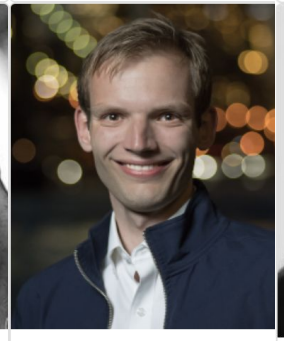
IRIS/HE Retreat: Analysis Systems

Kyle Cranmer (NYU)

Analysis Systems Team

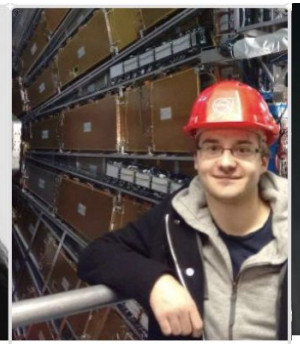
Institutions: NYU, Washington, Princeton, Cincinnati, Illinois

Kyle Cranmer
New York University



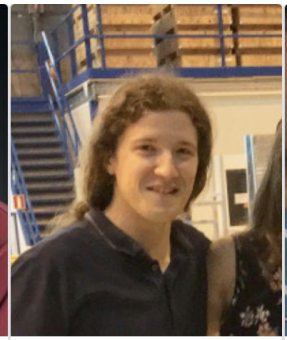
Johann Brehmer
New York University

Irina Espejo
New York University



Alexander Held
New York University

Gordon Watts
University of Washington

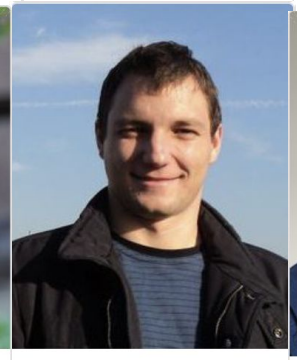


Mason Proffitt
University of Washington

Emma Torro
University of Washington

Ianna Osborne
Princeton University

Jim Pivarski
Princeton University



Vassil Vassilev
Princeton University

Henry Schreiner
Princeton University

Mike Sokoloff
University of Cincinnati

Ben Galwesky
National Center for Supercomputing Applications

Mark Neubauer
University of Illinois at Urbana-Champaign

Daniel S. Katz
University of Illinois at Urbana-Champaign

Matthew Feickert
University of Illinois at Urbana-Champaign

Prior to IRIS-HEP

Bulk Data Processing

Reconstruction Algorithms

Analysis Code

Analysis code in HEP is often more free-form with less organized development:

- one-off approach limits functionality
- slow iteration cycle
- slow on-boarding and lack of interoperability
- difficult to reproduce and reuse

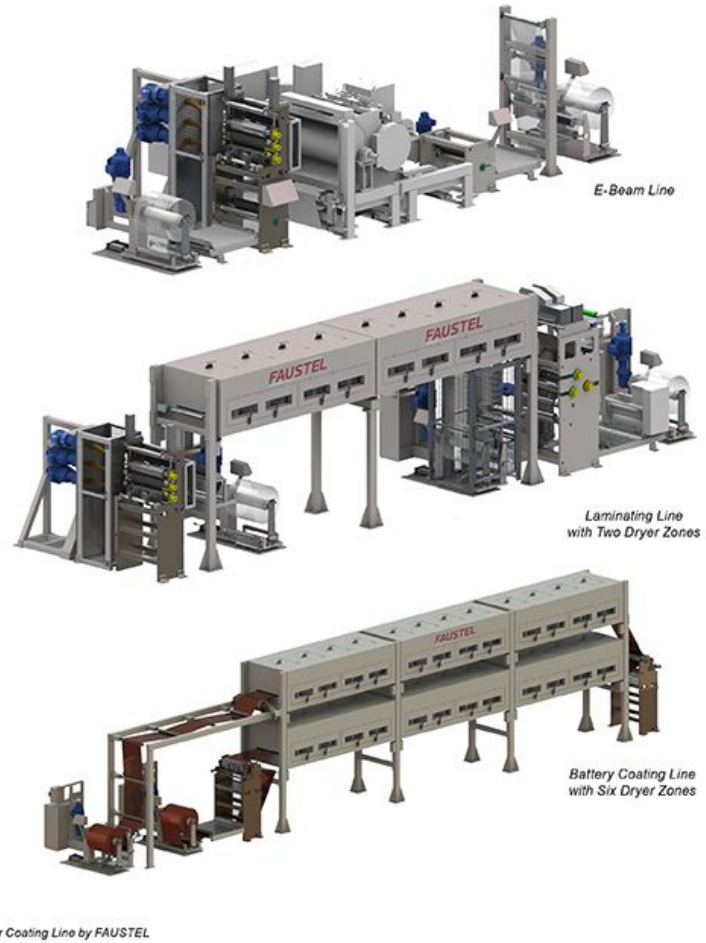
- primarily ROOT & C++
- lack of developer community
- overlapping solutions
- data redundancy

Analysis Systems

ad hoc analysis code

IRIS-HEP

Analysis Systems

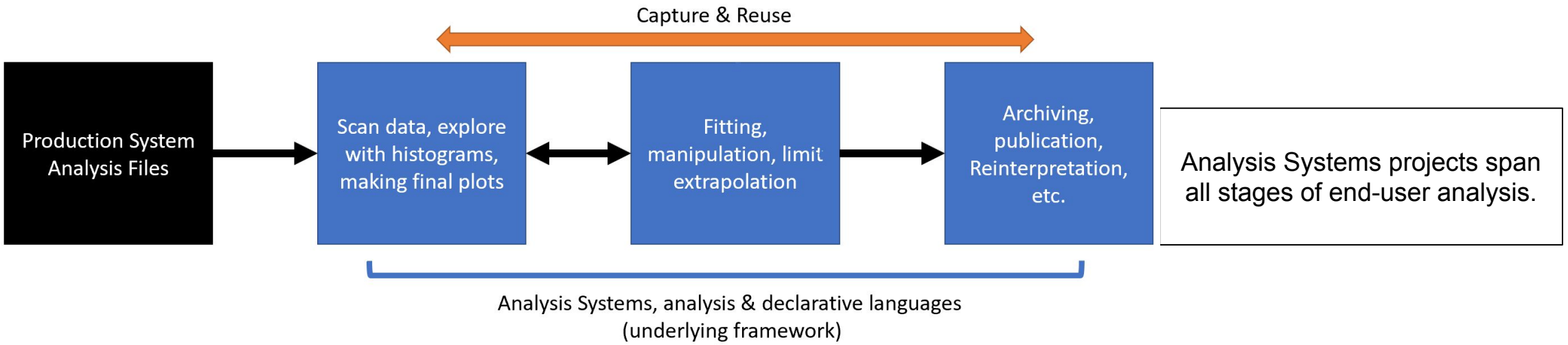


Analysis Systems strategies:

