Lumi Session

HELMHOLTZ ASSOCIATION
Helmholtz Institute Mainz

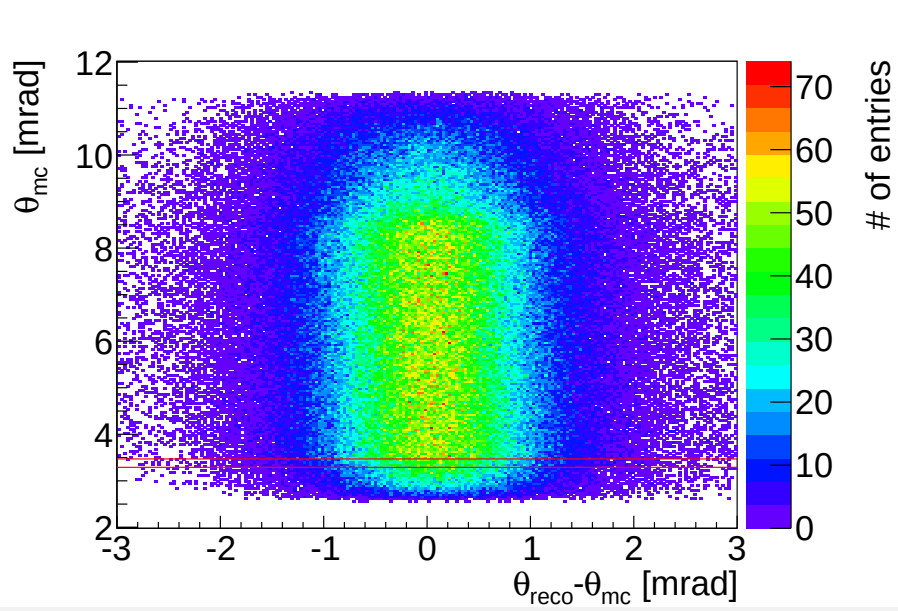
GSI

Luminosity Fit Status

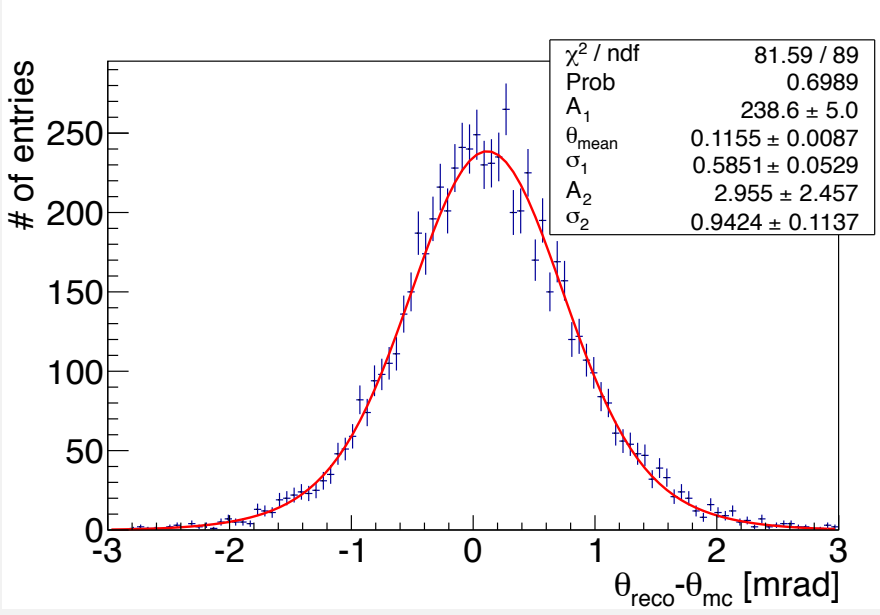
S. Pflüger

Resolution: Reminder

Θ Resolution vs Θ



Θ Resolution Slice



→ offset in resolution ($\sim +0.1$mrad)

$$A_1 e^{-\frac{1}{2}\left(\frac{\Delta\theta-\theta_{\text{mean}}}{\sigma_1}\right)^2} + A_2 e^{-\frac{1}{2}\left(\frac{\Delta\theta-\theta_{\text{mean}}}{\sigma_2}\right)^2}$$

