

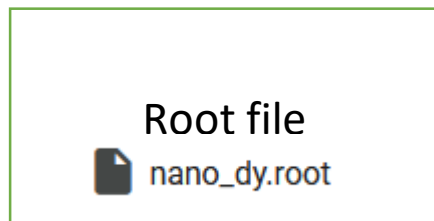
The development of coffea schemas

This project was made by the support of IRIS-HEP program

Speaker: Maxym Naumchyk

Mentors: Ianna Osborne, Peter Fackeldey, Iason Krommydas, Nick Smith (listing order is random)

HEP data analysis using coffea

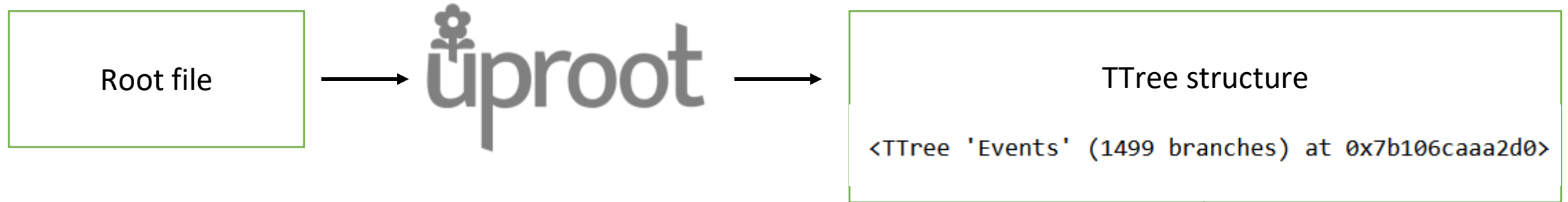


Awkward array

```
[{PSWeight: [1.01, 1.26, 0.99, 0.791], SoftActivityJetNjets2: 5, ...},  
{PSWeight: [2.06, 0.872, ..., 0.962], SoftActivityJetNjets2: 5, ...},  
{PSWeight: [1.07, 0.887, 0.933, 1.02], SoftActivityJetNjets2: 7, ...},  
{PSWeight: [0.833, 0.827, 1.15, 1.1], SoftActivityJetNjets2: 4, ...},  
{PSWeight: [0.936, 0.622, 1.04, 1.17], SoftActivityJetNjets2: 16, ...},  
{PSWeight: [1.17, 1, 0.86, 0.978], SoftActivityJetNjets2: 6, ...},  
{PSWeight: [1.12, 1.59, 0.906, 0.688], SoftActivityJetNjets2: 4, ...},  
{PSWeight: [0.946, 0.922, ..., 0.997], SoftActivityJetNjets2: 12, ...},  
{PSWeight: [0.88, 0.855, 1.09, 0.811], SoftActivityJetNjets2: 3, ...},  
{PSWeight: [0.854, 0.518, 1.11, 1.16], SoftActivityJetNjets2: 20, ...},  
...,
```

```
backend: cpu  
nbytes: 243.3 kB  
type: 40 * event
```