- improve functionality & interoperability
- more modular, less dependence on ROOT
- declarative: focus on what to do not how to do it
- align with modern data science practices

Analysis Systems

- Develop sustainable analysis tools to extend the physics reach of the HL-LHC experiments
  - *create greater functionality to enable new techniques,*
  - *reducing time-to-insight and physics,*
  - *lowering the barriers for smaller teams, and*
  - *streamlining analysis preservation, reproducibility, and reuse.*



Value of IRIS-HEP as an Institute

IRIS-HEP as a tugboat:

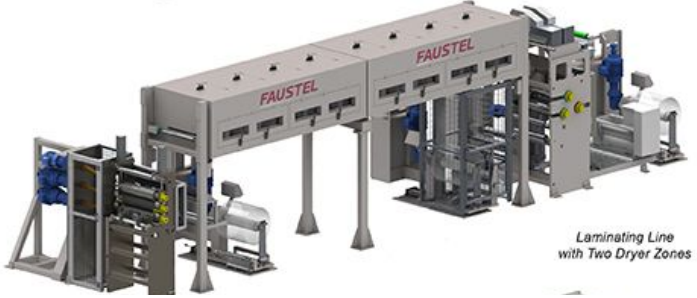
- direct and navigate large efforts in the collaborations with significant inertia
- take advantage of consistent presence and messaging within the large collaborations
- Examples:
  - *pythonic analysis tools*
  - *software practices*
  - *industry-standards*

Value of IRIS-HEP as an Institute

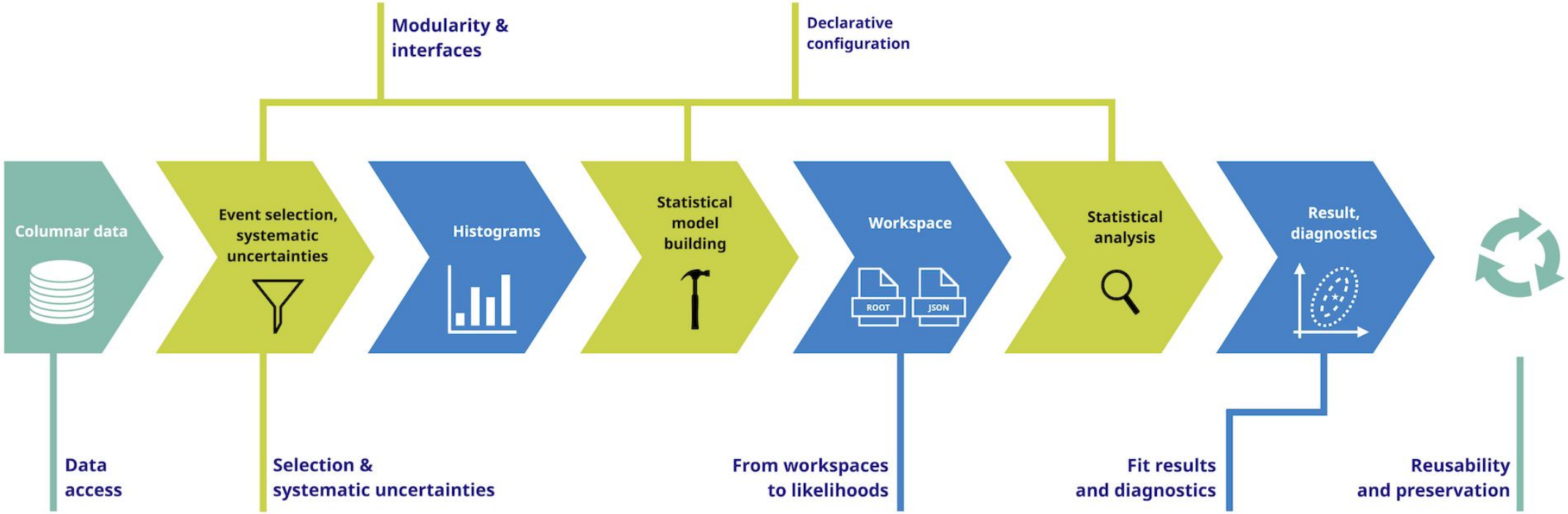
IRIS-HEP as a lighthouse:

- provide cohesive, long-term vision for how software should evolve to meet needs of HL-LHC
- take advantage of holistic perspective of the institute
- Examples:
  - *columnar analysis*
  - *declarative programming*
  - *differentiable programming*
  - *preservation & reuse*

