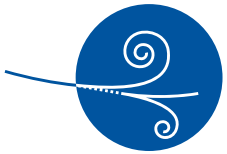
Rivet Usage at CMS

Markus Radziej

Thomas Hebbeker, Lars Sonnenschein



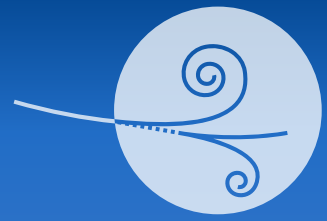
III. Physikalisches
Institut A

RWTH AACHEN
UNIVERSITY

Terascale 2014

Bundesministerium
für Bildung
und Forschung

Outline



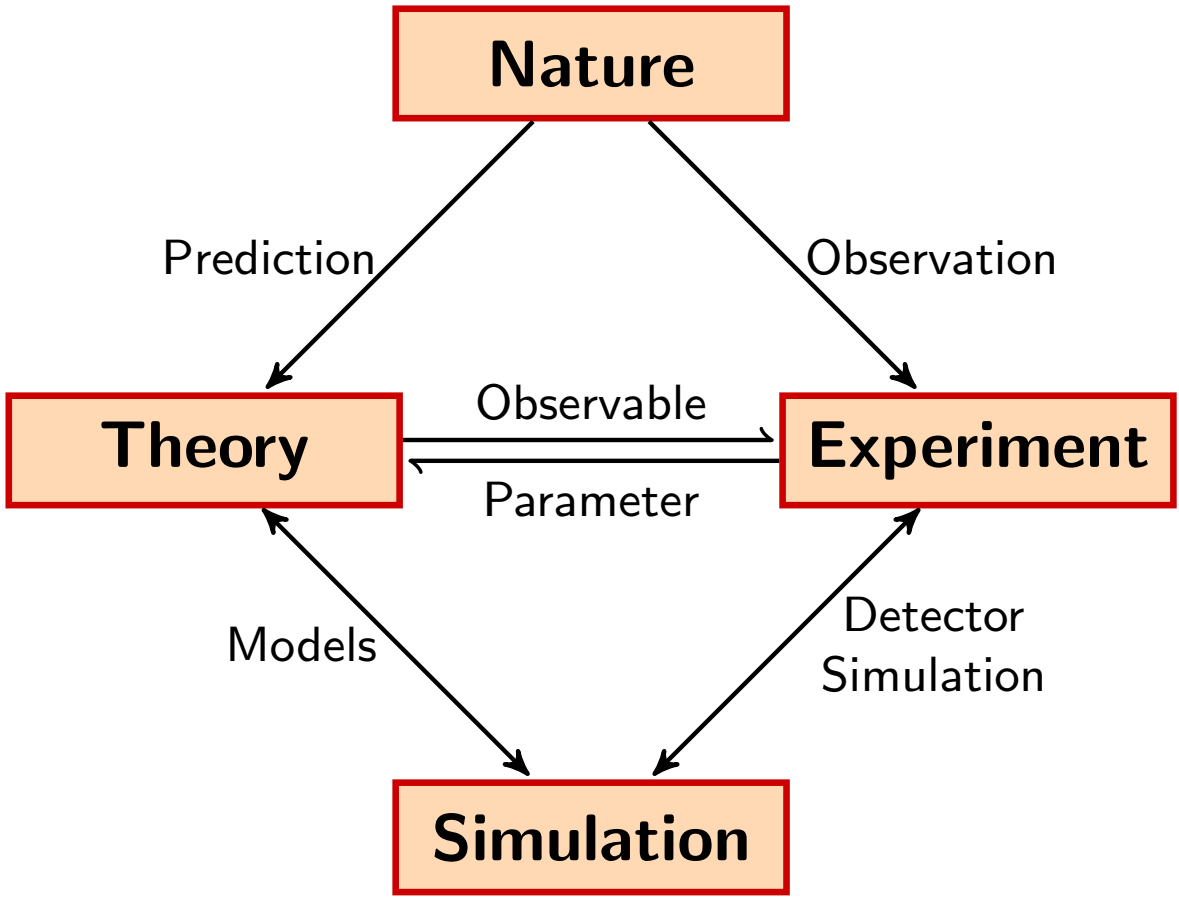
Robust
Independent
Validation of
Experiment and
Theory

