Preserving Collider Physics Analyses with Rivet

Christian Gütschow

MC4EIC 2025, Jefferson Lab

09 July 2025

Ensuring Long-Term Impact of EIC Results

- The EIC will explore **complex final states** and observables in a **new kinematic regime**.
- Robust validation of MC tools will be critical to understand detector effects and theoretical uncertainties in this precision frontier.
- **Future theory developments must be testable against today's measurements.**
- LHC experience shows that without structured analysis preservation, valuable insights are quickly lost:

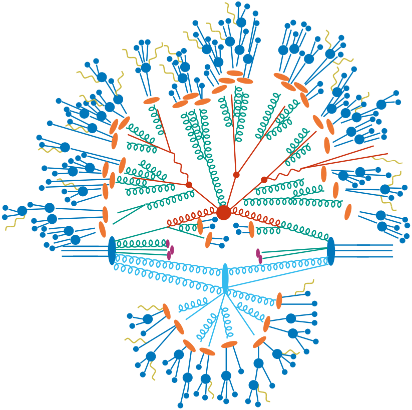
Key	ALICE	ATLAS	CMS	LHCb	Forward	HERA	e^+e^- (≥ 12 GeV)	e^+e^- (≤ 12 GeV)	Tevatron	RHIC	SPS
Rivet wanted (total):	384	498	588	201	19	473	647	73	1116	528	61
Rivet REALLY wanted:	87	66	106	21	0	14	1	0	9	2	5
Rivet provided:	44/428 = 10%	219/717 = 31%	140/728 = 19%	76/277 = 27%	10/29 = 34%	41/514 = 8%	243/890 = 27%	1013/1086 = 93%	61/1177 = 5%	11/539 = 2%	15/76 = 20%

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.

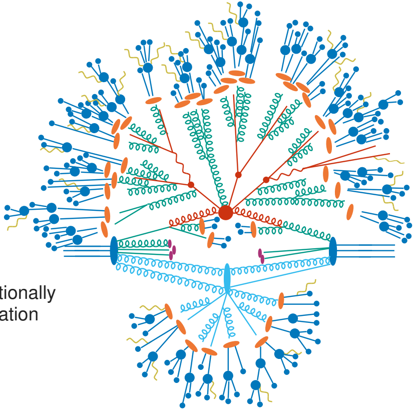
Connecting Theory Tools to Experiment

- 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.

