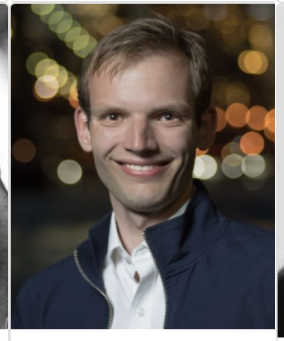
Response to Questions/Requests for Analysis Systems

Kyle Cranmer (NYU)

Analysis Systems Team

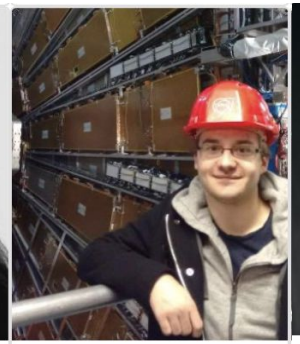
Institutions: NYU, Washington, Princeton, Cincinnati, Illinois

Kyle Cranmer
New York University



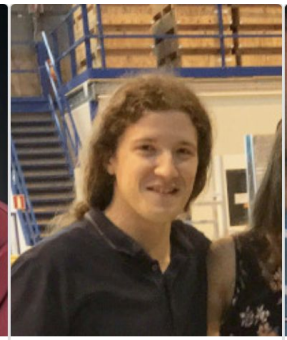
Johann Brehmer
New York University

Irina Espejo
New York University



Alexander Held
New York University

Gordon Watts
University of Washington

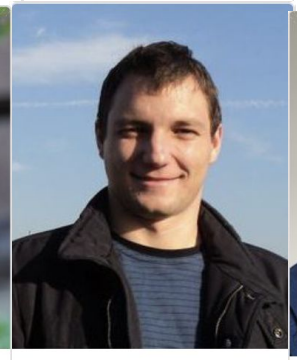


Mason Proffitt
University of Washington

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

Summarize the set of projects/activities and associated effort for your area

See next two pages

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

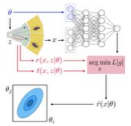
Functional Analysis Description Language

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

Summarize the set of projects/activities and associated effort for your area

It should be easier for area leads to see the FTEs

ServiceX activities overlap with DOMA/SSL, not sure how to deal with accounting for the purposes of this meeting.

Awkward, uproot:

- Jim P. Ianna O. (were funded largely on DIANA-HEP until ~now)

func_adl & ServiceX:

- Gordon + Mason @ UW in collaboration with Ben G & Marc W.

pyhf: Matthew Feickert (~80%)

MadMiner & Exploratory ML: Johann Brehmer (~50%)

Integrating tools & developing declarative specifications for end-user:

- Alexander Held (~80%)

Modernizing analysis support (eg. docker, analysis preservation, training, conda forge, etc...)

- Matthew Feickert (~20%), Alexander Held (~20%), Henry Schreiner (~25%?)

Cyberinfrastructure: Sinclert Perez (~30% starting 2020, HEPData and RECAST)

Core Scikit-HEP : boost-histogram, hist, vector, ... Henry Schreiner (~65%)

Misc Scikit-HEP: scikit-hep org, particle, decay language Henry Schreiner (10%), Daniel Vieira (???) not participating in biweekly meetings)

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**)

Which project/activities/goals are making progress and which are not? (Area lead's opinion) For those that are not, what is impeding progress?

Projects making good progress with good adoption:

- uproot and awkward usage in physics and by developers
- Service X & func_adl code development is active and groups seem to coordinate well
- pyhf adoption in physics analysis (example: ATLAS SUSY EWK using for official combinations)
- REANA growing in popularity and planning to deploy at facilities
- MadMiner interest is growing: ~100 person tutorial, had full of ATLAS & CMS people trying it out
- Scikit-hep community growing rapidly, interest from stats committees & physics groups of experiments

Some issues / concerns

- ServiceX for CMS had some bumps, but that is being addressed. Large effort should have big win.
- LHCb effort in GooFit/AmpGen/DecayLanguage not well integrated (though conceptually good fit)
- Role of "Analysis Facility" not part of discussions in Analysis Systems group, not well integrated.

Basically completed

- Awkward (1.0? clarify)
- Uproot 3.x (future is Uproot 4.x)
- PPX protocol
- Boost histogram (future is hist)
- MadMiner as a tool (user community growing)

analysis facility blueprint needed
A gap in planning.

How are each of these projects/activities connected to, being informed by or planning on delivering (eventually) to the experiments? Are there relevant blueprint meetings or workshops that should happen to make progress?

- IRIS-HEP stats tools making good progress in ATLAS
- RECAST, pyhf, analysis preservation making good progress in ATLAS
  - *Clemens Lange looking for RECAST contributors for CMS, added a Fellows project to encourage this.*
- MadMiner is having tires kicked by ATLAS & CMS experimentalists
  - *closely related reweighting based on CARL an ATLAS qualification project*
- Coffea being picked up increasingly in CMS
  - *corresponding effort needed for ATLAS and LHCb. IRIS-HEP fellow project added to encourage this*
  - *Alex Held and KyungEon working on TRExFitter analog using these tools*
- A blueprint connected to AS grand challenge & analysis facility may help
- ServiceX in R&D phase now, but need to check on planning for ServiceX for analysis facility: Skyhook, River, …
  - *answer requested on SLACK ServiceX channel (next page)*

KyungEon doesn't seem to formally be part of IRIS-HEP.
He should be.
Maybe a fellow?

How are each of these projects/activities connected to, being informed by or planning on delivering (eventually) to the experiments? Are there relevant blueprint meetings or workshops that should happen to make progress?

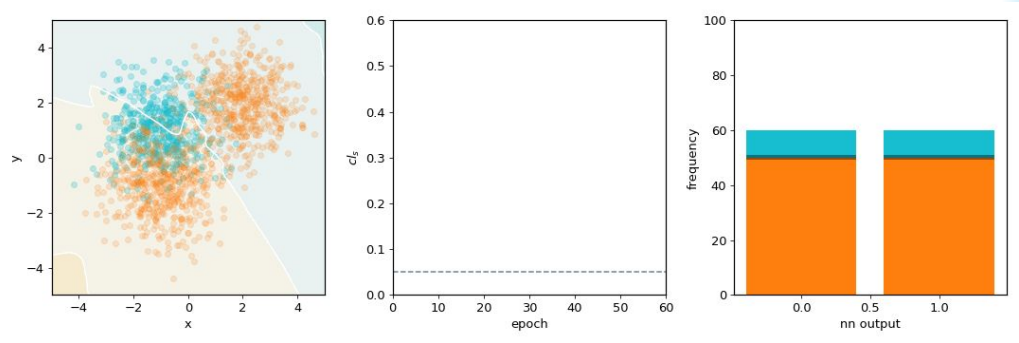
ServiceX is designed to facilitate high-performance array-based analyses. It does this by allowing users to construct sophisticated in-place data queries via an analysis description language, performing on-the-fly data transformations into convenient analysis formats, and connecting the output to future analysis facilities.

- It is closely connected to both ATLAS and CMS, and features the ability to handle experiment-specific input formats like xAOD (ATLAS) and miniAOD (CMS).
- The developers are currently in close contact with members of each experiment to ensure the service will improve time-to-insight in both cases, and to make it possible to easily develop additional transformers for new input formats.
- While ServiceX is currently in a prototyping phase, it seeks to reach production version 1.0 by late May, at which point the service will benefit significantly from formal blueprint meetings with both collaborations.

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 specification for high-level `TRExFitter`-like analysis
- Demonstration of differentiable analysis pipeline (eg new `TRExFitter`) ending with `pyhf` limit back-propagating through selection implemented with awkward, `func_adl`, etc.
  - *connect to `pyhf/neos` demo. Need autodiff-able awkward*
  - *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 sensitivyt studies)

autodiff blueprint
(may need to send gradients back to analysis facility via ServiceX)



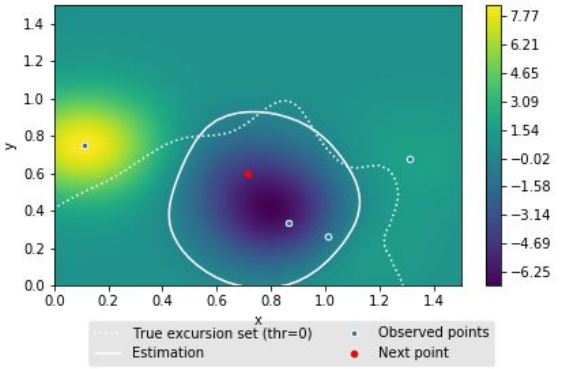
What “grand challenges” would be useful to organize involving your area during Year 3 of IRIS-HEP? How would these challenges depend on efforts from other areas of IRIS-HEP, the US LHC Ops programs or the experiments?

(Assuming not to be completed in Y3, but organized in Y3)

- Ability to process XX TB of data in YY minutes using columnar analysis tools
- End-to-end analysis optimization with automatic differentiation on large (~TB) simulated data sets with multiple signal and background components and systematic uncertainties. [touches on ServiceX]
- Ability to test a new theory by reinterpreting multiple analyses and performing a statistical combination of their results in ~1 day (+ the time it takes to generate new signal MC) [touches on LHC Ops b/c would use production system]
  - *extend by using* `excursion` *to streamline MC production for ATLAS or CMS reinterpretation campaign [touches on LHC Ops b/c would use production system]*
- Ability for new user to “fork” an analysis, make a modification, and obtain new results in an afternoon. [touches the experiments and training]

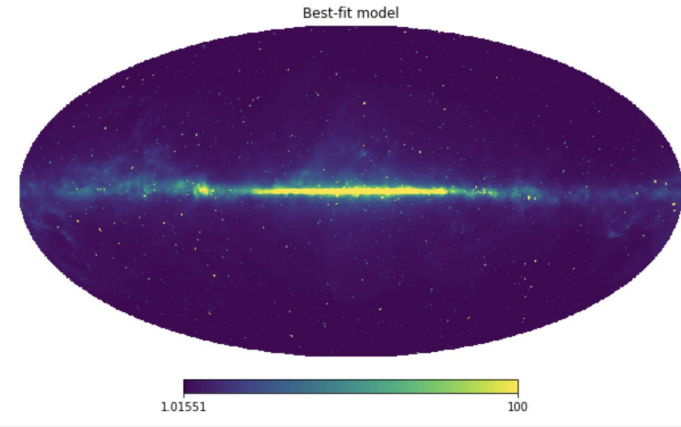
Are there new opportunities where effort from IRIS-HEP can make an impact? Is the alignment of the focus areas in IRIS-HEP appropriate?

