

Rivet: past and future

Chris Pollard
University of Glasgow

for the Rivet team

MCnet Meeting
2016 11 24



University
of Glasgow



What is Rivet?

Robust **I**ndependent **V**alidation
of **E**xperiment and **T**heory

In other words: “toolkit for robust comparison of physics models to experimental data”

<http://rivet.hepforge.org>



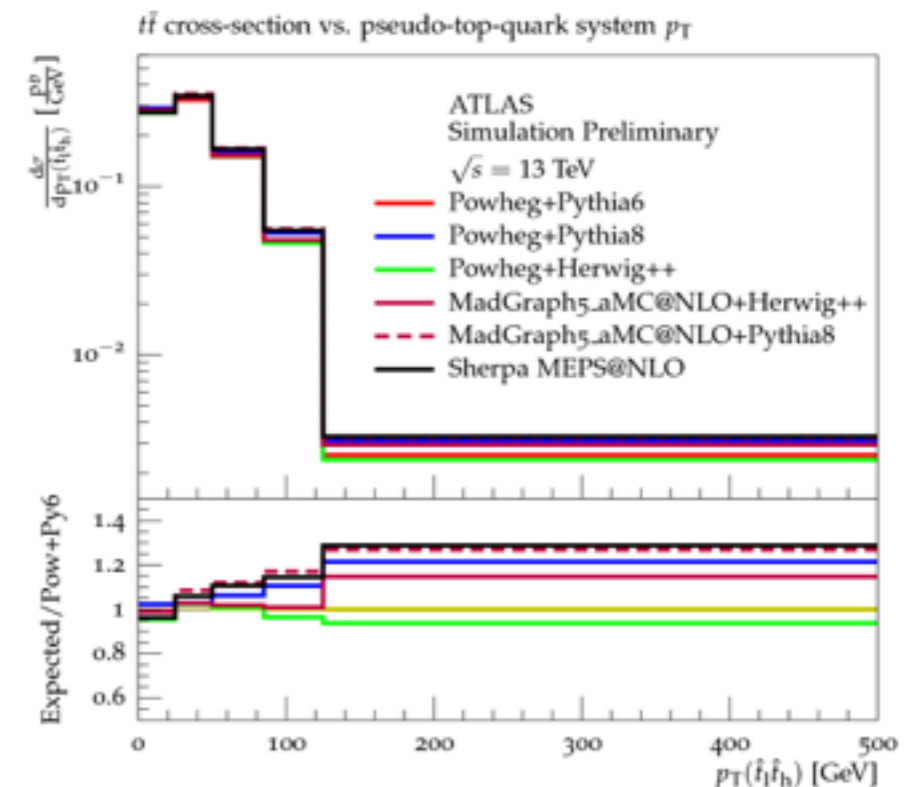
What it's made of

1. C++/python framework for **developing and documenting particle-level physics analyses**
2. Collection of **analyses** (code) and **corresponding data**
3. Tools for analysis object (e.g. histogram) manipulation, storage, visualization, etc. (**YODA**: <http://yoda.hepforge.org>)

