



# News

*Danilo Piparo (CERN, EP-SFT)*

*02-09-2024*



# ROOT's Strategic Goals

**Provide a unified software package for the storage, processing, visualisation and analysis of scientific data that is reliable, performant and supported, that is easy to use and obtain, and that minimises scientists' time needed to achieve scientific results.**

Update coming from feedback to the Vision for ROOT 7 Document and last meeting.

**The success of experiments and all ROOT users at large is our priority**



# Releases



- ▶ Short cycle: active until 6.36.00 is out
  - Date for 6.36.00 depends on the LHC Schedule and Experiments plans
- ▶ Date: 4-6 November 2024
- ▶ RC-1: 21-25 October
- ▶ Create v6-34-00-patches: 9-13 September
- ▶ Deadline for changes: 30 August ← missed due to LLVM issue
- ▶ We want LLVM18 in this release
  - Delay by 1/2 weeks the deadline for changes and v6-34-00-patches creation



# LLVM18 Current Status

Upgrade within reach,  
but we need a last effort.

- ▶ We have green builds [#15696](#)
  - Win, Mac, Linux: all flavours – heroic effort of Dev+Jonas last week!
- ▶ CMS kindly agreed to test master+llvm18 on CMSSW
  - Everything works on x86
  - **Almost nothing works on ARM64 (see status report doc)**



Ongoing Discussions that  
need to Converge



# Priority Items from CMS

From the AF Meeting

## High priority

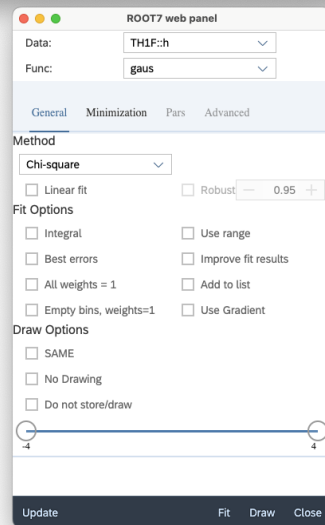
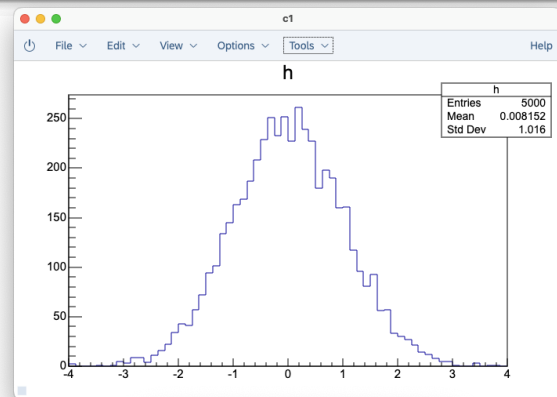
- [#7955](#) `GetMethodWithPrototype` return the incorrect function
- Schema evolution `IORule` not working for `std::auto_ptr<T> -> std::unique_ptr<T>` for a split TTree ([cms-sw/cmssw#43923](#)). This is necessary for the planned re-MiniAOD processing of Run 2 UL MC on `CMSSW_15_0_X`.
- Help in figuring out the cause and workarounds for “bytecount too large” errors seen in prompt reconstruction ([cms-sw/cmssw#45089](#), [cms-sw/cmssw#40132](#))
- Continue collaborating on RNTuple unsplit field behavior and API

## Medium priority

- Mechanism to read files with `std::auto_ptr<T>` without relying on existence on `std::auto_ptr<T>` ([cms-sw/cmssw#43422](#))
- Fix for string cut parser sometimes not seeing all methods of an object when using ROOT reflection ([cms-sw/cmssw#33084](#))
- RNTuple: Pattern for storing Structure-of-Array data structures
- RNTuple: schema evolution support
- Following two have workarounds (both related to RNTuple-inspired data format evolution)
- Schema evolution problem with a recursive class when reading a split TTree ([cms-sw/cmssw#43768](#))
- Schema evolution problem when removing an intermediate base class ([cms-sw/cmssw#43516](#)) priority



- ▶ This technology greatly improved in the last year
  - Security, functionality, test coverage...
- ▶ Default in master (Browser, FitPanel, plots)
- ▶ **Are we ready to enable it by default for the November release?**
  - To what degree? Browser only? Notebooks only? Everywhere?
  - If we cannot decide now, what is the plan to arrive to an informed decision?
- ▶ Set up ad-hoc meeting: **Wednesday 11 September 16:00, room 32/1-A01.**
  - TBC with Yasemin, it could be 15:30-16:30.
  - Follow up the week after with Serguei, who will be here at CERN