- 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
)
```

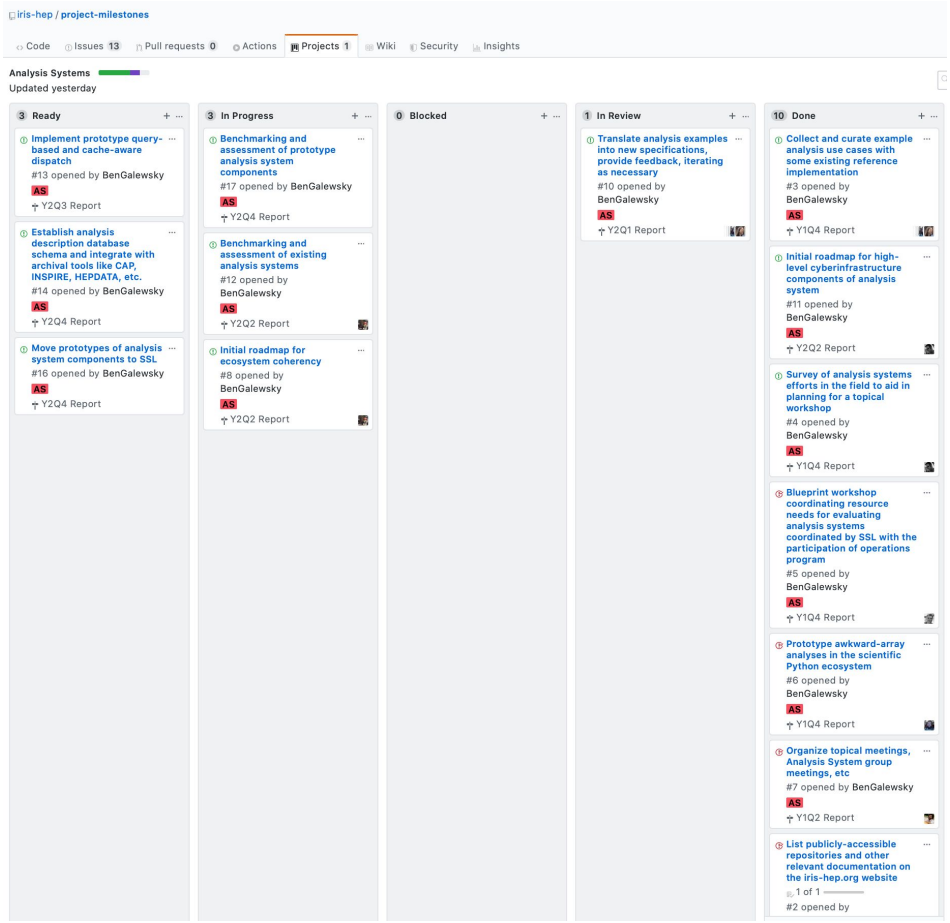
```
hp.mollview(m.expected_data(bestfit), max=100, title='Best-fit model')
```



How are projects currently managed in your area? What tools are being used? How is progress measured? How are risks recorded, identified and mitigated?

Progress tracked on GitHub: github.com/iris-hep/project-milestones/

	Analysis Systems		Done	Late	In progress	not started / Due soon				
Label	Description	Type (M/D)	Y1Q1	Y1Q2	Y1Q3	Y1Q4	Y2Q1	Y2Q2	Y2Q3	Y2Q4
G2.1	Organize topical meetings, Analysis System group meetings, etc.									
G2.2	List publicly-accessible repositories and other relevant doc- umentation on the iris-hep.org website									
G2.3	Collect and curate example analysis use cases with some existing reference implementation									
G2.4	Survey of analysis systems efforts in the field to aid in planning for topical workshop									
G2.5	Blueprint workshop coordinating resource needs for evaluating analysis systems coordinated by SSL with participation of operations program									
G2.6	Develop initial specifications for user-facing interface to analysis system components									
G2.7	Prototype awkward-array analyses in the scientific Python ecosystem									
G2.8	Initial roadmap for ecosystem coherency									
G2.9	Develop initial design for interface of analysis query system to the IDDS									
G2.10	Translate analysis examples into new specifications, provide feedback, iterating as necessary									
G2.11	Initial roadmap for high-level cyberinfrastructure components of analysis system									
G2.12	Benchmarking and assessment of existing analysis systems									
G2.13	Implement prototype query-based and cache-aware dispatch									
G2.14	Establish analysis description database schema and integrate with archival tools like CAP, INSPIRE, HEPDATA, etc.									
G2.15	GPU/accelerator-based implementation of statistical and other appropriate components									
G2.16	Move prototypes of analysis system components to SSL									
G2.17	Benchmarking and assessment of prototype analysis system components									

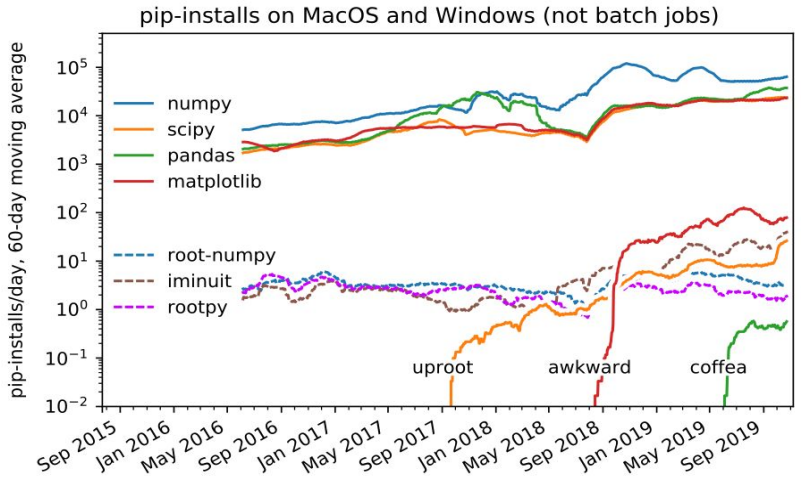


Are the metrics being used to measure success clearly defined? How well do metrics in your area measure progress, success or impact? Where can the metrics be improved or refined to better measure progress, success or impact?

Metrics listed on next page for reference.

- Metrics are clearly defined,
  - *I put some effort into defining reasonable targets for them*
  - *some targets are 'relative' (fraction of specs that are implemented) and some are 'absolute' (number of XXX)*
  - *Not clear if targets should be for current time or end of Y5*
- They are ok at measuring progress / effort, though not necessarily great at measuring "success".
  - *hard to assess meaningful measures of 'success' early in (R&D) phase*
  - *measures of user adoption & results using new tools will lag development >1 yr*
- Maybe better to enumerate AS components needed for a few vertical analysis use cases and track what fraction of those components are implemented (and connect with performance benchmarking of individual components)

Metrics



M.2.1: Number of specifications developed

- 12 thus far. Expect maybe 50 after 5 years

M.2.2: Number of implementations for corresponding specifications

- 5/12: ppx, func_adl, pyhf, aghast, histos, decay language

M.2.3: Throughput and latency metrics for analysis systems using SSL testbed

- Aiming for ~10x speedup for various analysis tasks. Seeing >100x in some cases

M.2.4: List of experiments using CAP and number of analyses stored in CAP

- 14 ATLAS analyses with workflows in CAP/REANA/RECAST-ready format

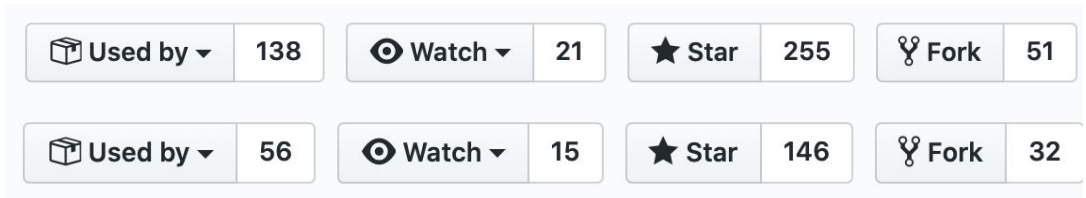
