Preserving LHC Analyses with Rivet: A Foundation for Reproducible and Reusable Particle Physics Research

Christian Gütschow

FOSDEM 2025, Brussels

01 February 2025

UCL ARC
ADVANCED RESEARCH COMPUTING

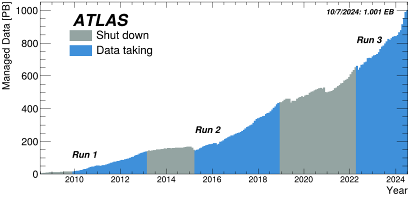
MCnet

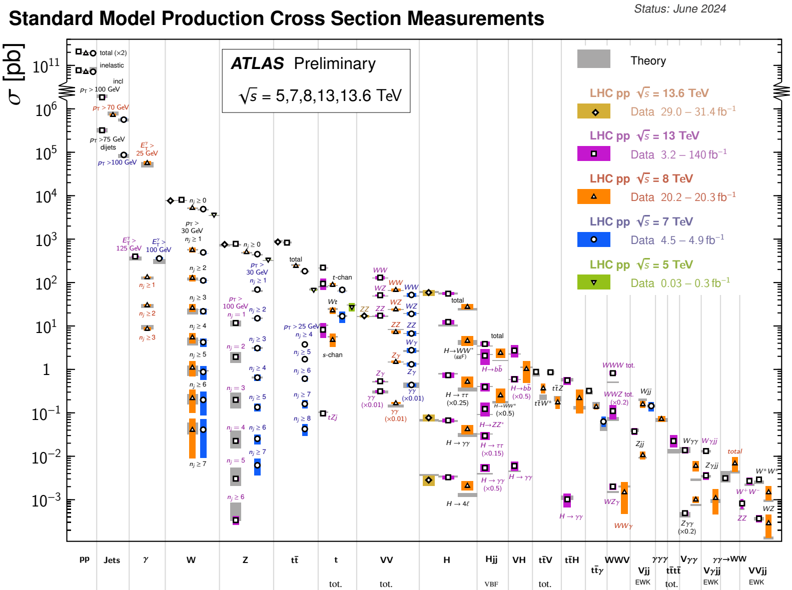
SWIFT HEP

Understanding the Building Blocks of the Universe

- ➜ The Large Hadron Collider (LHC) generates petabytes of data annually from **billions of collision events**.
- ➜ Each event records the properties of numerous particles, creating **complex, high-dimensional datasets**.
- ➜ To interpret these events, we rely heavily on Monte Carlo (MC) simulations based on theoretical models.
- ➜ Physics processes vary in frequency and complexity: From extremely rare events to more common interactions, requiring accurate simulations across a vast range of scenarios.

[CERN]





Standard Model Production Cross Section Measurements

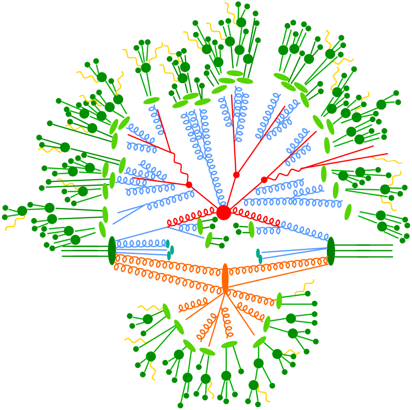
Status: June 2024

Connecting Theory Tools to Experiment

- Independent Development
  - Theory tools such as event generators and parton distribution functions (PDFs) are developed primarily by the theoretical physics community, separate from experimental collaborations.

Connecting Theory Tools to Experiment

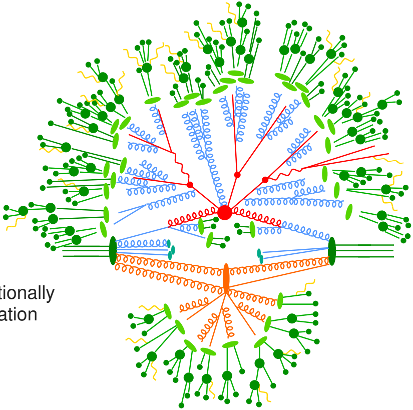
- Independent Development
  - Theory tools such as event generators and parton distribution functions (PDFs) are developed primarily by the theoretical physics community, separate from experimental collaborations.
- Continuous Improvements
  - These tools are updated with advancements in theoretical models and precision calculations, often after experimental results are published.
  - Retrospective reinterpretation of data becomes crucial for leveraging these improvements.