Motivation

Rivet's place in particle physics

Four Cornerstones

- Continuous exchange of information
- Simulation connects theory and experiment

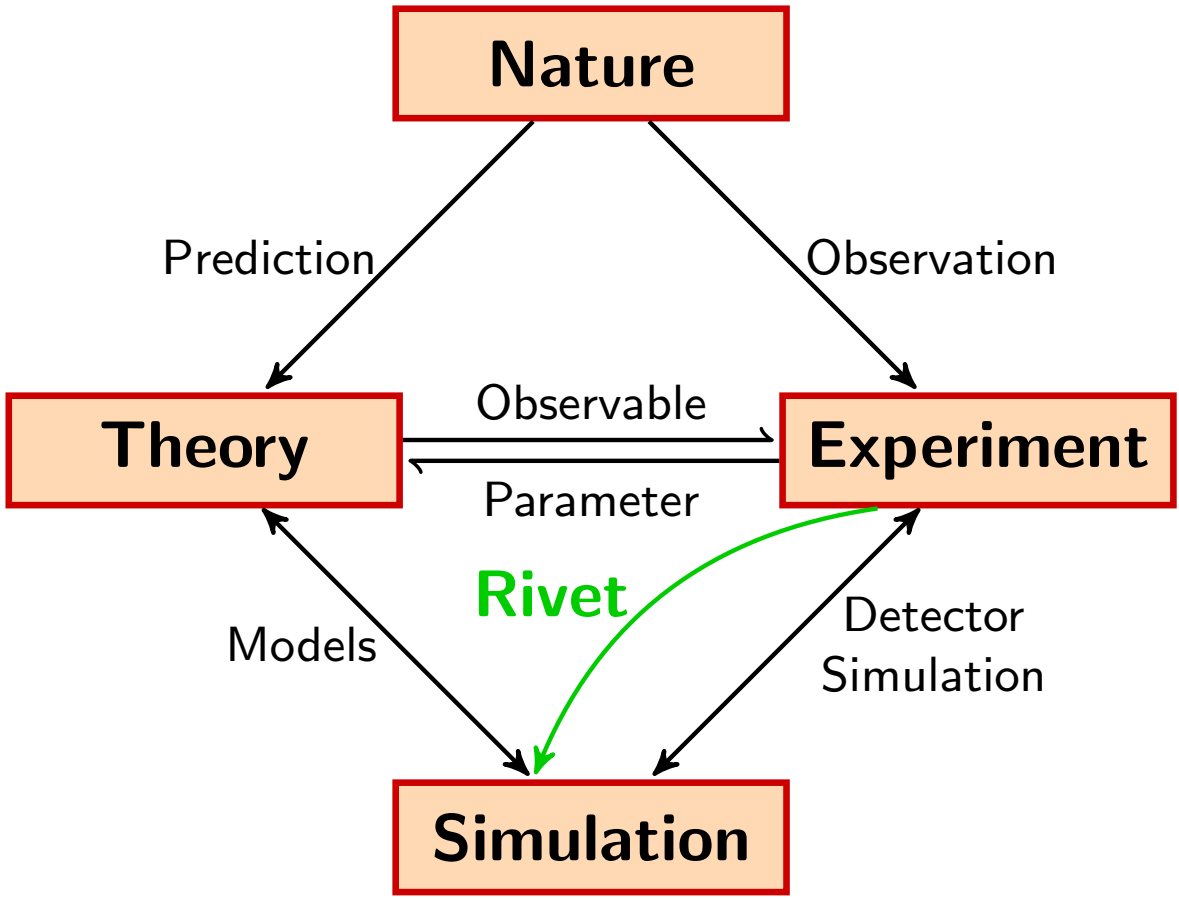


Motivation

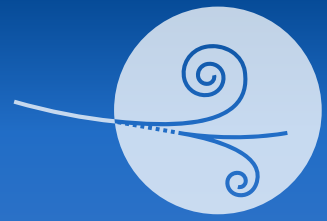
Rivet's place in particle physics

Four Cornerstones

- Continuous exchange of information
- Simulation connects theory and experiment
- **Rivet** supports simulation with data
- Focus lies on cross sections instead of observables
  e.g. mass, momentum



Event Generator Parameters



- Theory calculations are accurate up to a given order (LO, NLO, ... + eventually resummation)
- Generator implementation often combines phenomenological models with pertubative predictions
- Phenomenological parameters need to be determined from data

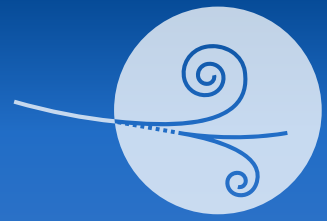
- Parton shower parameters
- Hadronisation - Lund string model, Herwig cluster model
- Parton distribution functions (PDFs)

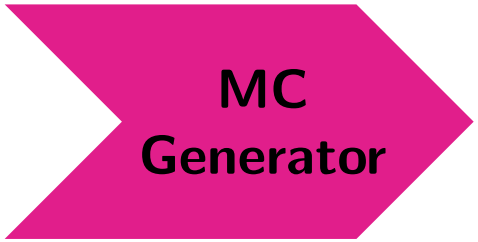
Using data for **validation & optimization** of **Monte Carlo** event generation is **essential**!

Section 2

Rivet Environment

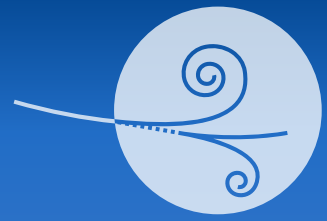
Input / Output
of the Rivet environment

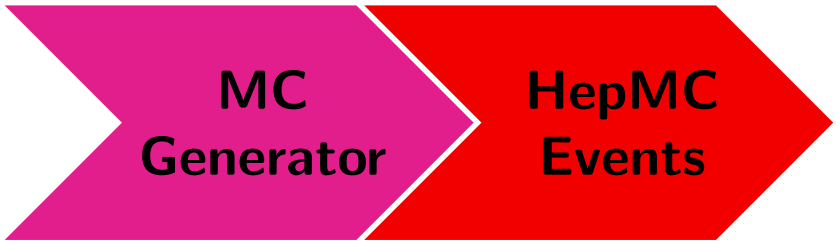




- Rivet works with almost any Monte Carlo event generator: Pythia 6/8, madgraph, Sherpa, ...

