Cross Section Software: MINERvA's Framework

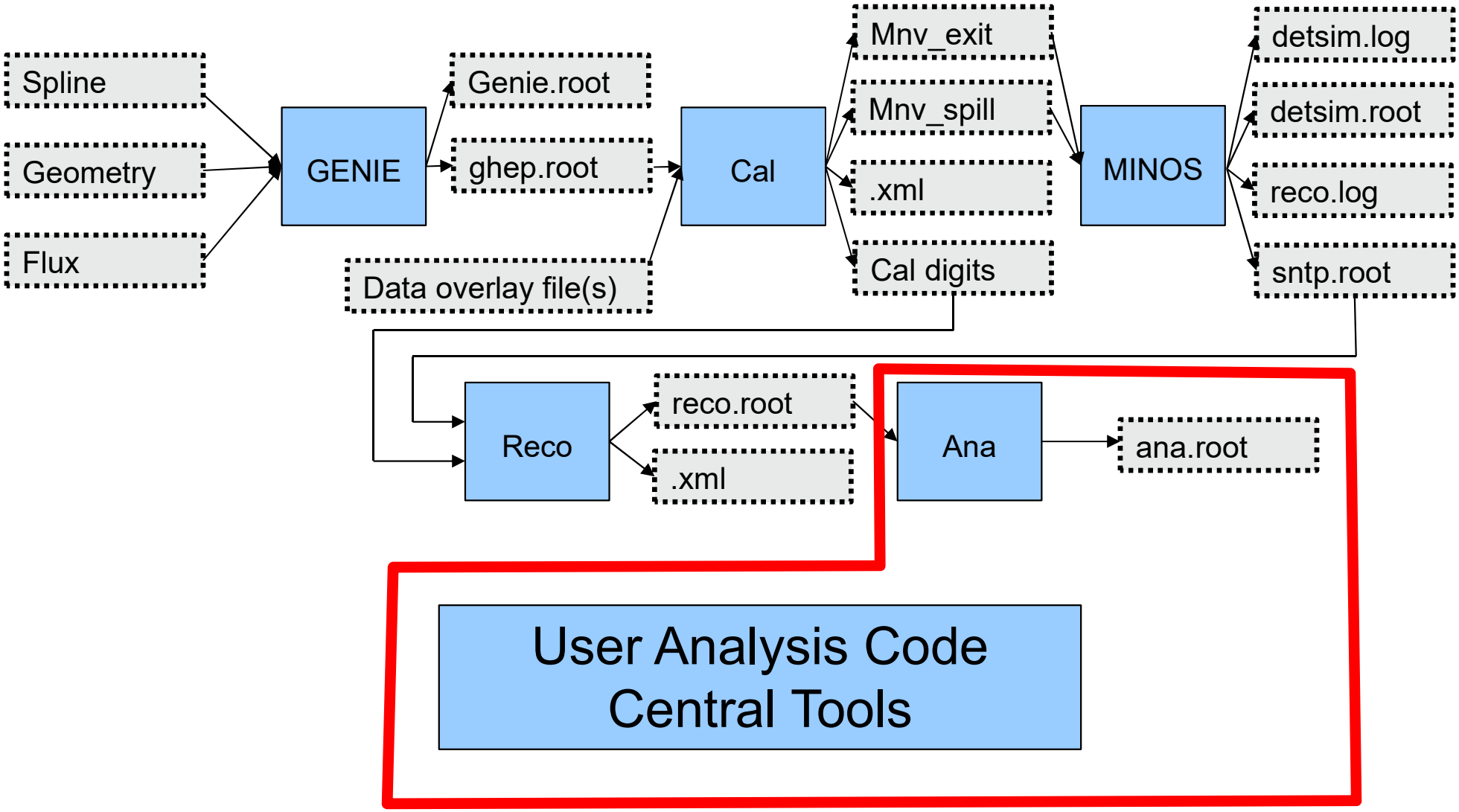
Daniel Ruterbories

July 24th, 2019

MINERvA Software

- Gaudi framework for calibration, general reconstruction
- Every analyst runs reconstruction/variable calculation
  - General code inserts classes of systematics weights/central value corrections/Truth information
  - User can add truth and reco level information
- “Flat” trees are produced which are used for analysis

Stages of Processing



Gaudi Stage (ana)

Gaudi Tool

- Provides users a “physics event” which they can choose to apply additional reconstruction algorithms to the hits/clusters etc.
- Allows for Hypothesis based on reconstruction
  - Issues with object mapping (used versus unused hits)
  - Ideally, you’d want to run Class A and then have two sub classes of events for hypothesis, but this isn’t very easy to do in our framework
- **Bottom line** – flexible reconstruction for analysis specific topologies is important

Macro Stage

ROOT with special classes (MnvH*D, MnvVertErrorBand, MnvLatErrorBand,…etc.)