```
// initialize physic object definitions
// and histograms
void init() {
    // use all final, visible particles with
    FinalState calofs(Cuts::abseta < 4.8);

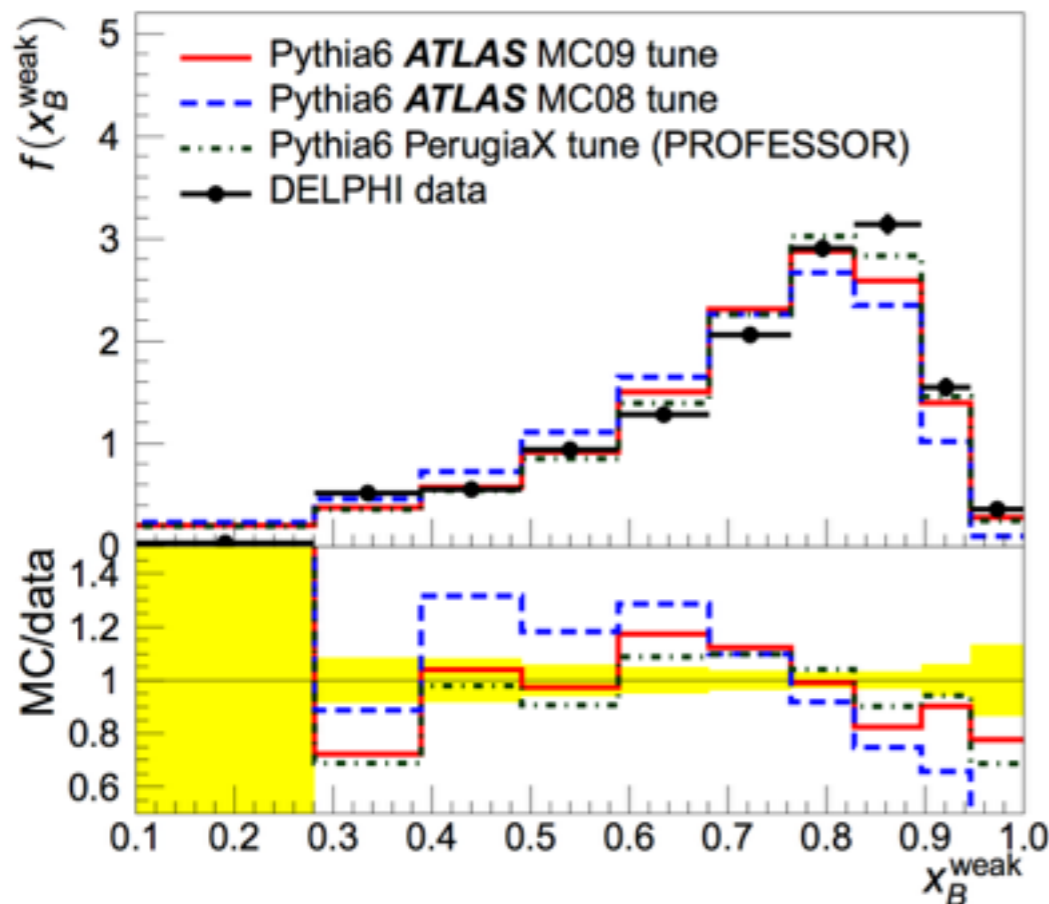
    // build anti-kt 0.4 jets out of these
    FastJets fj(calofs, FastJets::ANTIKT,

    // declare these jets for use in selection
    declare(fj, "TruthJets");
```



Analysis preservation

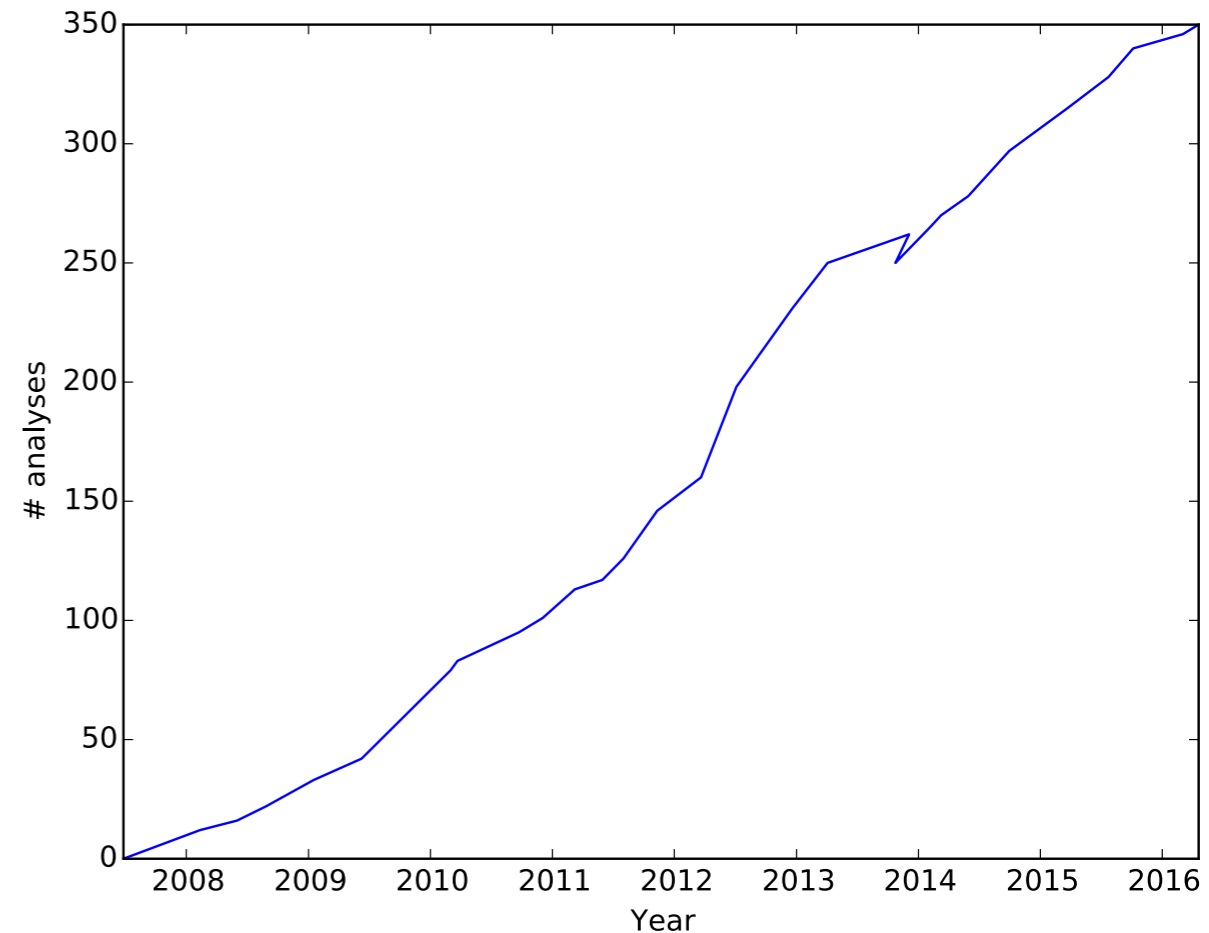
- Rivet releases include both **the recorded data** and **the corresponding fiducial definition** (code).
- We work with both theory and experimental communities in order to **simplify**