Resolution: Current Status

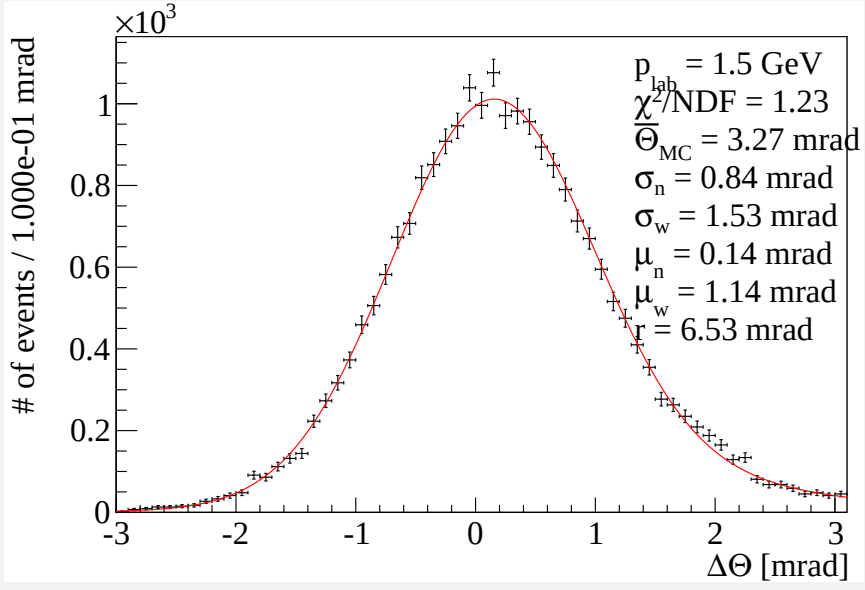
- switched to Pixel Sensors (including beampipe etc)!
- Modeling Framework used for resolution determination
- some work on backtracking software

Resolution: Current Status

- switched to Pixel Sensors (including beampipe etc)!
- Modeling Framework used for resolution determination
- some work on backtracking software

- offset remains ($\sim +0.15$mrad)
- additional asymmetry
  $\rightarrow$ absorbed by wide part

Θ Resolution Slice



$$\frac{r}{1+r}\frac{1}{\sigma_n\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{\Delta\theta-\mu_n}{\sigma_n}\right)^2} + \frac{1}{1+r}\frac{1}{\sigma_w\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{\Delta\theta-\mu_w}{\sigma_w}\right)^2}$$