M.2.5: Number of results / papers making use of CAP/REANA

- 3 thus far, more on the way

M.2.6: GitHub stars, forks, watch, contributor statistics

- 12 GitHub repos
- healthy statistics for core projects

eg. uproot & awkward



Backup

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

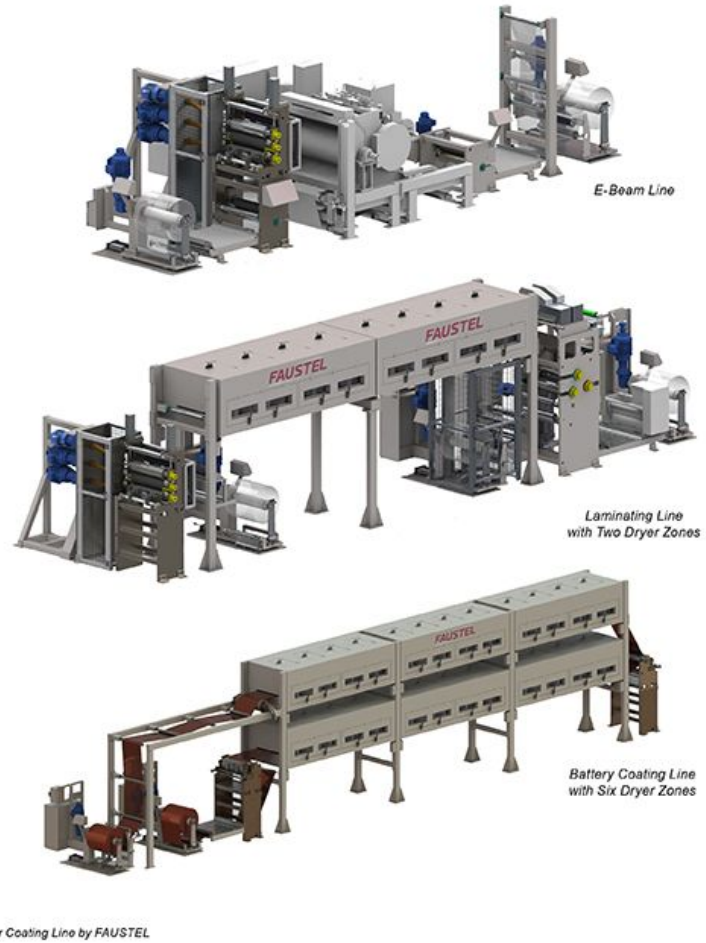
- primarily ROOT & C++
- lack of developer community
- overlapping solutions
- data redundancy

IRIS-HEP as an Institute

ad hoc analysis code

IRIS-HEP

Analysis Systems

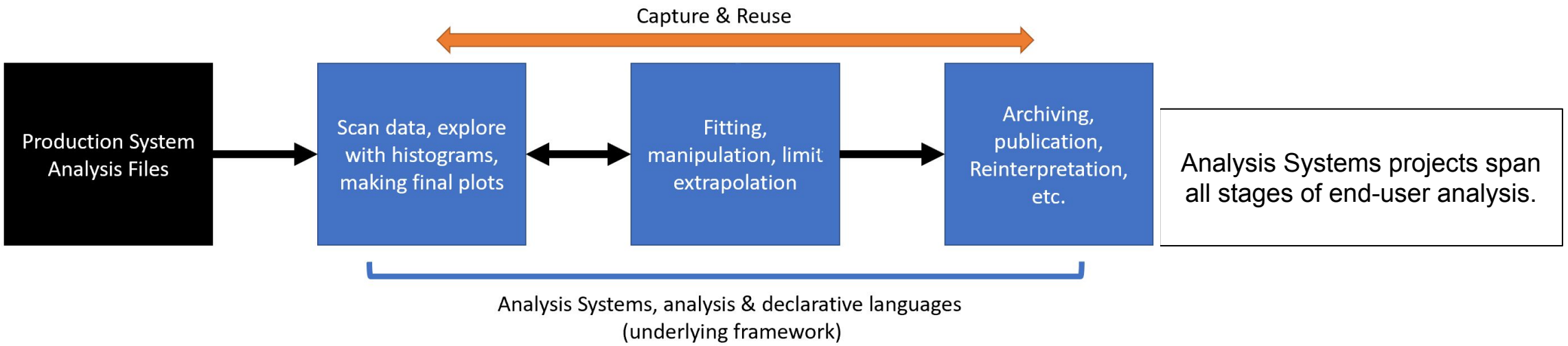


Analysis Systems strategies:

- 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

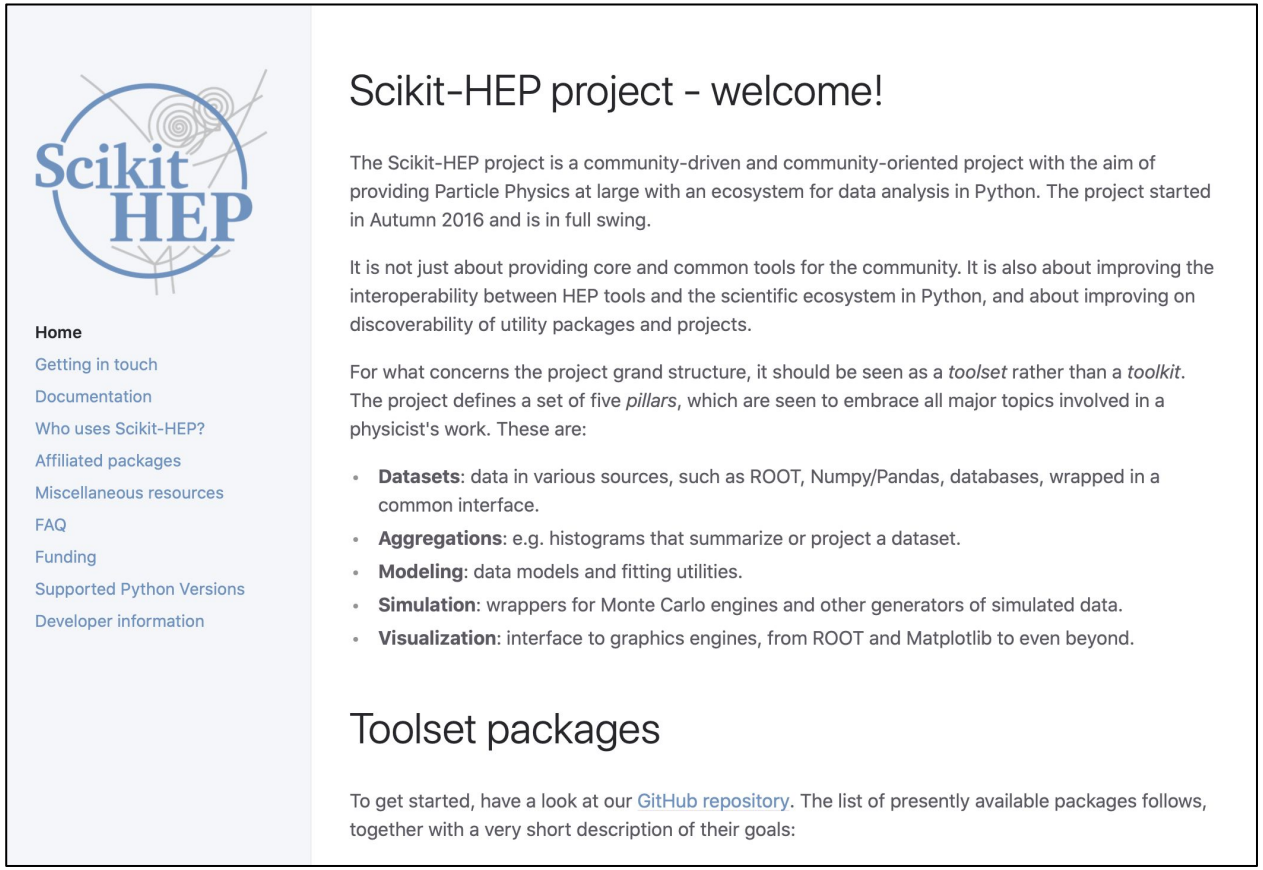
Analysis Systems

- 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.*



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:

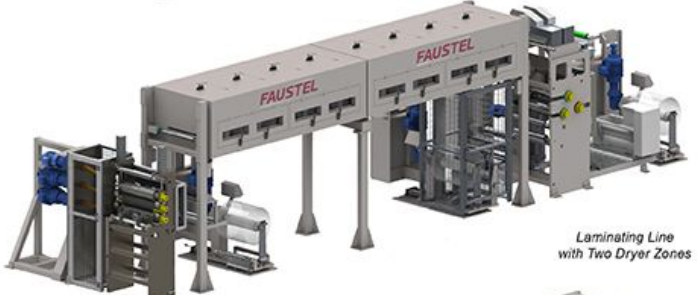
aghast

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

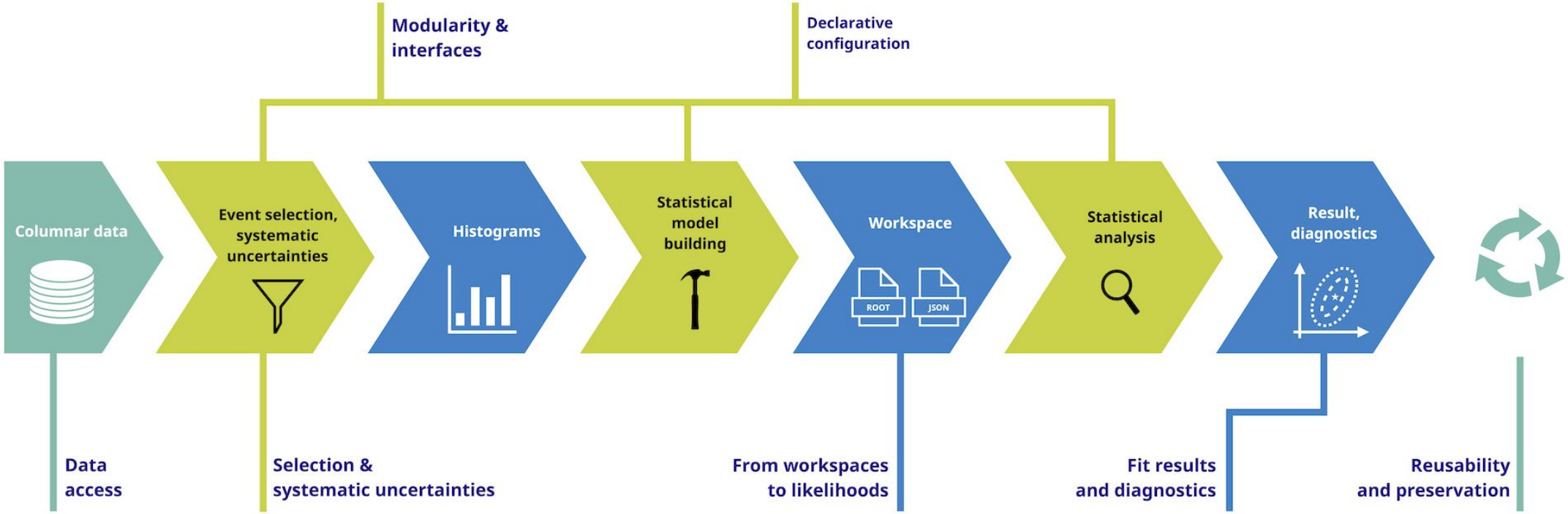
Boost Histogram

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

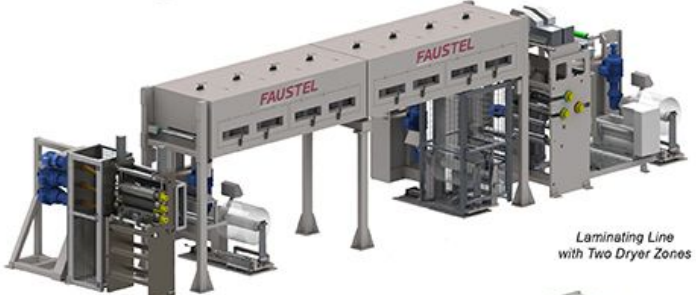
A coherent ecosystem



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



func_adl
formulate

Modularity & interfaces

Declarative configuration

Columnar data

Event selection, systematic uncertainties

Histograms

Statistical model building

Workspace

ROOT

JSON

Statistical analysis

Result, diagnostics

Data access

Selection & systematic uncertainties

From workspaces to likelihoods

Fit results and diagnostics

Reusability and preservation

ServiceX

uproot

Awkward Array

coffea

Boost Histogram

pyhf differentiable likelihoods

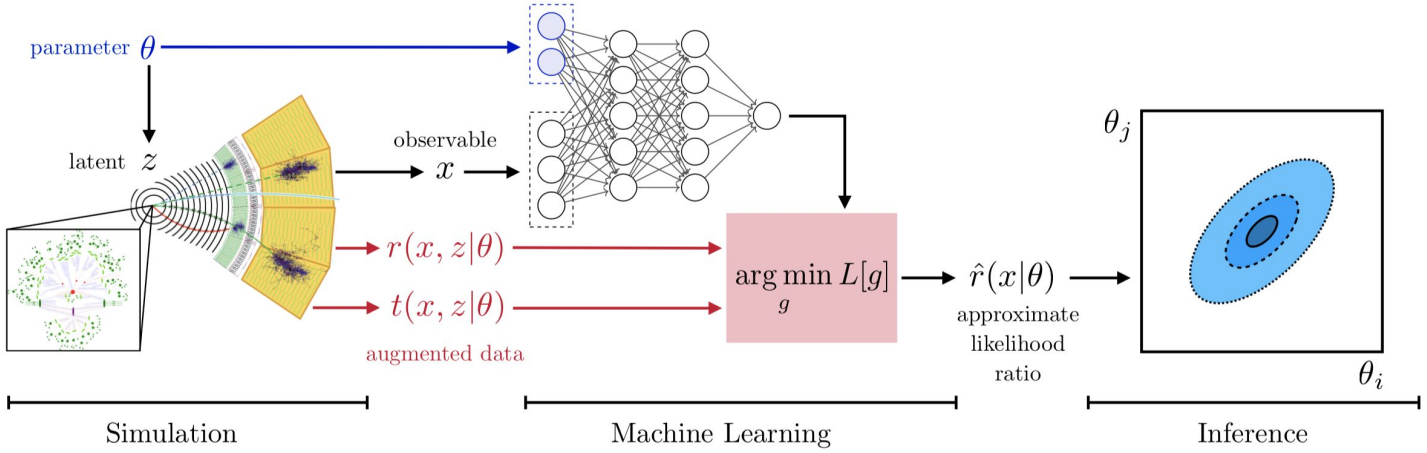
yadage

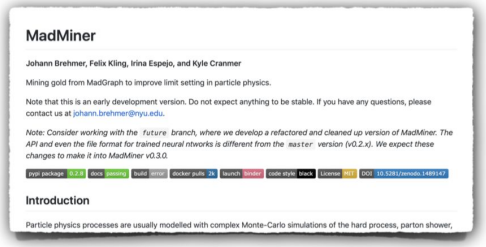
recast

The Future

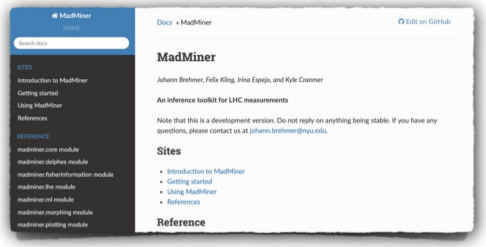
Tight integration of

- Simulation
- Machine Learning
- Statistical Inference

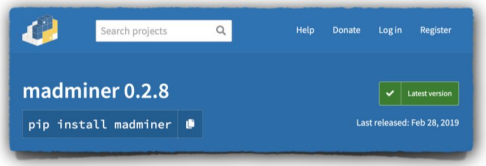




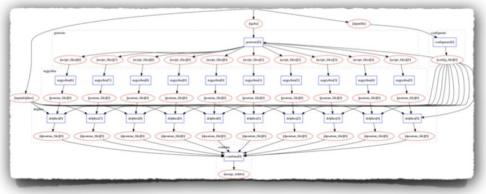
Repository and tutorials:
github.com/johannbrehmer/madminer



Documentation:
madminer.readthedocs.io

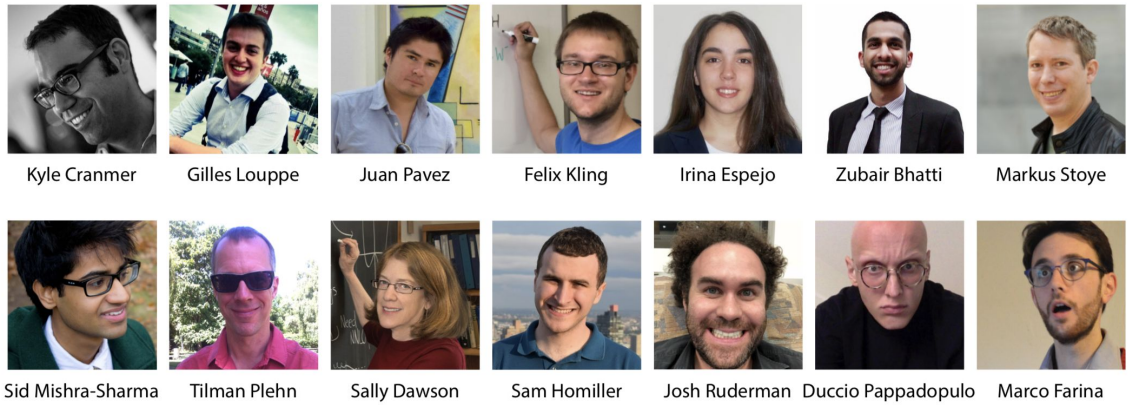


Installation:
`pip install madminer`



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

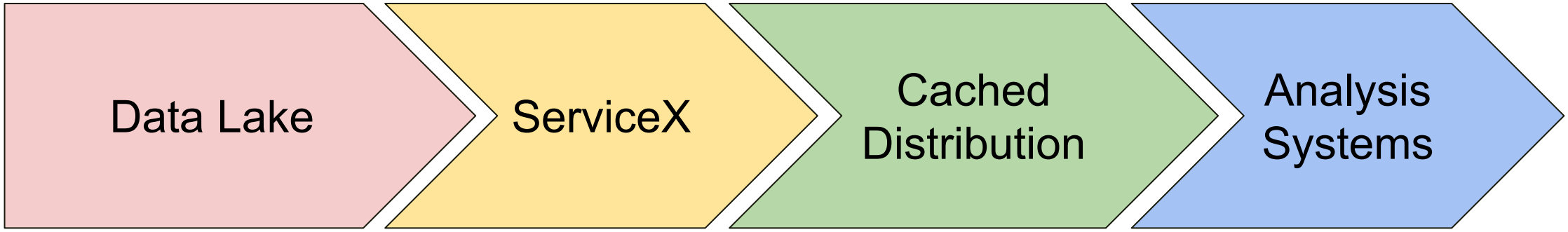
The SCAILFIN Project
scailfin.github.io

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

Connections to DOMA & SSL



ServiceX is part of DOMA's iDDS

- feeds data to downstream analysis systems
- uses components from analysis systems to:
  - *read ROOT-formatted data*
  - *transform analysis languages*
  - *export data formatted for downstream analysis*

ServiceX is being prototyped using IRIS-HEP's Scalable Systems Lab

- 10 TB xAOD data from ATLAS using IRIS functional analysis description language
- CMS example using uproot & awkward

B. Galewsky, R. Gardner, L. Gray, M. Neubauer,J. Pivarski, M. Proffitt, I. Vukotic, G. Watts, M. Weinberg

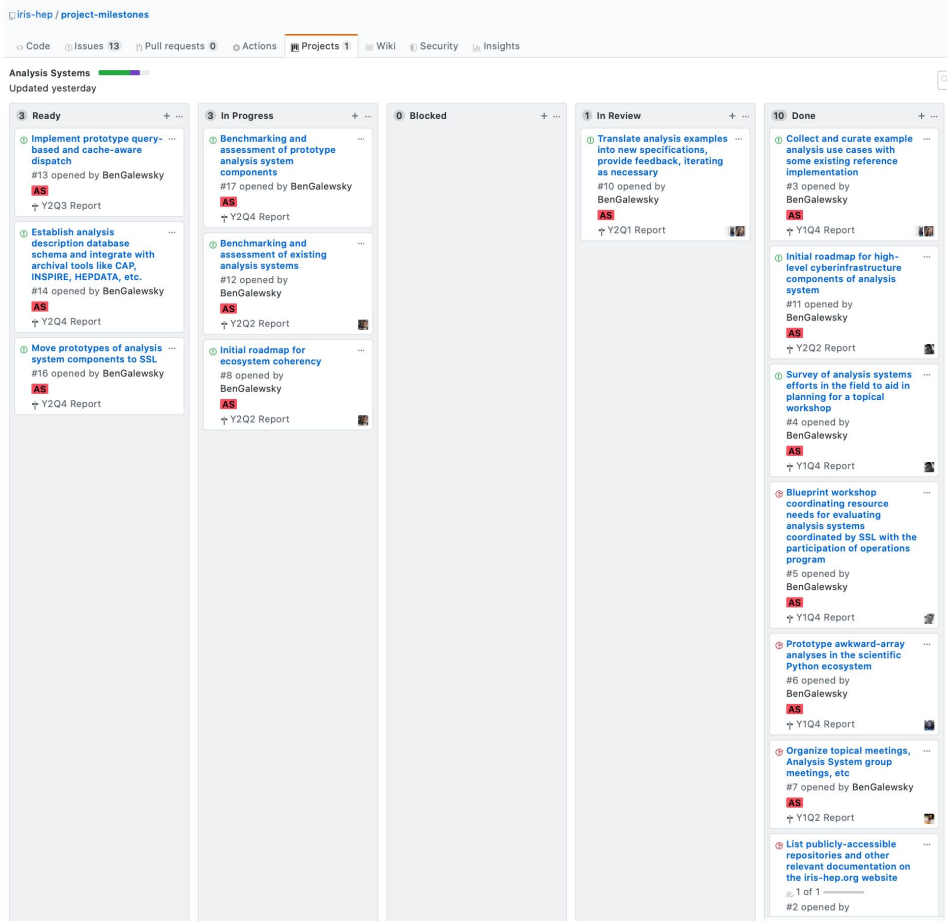
Milestones and Deliverables

	Analysis Systems		Done	Late	In progress	not started / Due soon				
Label	Description	Type (M/D)	Y1Q1	Y1Q2	Y1Q3	Y1Q4	Y2Q1	Y2Q2	Y2Q3	Y2Q4
G2.1	Organize topical meetings, Analysis System group meetings, etc.									
G2.2	List publicly-accessible repositories and other relevant doc- umentation on the iris-hep.org website									
G2.3	Collect and curate example analysis use cases with some existing reference implementation									
G2.4	Survey of analysis systems efforts in the field to aid in planning for topical workshop									
G2.5	Blueprint workshop coordinating resource needs for evaluating analysis systems coordinated by SSL with participation of operations program									
G2.6	Develop initial specifications for user-facing interface to analysis system components									
G2.7	Prototype awkward-array analyses in the scientific Python ecosystem									
