Thoughts on Julia in HEP

Philippe Gras [1], Ben Krikler [2]

[1] Université de Paris-Saclay, CEA/IRFU, France

[2] University of Bristol, UK

Sep 01, 21

Introduction

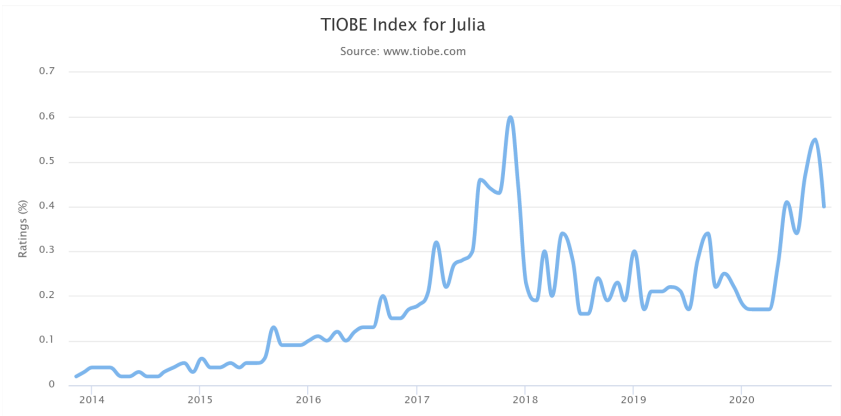
Ju... who?

- Julia is a programming language that bring together ease of Python with the performance of C/C++
  - Development started in 2012 by an MIT team;
  - Fast growing: at 26th position in the TIOBE index of August 2021

"Both Rust and Julia are strong candidates for a permanent top 20 position" – Paul Jansen, CEO TIOBE Software[1].

- Session dedicated to it at the latest PyHEP2021 workshop

In short Julia is a Python-like language with the performance of C/C++



[1] from https://www.tiobe.com/tiobe-index/ consulted on Sep. 1st, 2021

Why looking at another language?

Julia is not just another language

- Julia is unique by providing both high-performance and high-level programming in a single language
  - Makes it, **the ideal language for HEP application**.
- Rising popularity with a large ecosystem
  - In 5th position in the "Most loved language" ranking of the Stackoverflow 2021 survey. Python ranked 6th, C++ 25th;
  - 6,000+ Julia packages registered in JuliaHub;
- Scientific application oriented: Plots.jl and DataFrame.jl are the most used package according the 2021 Julia survey.
  - Adopted by the Climate Modelling Alliance, CliMA project to model the climate, www.nature.com/articles/d41586-019-02310-3.

Why now ?

- HSF looked first at Julia in 2016 during its childhood (4 years old).
- Has reached Today the maturity to be used for HEP applications

Julia specificities

- Easy code writing similar to Python
- Just-in-time compilation (to machine code), with cache mechanism to provide responsiveness
  - Designed for fast compilation
- Supported by Jupyter: the **Ju** of Jupyter.
- Dynamic-type system that approaches static-typed language performance
- Multiple dispatch: a generalisation of OO programming polymorphism
  - $\Rightarrow$ Extremely effective for code reusability, which can be seen in the development of the Julia ecosystem.
- Large ecosystem
- Reproducible environments

Julia in HEP

In a chick-and-egg problem situation

- Developers find the language awesome, but miss "consumers"
- Users aware of the language find it greats, but miss the HEP-specific tools

But already a small community and the interest is growing

- A JuliaHEP github repository
- Publication in last April in Computing and Software for Big Science, Performance of Julia for High Energy Physics Analyses
  M. Stanitzki and J. Strube.
- Julia for data analysis in High Energy Physics presented by M. Mikhasenko (LHCb) at JuliaCon 2021 conference;
- Keno Fischer from Julia Computing invited at the last month CaaS meeting .

A coming mini workshop

Potential of Julia for HEP will be studied within the HSF PyHEP WG

The effort will lead to a written report.

Half-a-day workshop on September 20th afternoon to launch the effort

- Workshop Goals:
  - Aggregate interested people;
  - Agree on the content of the report and on the scope of the study;
  - Identify points that need some work to be answered and start to address them.
- Will be hands-on like.

Interested people should contact Philippe Gras and Ben Krikler.