1. preservation for experimentalists: short, understandable code, easy-to-use API
2. interpretation: easy-to-run analyses, useful output, automatic data comparisons

How we do it

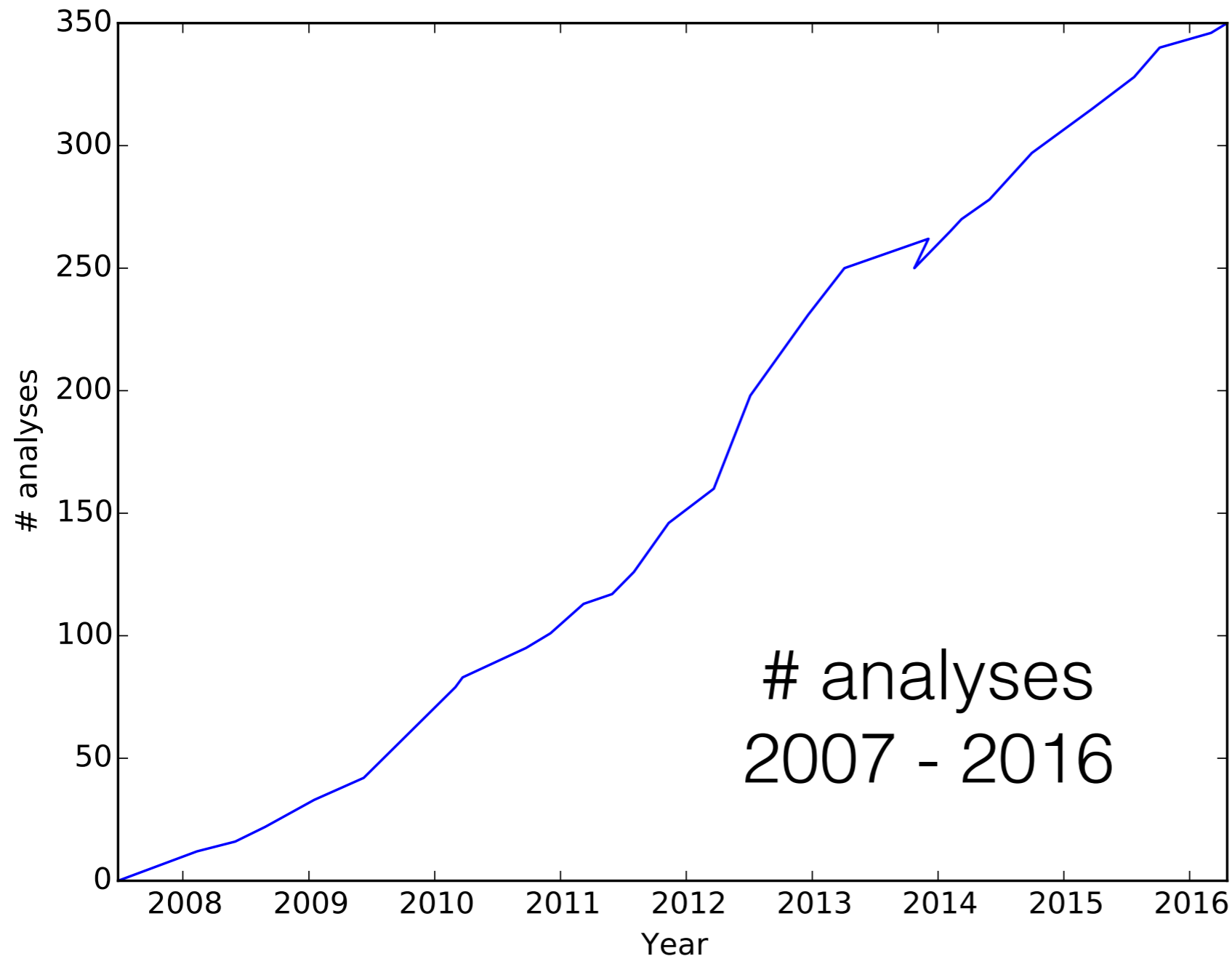
- Nowadays there is a dedicated **Rivet contact person** in each of the four major LHC experiments.
- This contact is responsible for **code style, validation** (analysis and data), **errata**, and **analysis submission**.
- After submission, **analyses are maintained by the Rivet team**.