Connecting Theory Tools to Experiment

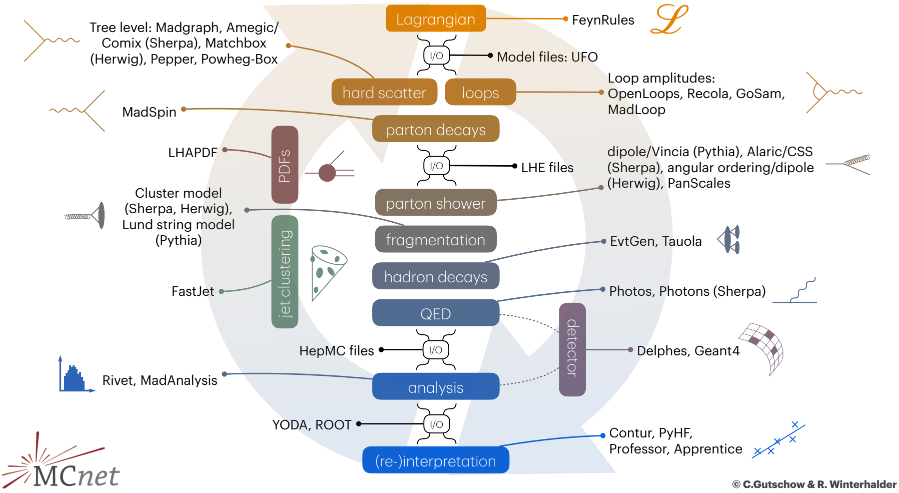
- Independent Development
  - Theory tools such as event generators and parton distribution functions (PDFs) are developed primarily by the theoretical physics community, separate from experimental collaborations.

- Continuous Improvements
  - These tools are updated with advancements in theoretical models and precision calculations, often after experimental results are published.
  - Retrospective reinterpretation of data becomes crucial for leveraging these improvements.



- Resource-Intensive Simulations
  - Large-scale Monte Carlo simulations are computationally and energetically expensive, requiring robust validation to ensure accuracy and efficiency.
  - Rigorous cross-validation with experimental data ensures reliable theoretical predictions.

Monte Carlo Event Generation and Analysis Workflow

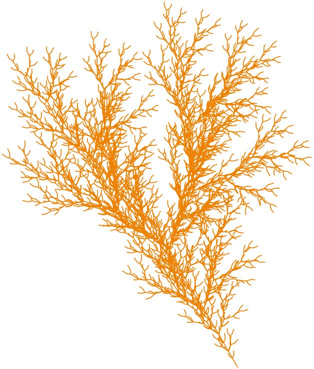

Lagrangian
FeynRules
I/O
Model files: UFO
Tree level: Madgraph, Amegic/ Comix (Sherpa), Matchbox (Herwig), Pepper, Powheg-Box
hard scatter
loops
Loop amplitudes: OpenLoops, Recola, GoSam, MadLoop
MadSpin
parton decays
LHAPDF
PDFs
I/O
LHE files
dipole/Vincia (Pythia), Alaric/CSS (Sherpa), angular ordering/dipole (Herwig), PanScales
parton shower
Cluster model (Sherpa, Herwig), Lund string model (Pythia)
fragmentation
jet clustering
EvtGen, Tauola
hadron decays
FastJet
Photos, Photons (Sherpa)
QED
detector
HepMC files
I/O
Delphes, Geant4
Rivet, MadAnalysis
analysis
YODA, ROOT
I/O
(re-)interpretation
Contur, PyHF, Professor, Apprentice
© C.Gutschow & R. Winterhalder

Current Challenges in HEP Analysis

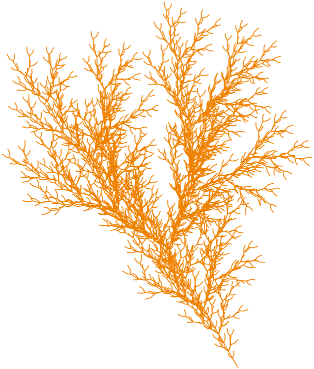
- Complex Analysis Workflows
  - Involve event generation, detector simulation, and statistical analysis pipelines.

Current Challenges in HEP Analysis

- Complex Analysis Workflows
  - Involve event generation, detector simulation, and statistical analysis pipelines.
- Complex Event Structures
  - MC event graphs contain hundreds of particles and vertices with some differences among generators, making consistent and efficient data analysis difficult.