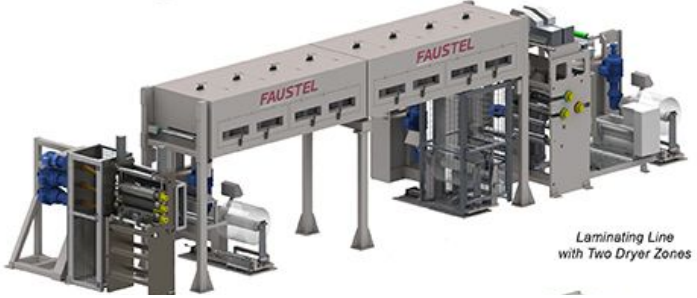
A coherent ecosystem

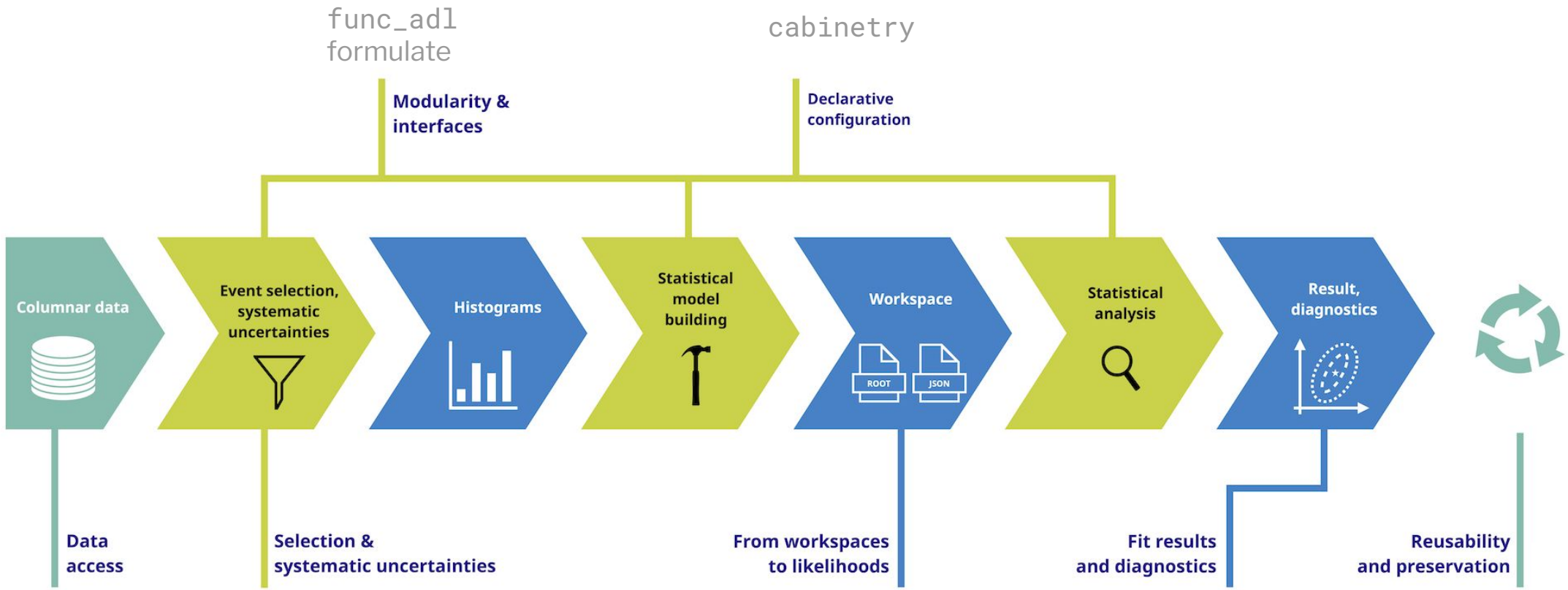


One of our analysis use cases involves a vertical slice from ServiceX to final limits for a real-world ATLAS Higgs analysis. See Alex Held's poster.



A coherent ecosystem





ServiceX

uproot

Awkward Array

coffea

Boost Histogram

pyhf differentiable likelihoods

yadage

recast

Y3 Milestones

Integration:

- Many of the individual tools are at beta stage or better.
- Increase our efforts towards integrating tools into systems (vertical slices)
  - *this is expected to expose areas where tools can be improved, modified, etc.*
- More explicit coordination and planning with Analysis Facility / DOMA / SSL
- An important step towards almost any Grand Challenge involving Analysis Systems

Adoption:

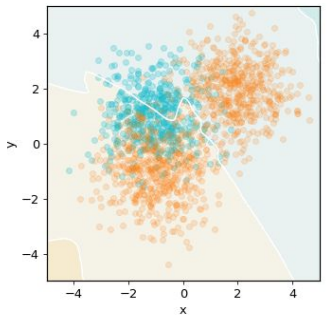
- Some of our tools and projects are at “tipping point,” rapidly gaining traction within experiments and in user communities.
  - *Example: pyhf adoption is rapid (papers, likelihood publishing, etc.)*
  - *Example: ATLAS is ramping up RECAST efforts (papers to come near end Y3)*
  - *Example: Scikit-hep as an example of community-driven software effort*
- Good to invest effort in these areas for results and to build IRIS-HEP reputation
  - *Development, Training, Documentation*
  - *Misc. experiment specific contributions also valuable for ”delivery to experiments”*

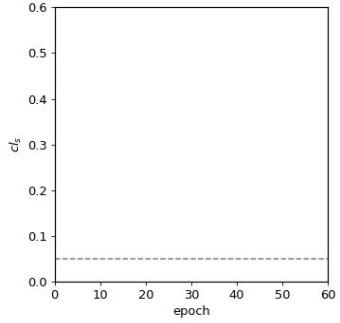
Questions from Area Lead Chat:

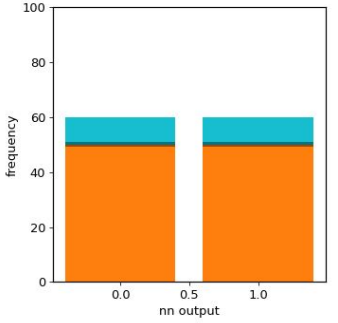
What would be potential Year 3 milestones for each of the projects? (First ideas, to be iterated with PIs and the whole team as this process moves forward.)

- Integration of `func_adl` specification for variable definition and selections with the emerging `cabinetry` specification for high-level template fit analysis
- Demonstration of differentiable analysis pipeline (eg `cabinetry`) ending with `pyhf` limit back-propagating through selection implemented with awkward, `func_adl`, etc.
  - *connect to `pyhf/neos` demo. Need autodiff-able analysis over awkward arrays*
  - *connect to histogramming projects*
  - *Discussion in Slack to connect this with the Sally algorithm in MadMiner*
- Documentation and training event using new tools
- Use of new IRIS-HEP tools (MadMiner, awkward, ...) for analysis in LHC experiment (may not be published by end of Y3)
- Snowmass (tools & REANA workflows for sensitivity studies)

gradHEP



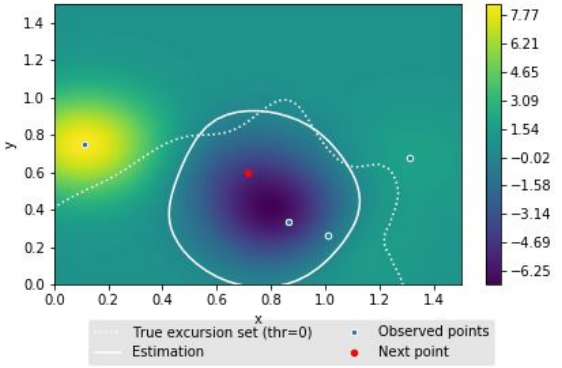




Questions from Area Lead Chat:

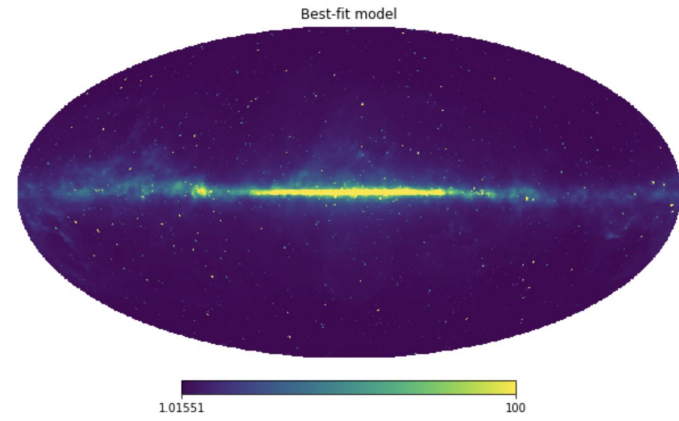
Are there new opportunities where effort from IRIS-HEP can make an impact? Is the alignment of the focus areas in IRIS-HEP appropriate?