How ffea works



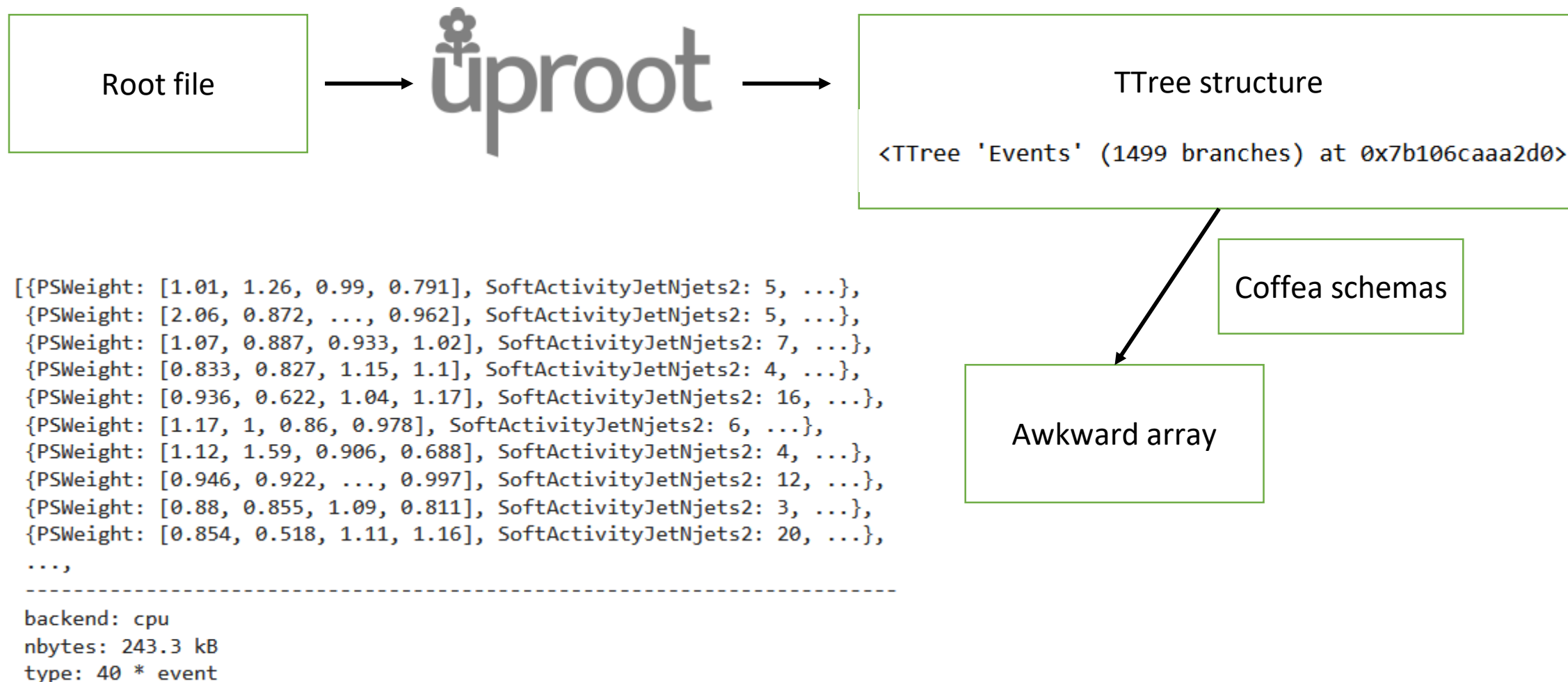
```
[{PSWeight: [1.01, 1.26, 0.99, 0.791], SoftActivityJetNjets2: 5, ...},  
{PSWeight: [2.06, 0.872, ..., 0.962], SoftActivityJetNjets2: 5, ...},  
{PSWeight: [1.07, 0.887, 0.933, 1.02], SoftActivityJetNjets2: 7, ...},  
{PSWeight: [0.833, 0.827, 1.15, 1.1], SoftActivityJetNjets2: 4, ...},  
{PSWeight: [0.936, 0.622, 1.04, 1.17], SoftActivityJetNjets2: 16, ...},  
{PSWeight: [1.17, 1, 0.86, 0.978], SoftActivityJetNjets2: 6, ...},  
{PSWeight: [1.12, 1.59, 0.906, 0.688], SoftActivityJetNjets2: 4, ...},  
{PSWeight: [0.946, 0.922, ..., 0.997], SoftActivityJetNjets2: 12, ...},  
{PSWeight: [0.88, 0.855, 1.09, 0.811], SoftActivityJetNjets2: 3, ...},  
{PSWeight: [0.854, 0.518, 1.11, 1.16], SoftActivityJetNjets2: 20, ...},  
...,
```

```
backend: cpu  
nbytes: 243.3 kB  
type: 40 * event
```

?

