

Feedback about Math and Stats

Yuka, Vasil, Sergey, Danilo, Jakob

ROOT

Data Analysis Framework

<https://root.cern>



- ▶ Math & Fits
- ▶ Histograms
- ▶ RooFit, RooStats
- ▶ TMVA

- Messages distilled from slides and interventions from the audience
- We also add our take on them



- ▶ Math is an area where we have unique functionality
 - Minimisation and HENP specific components (TLorentzVector)
 - A lot of perceived (?) competition: xtensor, numpy
- ▶ The quality of the code needs to be high
 - Smaller, atomic and revertible (no offense) PRs
 - Test coverage



- ▶ Little feedback: is everybody satisfied or a few do care?
- ▶ Can we preserve some “provenance” after a training?
- ▶ The interface seems anachronistic sometimes
 - Option strings
- ▶ Test coverage must be guaranteed
 - Avoid bugs like the result change in the MT case
- ▶ We think RDataFrame is appreciated and it was presented how to exploit it in TMVA
 - **Go ahead**, demonstrate the value of the plan
 - **Careful**: it looks good now, if we don't want it anymore, it will be hard to remove



- ▶ Too many matrix classes
 - A good candidate for the “deprecation” strategy?
- ▶ Big interest in the fitting: something unique
 - Clarify long term strategy and interplay with RooFit
 - Did we neglect it a bit wrt RooFit?
 - Can CLAD increase its value and make it more unique?
- ▶ We are not sure about R: not mentioned here
 - What is the strategy?



- ▶ Componentization of Math?
- ▶ We are unsure about the support on Windows
 - Mathmore? Be clear about what ROOT can and cannot: it builds confidence and the boundary can be shifter
- ▶ Expressed the wish to use Math and RooFit standalone
 - These are not really decoupled from ROOT
 - How can we respond to it?



- ▶ Componentization of Math?
- ▶ We are unsure about the support on Windows
 - Mathmore? Be clear about what ROOT can and cannot: it builds confidence and the boundary can be shifter
- ▶ Expressed the wish to use Math and RooFit standalone
 - These are not really decoupled from ROOT
 - How can we respond to it?
- ▶ QuantStack expressed a strong interest in helping (also) with RVec ad RTensor
 - Can we get more out of this than common patterns?



RooFit, RooStats and Minuit

- ▶ Total absence of feedback about RooStats!
- ▶ How to react to the perception of old/outdated documentation?
 - More text? More tutorials? More trainings?
- ▶ Separation from ROOT
 - How to react? Can we provide options to build “a la carte” waiting for *the ROOT package manager*
- ▶ TRooFit might lower the hurdle to use RooFit effectively
 - Is this a very well scoped effort? If yes, what can we learn from it?



- ▶ DQM: important customers
 - Can we help providing best practices to implement large histograms sets?
 - E.g. do we need here some extra convenient container?
- ▶ TH1::SetDirectory(nullptr)
 - put on title page of <https://root.cern/>
 - Overcome present automatic registration?



- ▶ The “v7” hat is being liked!
- ▶ Balance of old code maintenance VS new developments is critical
- ▶ New histograms require a lot: I/O, drawing, fitting
- ▶ Strategy of introducing new functionality:
 - All in one: require abruptly the new ones, burn bridges
 - Step by step (old and new histos coexisting in the same release?)
 - Communicate a clear timescale
- ▶ Does we support external tools?
 - boost flavor of c++11 histograms: identify our added value clearly



Interplay with PyROOT

- ▶ Python interface needs attention
 - Users do care about Python
- ▶ Focus on the programming model
 - Popular language
- ▶ PyRoofit, iMinuit
 - An opportunity for the new PyROOT?
- ▶ Performance was not stressed as so critical
 - We know it is, on the other hand
- ▶ **Risk:** investing in PyROOT without focussing on the underlying interfaces