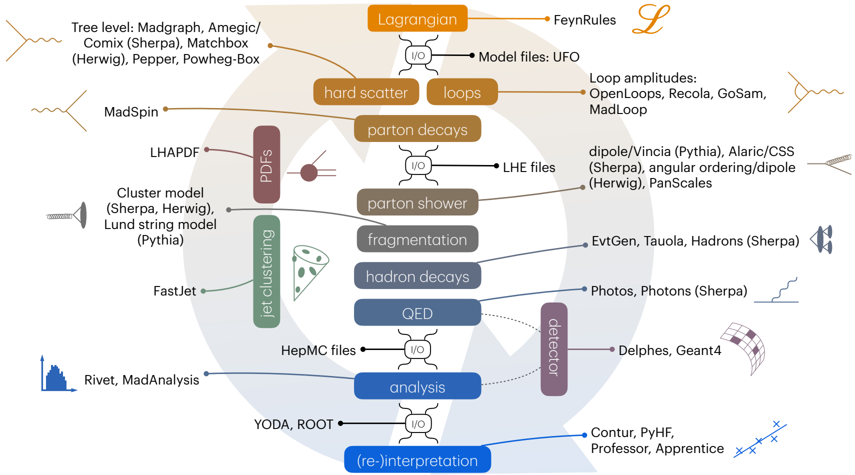


Connecting Theory Tools to Experiment

- 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

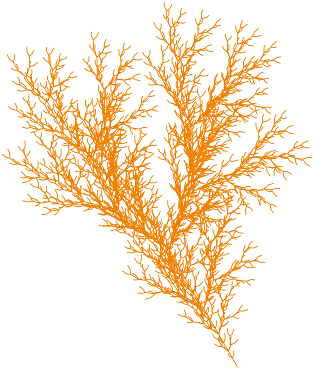


Current Challenges in HEP Analysis

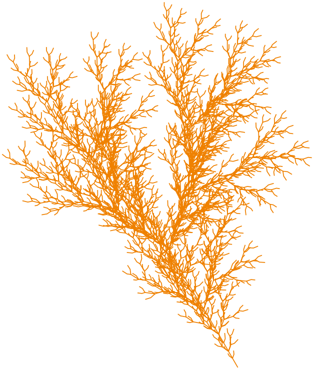
- Complex Analysis Workflows
  - Involve event generation, detector simulation, and statistical analysis pipelines.

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.

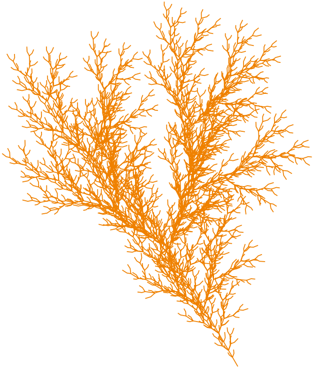


Current Challenges in HEP Analysis



- 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.

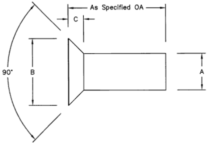
Introducing Rivet

- 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

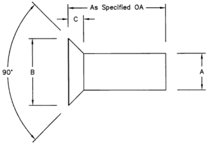
Christian Bierlich, Andy Buckley, Jonathan Butterworth, Christian Gutschow, Leif Lonnblad, Tomasz Procter, Peter Richardson, Yoran Yeh [arXiv:2404.15984]

Designing the Rivet



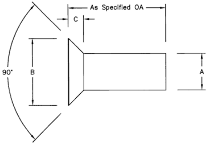
- Ease of Use
  - Focus on enabling physicists to concentrate on physics insights rather than technical details.
  - Minimal boilerplate code for cleaner, simpler analysis writing.
  - Familiar event loop structure and intuitive histogramming tools.
  - Streamlined integration for syncing results with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

Designing the Rivet



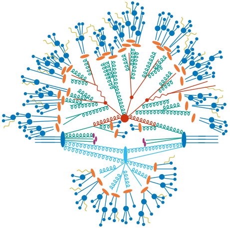
- Ease of Use
  - Focus on enabling physicists to concentrate on physics insights rather than technical details.
  - Minimal boilerplate code for cleaner, simpler analysis writing.
  - Familiar event loop structure and intuitive histogramming tools.
  - Streamlined integration for syncing results with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

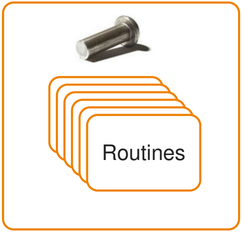


- Ease of Use
  - Focus on enabling physicists to concentrate on physics insights rather than technical details.
  - Minimal boilerplate code for cleaner, simpler analysis writing.
  - Familiar event loop structure and intuitive histogramming tools.
  - Streamlined integration for syncing results with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

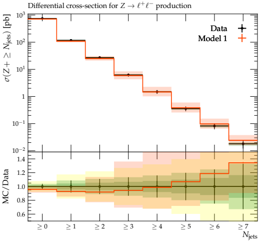
Rivet Workflow Overview



→



→



- 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.
  - Generator uncertainties via event weights handled automatically.

What is a Rivet routine?

- Analysis logic for processing simulated collision events, loadable at runtime from anywhere.
- 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.

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.