- Visualization tools (eg. altair like declarative visualization)
  - *yes for AS, but expansion of scope*
- `excursion` alg. to streamline MC production for ATLAS or CMS reinterpretation campaign
  - *yes for AS*
- Improve efficiency of event gen. with ML-inspired tools & techniques
  - *yes for AS, but an expansion into "theory" tools*
- pyhf and astrophysics (HEALPix for boost histogram)
  - *yes for AS, but secondary aim of IRIS-HEP*
- MadMiner like tools for EIC
  - *yes for AS, but secondary aim of IRIS-HEP. Brought up at 18 mo review*
- python library for fastjet that plays well with columnar analysis
- Documentation efforts
  - *yes, aligns with "lowering barriers" goal of AS*



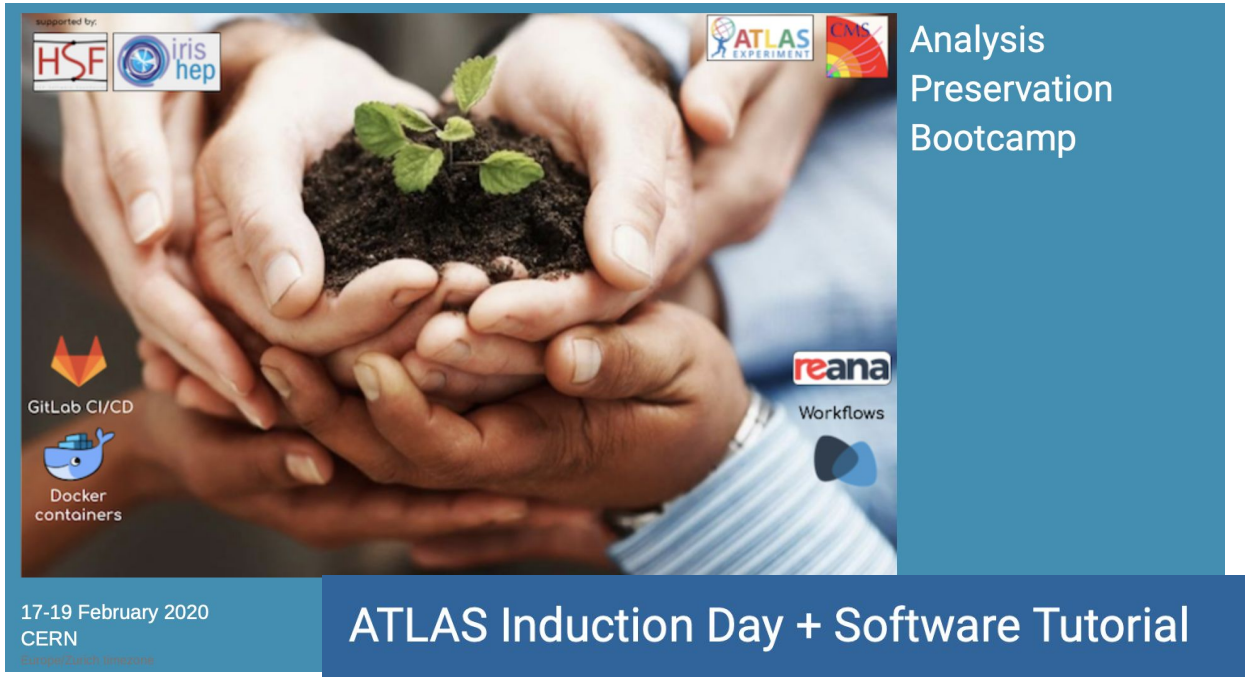

```
m = pyhf.Model(spec, poiname = 'mu_dm')
bestfit = pyhf.optimizer.minimize(
    lambda theta,data,m: -m.logpdf(theta, data),data,m,
    init_pars = [1]*5,
    par_bounds = [[0,20]]*5
)
```

```
hp.mollview(m.expected_data(bestfit), max=100, title='Best-fit model')
```



Backup

Training



ATLAS Induction Day + Software Tutorial

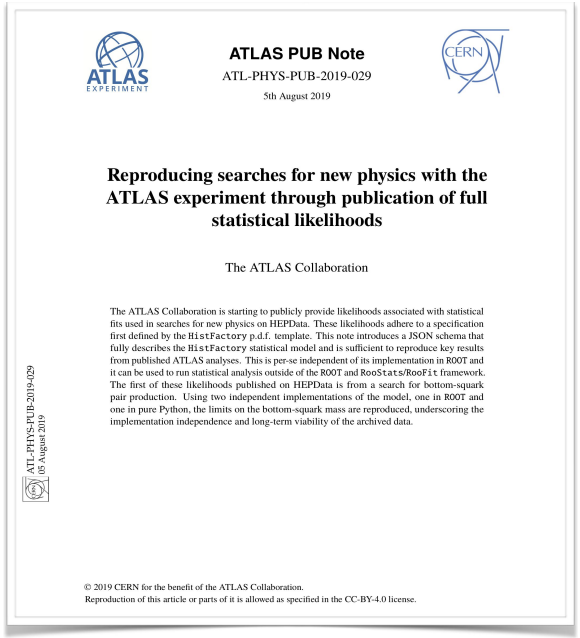
21-25 October 2019
CERN
Europe/Zurich timezone

Introduction to pyhf	*Giordon Holtsberg Stark et al.*	
222/R-001, CERN		14:00 - 14:30
Hands-on with pyhf	*Giordon Holtsberg Stark et al.*	
Docker Analysis Release Containers	*Lukas Alexander Heinrich*	
222/R-001, CERN		16:30 - 16:50
Using GitLab for Analysis Code Management	*Giordon Holtsberg Stark*	
222/R-001, CERN		17:00 - 17:20

Highlight

- The field is at a tipping point, DIANA/DASPOS/IRIS-HEP contributions have been transformational.
- First results using the RECAST reinterpretation framework and publishing full statistical likelihoods (using pyhf)