G2.8	Initial roadmap for ecosystem coherency									
G2.9	Develop initial design for interface of analysis query system to the IDDS									
G2.10	Translate analysis examples into new specifications, provide feedback, iterating as necessary									
G2.11	Initial roadmap for high-level cyberinfrastructure components of analysis system									
G2.12	Benchmarking and assessment of existing analysis systems									
G2.13	Implement prototype query-based and cache-aware dispatch									
G2.14	Establish analysis description database schema and integrate with archival tools like CAP, INSPIRE, HEPDATA, etc.									
G2.15	GPU/accelerator-based implementation of statistical and other appropriate components									
G2.16	Move prototypes of analysis system components to SSL									
G2.17	Benchmarking and assessment of prototype analysis system components									

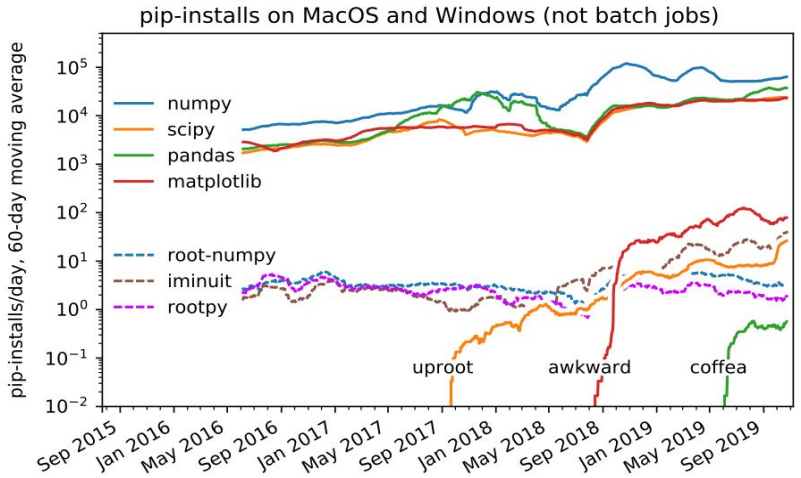
Milestones and Deliverables

Progress tracked on GitHub: github.com/iris-hep/project-milestones/

	Analysis Systems		Done	Late	In progress	not started / Due soon				
Label	Description	Type (M/D)	Y1Q1	Y1Q2	Y1Q3	Y1Q4	Y2Q1	Y2Q2	Y2Q3	Y2Q4
G2.1	Organize topical meetings, Analysis System group meetings, etc.									
G2.2	List publicly-accessible repositories and other relevant doc- umentation on the iris-hep.org website									
G2.3	Collect and curate example analysis use cases with some existing reference implementation									
G2.4	Survey of analysis systems efforts in the field to aid in planning for topical workshop									
G2.5	Blueprint workshop coordinating resource needs for evaluating analysis systems coordinated by SSL with participation of operations program									
G2.6	Develop initial specifications for user-facing interface to analysis system components									
G2.7	Prototype awkward-array analyses in the scientific Python ecosystem									
G2.8	Initial roadmap for ecosystem coherency									
G2.9	Develop initial design for interface of analysis query system to the IDDS									
G2.10	Translate analysis examples into new specifications, provide feedback, iterating as necessary									
G2.11	Initial roadmap for high-level cyberinfrastructure components of analysis system									
G2.12	Benchmarking and assessment of existing analysis systems									
G2.13	Implement prototype query-based and cache-aware dispatch									
G2.14	Establish analysis description database schema and integrate with archival tools like CAP, INSPIRE, HEPDATA, etc.									
G2.15	GPU/accelerator-based implementation of statistical and other appropriate components									
G2.16	Move prototypes of analysis system components to SSL									
G2.17	Benchmarking and assessment of prototype analysis system components									



Metrics



M.2.1: Number of specifications developed

- 12 thus far. Expect maybe 50 after 5 years

M.2.2: Number of implementations for corresponding specifications

- 5/12: ppx, func_adl, pyhf, aghast, histos, decay language

M.2.3: Throughput and latency metrics for analysis systems using SSL testbed

- Aiming for ~10x speedup for various analysis tasks. Seeing >100x in some cases

M.2.4: List of experiments using CAP and number of analyses stored in CAP

- 14 ATLAS analyses with workflows in CAP/REANA/RECAST-ready format

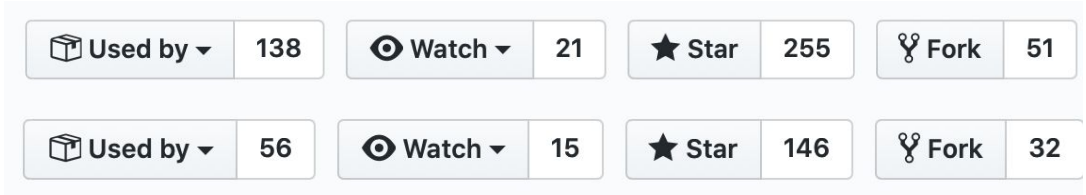
M.2.5: Number of results / papers making use of CAP/REANA

- 3 thus far, more on the way

M.2.6: GitHub stars, forks, watch, contributor statistics

- 12 GitHub repos
- healthy statistics for core projects

eg. uproot & awkward



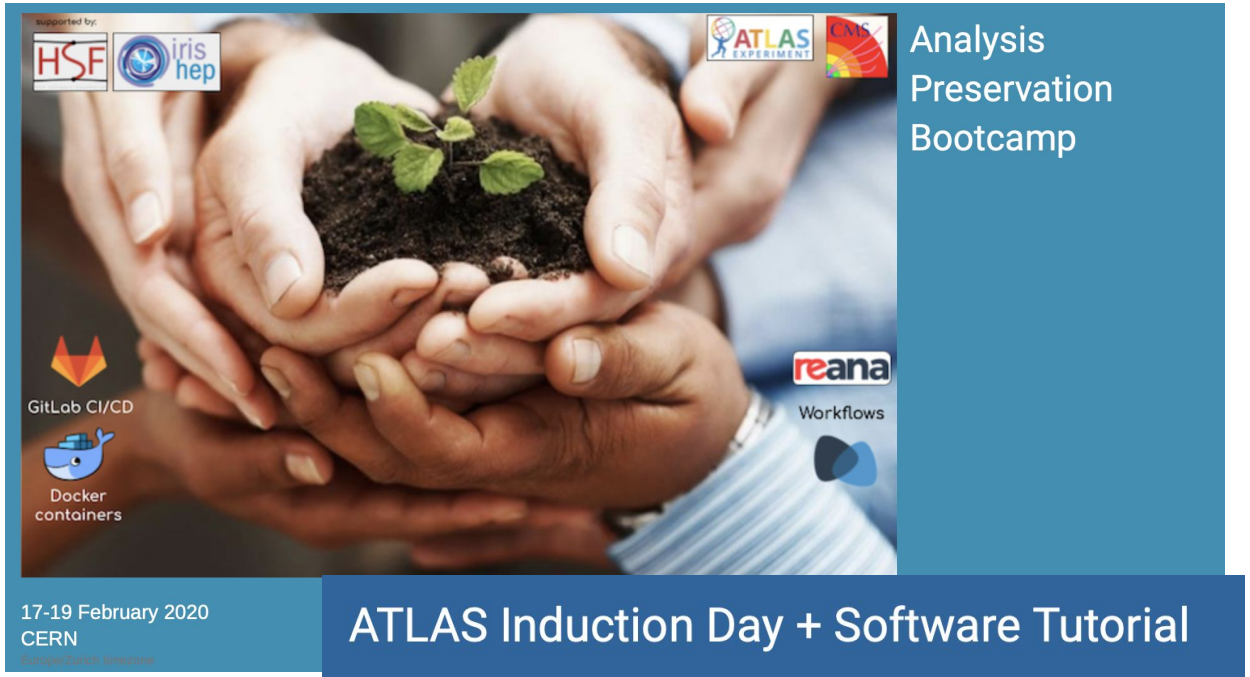
Community Building

"I just wanted to express my personal awe to you and your team working so hard on a bunch of wonderful projects. The talks delivered by Johann, Lukas and Gunes were excellent! In my personal opinion it was the best part of ACAT conference."