analyses
2007 - 2016

Analyses

> **350 analyses are included in latest release. (!!)**



several are added
with each release
*by the experiment
contacts*

**every new ATLAS
analysis from the
SM and top groups**

PS: that's a lot of
maintenance...

Why we need it

Rivet is the *de facto* standard for preserving HEP collider measurements (LHC+Tevatron+LEP+...).

At this point, **Rivet is critical for the preservation of HEP data for current and future interpretation.**



Rivet in experiments

Rivet is now also used for a variety of tasks **within the experiments**

- Validation of MC generation
- MC generator tuning
- Fiducial cross section and uncertainty determination
- Internal analysis preservation



User-facing improvements

One of Rivet's major selling points is that it is simple to encode even complicated analyses.

Over the past four years there have been many improvements on this front:

- Better analysis object handling with YODA
- 2D histograms, 2D profiles, counters...
- Improved FastJet integration & jet handling
- Cuts system
- and more...



Behind-the-scenes improvements

Of course, behind-the-scenes improvements are always welcome...



- C++11 support and removal of BOOST dependency
- Faster I/O for YODA objects
- Improved validation of analyses release-to-release
- Many, many more...

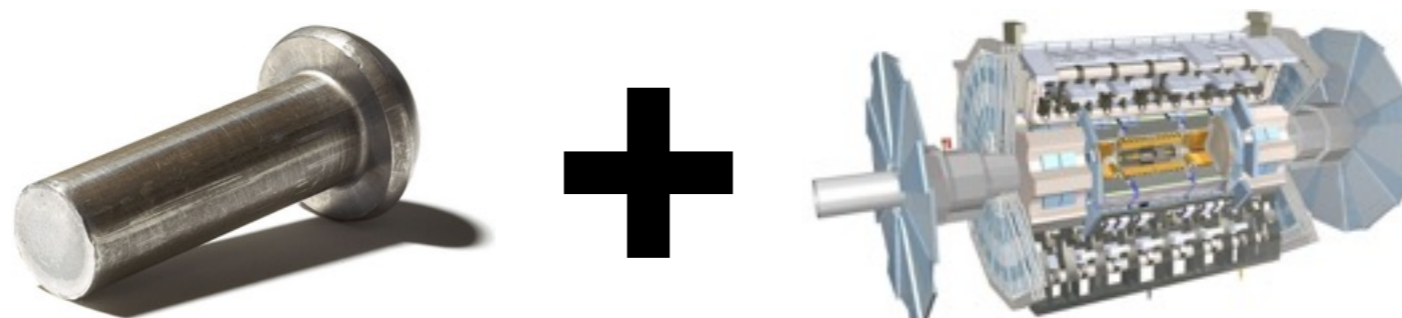
New and future developments

1. Preservation of data that is **not** unfolded to particle-level
2. Automatic handling of weight-based systematic uncertainties
3. Laundry list of other features