External packages – RooUnfold (arXiv:1105.1160)

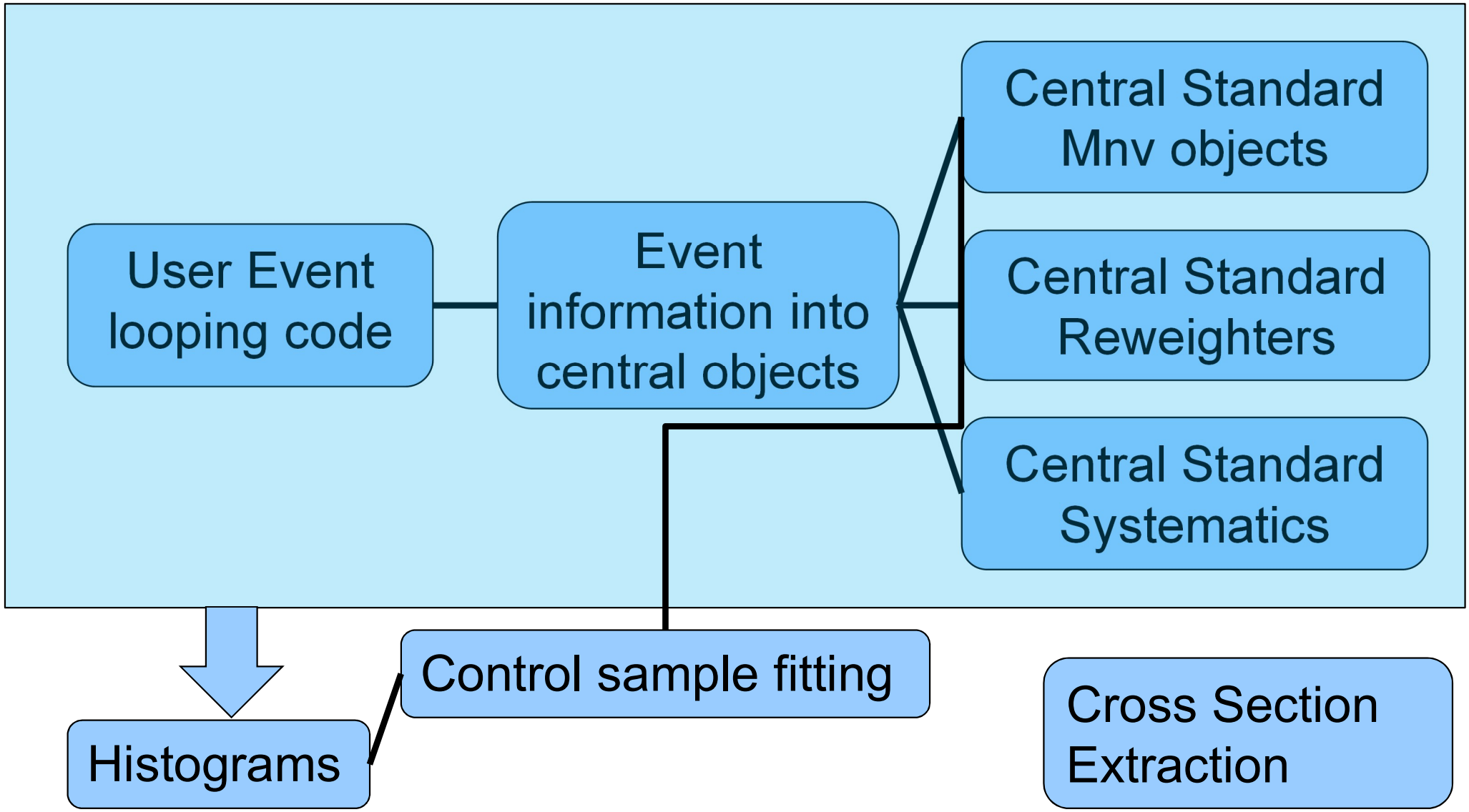
MINERvA's Extraction Method

- MINERvA only does unfolded cross sections

Migration matrices
Event selection
Control sample selection
Acceptance tables

$$\left(\frac{d\sigma}{dx}\right)_\alpha = \frac{\sum_j U_{j\alpha}\left(N_{data,j} - N^{bkgd}_{data,j}\right)}{A_\alpha(\Phi T)(\Delta x)}$$