ROOT: 10+ hours
pyhf: < 30 minutes



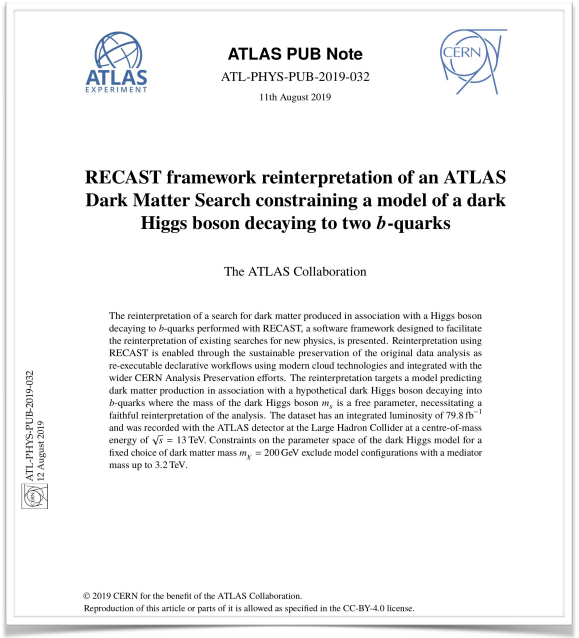
ATLAS PUB Note
ATL-PHYS-PUB-2019-029
5th August 2019

Reproducing searches for new physics with the ATLAS experiment through publication of full statistical likelihoods

The ATLAS Collaboration

The ATLAS Collaboration is starting to publicly provide likelihoods associated with statistical fits used in searches for new physics on HEPData. These likelihoods adhere to a specification first defined by the `HistFactory` p.d.f. template. This note introduces a JSON schema that fully describes the `HistFactory` statistical model and is sufficient to reproduce key results from published ATLAS analyses. This is per-se independent of its implementation in ROOT and it can be used to run statistical analysis outside of the ROOT and `RooStats`/`RooFit` framework. The first of these likelihoods published on HEPData is from a search for bottom-squark pair production. Using two independent implementations of the model, one in ROOT and one in pure Python, the limits on the bottom-squark mass are reproduced, underscoring the implementation independence and long-term viability of the archived data.

ATL-PHYS-PUB-2019-029
05 August 2019



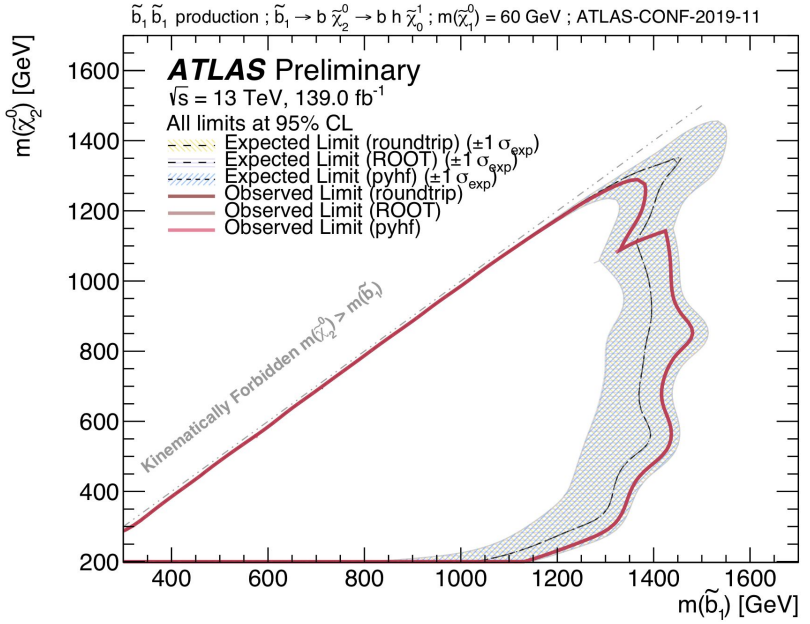
ATLAS PUB Note
ATL-PHYS-PUB-2019-032
11th August 2019

RECAST framework reinterpretation of an ATLAS Dark Matter Search constraining a model of a dark Higgs boson decaying to two *b*-quarks

The ATLAS Collaboration

The reinterpretation of a search for dark matter produced in association with a Higgs boson decaying to *b*-quarks performed with RECAST, a software framework designed to facilitate the reinterpretation of existing searches for new physics, is presented. Reinterpretation using RECAST is enabled through the sustainable preservation of the original data analysis as re-executable declarative workflows using modern cloud technologies and integrated with the wider CERN Analysis Preservation efforts. The reinterpretation targets a model predicting dark matter production in association with a hypothetical dark Higgs boson decaying into *b*-quarks where the mass of the dark Higgs boson m_s is a free parameter, necessitating a faithful reinterpretation of the analysis. The dataset has an integrated luminosity of 79.8 fb^{-1} and was recorded with the ATLAS detector at the Large Hadron Collider at a centre-of-mass energy of $\sqrt{s}$ = 13 TeV. Constraints on the parameter space of the dark Higgs model for a fixed choice of dark matter mass m_χ = 200 GeV exclude model configurations with a mediator mass up to 3.2 TeV.