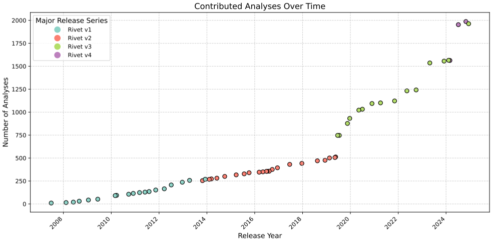
Main event loop: Retrieve event data, apply selections, fill histograms.

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

```
#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);
}
```

Collider Physics Analysis Preservation in Practice

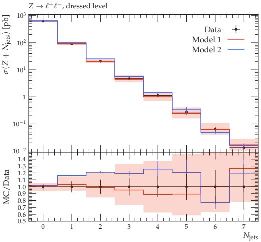
- Validated Repository
  - Central library of hundreds of published HEP analyses (and more).
- Transparency in Scientific Workflow
  - Open-source routines allow full inspection and replication of results.
  - Clear documentation of selection criteria and observable definitions.



- 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.

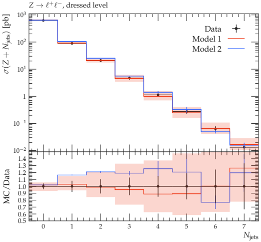
Community and Collaboration

- Bridging Theory & Experiment
  - Rivet is the **de facto standard** for comparing MC event generators with collider physics 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 collider physics 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 (HEP experiments, MC developers, theorists) helps **align best practices** and helps estasblish standards for event representation that best align with technology and physics principles.



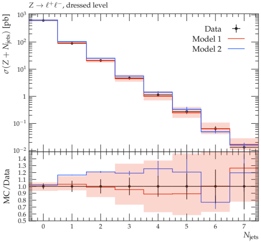
A standard convention for particle-level Monte Carlo event-variation weights

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 collider physics 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 (HEP experiments, MC developers, theorists) helps **align best practices** and helps estasblish standards for event representation that best align with technology and physics principles.
- 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

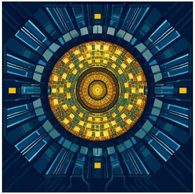
Enrico Bothmann, Andy Buckley, Christian Gütschow, Stefan Prestel, Marek Schönherr, Peter Skands, Jeppe Andersen, Saptaparna Bhattacharya, Jonathan Butterworth, Gurpreet Singh Chahal, Louie Corpe, Leif Gellersen, Matthew Gignac, Deepak Kar, Frank Krauss, Jan Kretzschmar, Leif Lönnblad, Josh McFayden,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

[arXiv:2203.08230]

Fiducial Cross Sections: Simple Idea, Big Impact

- A surprisingly powerful idea
  - Define cross-sections using observable final-state particles, within a clearly specified kinematic region.
  - Makes it easy to reproduce key plots, enabling real understanding, catching issues early, and improving MCs.
  - Establishes a shared language between theory and experiment – essential for tuning, fits, and reinterpretation.

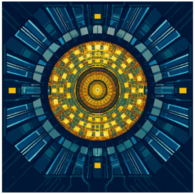
[fiducial]



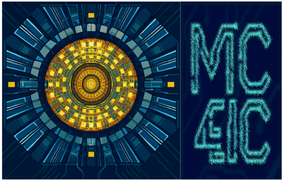
Fiducial Cross Sections: Simple Idea, Big Impact

- A surprisingly powerful idea
  - Define cross-sections using observable final-state particles, within a clearly specified kinematic region.
  - Makes it easy to reproduce key plots, enabling real understanding, catching issues early, and improving MCs.
  - Establishes a shared language between theory and experiment – essential for tuning, fits, and reinterpretation.
- But it's tempting to cheat. . .
  - Partons, bosons, and other "truth" objects in the event record look easy to use.
  - In practice, they're often ambiguous, model-dependent, or even non-existent (e.g. in higher-order simulations).
  - Focus on physical final states (hadrons, leptons, photons) as the reliable basis for comparisons.