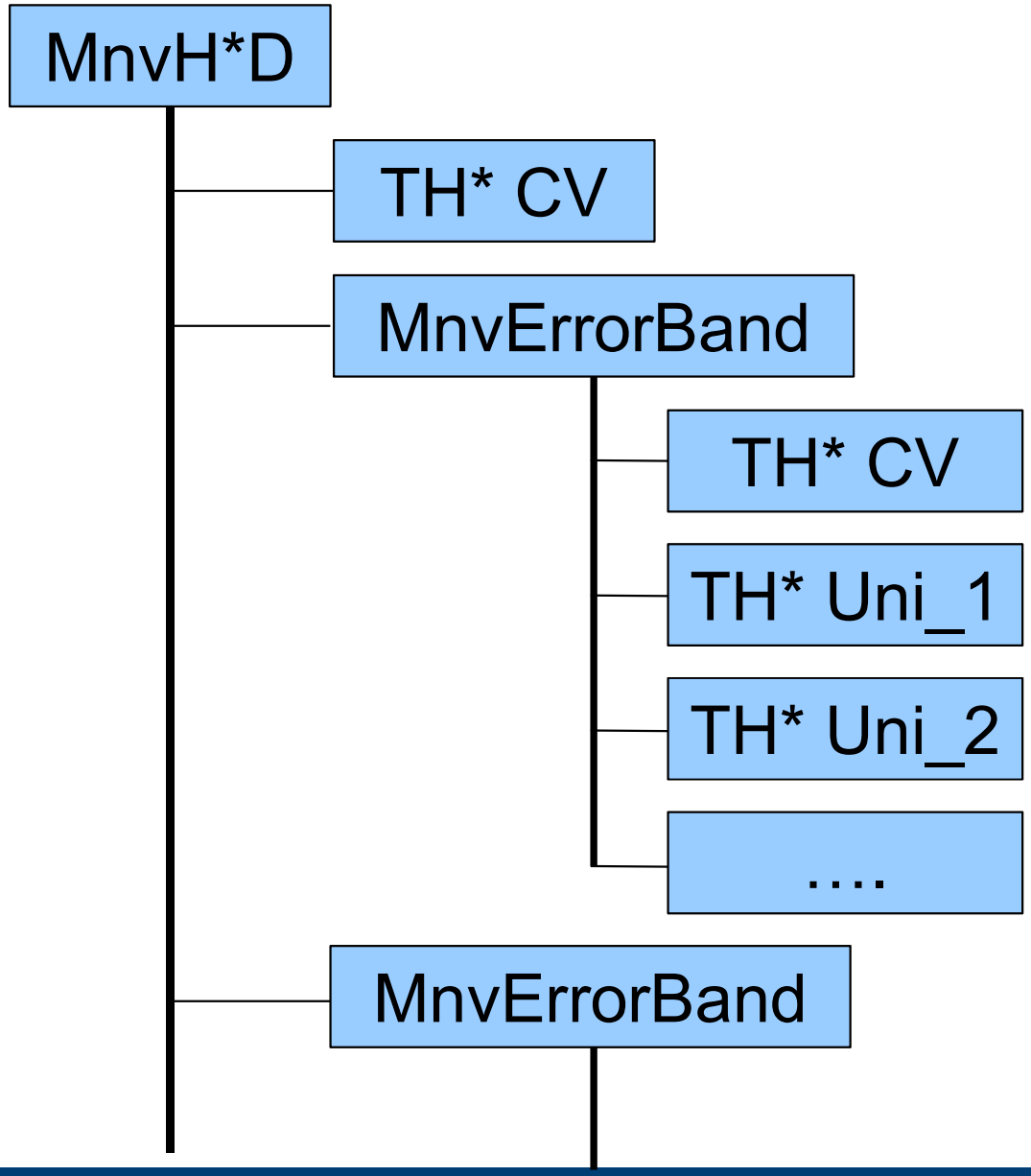
Rough diagram code



Some standard classes

- **Reweighter classes** – particle response, process specific reweighters (RPA, 2p2h, QE etc.)
- **FluxReweighter** – Modifies central value of MC, stores all flux universe weights, applies flux constraints
- **MnvH*D** – MINERvA created classes which inherit from ROOT TH*D classes
- **MnvVertErrorband** – MINERvA uncertainty class
- **MnvLatErrorband** – MINERvA uncertainty class
- **MnvPlotter** – Plotting class with common utilities

What is an MnvH*D ?



Portable results

- A single MnvH*D contains all the information necessary to produce a measurement with statistical and systematic uncertainty break downs
- Everything is stored in:
  - Universes
  - External covariance are possible

Powerful bookkeeping

- Add, Multiply, Divide, Scale operations are applied to the CV and all universes
- Covariance, correlation matrices can be computed on the fly
  - Options for different ways of calculating uncertainty

Things I'd change

- Each errorband shouldn't need to own its own CV histogram
  - If a user fills these incorrectly the uncertainty (fractional) will be wrong
- There are other uncertainty metrics which could be useful
  - Average difference
  - RMS
  - Max difference
  - Etc.

Technical Considerations

- The ND will produce incredibly detailed results
  - Large exposures
  - Better resolutions
  - End result is **more** analysis bins
- Object size limits in ROOT
- Memory allocation
- 1D results will almost never be an issue
- 2D results push limits
- ND break things

Technical Considerations

- Consider fields which are not necessary in systematics
  - Sumw2 values are NOT typically necessary in a systematic universe
  - Are there others?
- With big objects having tools to divide up the object into smaller pieces and reassemble them as necessary is useful
  - Probably more clever, technically correct ways to handle this…

Technical Considerations

- External packages (e.g. RooUnfold) need an interface between Mnv type objects and the standard TH objects they expect.

Other Tools

- Standardized ways of handling the volume of input files
  - Event loops slow down with the number of files -> Merge files.
  - Computing failures (transient) can cause nasty issues
- Histogram validation tools
- Macro level standard geometry tools
- Macro level number of scattering center calculators

Sidenote

- Unfolding isn't the only cross section extract method – lots of discussion of the pros/cons over a variety of methods
  - Forward folding (https://remu.readthedocs.io/en/latest/introduction/README.html)
  - Likelihood fitting
  - Unregularized unfolding