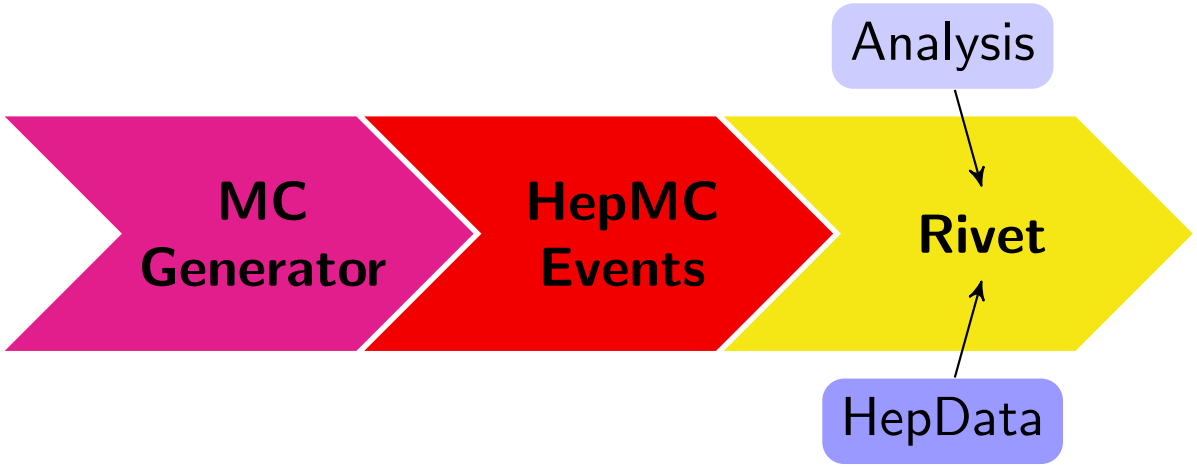
Input / Output
of the Rivet environment





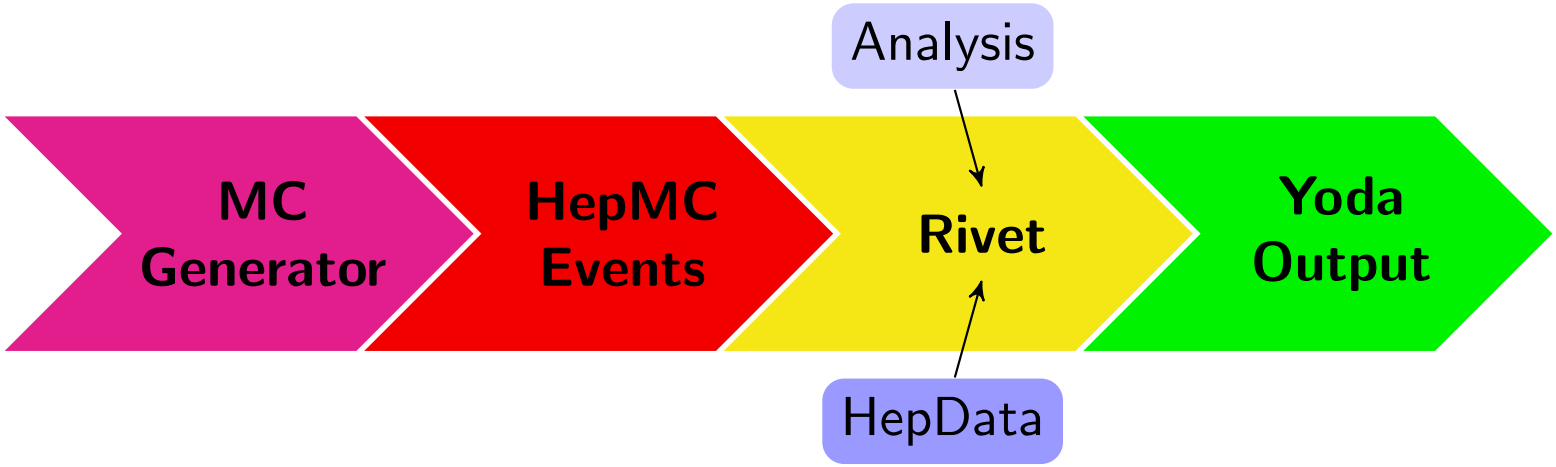
- Rivet works with almost any Monte Carlo event generator: Pythia 6/8, madgraph, Sherpa, ...
- HepMC is the basic input format
  (Can be piped to avoid storing large files)

Input / Output
of the Rivet environment



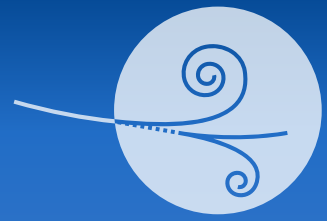
- Rivet works with almost any Monte Carlo event generator: Pythia 6/8, madgraph, Sherpa, ...
- HepMC is the basic input format
  (Can be piped to avoid storing large files)
- Rivet uses data of **HepData** database
  Needs **your analysis** as input!