[fiducial]



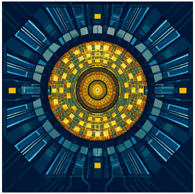
[extrapolated]



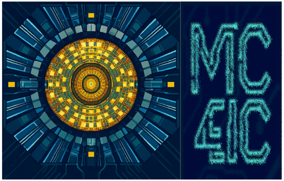
Fiducial Cross Sections: Simple Idea, Big Impact

- A surprisingly powerful idea
  - Define cross-sections using observable final-state particles, within a clearly specified kinematic region.
  - Makes it easy to reproduce key plots, enabling real understanding, catching issues early, and improving MCs.
  - Establishes a shared language between theory and experiment – essential for tuning, fits, and reinterpretation.
- But it's tempting to cheat. . .
  - Partons, bosons, and other "truth" objects in the event record look easy to use.
  - In practice, they're often ambiguous, model-dependent, or even non-existent (e.g. in higher-order simulations).
  - Focus on physical final states (hadrons, leptons, photons) as the reliable basis for comparisons.

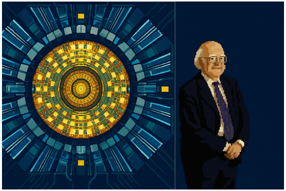
[fiducial]



[extrapolated]



[nature]



Why it matters

- Standardisation sounds boring—but agreeing on things like status codes PDG IDs, and weight schemes saves huge headaches later.
- Avoiding unphysical shortcuts has led to genuine physics insight, including:
  - Recognition of hadronisation as a “decoherence barrier”, guiding analysis definitions
  - Better truth tagging strategies: e.g. using fragmenting heavy-flavour hadrons rather than ancestry of soft partons
  - The shift to dressed leptons by default, i.e. truth leptons with their photon halos attached
- Common pitfalls to avoid:
  - Coloured objects aren’t final-state particles: “final-state tops” aren’t physical!
  - Electroweak bosons aren’t final states either: prefer leptonic or hadronic decay products.
  - Missing energy $\neq$ sum of neutrino momenta: use particle-level missing transverse momentum instead.
  - Hidden cuts hide physics: all vetoes and selections should be encoded in the fiducial definition, not just the code.

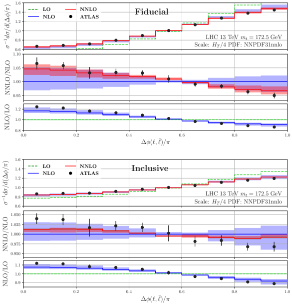


FIG. 1: NNLO QCD predictions for the fiducial (top) and inclusive selections (bottom) of the normalized $\Delta\phi_{\ell\ell}$ distribution versus ATLAS data [20]. Uncertainty bands are from 7-point scale variation.

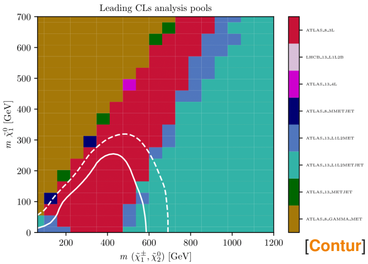
[arXiv:1901.05407]

Constraining QCD Models with EIC Data

- Testing and Refining Models
  - EIC will deliver high-precision data across a wide range of observables and beam configurations.
  - Rivet enables consistent comparison of QCD model predictions–PDFs, TMDs, nPDFs, hadronisation, and flow–with published measurements.
  - Discrepancies can highlight model limitations or the need for improved non-perturbative inputs.