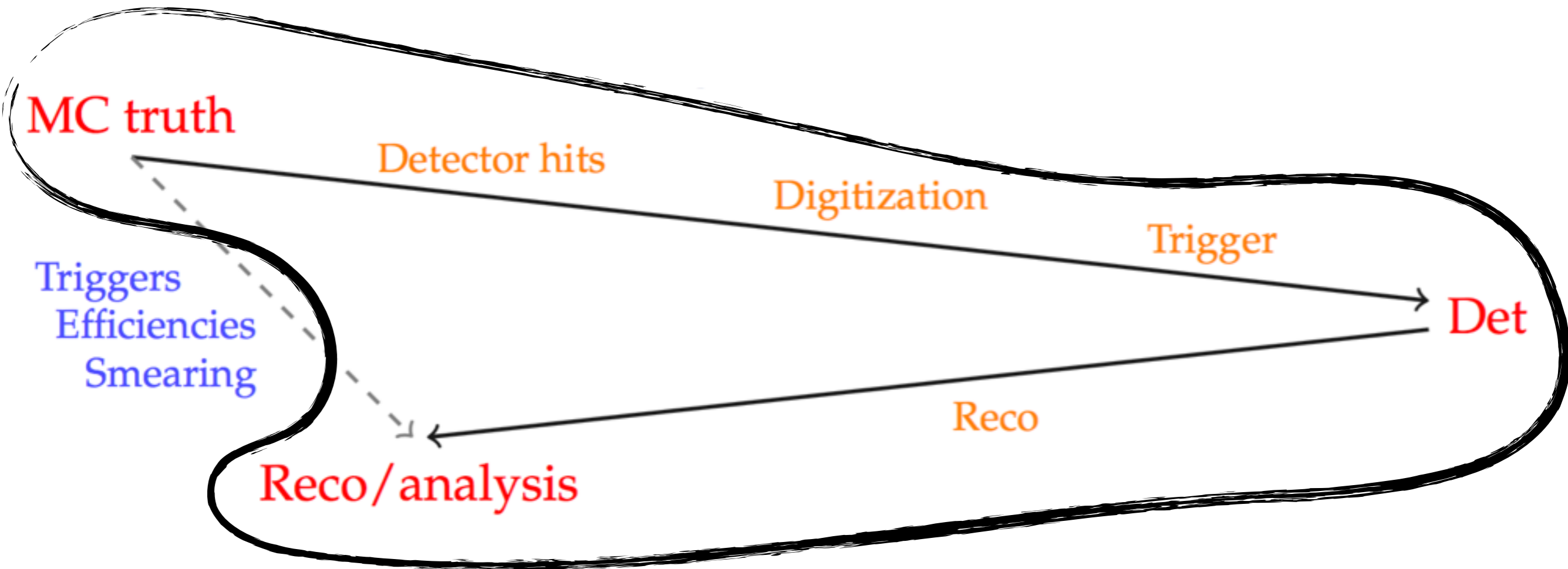


Analysis preservation of reco-level results

- There are many analyses that do *not* unfold their data, but we think they ought to be preserved nonetheless.
- BSM searches and searches for difficult/rare SM processes (eg. $H \rightarrow b\bar{b}$) are often not unfolded to particle level.
- HepData entry + publication seldom provides the info needed to reliably reproduce analysis (i.e. it's ~not useful beyond the scope of the original publication).
- With Rivet 2.5 we began providing simple extensions to the current infrastructure to help preserve such analyses.