# ROOT 2025 PoW Discussion

- ▶ [Indico](#): Wed 25-9, 15:00-18:00,
  - 2023 edition [here](#): have a look to be inspired

Format:

- ▶ 6 talks: I/O, Graphics, Infrastructure, Statistics/ML, Analysis, Interpreters
- ▶ Talks *not* given by the person mostly focussing on the topic, but by other colleagues (with a bias towards early career)
- ▶ Requires discussions with experts to prepare the talk, e.g. Analysis: talk to Marta and Vincenzo, Statistics talk to Jonas R, ML Lorenzo, Interpreter C++ Vassil, JonasH, Dev...
- ▶ Enforces clarity in the explanations and listing of points



# ROOT 2025 PoW Discussion

- ▶ Final deliverable: a table with items, with priority 1 or 2 (see [pow sheet](#)), and, if available, stretch-goals, i.e. "good-to-have"s for students and other contributors
  - Make sure the proposed PoW points stick out and can be easily summarised
- ▶ The PoW will be further refined until November, and exposed to experiments, users and SFT – there will be time to perfect it.

	5	Address residual scaling issues with MT writing	2	0.5	10 %
<b>RNTuple</b>	1	Complete implementation of merging	1	0.5	
	2	Complete implementation of datasets chains	1	0.5	
	3	Limit testing in collaboration with CERN IT	1	0.5	
	4	Follow-up on API review by HEP-CCE	1	0.5	
	5	Implement unsplit ("blobbified") encoding	1	1	
	6	Support for unaligned friends and joins	1	0	
	7	(E) RNTuple: schema evolution	1	0	
	8	Further develop support for lossy compression with low-precision floats	2	0	
	9	Design compression dictionaries and understand implications for the spe	2	0.5	
	10	First implementation of highly-scalable parallel writing	2	1	
	11	Organise a Design Workshop to discuss intra-link events, metadata, nati	2	0	40.9 %



## Proposed speakers

- ▶ I/O: Marta
- ▶ Graphics: Stephan
- ▶ Infrastructure: Florine
- ▶ Statistics/ML: Vincenzo
- ▶ Analysis: Monica
- ▶ Interpreters: Giacomo



- ▶ Email sent on Friday about the [Vision for ROOT 7 document](#)
- ▶ A base for the plan that will lead us to the ROOT 7.00.00 in 2027
  - 2y before the Run 4 data taking according to the current LHC schedule
- ▶ **Your feedback is being integrated: thanks for it.**
- ▶ Next step: expose the document to a selected number of representatives of our users (experiments, "Analysis Tools" groups, ...)

ROOT Internal Draft - Please do not disseminate

## Vision for ROOT 7

v0.2 23-8-24

At the start of LHC Run 2, ten years ago, ROOT 6 represented a major modernisation of ROOT. The most visible element of this modernisation was Cling, the new LLVM-based C++ interpreter, a software product still unique in its kind. Since then, innovation in ROOT continued year after year, with components like RDataFrame and RNTuple, or with the major renovation of RooFit and PyROOT. ROOT was also adapted to multicore CPUs, allowing users to express parallelism through a highly efficient multithreading paradigm and by making ROOT components thread friendly, to use ROOT as a library in highly parallelised environments.

The end of Run 3 and the start of Long Shutdown 3 represent an opportunity to formalise the aforementioned incremental evolution into a new release cycle, ROOT's seventh. With ROOT 7, we want first of all to further improve the ROOT user experience, for the use cases of data analysis as well as frameworks and data processing. In addition, we aim to make ROOT an even more sustainable software product, to guarantee seamless support of the HEP community, and scientific community at large, in the years to come. Finally, the freedom offered by a new release cycle can be used to further innovate our software solution. While ROOT's most-trusted features will continue to function as before, we plan to introduce change also through selected backward-incompatible upgrades, but in an evolutionary manner, communicating clearly with users and developers.†