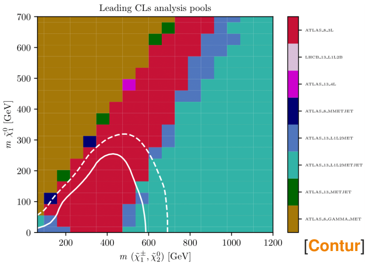
Constraining QCD Models with EIC Data

- Testing and Refining Models
  - EIC will deliver high-precision data across a wide range of observables and beam configurations.
  - Rivet enables consistent comparison of QCD model predictions–PDFs, TMDs, nPDFs, hadronisation, and flow–with published measurements.
  - Discrepancies can highlight model limitations or the need for improved non-perturbative inputs.
- Comprehensive and Reusable Coverage
  - Rivet provides a synoptic view across many final states, energies, and kinematic regions.
  - Once a Rivet routine exists, it can be immediately reused in global fits, model tuning, and generator benchmarking.



Constraining QCD Models with EIC Data

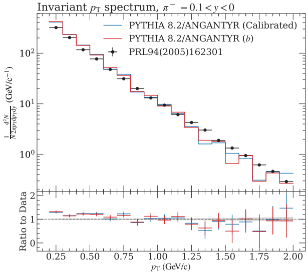
- Testing and Refining Models
  - EIC will deliver high-precision data across a wide range of observables and beam configurations.
  - Rivet enables consistent comparison of QCD model predictions–PDFs, TMDs, nPDFs, hadronisation, and flow–with published measurements.
  - Discrepancies can highlight model limitations or the need for improved non-perturbative inputs.
- Comprehensive and Reusable Coverage
  - Rivet provides a synoptic view across many final states, energies, and kinematic regions.
  - Once a Rivet routine exists, it can be immediately reused in global fits, model tuning, and generator benchmarking.
- Towards Long-Term Interpretability
  - Future theoretical advances–e.g. improved factorisation schemes or resummation frameworks–can be rapidly tested against preserved EIC measurements.
  - High-quality, model-independent data ensures enduring scientific value.

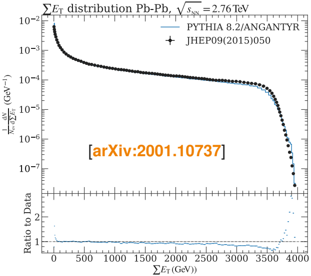


[Contur]

Heavy-Ion Support

- Supporting HI observables might sound simple – but in practice, it's a significant extension.
- HI analyses often require features like:
- Centrality calibration curves, which mandate 2-pass processing
- Event–event correlations, including centrality binning
- Swappable centrality definitions: few HI generators support all options (e.g. forward E_T and jet quenching).
- With HI MC standards still evolving, having a shared toolkit helps the community converge on best practices.

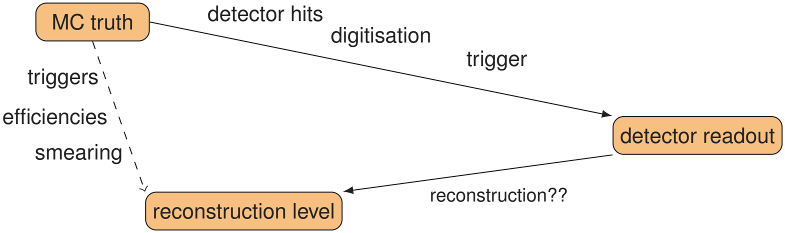




[arXiv:2001.10737]

Detector Smearing via Rivet Projections

- Built on Rivet's modular projection system:
  allows reco-level analysis without a full detector simulation.
- More flexible than Delphes: Smearing is analysis-specific,
  not hard-coded to a particular detector
- Enables studies like tunable jet-substructure smearing, or systematic variations.



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.

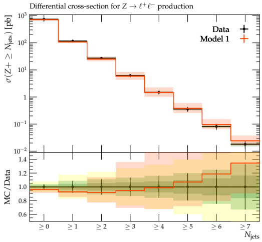
Challenges and Lessons Learned

- 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 HEP 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 colliders
  - Strengthening analysis preservation efforts

[**rivet.hepforge.org**]

[**gitlab.com/hepcedar/rivet**]

[**docker:hepstore/rivet**]

[**arXiv:2404.15984**]

Backup