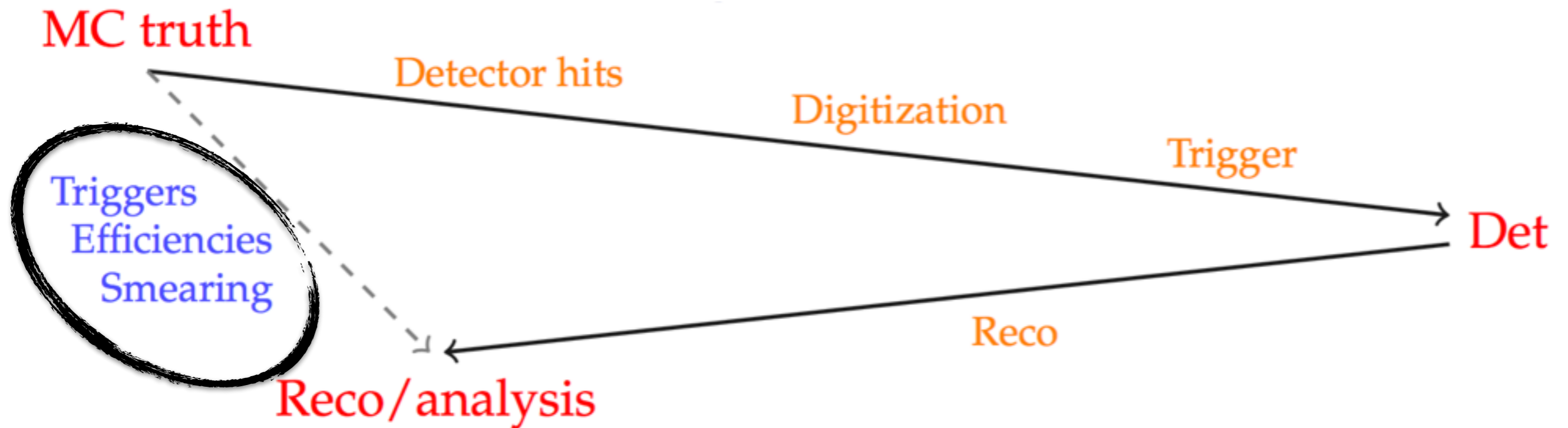


Reco-level analyses: experiments



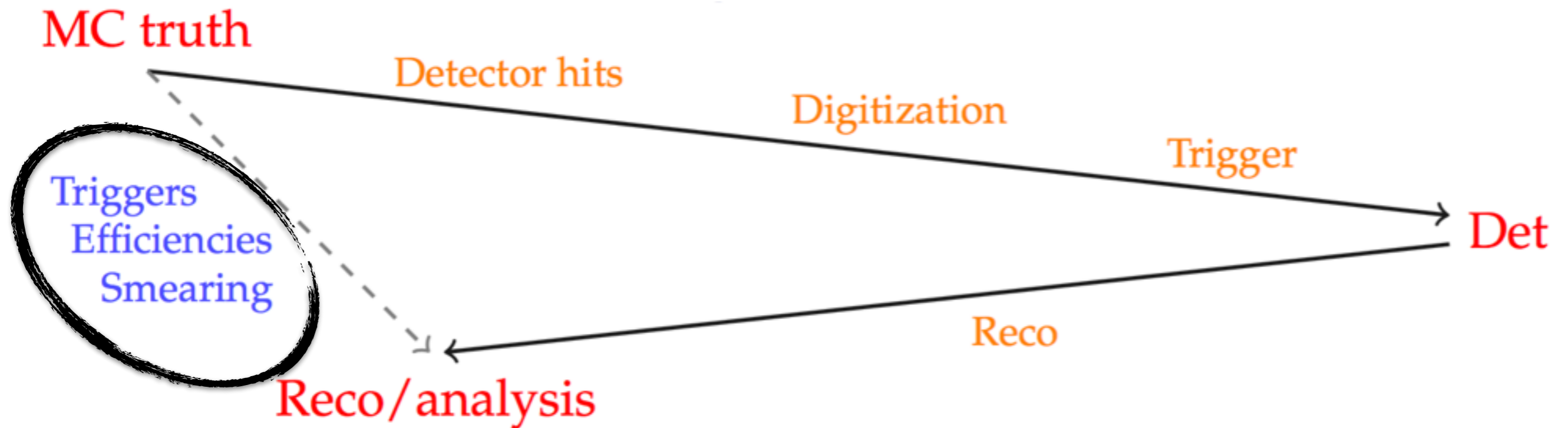
How predictions are propagated through
detector experiments

Reco-level analyses: Rivet



- Experiments encode their selection and *approximate* detector effects **into the Rivet analysis.**
- Include background-subtracted data **at reco level** (should already be on HepData) in Rivet releases