ATL-PHYS-PUB-2019-032
12 August 2019



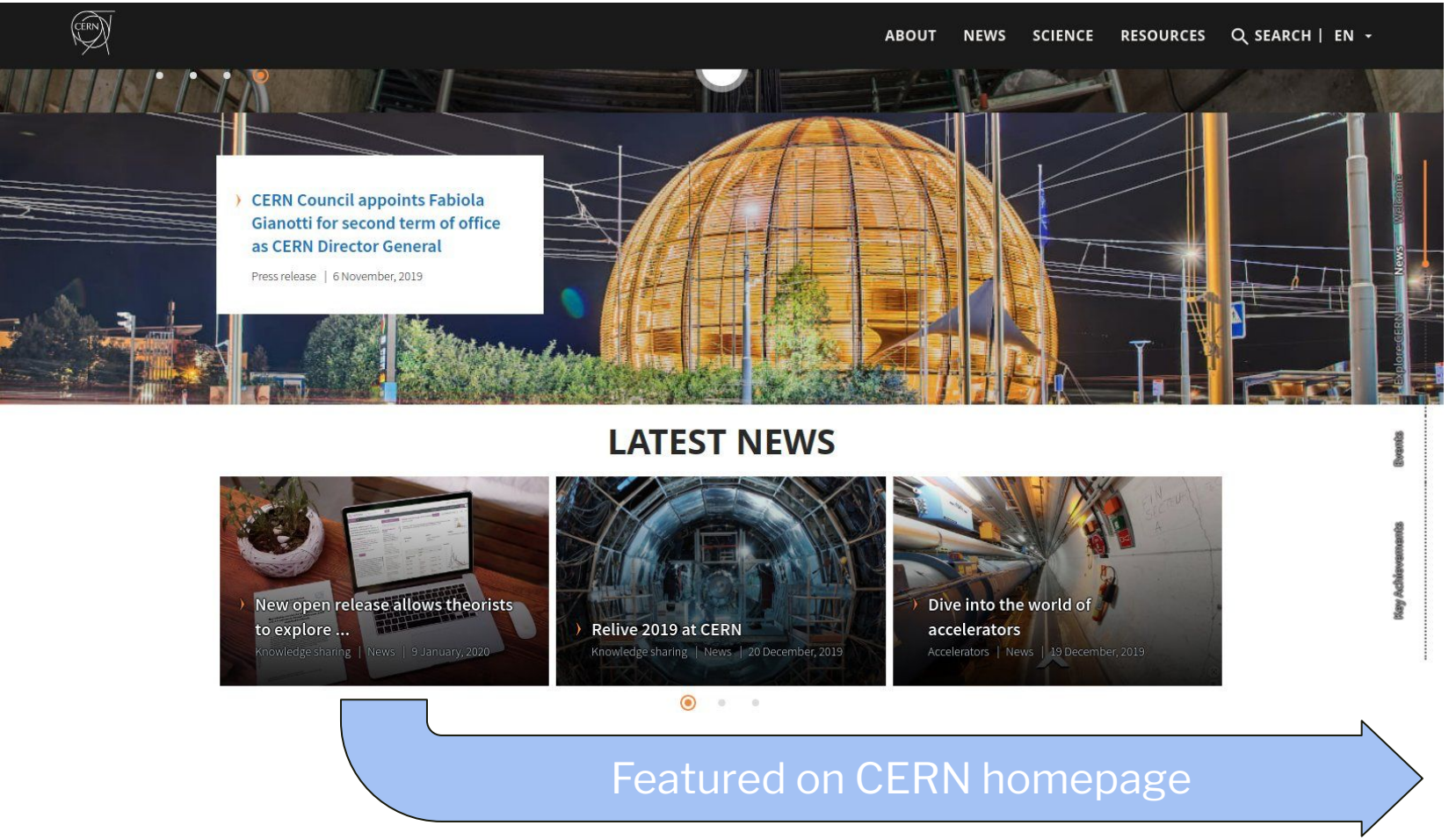
Highlight

HEPData
@HEPData

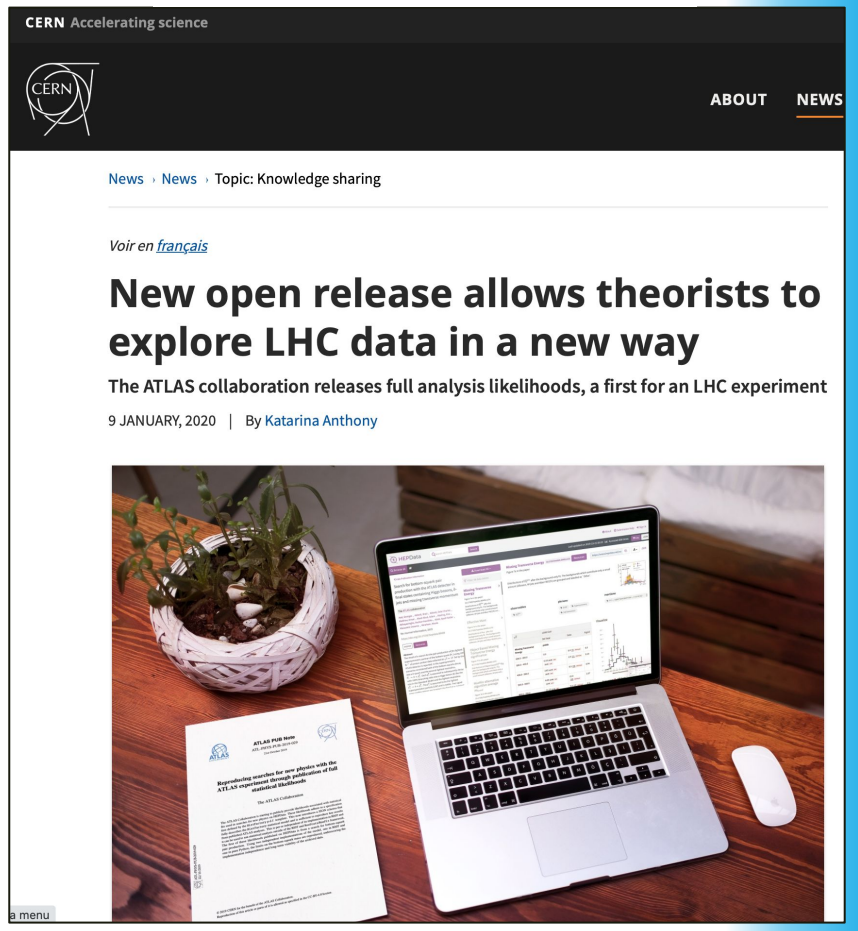
Thanks @KyleCranmer for your support and promotion of @HEPData over several years. Looking forward to future collaboration with @iris_hep on #pyhf likelihoods and more.

> **Kyle Cranmer** @KyleCranmer · Jan 29
> I would like to applaud @STFC_Matters for funding @HEPData, a vital piece of cyberinfrastructure for HEP. The @NSF has been supporting HEP software and cyberinfrastructure with DASPOS, @diana_hep and @iris_hep. @iris_hep looks forward to collaborating with you! twitter.com/HEPData/status...