Input / Output
of the Rivet environment



- Rivet works with almost any Monte Carlo event generator: Pythia 6/8, madgraph, Sherpa, ...
- HepMC is the basic input format
  (Can be piped to avoid storing large files)
- Rivet uses data of **HepData** database
  Needs **your analysis** as input!
- Yoda is a human readable plain text histogram format

HepData database



HepData - High Energy Particle physics Data

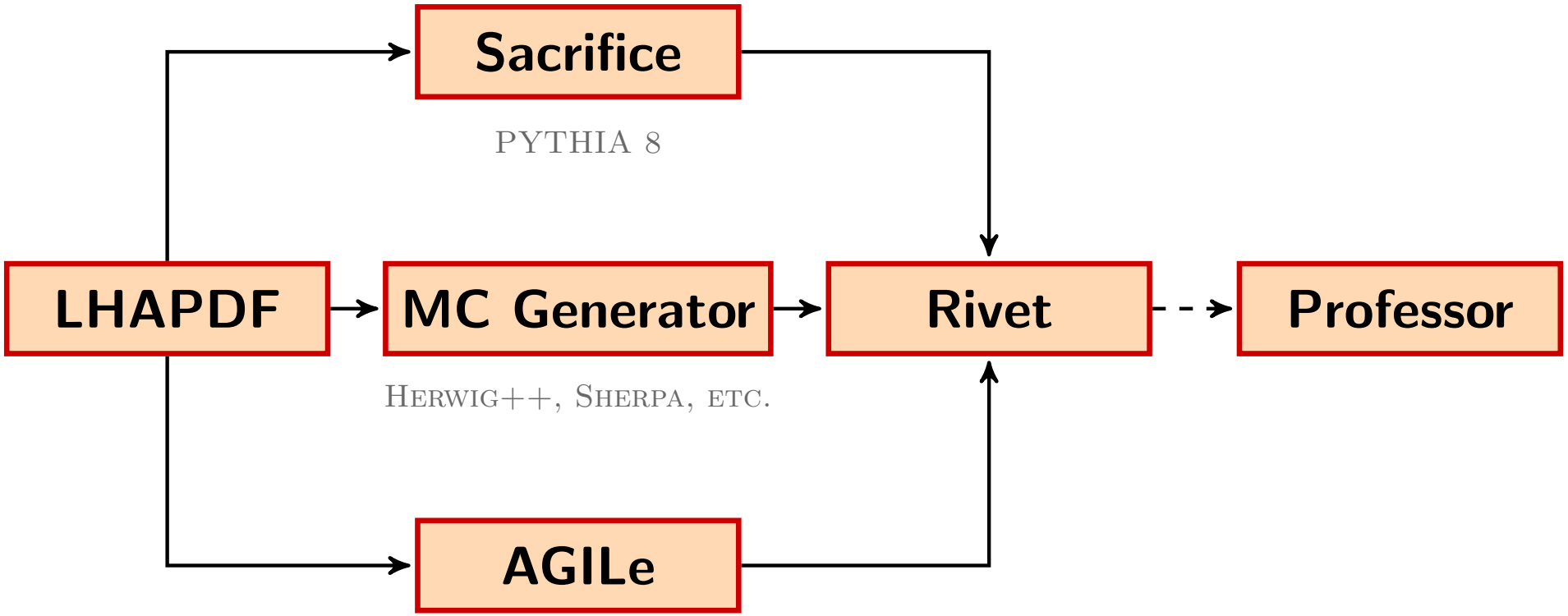
`http://hepdata.cedar.ac.uk/`

- Encompasses over 30 years of experimental results
  $\Rightarrow$ Mostly **corrected for detector effects**
- Focus on **cross sections** and **shapes of distributions**
  $\Rightarrow$ Complementary to PDG

Rivet

- **Rivet includes copy of HepData data** for existing analyses
- Rivet analyses work best with data corrected for detector effects
  $\Rightarrow$ **Corrected data** will be **always be useful**!
- Publishing authors can also provide their data for this database
  ($\Rightarrow$ contact respective Rivet Team)

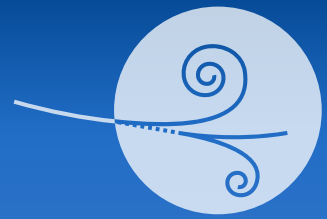
Software Tree



- LHAPDF allows for **different PDF sets**; Also **different tunes**
- **Generate events** on your own or use SACRIFICE & AGILE for steering
- PROFESSOR for Monte Carlo **event generator tuning**

Use Cases

Why you should care



- **Ease of use**
- **Particle level MC predictions** for any analysis, including your own
- Can reuse existing/old samples
- **Fast results** ($\sim 10\,\text{min}$) for low numbers of events because of omitting detector simulation
- **Simple to add more samples** produced with different PDF sets, tunes, MC generators