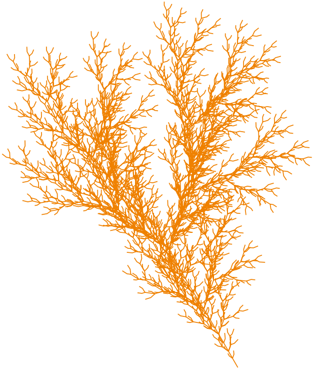


Current Challenges in HEP Analysis



- Complex Analysis Workflows
  - Involve event generation, detector simulation, and statistical analysis pipelines.
- Complex Event Structures
  - MC event graphs contain hundreds of particles and vertices with some differences among generators, making consistent and efficient data analysis difficult.
- Limited Software Portability
  - Analysis tools often tightly integrated with experiment-specific frameworks, making sharing difficult.

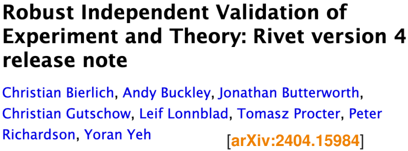
Current Challenges in HEP Analysis



- Complex Analysis Workflows
  - Involve event generation, detector simulation, and statistical analysis pipelines.
- Complex Event Structures
  - MC event graphs contain hundreds of particles and vertices with some differences among generators, making consistent and efficient data analysis difficult.
- Limited Software Portability
  - Analysis tools often tightly integrated with experiment-specific frameworks, making sharing difficult.
- Risk of Knowledge Loss
  - Without proper preservation, critical analysis insights may be lost over time.

Introducing Rivet

- Robust and Independent Validation of Experiment and Theory! [rivet.hepforge.org]
- Widely adopted by both experimental and theoretical particle physics communities as the common "language" for MC analysis.
- First released in 2007, fourth major version available as of 2024. [gitlab.com/hepcedar/rivet]
- Written in C++, with Python-based command-line tools for flexible workflows.
- Ensures consistent and robust comparison of theoretical predictions and experimental measurements.

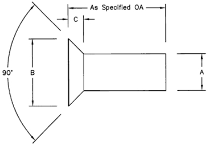


Robust Independent Validation of Experiment and Theory: Rivet version 4 release note

Christian Bierlich, Andy Buckley, Jonathan Butterworth, Christian Gutschow, Leif Lonnblad, Tomasz Procter, Peter Richardson, Yoran Yeh

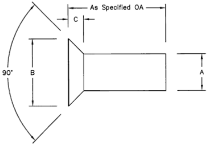
[arXiv:2404.15984]

Designing the Rivet



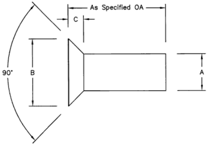
- Ease of Use
  - Focus on enabling physicists to concentrate on physics insights rather than technical details.
  - Minimal boilerplate code for cleaner, simpler analysis writing.
  - Familiar event loop structure and intuitive histogramming tools.
  - Streamlined integration for syncing results with external data sources like [**HepData**].

Designing the Rivet



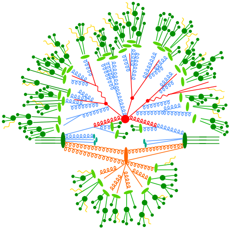
- Ease of Use
  - Focus on enabling physicists to concentrate on physics insights rather than technical details.
  - Minimal boilerplate code for cleaner, simpler analysis writing.
  - Familiar event loop structure and intuitive histogramming tools.
  - Streamlined integration for syncing results with external data sources like [**HepData**].
- Flexible and Embeddable
  - Core functionality in modern C++ with Python bindings for enhanced scripting flexibility.
  - Works with any event generator using the standard [**HepMC**] format for seamless integration.
  - Analyses are modular and dynamically loaded as “plugins”, promoting code reuse and clarity.

Designing the Rivet

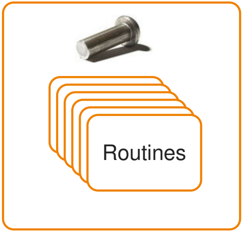


- Ease of Use
  - Focus on enabling physicists to concentrate on physics insights rather than technical details.
  - Minimal boilerplate code for cleaner, simpler analysis writing.
  - Familiar event loop structure and intuitive histogramming tools.
  - Streamlined integration for syncing results with external data sources like [HepData].
- Flexible and Embeddable
  - Core functionality in modern C++ with Python bindings for enhanced scripting flexibility.
  - Works with any event generator using the standard [HepMC] format for seamless integration.
  - Analyses are modular and dynamically loaded as “plugins”, promoting code reuse and clarity.
- Efficient and Scalable
  - Built-in caching system to avoid redundant computations during event processing.