1:15 PM · Jan 30, 2020 · Twitter Web App

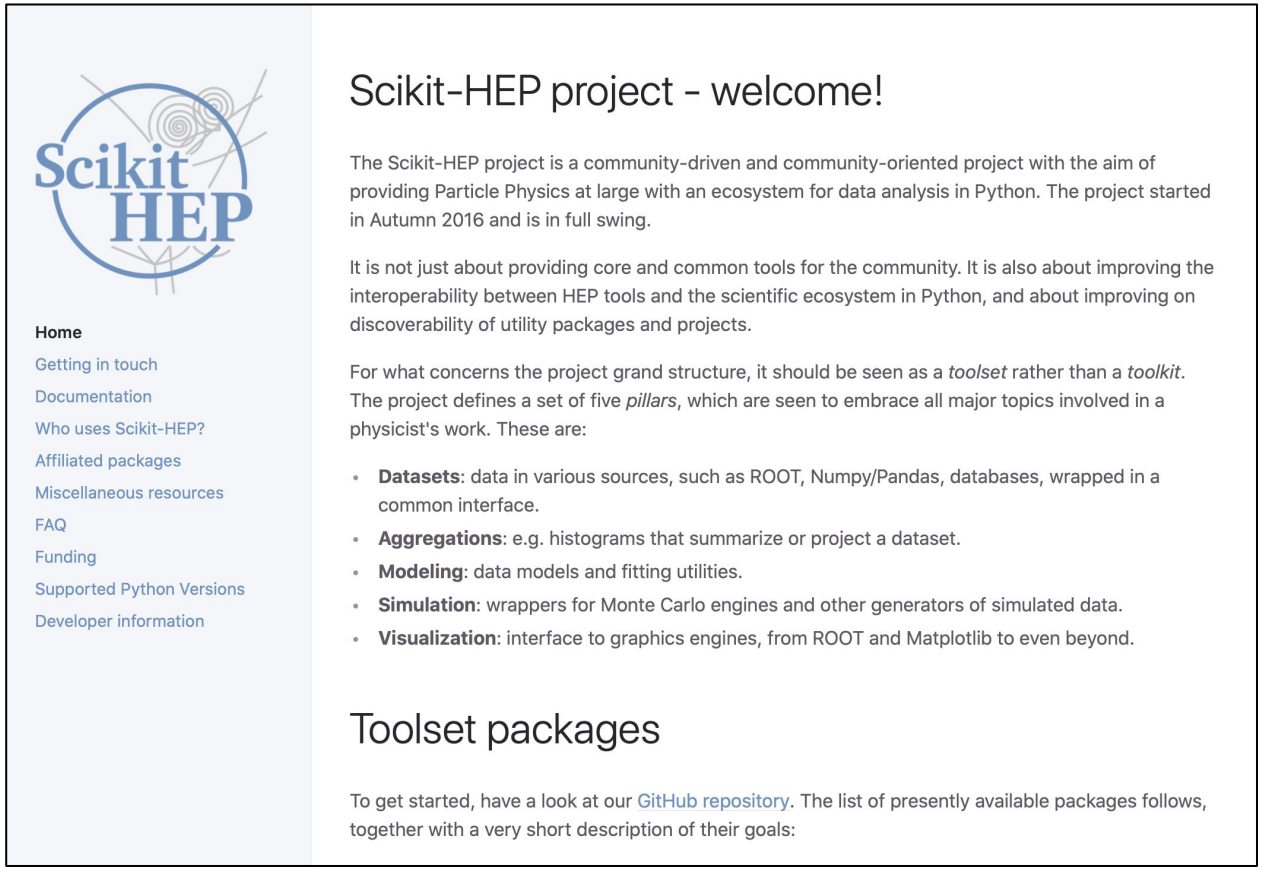


Featured on CERN homepage



Scikit-HEP

A broad community project with heavy IRIS-HEP involvement.



Scikit HEP

Home
Getting in touch
Documentation
Who uses Scikit-HEP?
Affiliated packages
Miscellaneous resources
FAQ
Funding
Supported Python Versions
Developer information

Scikit-HEP project - welcome!

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. The project started in Autumn 2016 and is in full swing.

It is not just about providing core and common tools for the community. It is also about improving the interoperability between HEP tools and the scientific ecosystem in Python, and about improving on discoverability of utility packages and projects.

For what concerns the project grand structure, it should be seen as a *toolset* rather than a *toolkit*. The project defines a set of five *pillars*, which are seen to embrace all major topics involved in a physicist's work. These are:

- **Datasets**: data in various sources, such as ROOT, Numpy/Pandas, databases, wrapped in a common interface.
- **Aggregations**: e.g. histograms that summarize or project a dataset.
- **Modeling**: data models and fitting utilities.
- **Simulation**: wrappers for Monte Carlo engines and other generators of simulated data.
- **Visualization**: interface to graphics engines, from ROOT and Matplotlib to even beyond.

Toolset packages

To get started, have a look at our GitHub repository. The list of presently available packages follows, together with a very short description of their goals:

Basics:

Awkward Array

awkward-array : Manipulate arrays of complex data structures as easily as Numpy.
pypi v0.12.20 conda-forge v0.12.20

hepunits : Units and constants in the HEP system of units.
pypi v1.1.1

Data manipulation and interoperability:

formulate : Easy conversions between different styles of expressions.
pypi v0.0.8

root_numpy : Interface between ROOT and NumPy.
pypi v4.8.0 conda-forge v4.8.0

root_pandas : Module for conveniently loading/saving ROOT files as pandas DataFrames.
pypi v0.7.0 conda-forge v0.7.0

uproot

uproot : Minimalist ROOT I/O in pure Python and Numpy.
pypi v3.11.3 conda-forge v3.11.3

uproot-methods : Pythonic behaviours for non-I/O related ROOT classes.
pypi v0.7.3 conda-forge v0.7.3

Histogramming:

aghast

aghast : Convert between histogram representations
pypi v0.2.1 conda-forge v0.2.1

Boost Histogram

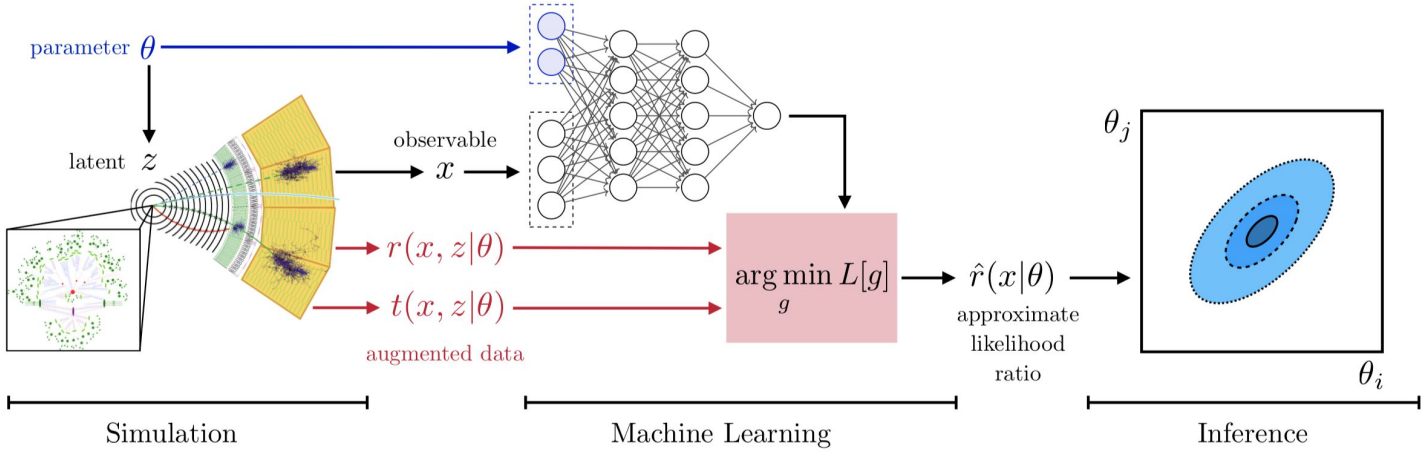
boost-histogram : Python bindings for the C++14 Boost::Histogram library.
pypi v0.6.2 conda-forge v0.6.2

