The ROOT Project: Status and Future Directions

Danilo Piparo for the ROOT team (CERN, EP-SFT)

14-05-2024

A new ROOT Project Leader

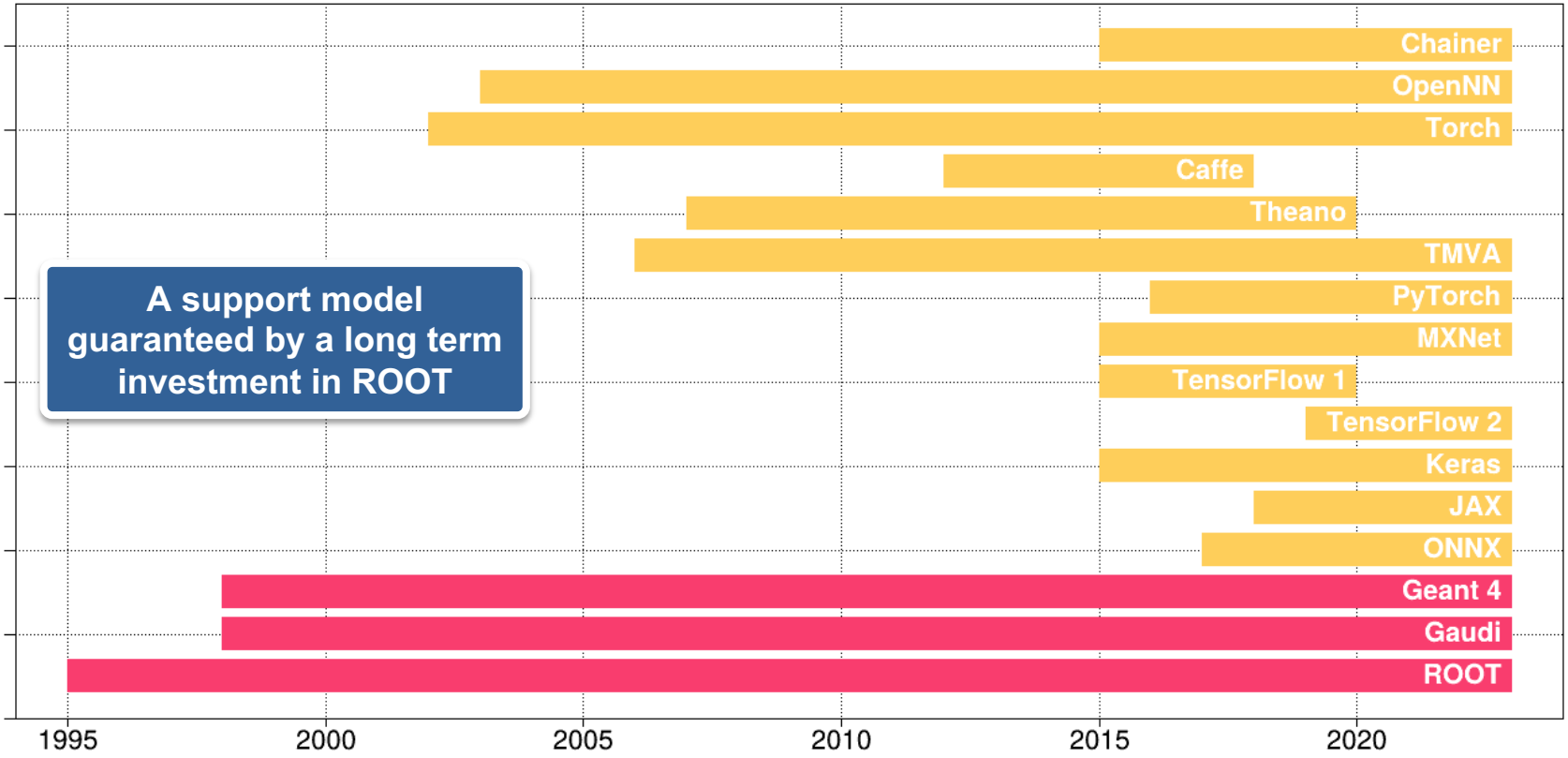
- Danilo Piparo, starting from January 1st 2024
- After 6 years as ROOT Project Leader, Axel Naumann moved to a coordination role in the Next Generation Triggers project at CERN
- A change supported by the ROOT Team, Axel and Danilo
- It was a pleasure and honour to work with Axel. We'll miss his competence, kindness and creativity!