# ROOT Components Table

Release 6.34.00							
Component	Under Review	Modularity	Legacy	Deprecated	Removed	Maintainer	Notes
THtml	<input type="checkbox"/>	Yes, Off	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	To be removed in 6.36
PROOF	<input type="checkbox"/>	Yes, Off	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Superseded by RDF
CMake opt <i>cxxmodules</i>	<input type="checkbox"/>	Yes, Off	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	PR <a href="#">#16344</a> removed it.
TPython	<input type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	PR <a href="#">#16337</a>
Splashscreen	<input checked="" type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	PR <a href="#">#15056</a> , comment proposes to make it modular and off by default.
TSpectrum	<input checked="" type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	PR <a href="#">#16351</a> turns it off by default: should this become a posterchild for Legacy?
Geometry	<input checked="" type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Can be made modular (here an <a href="#">abandoned attempt from 2017</a> )
TH1K	<input checked="" type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
R Bindings	<input checked="" type="checkbox"/>	Yes, Off	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
TThread	<input checked="" type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A candidate for the ROOT::Legacy namespace?
TRolke	<input checked="" type="checkbox"/>	No	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A candidate for the ROOT::Legacy namespace?
Legend							
<b>Component</b>	the name of the ROOT component being described in the row						
<b>Under Review</b>	the future of the component is being reviewed, e.g. whether it should be made modular, deprecated or removed.						
<b>Modular No/On/Off</b>	the component not modular (No), or modular and by default it is built as part of ROOT (On) or not (Off)						
<b>Legacy</b>	the component will continue to work, bug fixes are decided on a case by case basis, no new features						
<b>Deprecated</b>	the component will be removed in a future release						
<b>Removed</b>	A reminder for the components that have been removed for this release						
<b>Maintainer</b>	Whether the component has a maintainer or not						

Find it at <https://cern.ch/root-components>



# Important Forthcoming Events

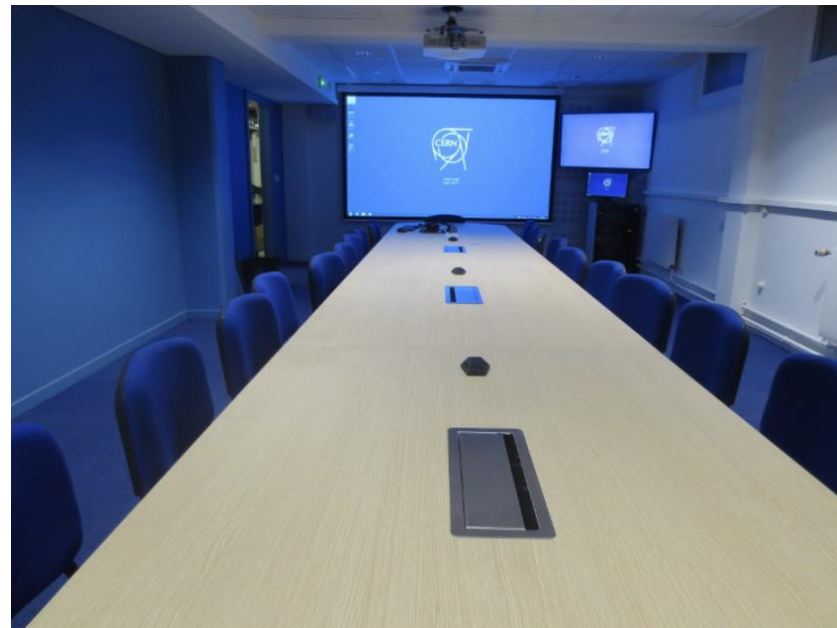


# Q3 Report with Experiments and Funders

- ▶ 24 September 2024 16:00-17:00 4/S-030
- ▶ Usual (by now) report about our progress
- ▶ Discussion about releases in 2025
- ▶ Any other item to be discussed with our stakeholders?



- ▶ CHEP: 19-25 October
- ▶ Dates:
  - Rehearsals between 7-11 October
  - 7-8-9 Oct: Mon, Tue, Wed
  - 15:00-17:00
  - Room 4/S-030







# PoW 2025 Jamboree and SFT Retreat

- ▶ SFT Retreat: November 1<sup>st</sup>
- ▶ Input for ROOT's talk needs to be collected *before CHEP*
  - The talk will have to incorporate important trends/information gathered at the conference, if any
- ▶ Two options
  - **A:** Collection of input for SFT at large + PoW Jamboree at the end of September: the talk will be forward looking, a preliminary PoW can be presented → **Now fixed for September 25th**
  - ~~**B:** Collection of input for SFT at large at the end of September, PoW Jamboree in November: talk focus on 2024 achievements, only general directions about 2025, need a Jamboree in November anyway.~~



- ▶ Mon 25, Tue 26 November Idea2
  - Will ask for 27 too, currently conflicting with another event.
- ▶ Same format as February
  - Lunches by us, social dinner on Mon (Tue if 3 days?)
- ▶ We need a concept to announce the event. A proposed narrowed set of topics:
  - PyROOT, Analysis and Documentation
  - Ask participants to make proposals: if not an abstract, to select where they would like to contribute and whether they want to specify a topic