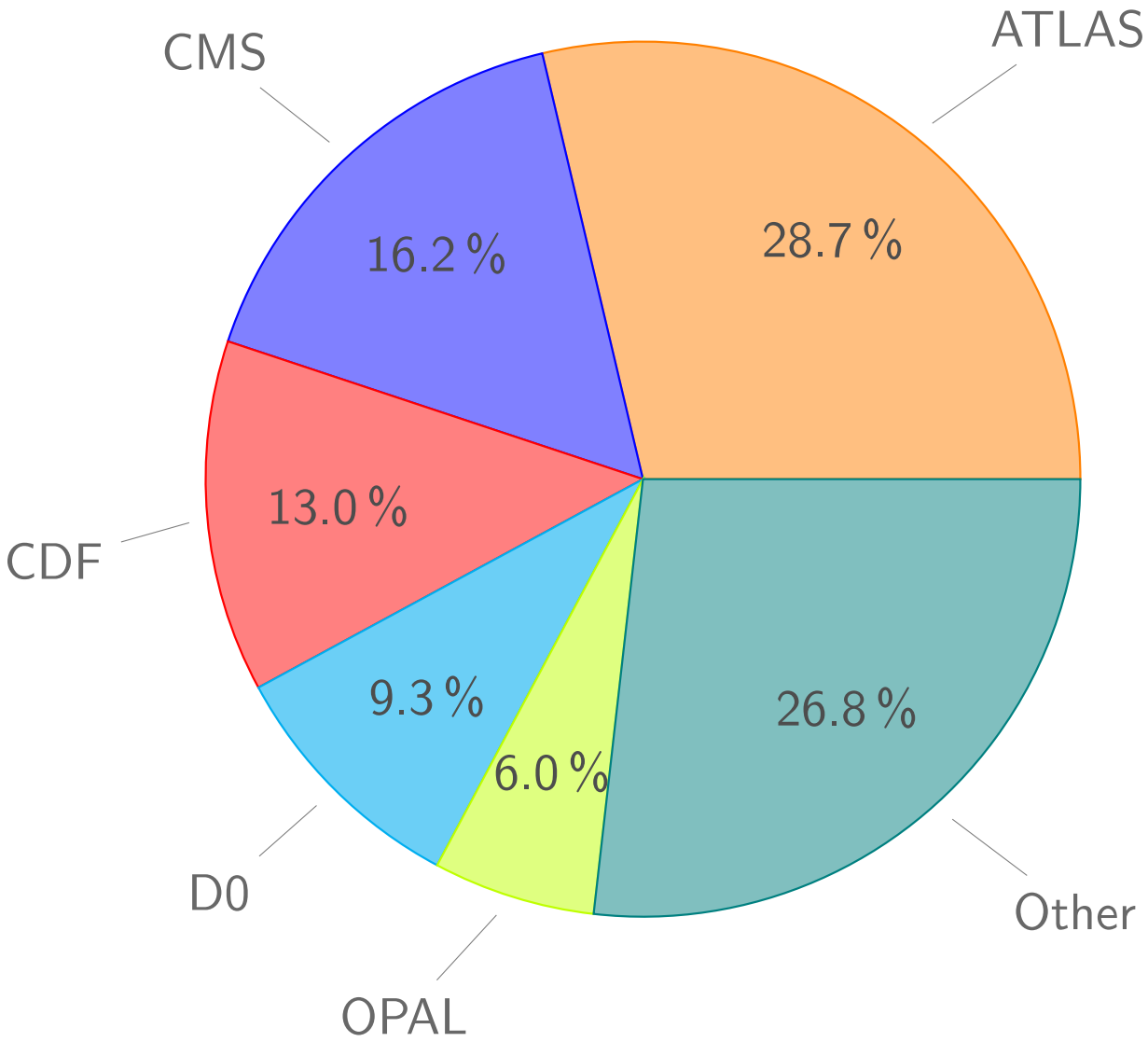
---

- Contribute to the community! Submit **your analysis**!

Section 3

Rivet Analyses

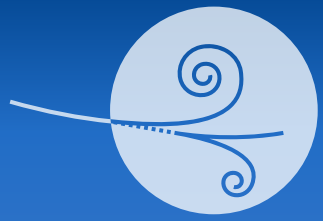
Overview of all Analyses



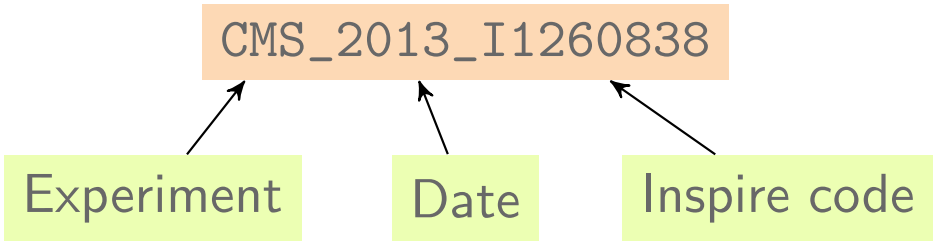
- Overall 216 Analyses in Rivet 2.2.0
- Major contribution from LHC analyses
- More analyses in queue already

Analysis Files

SMP-13-002



Naming scheme for SMP-13-002



- Analysis code, which yields histogrammed output

  `CMS_2013_I1260838.cc`

- General information regarding the publication, analysis, event generation, requirements, etc.