- Andrey Ustyuzhanin (LHCb & Yandex School of Data Analysis)

- Active participation in relevant venues:
  - *HEP Software Foundation, PyHEP, CHEP, ACAT, HOW, …*
  - *Internal experiment meetings, IRIS-HEP topical meetings*
  - *Partner projects: DIANA, SCAILFIN, CERN-IT, …*
  - *>100 presentations and 20 publications thus far*
- High-profile projects that provide clarity of vision and leadership
  - *Scikit-hep, uproot, awkward-array, histos, (Coffea)*
  - *MadMiner, AmpGen, functional analysis description language*
  - *pyhf, RECAST*
- Growing community of early adopters using tools now
  - *>1000 downloads / week for uproot*

Training



Analysis Preservation Bootcamp

17-19 February 2020
CERN
Europe/Zurich timezone

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

Presentations & Publications

108 presentations and 22 publications thus far

AS Presentations

- 19 Nov 2019 - Flows three ways, Kyle Cranmer (New York University), Deep Learning for Physics Seminar Series at Princeton Center for Theoretical Physics
- 07 Nov 2019 - Likelihood preservation and statistical reproduction of searches for new physics, Matthew Feickert (University of Illinois at Urbana-Champaign), CHEP 2019 Conference
- 07 Nov 2019 - HEP Data Query Challenges, Mason Proffitt (University of Washington), CHEP 2019
- 07 Nov 2019 - Recent developments in histogram libraries, Henry Schreiner (Princeton University), CHEP 2019
- 07 Nov 2019 - Aligning the MATHUSLA Detector Test Stand with Tensor Flow, Gordon Watts (University of Washington), CHEP 2019
- 07 Nov 2019 - Constraining effective field theories with machine learning, Alexander Held (New York University), 24th International Conference on Computing in High Energy & Nuclear Physics
- 05 Nov 2019 - pyhf: a pure Python implementation of HistFactory with tensors and autograd (poster), Matthew Feickert (University of Illinois at Urbana-Champaign), CHEP 2019 Conference
- 05 Nov 2019 - Using Analysis Declarative Languages for the HL-LHC, Gordon Watts (University of Washington), CHEP 2019
- 05 Nov 2019 - A Functional Declarative Analysis Language in Python (poster), Emma Torro (University of Washington), CHEP 2019
- 02 Nov 2019 - Analysis workflows, Gordon Watts (University of Washington), CHEP 2019
- 29 Oct 2019 - Harmonizing statistics tools - ideas, Alexander Held (New York University), ATLAS Statistics Committee Meeting
- 18 Oct 2019 - pyhf: pure-Python implementation of HistFactory, Matthew Feickert (University of Illinois at Urbana-Champaign), PyHEP 2019 Workshop
- 17 Oct 2019 - Simulation-based inference, interpretability, and experimental design, Kyle Cranmer (New York University), Workshop on Interpretable Learning in Physical Sciences Part of the Long Program Machine Learning for Physics and the Physics of Learning
- 17 Oct 2019 - Awkward 1.0, Jim Pivarski (Princeton University), PyHEP Workshop
- 17 Oct 2019 - Python Histogramming Packages, Henry Schreiner (Princeton University), PyHEP 2019
- 17 Oct 2019 - Python 3.8: What's new, Henry Schreiner (Princeton University), PyHEP 2019
- 17 Oct 2019 - Boost-Histogram: Hands-on, Henry Schreiner (Princeton University), PyHEP 2019
- 16 Oct 2019 - Lightning Talk: A Living HEP Analysis, Gordon Watts (University of Washington), PyHEP
- 05 Oct 2019 - Particle Physics in the context of Data Science, Kyle Cranmer (New York University), The 6th IEEE International Conference on Data Science and Advanced Analytics
- 30 Sep 2019 - What does the Revolution in Artificial Intelligence Mean for Physics?, Kyle Cranmer (New York University), Joint PITT-CMU Physics Department Colloquium
- 30 Sep 2019 - Run 3/Run 4 Perspectives - Event Delivery Impacts?, Gordon Watts (University of Washington), HSF & ATLAS Joint Event Delivery Workshop