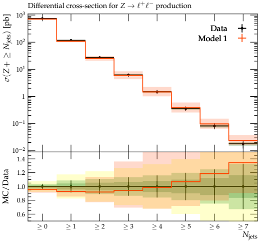
Rivet Workflow Overview



➜



➜

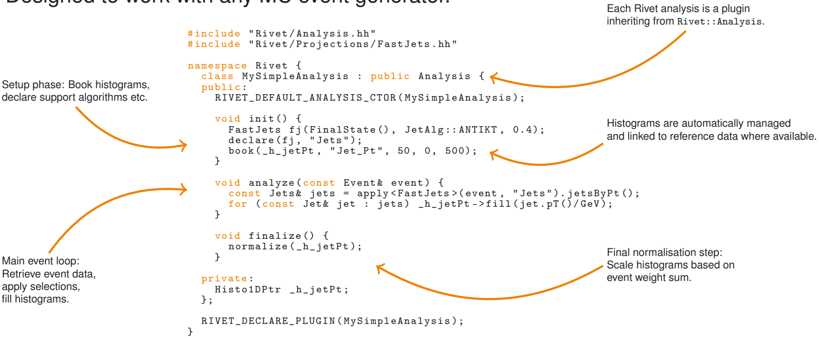


- ➜ Monte Carlo event generators produce simulated collision events
- ➜ Rivet reads these events, applies analysis routines, and fills histograms.
  - ➜ Analysis routines automatically loaded and executed by Rivet’s event loop framework.
- ➜ Histograms are stored in YODA format, facilitating further analysis, visualisation and reinterpretation studies.

[**Check out YODA@FOSDEM25 tomorrow!**]

What is a Rivet routine?

- A modular analysis plugin to process simulated collision events.
- Encodes physics logic for event selection, kinematic calculations, and histogramming.
- Pre-built analysis functions for standard observables.
- Designed to work with any MC event generator.



```
#include "Rivet/Analysis.hh"
#include "Rivet/Projections/FastJets.hh"

namespace Rivet {
  class MySimpleAnalysis : public Analysis {
  public:
    RIVET_DEFAULT_ANALYSIS_CTOR(MySimpleAnalysis);

    void init() {
      FastJets fj(FinalState(), JetAlg::ANTIKT, 0.4);
      declare(fj, "Jets");
      book(_h_jetPt, "Jet_Pt", 50, 0, 500);
    }

    void analyze(const Event& event) {
      const Jets& jets = apply<FastJets>(event, "Jets").jetsByPt();
      for (const Jet& jet : jets) _h_jetPt->fill(jet.pT()/GeV);
    }

    void finalize() {
      normalize(_h_jetPt);
    }

  private:
    Histo1DPtr _h_jetPt;
  };

  RIVET_DECLARE_PLUGIN(MySimpleAnalysis);
}
```

Each Rivet analysis is a plugin inheriting from `Rivet::Analysis`.

Setup phase: Book histograms, declare support algorithms etc.

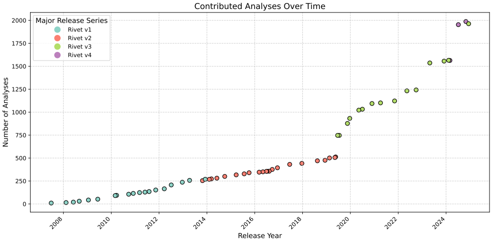
Histograms are automatically managed and linked to reference data where available.

Main event loop: Retrieve event data, apply selections, fill histograms.

Final normalisation step: Scale histograms based on event weight sum.

LHC Analysis Preservation in Practice

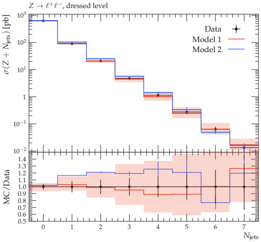
- Validated Repository
  - Central library of hundreds of published LHC analyses (and more).
- Transparency in Scientific Workflow
  - Open-source routines allow full inspection and replication of results.
  - Clear documentation of selection criteria and observable definitions.


Contributed Analyses Over Time

- Cross-Checking Experimental Results
  - Independent validation of collider results as well as benchmarking of MC event generators.
  - Direct reproduction of key measurements across experiments, boosting citations.
- Theory Reinterpretation Studies
  - Quickly test new theory models against archived analysis data.
  - Enables discovery potential for new physics.

