

multiweighted  
+ subevented  
rivet

# multiweighted events: what's the point?

- We often encode theoretical systematic uncertainties by weighting events according to how likely they are to occur under some variation w.r.t. the nominal.
- We want to make it easy (automatic!) to test a nominal prediction with an uncertainty band against the unfolded data.

# multiweighted events: how we do this

in a nutshell, we keep several copies of each booked histogram:

- **book()** now returns a “smart pointer” to an analysis object that **knows how many copies** to instantiate: one per event weight.
- **analyze()** and **finalize()** are now run **once per event weight**, with **fill()** (and similar) acting on the internal histogram corresponding to the correct weight.
- A **copy of each histogram** is stored in the **outgoing yoda file**. Currently **not providing plotting scripts** to make an uncertainty band: the band **depends on the meaning of the systematic variations**.

# multiweighted events: pitfalls

several implications of `analyze()` and `finalize()` being run once per event weight:

- most importantly, **`analyze()` can no longer depend on the value of an event's weight**: the underlying generator weight can't be accessed or modified by the user.
- **`fill()`** now implicitly looks up the event weight to adjust the histogram; any additional weight can be included by the user.
- The sum of weights is also stored for each weight variation and accessible in **`finalize()`**.
- Will people use the total **predicted cross section** correctly (including variations)?
- Any **storage of info outside multiweighted AnalysisObjects** will have **undefined behavior**.

# counter events: overview

Next-generation calculations throw **several “subevents”** that should correspond to one **“real” MC event...**

- In practice it's a bit tricky to get the statistics correct...
- We have to **maintain a list of fills of subevents** and **mimic a “single event” fill** once all subevents have been collected.
- Basic handling of this is **implemented for 1D histograms** (2+ dimensions is quite challenging).

# status

Implementation of multiweighted histograms and counter events (1D histos) is **stable** and **validated** for the **nominal histograms**

- **Small discrepancies** still for the **variation histograms**, but these are being understood (David?)
- There are about **15% of analyses** that do something that is **not “multiweight safe”**, e.g. keeping a `vector<double>` of fills, having the fiducial definition depend on the event weight, etc.
- Generically, analyses like this are not physically well-defined...