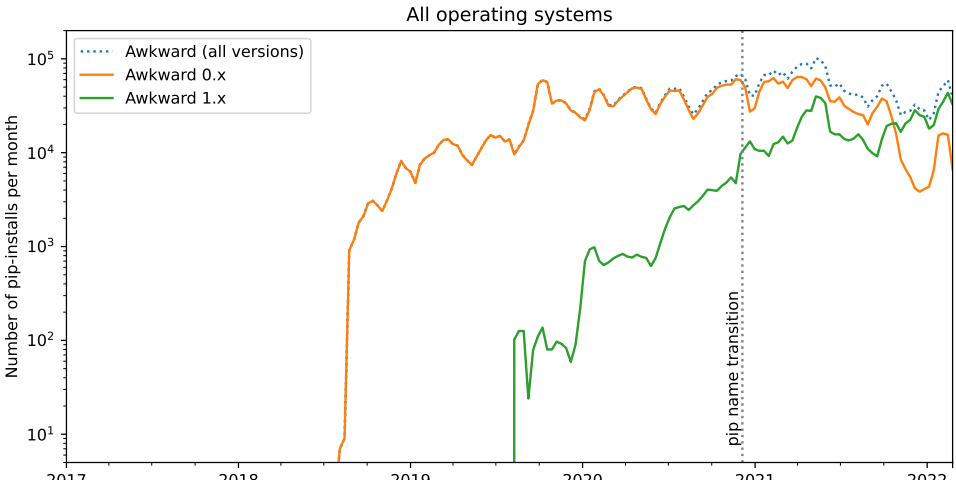
This community is also tolerant of significant API changes

PyPI download statistics of Uproot 3 → 4



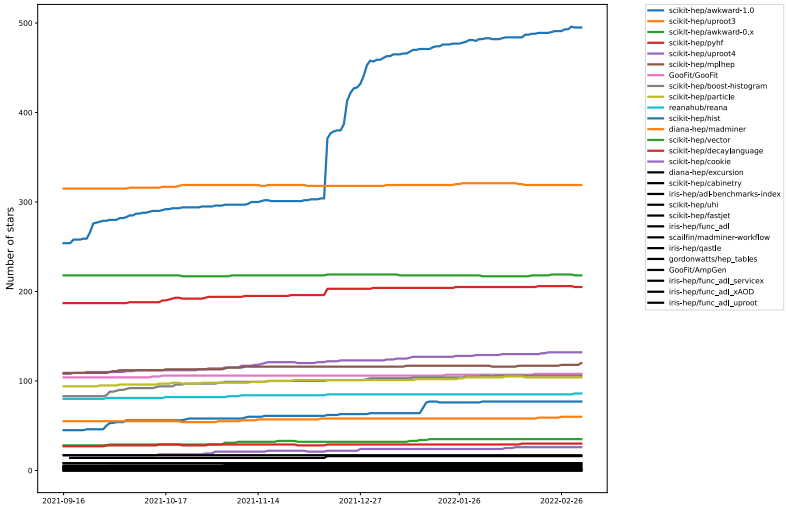
This community is also tolerant of significant API changes

PyPI download statistics of Awkward Array 0 → 1



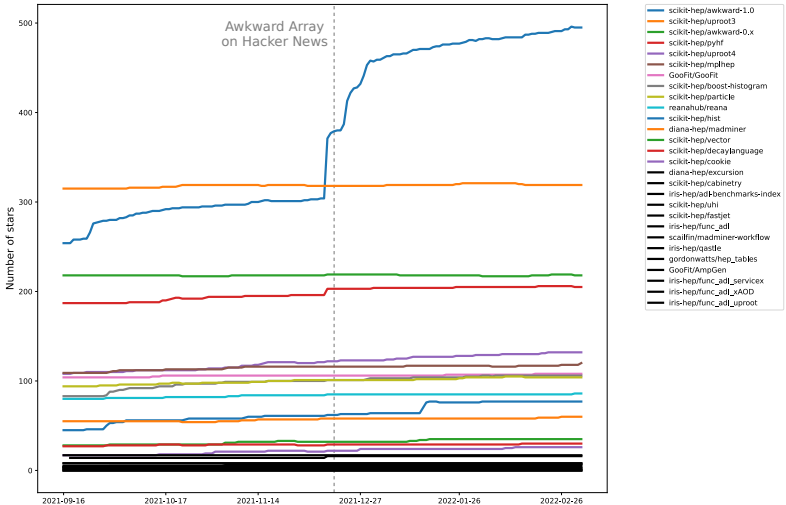
And we're seeing interest from outside the HEP community, too

GitHub stars versus time



And we're seeing interest from outside the HEP community, too

GitHub stars versus time



https://news.ycombinator.com/item?id=29576323