  `CMS_2013_I1260838.info`

- Options for generating graphical version of histograms

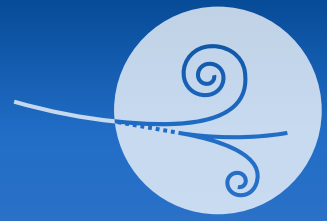
  `CMS_2013_I1260838.plot`

- (Unfolded) data histograms

  `CMS_2013_I1260838.yoda`

Code Structure

3 functions to fill



- Setting up the analysis, particle sets and histograms

```
void init() { [...] }
```

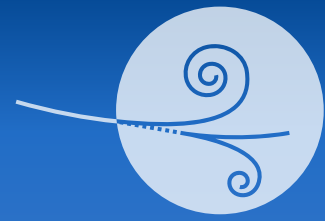
- Performing the per-event analysis

```
void analyze(const Event& event) { [...] }
```

- Normalize or manipulate histograms

```
void finalize() { [...] }
```

init()



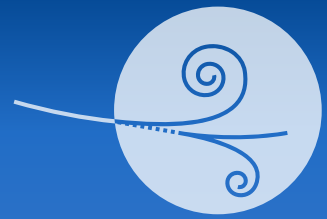
Projections

```
FinalState fs;

IdentifiedFinalState muon_fs(-2.4, 2.4, 5.0*GeV);
muon_fs.acceptIdPair(PID::MUON);
VetoedFinalState no_ele_fs;
no_ele_fs.addVEtoPairDetail(PID::ELECTRON, 20*GeV);

FastJets jetsak5(fs, FastJets::ANTIKT, 0.5);
addProjection(jetsak5, "JetsAK5");
ZFinder zfinder(fs, -2.4, 2.4, 5*GeV, PID::MUON, 60.0*GeV, 120*GeV);
addProjection(zfinder, "ZFinder");
```

Histograms

```
_h_pt_00_05_ak5 = bookHisto1D(1, 1, 1);
_h_pt_00_05_ratio= bookScatter2D(1, 1, 1);

BinnedHistogram<double> _ht_pt;
_h_pt.addHistogram(0.0, 0.5, bookHisto1D(1, 1, 1));
_h_pt.addHistogram(0.5, 1.0, bookHisto1D(2, 1, 1));
```

analyze()



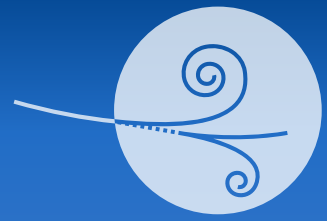
Applying Projections

```
const Jets& jetsak5 = applyProjection<FastJets>(event, "JetsAK5").
    jetsByPt(56.0*GeV);

const ZFinder& zfinder = applyProjection<ZFinder>(event, "ZFinder");
```

Filling Histograms

```
// Filling R = 0.5 jets
foreach(const Jet& jet, jetsak5) {
  FourMomentum jmom = jet.momentum();

  if (jmom.absrapidity() < 0.5) {
    _h_pt_05_ak5.fill(jmom.pT()/GeV, weight);
  }

  _h_pt.fill(jmom.absrapidity(), jmom.pT()/GeV, weight);
}
```

finalize()