Model Framework: Reminder

Requirements

construction of "complex" luminosity fit model

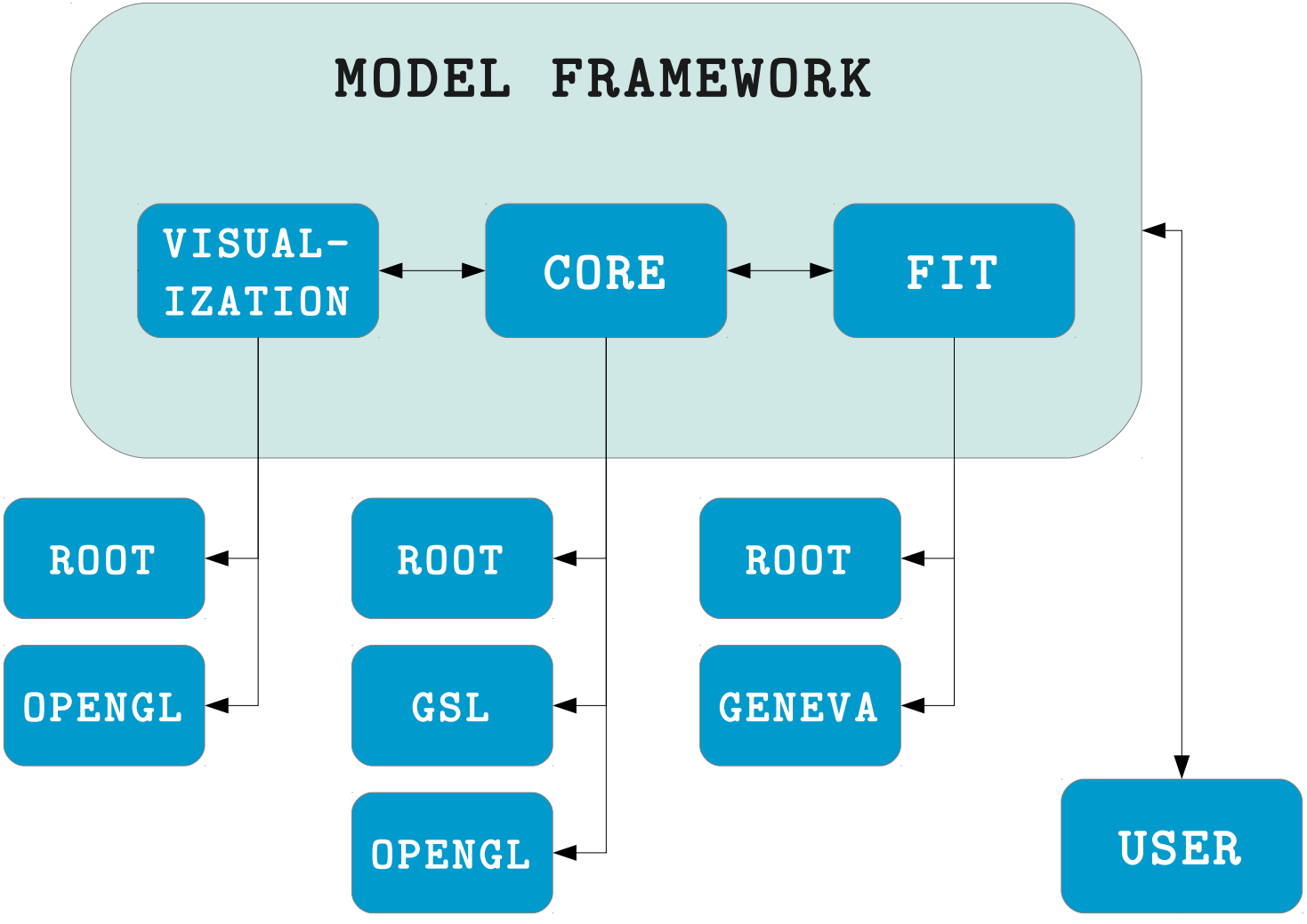
$$\frac{dN(\theta)}{d\theta} = L \cdot \left[\frac{d\sigma(\theta)}{d\theta} \cdot \epsilon(\theta) \right] \otimes Res(\theta)$$

Possible Solutions

- ROOT (TF1): tedious and not elegant
- RooFit: unable to construct lumi fit model
- others: non existing

$\rightarrow$ new Modeling Framework!

Concept


MODEL FRAMEWORK
VISUAL-IZATION
CORE
FIT
ROOT
OPENGL
ROOT
GSL
OPENGL
ROOT
GENEVA
USER

LMD Example: Walkthrough

Listing 1 : Constructing the Lumi Models (PndLmdModelFactory)

```
1  shared_ptr<Model1D> current_model;
2  if (fit_options->isFitRaw()) {
3    current_model.reset(new PndLmdDPMMTModel1D("dpm_mt_1d"));
4    // set free parameters
5    ...
6  } else {
7    current_model.reset(new PndLmdDPMAngModel1D("dpm_angular_1d"));
8    // finally set all parameters free according to the fit options
9    // set free parameters
10   ...
11 }
12 // add parametrization
13 shared_ptr<Parametrization> dpm_parametrization(new PndLmdDPMModelParametrization(
       current_model->getModelParameterSet()));
14 current_model->getModelParameterHandler().registerParametrizations(current_model->
       getModelParameterSet(), dpm_parametrization);
```

LMD Example: Walkthrough

Listing 2 : Constructing the Lumi Models (PndLmdModelFactory)