- 27 Sep 2019 - Simulation-based inference, causality, and active learning, Kyle Cranmer (New York University), AI and the Scientific Method, ETH, Zurich
- 26 Sep 2019 - Declarative programming: A paradigm shift in data analysis in preparation for the HL-LHC, Gordon Watts (University of Washington), eScience2019
- 14 Sep 2019 - Jagged, ragged, awkward arrays, Jim Pivarski (Princeton University), Strange Loop 2019
- 13 Sep 2019 - Histogramming and more, Henry Schreiner (Princeton University), 2019 IRIS-HEP Institute Retreat
- 13 Sep 2019 - func-adl to C++/xAOD backend, Gordon Watts (University of Washington), IRIS-HEP Institute Retreat
- 12 Sep 2019 - pyhf Roadmap: 2019 into 2020, Matthew Feickert (University of Illinois at Urbana-Champaign), 2019 IRIS-HEP Institute Retreat

AS Publications

- The frontier of simulation-based inference, K. Cranmer, J. Brehmer and G. Louppe, arXiv 1911.01429 (Submitted to National Academy of Sciences) (04 Nov 2019).
- Extending RECAST for Truth-Level Reinterpretations, A. Schuy, L. Heinrich, K. Cranmer and S. Hsu, arXiv 1910.10289 (Submitted to DPF2019) (22 Oct 2019).
- Hamiltonian Graph Networks with ODE Integrators, A. Sanchez-Gonzalez, V. Bapst, K. Cranmer and P. Battaglia, arXiv 1909.12790 (27 Sep 2019).
- Mining for Dark Matter Substructure: Inferring subhalo population properties from strong lenses with machine learning, J. Brehmer, S. Mishra-Sharma, J. Hermans, G. Louppe and K. Cranmer, arXiv 1909.02005 (04 Sep 2019).
- Benchmarking simplified template cross sections in WH production, J. Brehmer, S. Dawson, S. Homiller, F. Kling and T. Plehn, JHEP 11 034 (2019) (19 Aug 2019).
- RECAST framework reinterpretation of an ATLAS Dark Matter Search constraining a model of a dark Higgs boson decaying to two b-quarks, ATL-PHYS-PUB-2019-032 (12 Aug 2019).
- Reproducing searches for new physics with the ATLAS experiment through publication of full statistical likelihoods, ATL-PHYS-PUB-2019-029 (05 Aug 2019).
- MadMiner: Machine learning-based inference for particle physics, J. Brehmer, F. Kling, I. Espejo and K. Cranmer, arXiv 1907.10621 (24 Jul 2019).
- Etalumis: Bringing Probabilistic Programming to Scientific Simulators at Scale, A. Baydin, L. Shao, W. Bhimji, L. Heinrich, L. Meadows et. al., arXiv 1907.03382 (07 Jul 2019).
- A hybrid deep learning approach to vertexing, R. Fang, H. Schreiner, M. Sokoloff, C. Weisser and M. Williams, arXiv 1906.08306 (Submitted to ACAT 2019) (19 Jun 2019).
- Effective LHC measurements with matrix elements and machine learning, J. Brehmer, K. Cranmer, I. Espejo, F. Kling, G. Louppe et. al., arXiv 1906.01578 (04 Jun 2019).
- FPGA-accelerated machine learning inference as a service for particle physics computing, J. Duarte, P. Harris, S. Hauck, B. Holzman, S. Hsu et. al., Comput.Softw.Big Sci. 3 13 (2019) (18 Apr 2019).
- Machine learning and the physical sciences, G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld et. al., arXiv 1903.10563 (25 Mar 2019).
- Open is not enough, X. Chen, S. Dallmeier-Tiessen, R. Dasler, S. Feger, P. Fokianos et. al., Nature Phys. 15 (2019) (15 Nov 2018).
- Analysis Preservation and Systematic Reinterpretation within the ATLAS experiment, K. Cranmer and L. Heinrich, J.Phys.Conf.Ser. 1085 042011 (2018) (18 Oct 2018).
- Efficient Probabilistic Inference in the Quest for Physics Beyond the Standard Model, A. Baydin, L. Heinrich, W. Bhimji, L. Shao, S. Naderiparizi et. al., arXiv 1807.07706 (20 Jul 2018).
- Machine Learning in High Energy Physics Community White Paper, K. Albertsson, P. Altoe, D. Anderson, J. Anderson, M. Andrews et. al., J.Phys.Conf.Ser. 1085 022008 (2018) (08 Jul 2018).
- Adversarial Variational Optimization of Non-Differentiable Simulators, G. Louppe, J. Hermans and K. Cranmer, arXiv 1707.07113 (22 Jul 2017).