Smearing functions



```
// build anti-kt 0.4 jets out of these particles  
FastJets fj(calofs, FastJets::ANTIKT, 0.4);
```

```
// declare these jets for use in selection  
declare(fj, "TruthJets");
```

```
// smear the TruthJets' four-momenta by function defined in JET_SMEAR_ATLAS_RUN2  
// apply b-tag/mistag rates from JET_BTAG_ATLAS_RUN2_MV2C20  
declare(SmearedJets(fj, JET_SMEAR_ATLAS_RUN2, JET_BTAG_ATLAS_RUN2_MV2C20), "RecoJets");
```

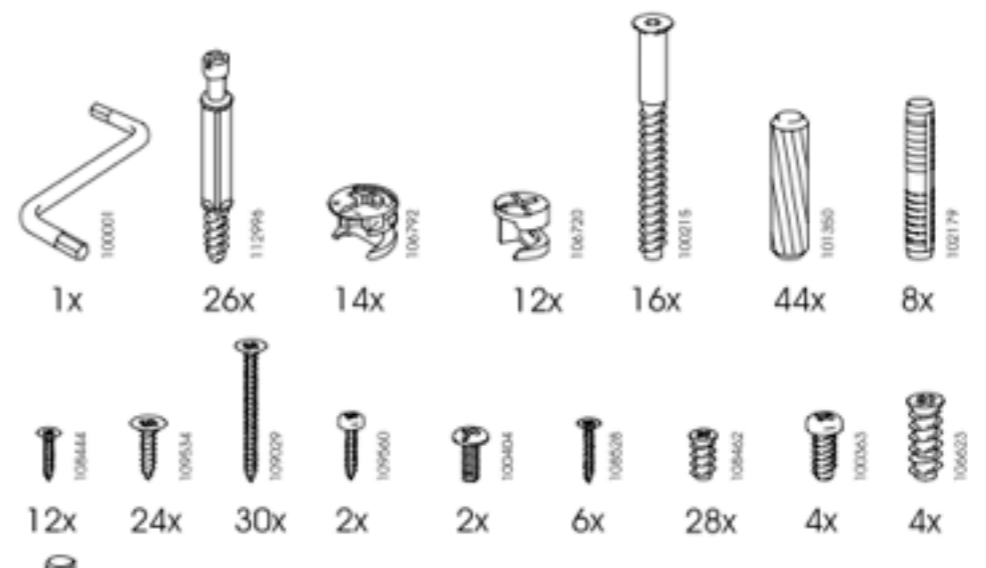
Reco-level analyses: summary

Just to reiterate...

- We think it's important to have the tools in place to properly preserve searches.
- There are several similar tools on the market for this, but they are not used uniformly like Rivet is for measurements.
- Currently Rivet has ~25 validated BSM analyses, but mostly Run I (or older); we'd like to improve this.
- We are already getting some feedback from experiments on our implementation.

Rivet: systematic weights

- Next major release (Rivet 3.0) will provide automatic handling of events with
 - **multiple weights** (eg. factorization scale, PDF variations)
 - **NLO sub events** (to be treated statistically as fully correlated)
- Each histogram, profile, etc. will have several versions stored—one corresponding to each event weight variation.
- **Rivet 3.0alpha1** already released with an implementation for 1D histograms.
- Analysis code will *change very little*: won't explicitly handle the event weight.
- As always, **analyses already in Rivet will be updated by the Rivet team.**



Rivet: other plans I

There are several other new features/enhancements we are working on:

- “Re-entrant histogramming”
 - Can merge Rivet runs (even those with eg. histogram divisions)
 - Can plot histograms during the Rivet run to make sure they look ok (already partially implemented)
- Better and faster plotting tools based on common python libraries
- Much faster compile times



Rivet: other plans II

There are several other new features/enhancements we are working on:

- HepMC 3 support
- Scatter plots with more than one associated uncertainty (eg. separate syst, stat)
- Thread safety for large parallel runs (already there to some degree)
- Luminosity info in analyses
- Support for running groups of analyses by *tag* (eg. all ttbar analyses, all dijet analyses)



Rivet: summary

- Hopefully that was a useful summary of past and future developments in Rivet.
- As always, we are happy for feedback “both sides of the aisle”.
- Rivet and YODA websites with documentation:
<http://rivet.hepforge.org>, <http://yoda.hepforge.org>
- Rivet developer mailing list:
rivet@projects.hepforge.org