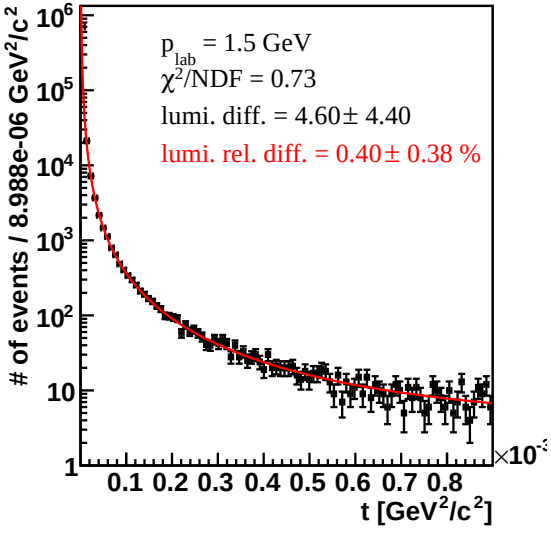
```
1  if (fit_options->isAcceptanceCorrOn()) { // with acceptance corr
2    // translate acceptance interpolation option
3    PndLmdROOTDataModel1D::interpolation_type intpol_type = PndLmdROOTDataModel1D::LINEAR;
4    if (fit_options->getAcceptanceInterpolationType() == 0) {
5      intpol_type = PndLmdROOTDataModel1D::CONSTANT;
6    }
7    // other interpolations
8    ....
9    shared_ptr<Model1D> acc(new PndLmdROOTDataModel1D("acceptance_1d", acceptance->
         getAcceptance1D(fit_options->isFitRaw()), intpol_type));
10   current_model.reset(new ProductModel1D("acceptance_corrected_1d", current_model, acc));
11 }
12 if (fit_options->isSmearingOn()) { // with resolution smearing
13   // ok since we have smearing on, generate smearing model
14   current_model.reset(new NumericConvolutionModel1D("smeared_acceptance_corrected_1d",
         current_model, generate1DResolutionModel(fit_options)));
15 }
```

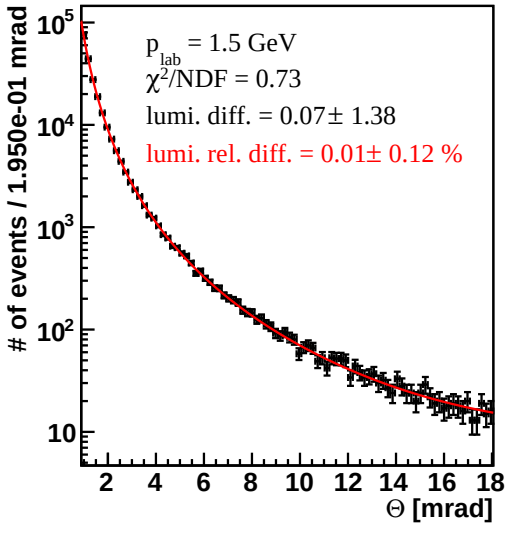
LMD Example: Walkthrough

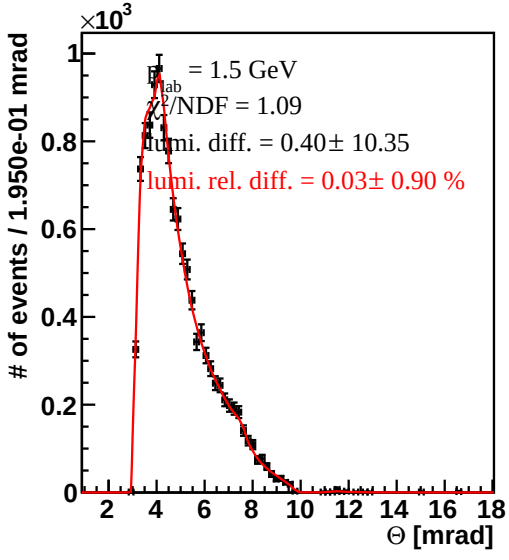
Listing 3 : Fitting Data with Lumi Models (PndLmdData)