- Yadage and Packtivity - analysis preservation using parametrized workflows, K. Cranmer and L. Heinrich, J.Phys.Conf.Ser. 898 102019 (2017) (06 Jun 2017).
- HEPData: a repository for high energy physics data, E. Maguire, L. Heinrich and G. Watt, J.Phys.Conf.Ser. 898 102006 (2017) (18 Apr 2017).
- QCD-Aware Recursive Neural Networks for Jet Physics, G. Louppe, K. Cho, C. Becot and K. Cranmer, JHEP 01 057 (2019) (02 Feb 2017).

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

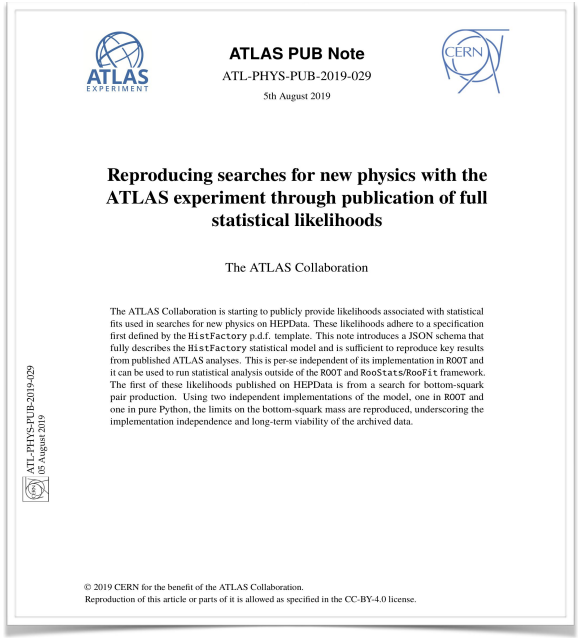
IRIS-HEP as a lighthouse:

- 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*
  - *preservation & reuse*

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



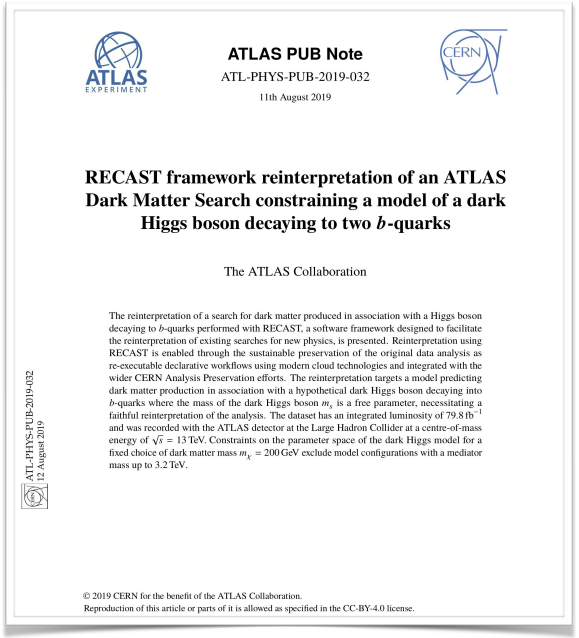
ATLAS PUB Note
ATL-PHYS-PUB-2019-029
5th August 2019

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



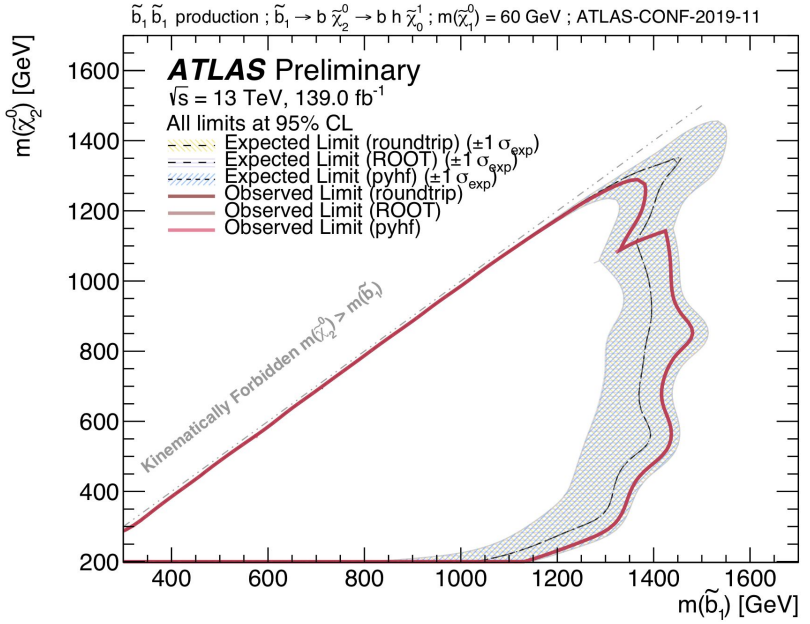
ATLAS PUB Note
ATL-PHYS-PUB-2019-032
11th August 2019

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



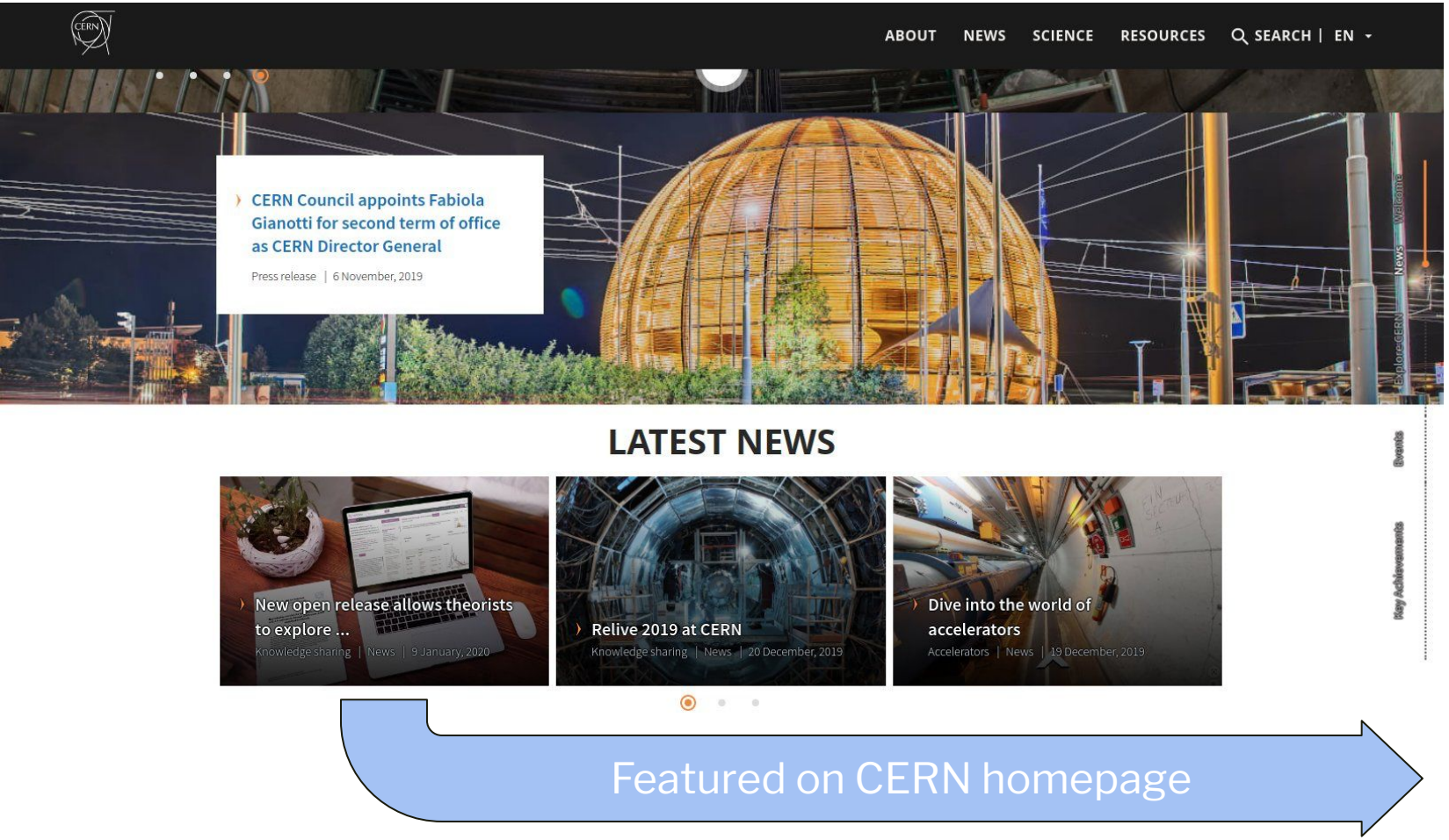
Highlight

HEPData
@HEPData

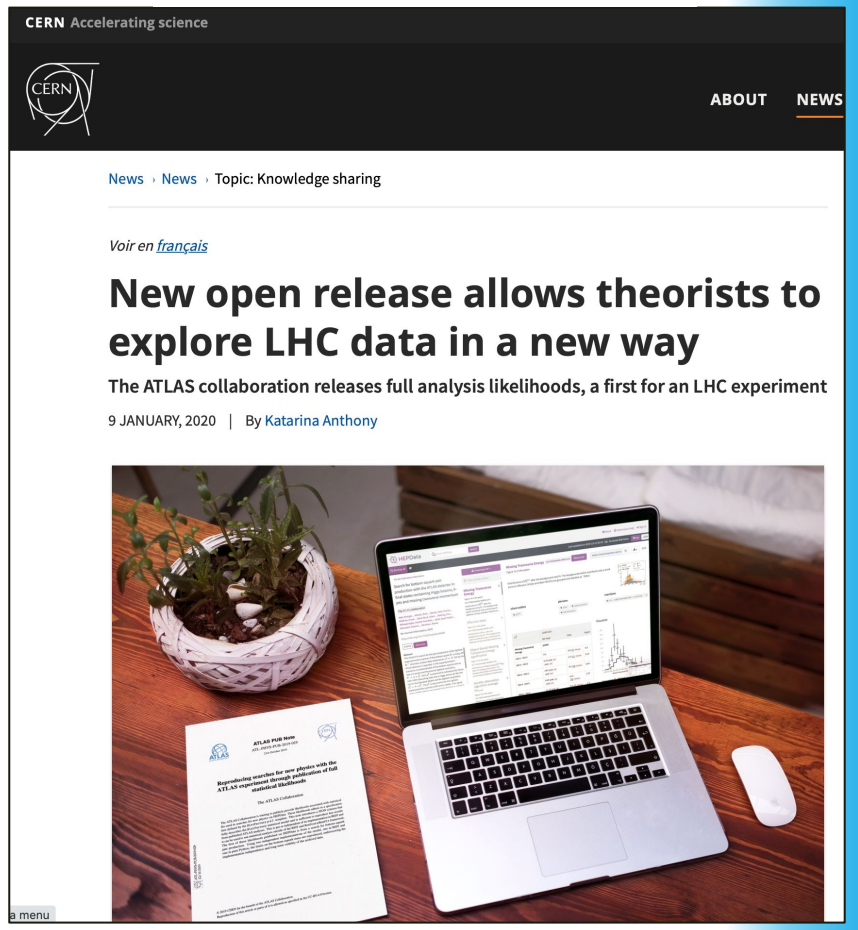
Thanks @KyleCranmer for your support and promotion of @HEPData over several years. Looking forward to future collaboration with @iris_hep on #pyhf likelihoods and more.

> **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

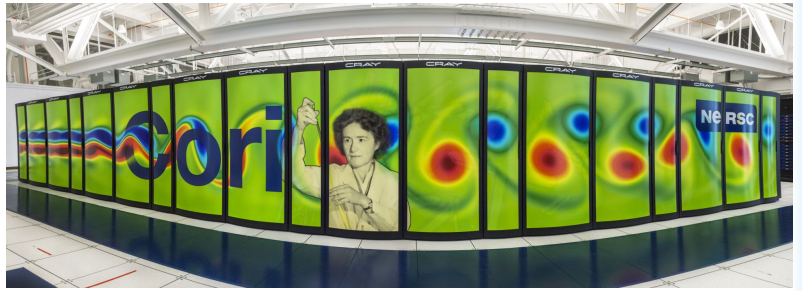


Featured on CERN homepage



Highlight

https://arxiv.org/abs/1907.03382

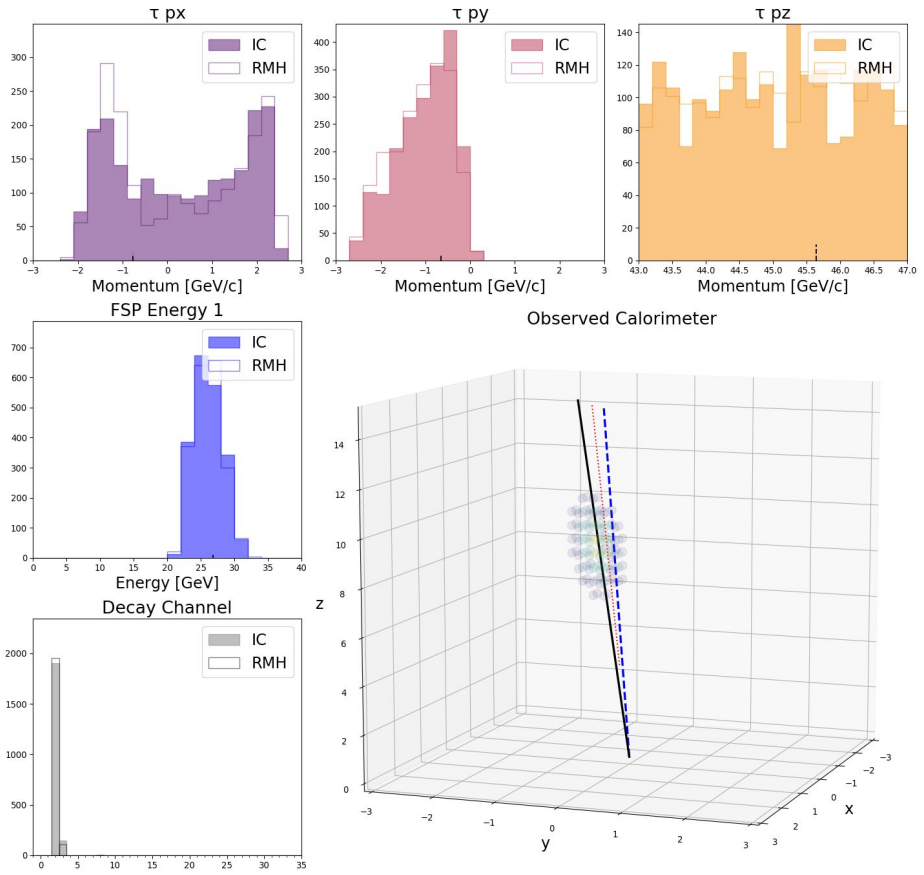


Finalist for best paper award at SC19 (Super Computing)

- Largest scale Bayesian inference ever using in a universal probabilistic programming language
  - ***Applied to complex LHC Physics use case: Sherpa code base of ~1M lines of code in C++***

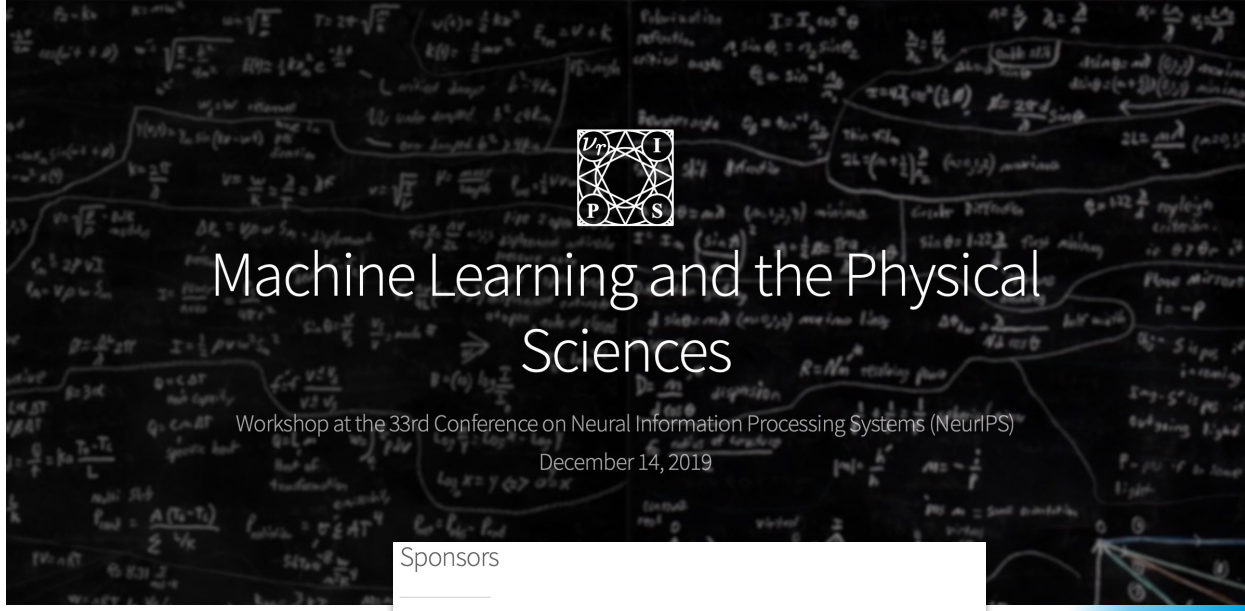
- 230x speedup for synchronous data parallel training of a 3DCNN-LSTM neural network
  - ***1,024 nodes (32,768 CPU cores)***
  - ***128k minibatch size, largest for this NN architecture***
  - ***One of the largest-scale use of PyTorch built-in MPI***

- Novel protocol (PPX) to execute & control existing, large-scale, scientific simulator code bases



Beyond HEP

- Co-organized by IRIS-HEP members
- 188 Paper submissions
- University + Industry + Labs
- Diversity in topics and participants



Sponsors

VECTOR INSTITUTE
FLATIRON INSTITUTE
iris hep
Institute for Research & Innovation in Software for High Energy Physics
DeepMind

Steering Committee


Anima Anandkumar
California Institute of Technology / NVIDIA


Kyle Cranmer
New York University

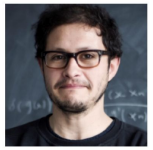

Roger Melko
University of Waterloo

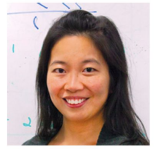

Prabhat
NERSC, Berkeley Lab


Frank Wood
University of British Columbia

Organizers


Atılım Güneş Baydin
University of Oxford


Juan Felipe Carrasquilla
Vector Institute / University of Waterloo


Shirley Ho
Flatiron Institute / Princeton University


Karthik Kashinath
NERSC, Berkeley Lab


Michela Paganini
Facebook AI Research


Savannah Thais
Princeton University / IRIS-HEP

Invited speakers


Alan Aspuru-Guzik
University of Toronto


Yasaman Bahri
Google Brain


Katie Bouman
California Institute of Technology

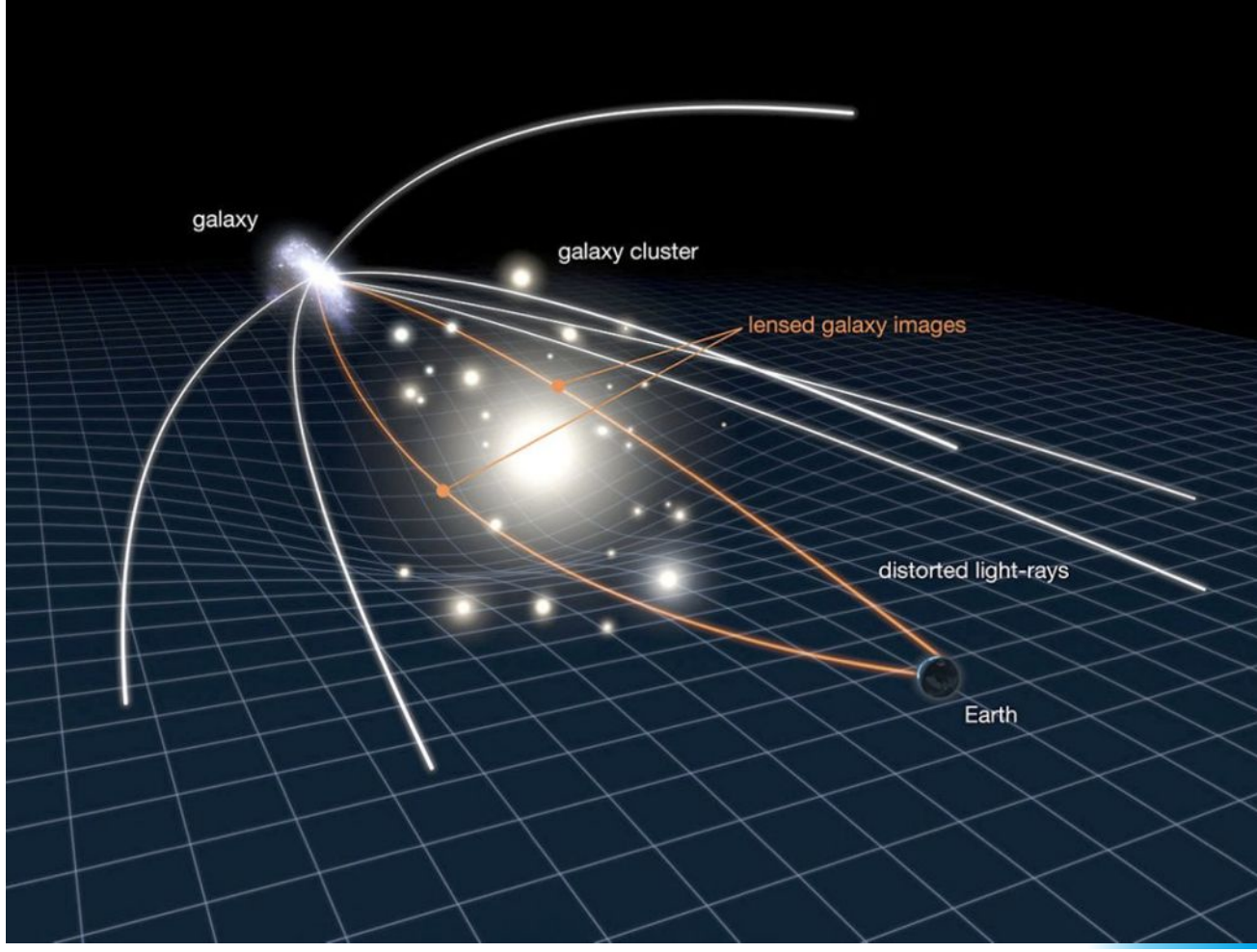

Bernhard Schölkopf
Max Planck Institute for Intelligent Systems


Maria Schuld
Xanadu

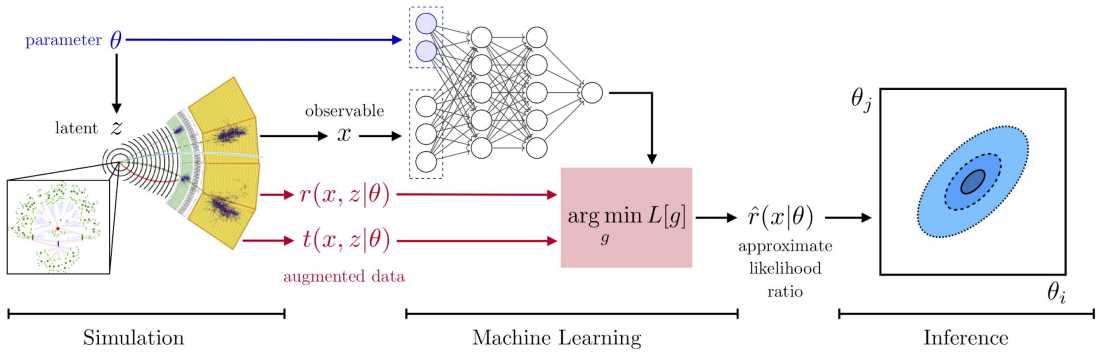

Lenka Zdeborova
Institut de Physique Théorique

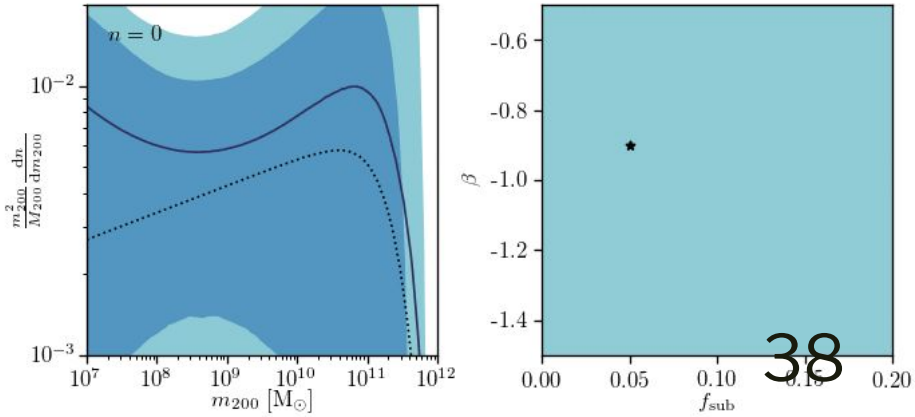
Beyond HEP

Machine learning & statistical techniques originally developed for LHC now being used to probe Dark Matter with gravitational strongly lensing



arXiv:1909.02005 published in The Astrophysical Journal.





Beyond HEP

in collaboration with Google DeepMind

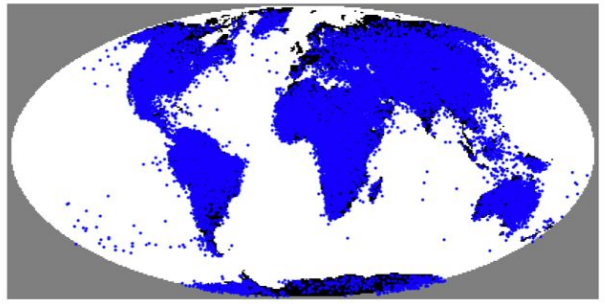