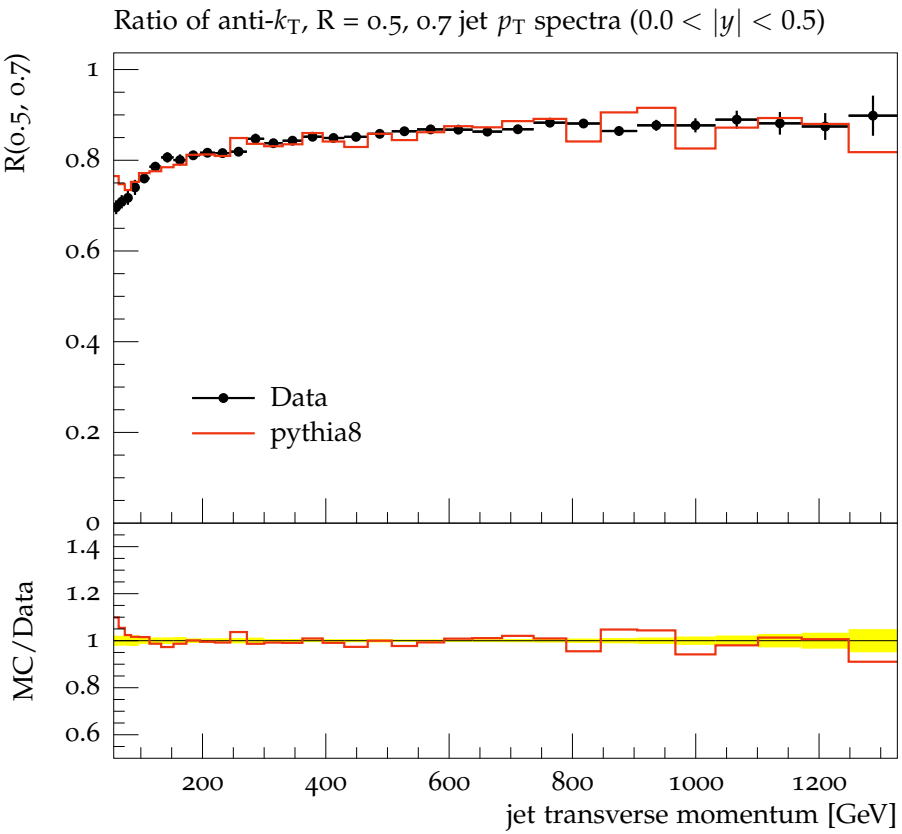
Histogram Manipulation

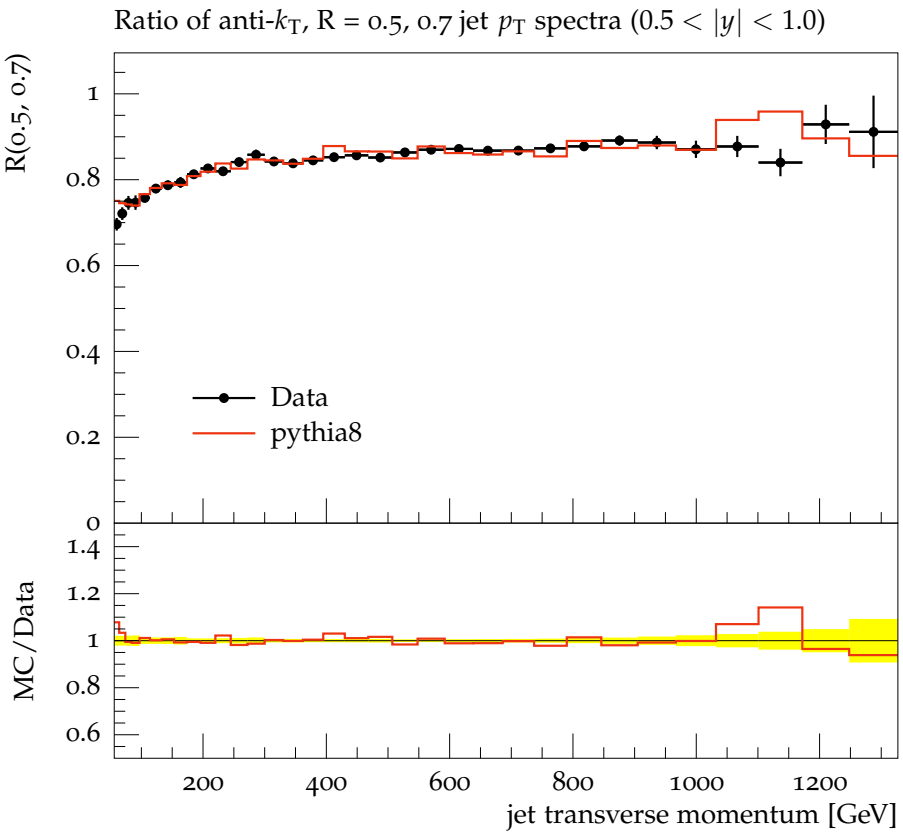
```
scale(_h_pt_00_05_ak5, crossSection() / sumOfWeights());

normalize(_h_pt_00_05_ak5, 1.);

divide(_h_pt_00_05_ak5, _h_pt_00_05_ak7, _h_pt_00_05_ratio);
```

Remaining steps

- Compiling the source code
  `rivet-buildplugin CMS_2013_I1260838.cc`
- Calling the routine
  `rivet -a CMS_2013_I1260838 /path/to/mc_file.hepmc`
- Generate plots and html overview
  `rivet-mkhtml Rivet.yoda`