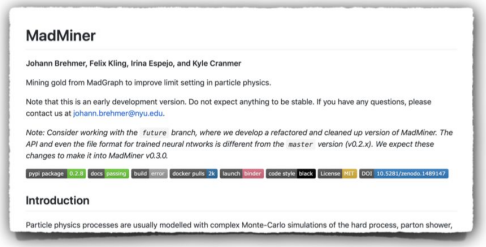
NSF

The Future

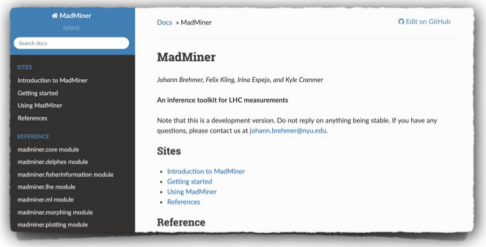
Tight integration of

- Simulation
- Machine Learning
- Statistical Inference

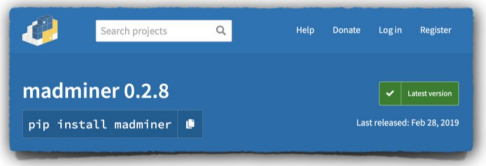




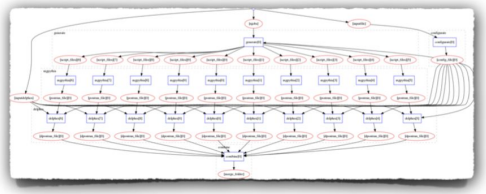
Repository and tutorials:
github.com/johannbrehmer/madminer



Documentation:
madminer.readthedocs.io

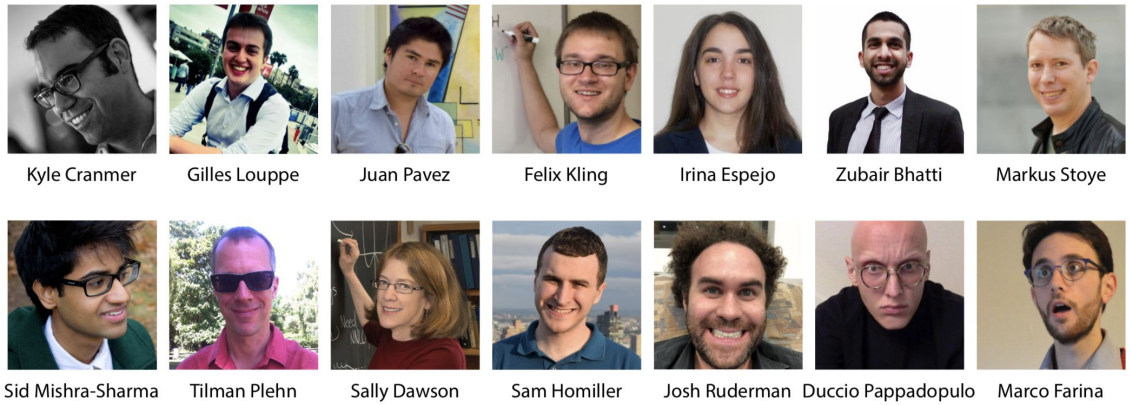


Installation:
`pip install madminer`



Deployment with Docker, yadage, REANA:
github.com/irinaespejo/workflow-madminer

34/40



Thanks to Kyle, Gilles, Felix, Irina, and Sam for material and inspiration for slides!

NYU MSDSE dianahep iris hep Institute for Research and Innovation in Software for High Energy Physics

The SCAILFIN Project
scailfin.github.io

Slides from Johann Brehmer's Keynote talk at ACAT on Constraining Effective Field Theories with Machine Learning

Major Activities

- Development of declarative specifications for different stages of analysis
- Identification and benchmarking of traditional implementations for benchmark example use-cases that span the scope of AS
- Implementation of prototype components & integration
  - *connection with DOMA (particularly ServiceX)*
- Benchmarking and assessment of prototype implementations and declarative specifications for the same example use cases
  - *connection with SSL (dedicated Blueprint Activity)*
- Exploratory research in machine learning that may impact how analysis is performed
- Engagement with community of early adopters and developers

Are there internal or external collaborations associated with each project or activity? For external collaborations, is IRIS-HEP leading, contributing or simply “connecting/liaising”?

Internal:

- **SSL**: benchmarking and scaling, REANA testbeds, etc.
- **SSL** & **DOMA**: ServiceX

External:

- **DIANA/HEP:** last bits of funding on NCE supporting various items very aligned
- **SCAILFIN**: developing products, good synergy w/ IRIS-HEP. **REANA** dev team
- **INSPIRE-HEP, HEPData, CAP, Invenio:** Advisory boards, join in development
- **ATLAS** stats effort: docker containers for RooFit-based statistical analysis & combinations and development of pyhf tools. IRIS-HEP (Matthew, Kyle, Alex) & Lukas & Giordon are leading
- HEP Statistics Serialization Standard (HS3) similar cast of characters
- **scikit-hep:** useful umbrella (not seen as US, ATLAS/CMS, or HSF) IRIS-HEP leading by example
  - *Awkward:*
    - formal collaboration with Amy Roberts at UC Denver on **Kaitai Structs**
    - frequent collaboration with **LPC/Coffea** (Lindsey Gray)
    - close liaisons with **Anaconda.com**: Numba and Dask developers
    - intermittent contact with **Oxford Big Data Institute** (genetics, developers of **Zarr**)

Projects

- Analysis systems are connected to analysis use cases
- Systems are composed of components
- Most of these projects refer to those components
  - *many projects include people beyond IRIS-HEP*
- Milestones and activities mainly oriented towards integration, evaluation, with a global overview of the vertical slice

ADL Benchmarks

Functionality benchmarks for analysis description languages

More information

AmpGen

Generation and fitting for multibody hadron decays

More information

Awkward Array

Manipulate arrays of complex data structures

More information

DecayLanguage

Describe and convert particle decays

More information

Functional ADL

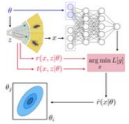
Functional Analysis Description Language

More information

Histogram projects

Histogramming efforts

More information



MadMiner

Likelihood-free Inference

More information

Particle

Pythonic particle information

More information

ROOT on Conda Forge

Use ROOT in Conda through Conda-Forge

More information

Scikit-HEP

pythonic analysis tools

More information

awesome-hep

A curated list of awesome high energy and particle physics software

More information

exploratory-ml

Analysis Reinterpretation

More information

ppx

cross-platform Probabilistic Programming eXecution protocol

More information

pyhf

Differentiable likelihoods

More information

recast

Analysis Reinterpretation

More information

uproot

Read and write ROOT files in Python

More information