Collaboration with DeepMind on AI techniques inspired by physics

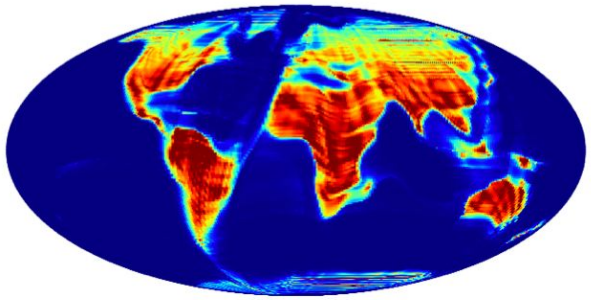
---

Normalizing Flows on Tori and Spheres

Danilo Jimenez Rezende [*1] George Papamakarios [*1] Sébastien Racanière [*1] Michael S. Albergo [2]
Gurtej Kanwar [3] Phiala E. Shanahan [3] Kyle Cranmer [2]

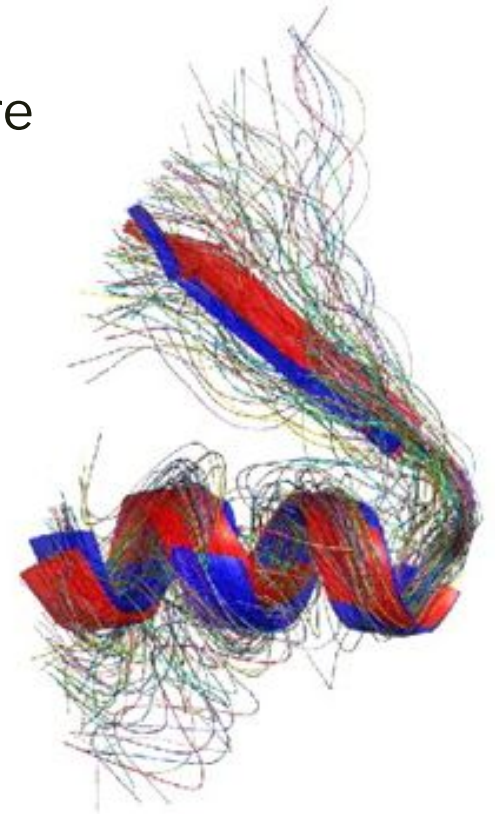
https://arxiv.org/abs/2002.02428





Relevant for:

- HEP
- nuclear physics (lattice QCD)
- cosmology
- geology
- protein structure
- robotics



Protein figure from Boomsma

Beyond HEP

See Sebastian Macaluso's poster highlighting exploratory machine learning projects.

- examples of use-inspired research
- connections to natural language processing (NLP) and genomics

Ginkgo: Toy Generative Model for Jets

Generative model to aid in ML research for jet physics.

NLP analogy: ground-truth parse trees with a known language model

K. Cranmer, S. Macaluso & D. Pappadopulo

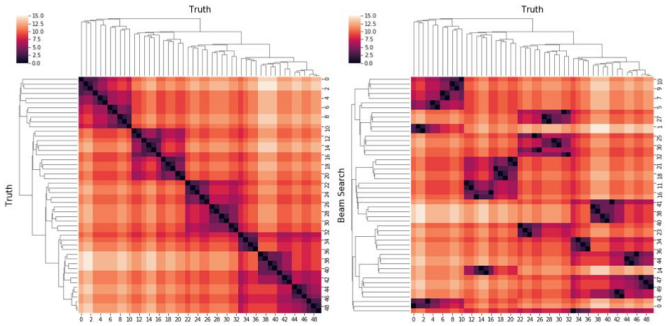
github.com/SebastianMacaluso/ToyJetsShower

Greedy Algorithm

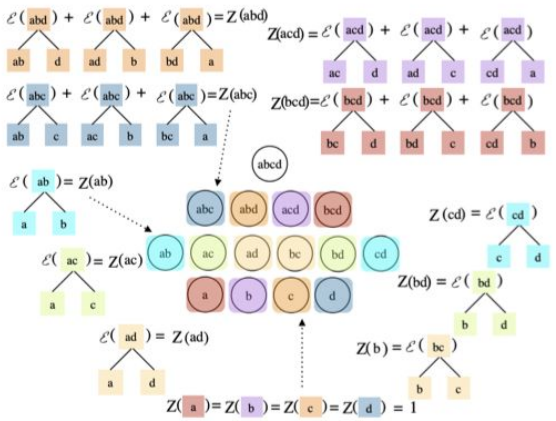
Locally maximizing the likelihood at each step.

Beam Search Algorithm

Maximize the likelihood of multiple steps before choosing the latent path.



Use-inspired Research: Hierarchical Cluster Trellis for Exact Inference



Standard clustering algorithm in HEP (anti-kT) is Greedy

$$p_{\text{Greedy}} < p_{\text{Beam Search}} < p_{\text{Trellis}}$$

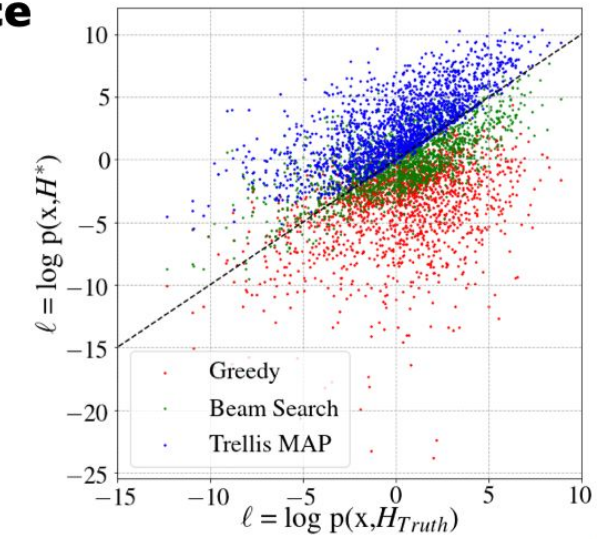
Use inspired-research

Seminar at UMass Amherst, Center for Data Science, College of Information & Computer Sciences led to collaboration for hierarchical clustering algorithms.

S. Macaluso, C. Greenberg, N. Monath, J. Lee, P. Flaherty, K. Cranmer, A. McGregor, A. McCallum

Applications in other domains, e.g. cancer genomics.

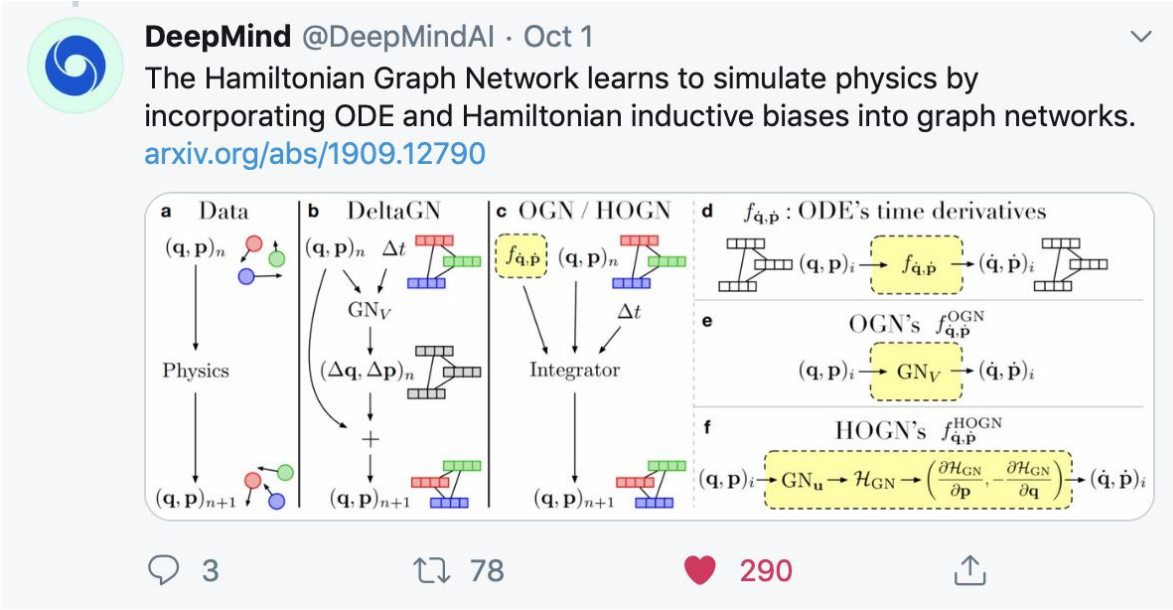
https://arxiv.org/abs/2002.11661



Beyond HEP

Collaboration with DeepMind on AI techniques inspired by physics

Models that incorporate physics generalize to unseen systems (zero-shot learning)



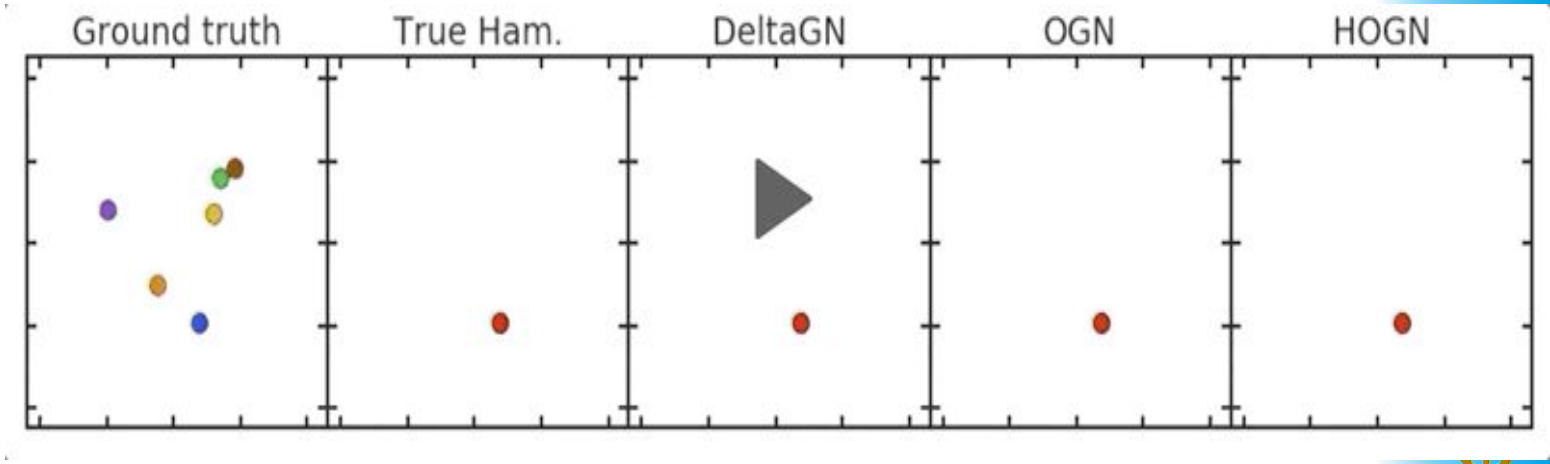
Hamiltonian Graph Networks with ODE Integrators

Alvaro Sanchez-Gonzalez
DeepMind
London, UK
alvarosg@google.com

Victor Bapst
DeepMind
London, UK
vbapst@google.com

Kyle Cranmer
NYU
New York, USA
kc90@nyu.edu

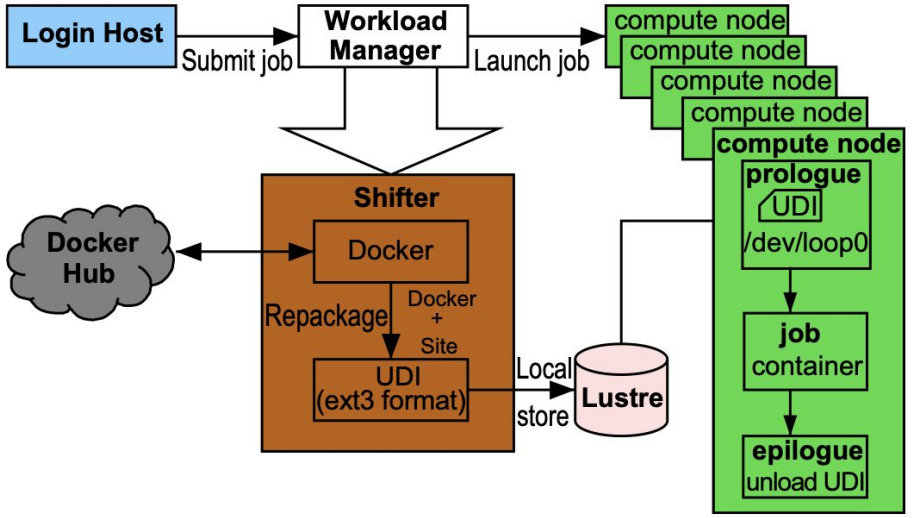
Peter Battaglia
DeepMind
London, UK
peterbattaglia@google.com



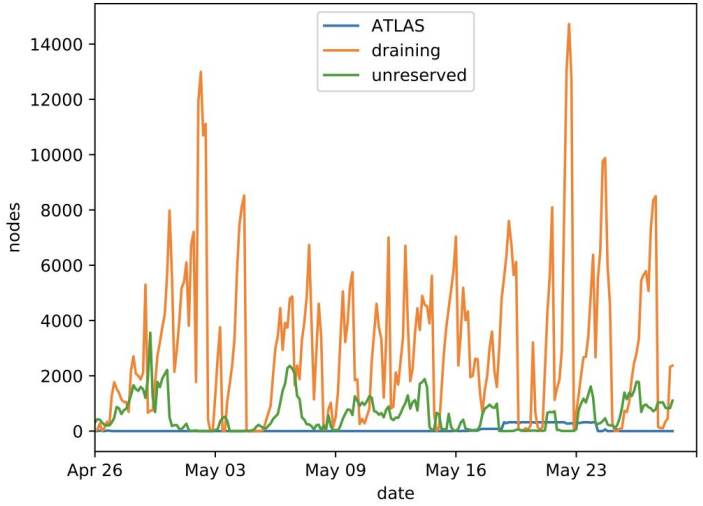
Beyond HEP

Collaborating with CS & astrophysics on computing models and tools to use HTC and HPC together, published as:

E. A. Huerta, R. Haas, S. Jha, M. Neubauer, D. S. Katz, "Supporting High-Performance and High-Throughput Computing for Experimental Science," *Computing and Software for Big Science* 3:5, 2019. doi: 10.1007/s41781-019-0022-7



Components involved in starting a Shifter job on Blue Waters (HPC). Jobs are submitted to workload manager on Blue Waters' login nodes, which launches jobs on compute nodes. When job requests use containers, workload manager first uses Shifter runtime environment to pull an up-to-date copy of the container image from Docker Hub. This image is repackaged as a user-defined image, then pre-mounted (prologue) by the jobs on the compute nodes and unloaded post-job (epilogue).



Left: period of time during which 35 million ATLAS events were processed using 300 Blue Waters nodes. Utilization during this period averaged 81%, typical for Blue Waters. Right: backlog of queued jobs for the same period in requested nodes, with colors indicating user accounts. During this period, the queued workload never dropped below 80,000 nodes i.e., four times the number of nodes in Blue Waters. The red and blue curves below the horizontal axis are nodes available for work scavenging during this period.