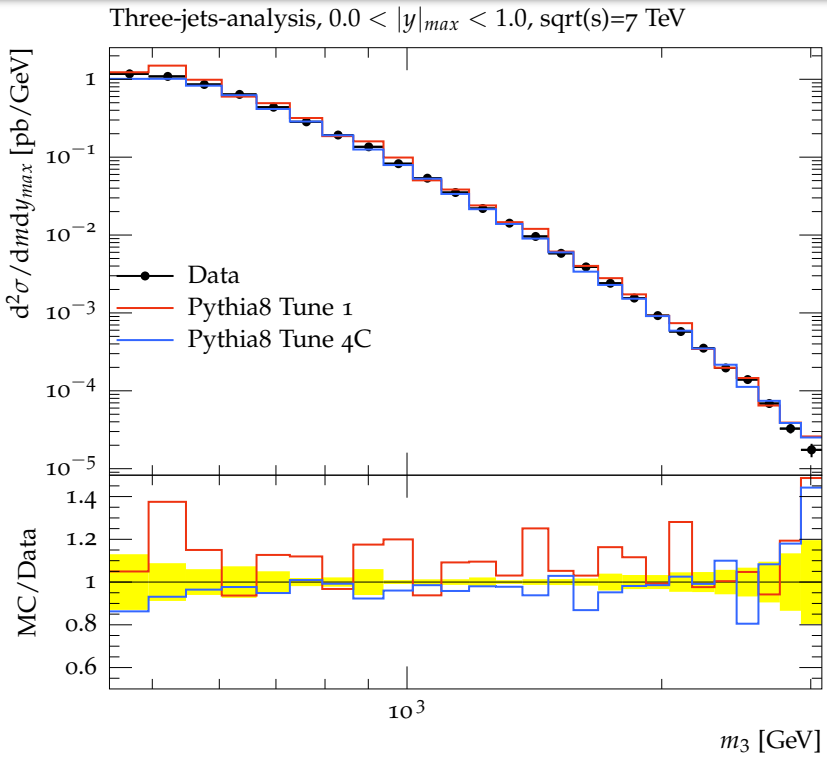
Results

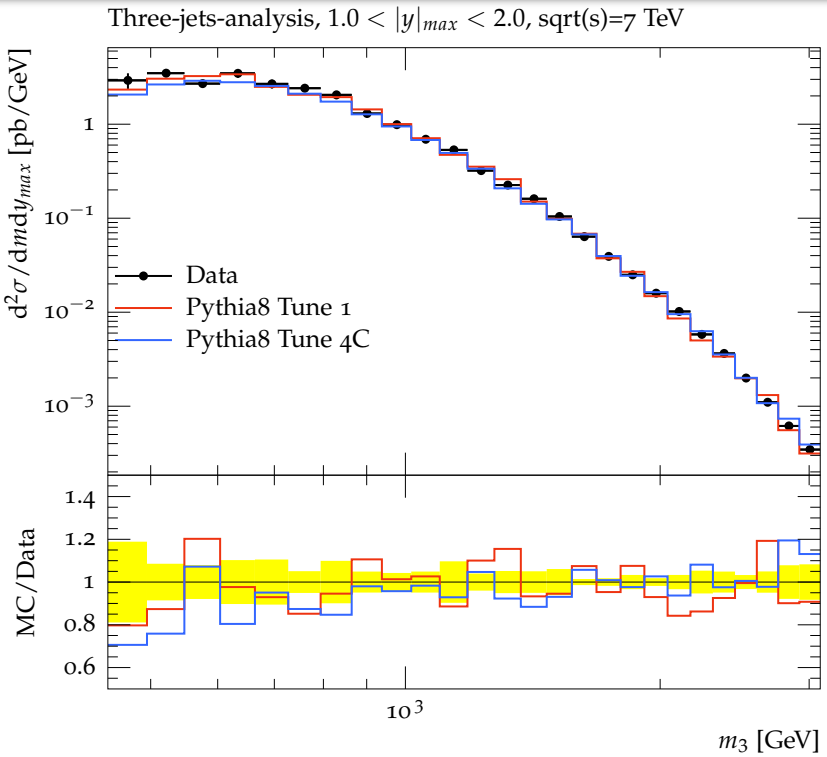




- 5M events generated with PYTHIA 8, analyzed in 90 minutes
- Produced with "flat" momentum prioritization to ensure sufficient statistics for high momenta

Future of Rivet





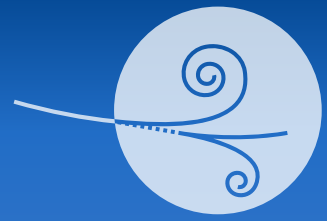
- Large pool of analyses available (200+)
- Many more already in progress or in queue

New analysis types

- **Top quark analyses**: Implementation test with TOP-10-028
- **Higgs analyses**: Being tested for viability

Resources

If Rivet piqued your interest ...



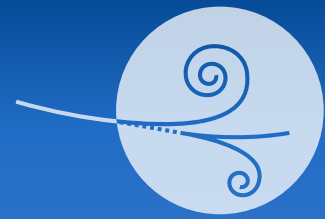
`rivet.hepforge.org`

- Standalone version — Easy installation thanks to "bootstrap"
  `https://rivet.hepforge.org/trac/wiki/GettingStarted`
- Instructions on how to write an analysis
  `https://rivet.hepforge.org/trac/wiki/WritingAnAnalysis`

  Also consult the existing code in `Rivet-2.X.X/src/Analyses`

- Bundled with CMS software package
  `https://twiki.cern.ch/twiki/bin/viewauth/CMS/Rivet`

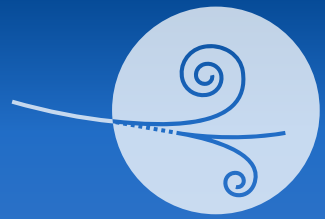
References



M.R. Whalleyp
The Durham-RAL high-energy physics databases: HEPDATA.
Inspire:289771

T. Sjöstrand et. al.
An Introduction to PYTHIA 8.2.
arxiv:1410.3012 [hep-mc]

A. Buckley et. al.
Systematic event generator tuning for the LHC
arxiv:0907.2973 [hep-mc]



Thank you for your attention

Questions ?

Markus Radziej
radziej@physik.rwth-aachen.de