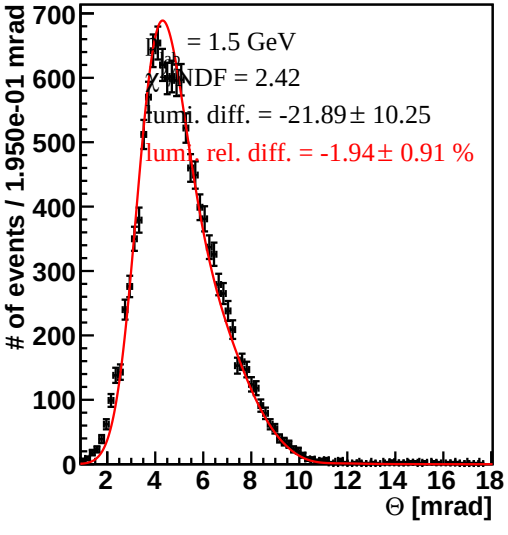
```
1  // create a new model via the factory
2  shared_ptr<Model1D> model1d = signal_model_fac.generate1DModel(fit_options, getLabMomentum()
       , lmd_acc);
3  // create chi2 estimator
4  Chi2Estimator chi2_est;
5  // set model
6  chi2_est.setModel(model1d);
7  // create and set data
8  ROOTDataHelper data_helper;
9  chi2_est.setData(data_helper.createBinnedData(getMeasuredHist1D(fit_options)));
10 // create estimator options
11 EstimatorOptions est_opt;
12 std::pair<double, double> fit_range = ...
13 est_opt.setFitRangeX(fit_range);
14 est_opt.setWithIntegralScaling(true);
15 // apply estimator options
16 chi2_est.applyEstimatorOptions(est_opt);
17 // now set better starting lumi value
18 ...
19 // create minimizer instance with control parameter
20 ROOTMinimizer fitter(chi2_est);
21 // do actual minimization
22 int fit_status = fitter.doMinimization();
```

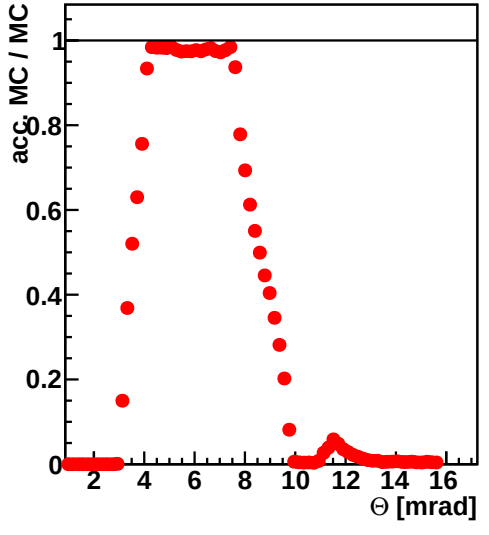
LMD Example: Fit Results

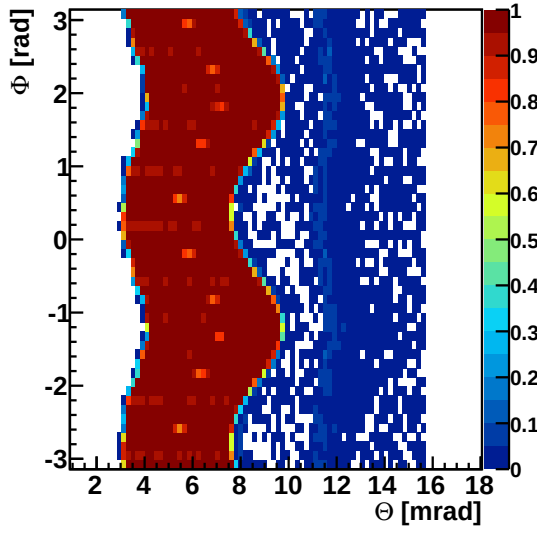












Conclusions

Resolution Parametrization

- Resolution remains "complex" $\rightarrow$ investigate
- but: work on automated parametrization determination almost finished!

Modeling

- core development finished
- well working and tested for complex 1D models
- try it out (currently in PandaRoot under lmd/LmdFit)
- however: still under development (report bugs please)
- important: give feedback! (email: s.pflueger(at)gsi.de)

End

Thanks for Your Attention!

Theta Resolution