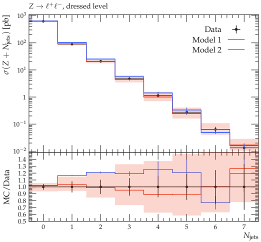
Community and Collaboration

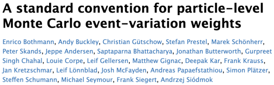
- Bridging Theory & Experiment
  - Rivet is the **de facto standard** for comparing MC event generators with LHC data.
  - Provides a **common framework** that ensures **consistent validation** of theoretical models.
  - Enables a **shared language** between theorists and experimentalists.



Community and Collaboration

- Bridging Theory & Experiment
  - Rivet is the **de facto standard** for comparing MC event generators with LHC data.
  - Provides a **common framework** that ensures **consistent validation** of theoretical models.
  - Enables a **shared language** between theorists and experimentalists.
- Fostering Common Standards
  - Having a unified toolset facilitates discussions on **consistent methodologies**.
  - Adoption by multiple communities (LHC experiments, MC developers, theorists) helps **align best practices**.





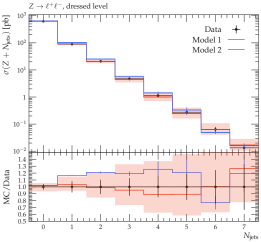
A standard convention for particle-level Monte Carlo event-variation weights

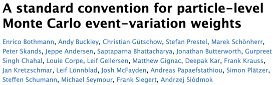
Enrico Bothmann, Andy Buckley, Christian Gütschow, Stefan Prestel, Marek Schönherr, Peter Skands, Jeppe Andersen, Saptaparna Bhattacharya, Jonathan Butterworth, Gurpreet Singh Chahal, Louie Corpe, Leif Gellersen, Matthew Gignac, Deepak Kar, Frank Krauss, Jan Kretzschmar, Leif Lönnblad, Josh McFayden, Andreas Papaefstathiou, Simon Plätzer, Steffen Schumann, Michael Seymour, Frank Siegert, Andrzej Siódmok

[arXiv:2203.08230]

Community and Collaboration

- Bridging Theory & Experiment
  - Rivet is the **de facto standard** for comparing MC event generators with LHC data.
  - Provides a **common framework** that ensures **consistent validation** of theoretical models.
  - Enables a **shared language** between theorists and experimentalists.
- Fostering Common Standards
  - Having a unified toolset facilitates discussions on **consistent methodologies**.
  - Adoption by multiple communities (LHC experiments, MC developers, theorists) helps **align best practices**.
- Open & Evolving
  - Actively maintained by the **HEP software community** with open-source contributions.
  - Regular **workshops, training sessions**, and discussions to drive future improvements.
  - Integrated into **analysis preservation efforts** for long-term impact.





A standard convention for particle-level Monte Carlo event-variation weights

Enrico Bothmann, Andy Buckley, Christian Gütschow, Stefan Prestel, Marek Schönherr, Peter Skands, Jeppe Andersen, Saptaparna Bhattacharya, Jonathan Butterworth, Gurpreet Singh Chahal, Louie Corpe, Leif Gellersen, Matthew Gignac, Deepak Kar, Frank Krauss, Jan Kretzschmar, Leif Lönnblad, Josh McFayden, Andreas Papaefstathiou, Simon Plätzer, Steffen Schumann, Michael Seymour, Frank Siegert, Andrzej Siódmok

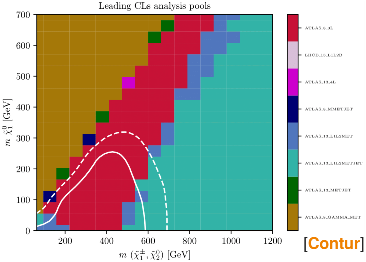
[**arXiv:2203.08230**]

Reinterpreting LHC Data for New Physics

- Testing new physics hypotheses
  - LHC experiments produce vast datasets of precision measurements.
  - Rivet allows injecting new-physics signals on top of Standard Model predictions.
  - If a Standard Model extension leads to statistically significant deviations, it can be ruled out.