Awkward array

How ffea works



coffea schemas

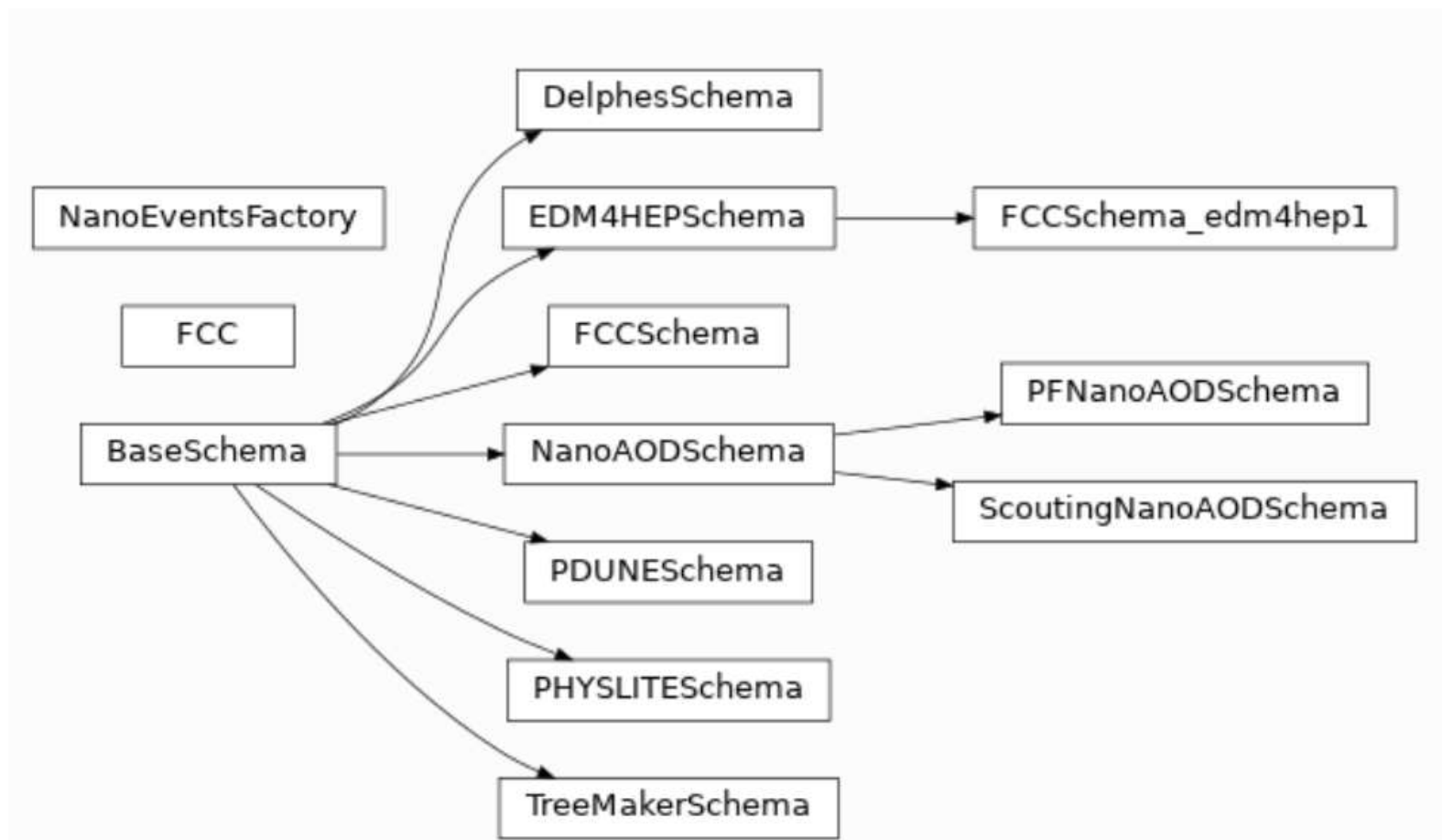


Image link: <https://coffea-hep.readthedocs.io/en/latest/modules/coffea.nanoevents.html#classes>

Awkward-zipper

awkward-zipper Public


Pin Unwatch 3 Fork 1 Star 4

main 3 Branches 0 Tags Go to file Add file Code

maxymnaumchyk Update README.md ✓ c25fb37 · 1 hour ago 97 Commits

.github	add a comment	yesterday
docs	add favicon and html page logo	20 hours ago
src/awkward_zipper	fix: handle collections that have underscore in their names	3 weeks ago
tests	style fix	3 weeks ago
.gitignore	initial commit	3 months ago
.pre-commit-config.yaml	add jupyter notebooks to documentation	yesterday
LICENSE	Initial commit	3 months ago
README.md	Update README.md	1 hour ago
README.rst	Update README.rst	19 hours ago
pyproject.toml	feat: add tests	3 weeks ago

README BSD-3-Clause license



Link: <https://github.com/maxymnaumchyk/awkward-zipper>

About

Documentation page at the link below:

maxymnaumchyk.github.io/awkward-zip...

- Readme
- BSD-3-Clause license
- Activity
- 4 stars
- 3 watching
- 1 fork

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Contributors 2

- maxymnaumchyk
- pfackeldey Peter Fackeldey

Deployments 31

github-pages 1 hour ago

[+ 30 deployments](#)

Documentation

awkward-zipper

Awkward zipper

Search docs

DOCUMENTATION

Examples

- How awkward_zipper works
 - Quick start
 - How awkward-zipper works internally
 - Usage example of awkward_zipper
- Benchmarks
- Documentation

Examples / Quick start [View page source](#)

Quick start

NOTE: You can find this whole page as a jupyter notebook and try it out yourself • ω • here: [notebooks/Usage_example_of_awkward_zipper.ipynb](#)

Before even starting to work with awkward-zipper we have to use `uproot` library to load the data. First we load our root file:

```
# Create a TTree from root
tree = uproot.open("nano_dy.root")["Events"]
```

Then we use `TBranch.arrays` function to convert the TTree format we got from the root file into an awkward array:

```
# TTree -> awkward.Array[awkward.Record[str, awkward.Array]]
array = tree.arrays(ak_add_doc=True)
```

Finally we can pass the resulting array to awkward-zipper.

```
from awkward_zipper import NanoAOD
restructure = NanoAOD(version="latest")
result = restructure(array)
```

Link: https://maxymnaumchyk.github.io/awkward-zipper/usage_example.html

Advantages of a standalone package

- Smaller package
- Only a few dependencies
- Improvements(simplifications) in how the 'schemas' work internally (makes it easier for users to create their own 'schemas')

Plans of development

- Nanoaod format works with eager(materialized) arrays
- Nanoaod format works with virtual(not materialized) arrays
- Eager and virtual modes are well tested and documented

Currently here ->

- First release
- Awkward-zipper supports all '*schemas*' from coffea
- Coffea uses awkward-zipper instead of its own '*schemas*'
- All the new schemas are developed on zipper directly

Thank you for attention!

Full presentation link: <https://github.com/maxymnaumchyk/Awkward-zipper-project-presentation>