

Python and Fast Imperative Code: Lowering the Barriers

Monday 9 October 2023 17:30 (30 minutes)

In a typical HEP data analysis process, data is explored by a physicist loading large amounts of data into an interactive Python environment. The physicist performs various analyses of this data. The results of the first analysis tell the physicist what the next steps should be. Python as a dynamically typed language is ideal for this task. The downside is that Python is not very fast.

C++ as a statically typed language is fast. It is perfect for writing the performance critical components that speed things up. Python is used to arrange and connect these components. Thus at runtime the physicist can rearrange these components interactively, without reloading the data.

We will look at a few examples how to write your own analysis components and connect them via:

- * Conversions of Awkward Arrays to and from RDataFrame (C++)
- * Standalone cppy (C++)
- * Passing Awkward Arrays to and from Python functions compiled by Numba
- * Passing Awkward Arrays to Python functions compiled for GPUs by Numba
- Header-only libraries for populating Awkward Arrays from C++ without any Python dependencies

We will introduce Awkward Arrays in Julia via a recent development of Awkward Arrays PyJulia/PyCall.jl-based bridges.

Authors: OSBORNE, Ianna (Princeton University); PIVARSKI, Jim (Princeton University)

Presenter: OSBORNE, Ianna (Princeton University)

Session Classification: Plenary Session Monday