Reinterpreting LHC Data for New Physics

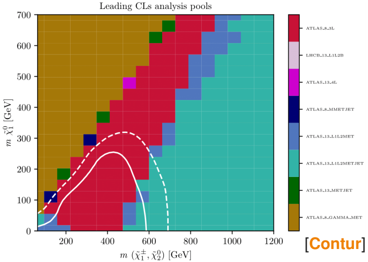
- Testing new physics hypotheses
  - LHC experiments produce vast datasets of precision measurements.
  - Rivet allows injecting new-physics signals on top of Standard Model predictions.
  - If a Standard Model extension leads to statistically significant deviations, it can be ruled out.
- Fast and Broad Coverage
  - Rivet provides a synoptic view – many independent measurements across different energies and processes.
  - New Rivet-based analyses can be used in a likelihood fit within hours.



[Contur]

Reinterpreting LHC Data for New Physics

- Testing new physics hypotheses
  - LHC experiments produce vast datasets of precision measurements.
  - Rivet allows injecting new-physics signals on top of Standard Model predictions.
  - If a Standard Model extension leads to statistically significant deviations, it can be ruled out.
- Fast and Broad Coverage
  - Rivet provides a synoptic view – many independent measurements across different energies and processes.
  - New Rivet-based analyses can be used in a likelihood fit within hours.
- Extending the New Physics Search Frontier
  - Increasingly precise calculations and new measurements expand the space of possible new physics models.
  - Reusable, high-quality data ensures long-term discovery potential.



Challenges and Lessons Learned

⚖ Balancing Generality & Usability

➜ Modular design with plugin-based analysis routines.

Challenges and Lessons Learned

⚖ Balancing Generality & Usability

→ Modular design with plugin-based analysis routines.

⚙ Interfacing with Experiment & Theory

→ Rivet uses HepMC as a standard interface and maintains version compatibility.

Challenges and Lessons Learned

⚖ Balancing Generality & Usability

- → Modular design with plugin-based analysis routines.

⚙ Interfacing with Experiment & Theory

- → Rivet uses HepMC as a standard interface and maintains version compatibility.

⟳ Analysis Preservation & Long-Term Reusability

- → Rigorous archival of validated analyses and automated tests to ensure compatibility.

Challenges and Lessons Learned

Balancing Generality & Usability

➜ Modular design with plugin-based analysis routines.

Interfacing with Experiment & Theory

➜ Rivet uses HepMC as a standard interface and maintains version compatibility.

Analysis Preservation & Long-Term Reusability

➜ Rigorous archival of validated analyses and automated tests to ensure compatibility.

Standardisation matters

➜ Ensuring consistent validation reduces discrepancies in MC predictions.

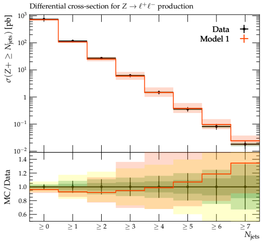
Challenges and Lessons Learned

- Balancing Generality & Usability
  - → Modular design with plugin-based analysis routines.
- Interfacing with Experiment & Theory
  - → Rivet uses HepMC as a standard interface and maintains version compatibility.
- Analysis Preservation & Long-Term Reusability
  - → Rigorous archival of validated analyses and automated tests to ensure compatibility.
- Standardisation matters
  - → Ensuring consistent validation reduces discrepancies in MC predictions.
- Community-driven design is key
  - → User feedback has shaped features like multi-pass execution for heavy-ion analyses.

Challenges and Lessons Learned

- Balancing Generality & Usability
  - → Modular design with plugin-based analysis routines.
- Interfacing with Experiment & Theory
  - → Rivet uses HepMC as a standard interface and maintains version compatibility.
- Analysis Preservation & Long-Term Reusability
  - → Rigorous archival of validated analyses and automated tests to ensure compatibility.
- Standardisation matters
  - → Ensuring consistent validation reduces discrepancies in MC predictions.
- Community-driven design is key
  - → User feedback has shaped features like multi-pass execution for heavy-ion analyses.
- Sustainability needs effort
  - → Workshops, onboarding & continuous development keep Rivet relevant.

Summary: Rivet for LHC Analysis & Beyond



- Standard tool for MC validation & analysis preservation
- Bridges experiment & theory with a common framework
- Designed for usability, flexibility, and long-term reusability
- Supports evolving physics needs
- Sustained by an active community – contributions welcome!
- Looking ahead:
  - Expanding automation & usability
  - Adapting MC pipelines for modern computing to handle next-generation LHC data
  - Strengthening analysis preservation efforts

[rivet.hepforge.org]

[gitlab.com/hepcedar/rivet]

[docker:hepstore/rivet]

[arXiv:2404.15984]

Backup