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 the computing resources needed to achieve scientific results.

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

HEP Software Support Timeline



Plot inspired by M. Mazurek

Who is ROOT?

- **ROOT is its user community, contributors, and developers**
- ROOT is an international collaboration, where a sizeable effort is provided by CERN
  - Institutional responsibilities, but also precious contributions coming from the user community!

Team from https://root.cern , PhD students onwards

4 new team members starting between May and August

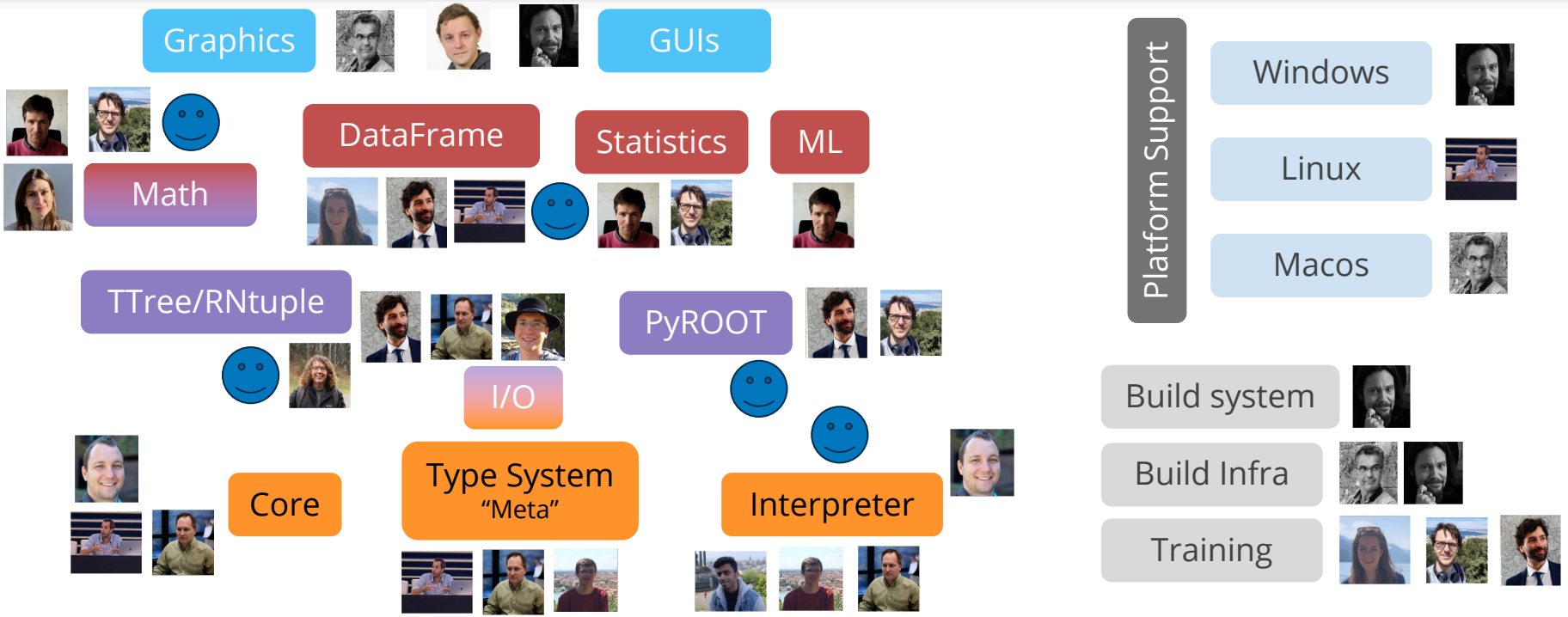
From left to right, starting from the first row (the affiliation is CERN if not specified):

1. Devajit Valaparambil, Bertrand Bellenot, Danilo Piparo, Florine de Geus, Jakob Blomer
2. Jonas Hanfeld, Jonas Rembser, Marta Czurylo, Olivier Couet, Philippe Canal (FNAL),
3. Vasil Vassilev (Princeton), Lorenzo Moneta, Monica Dessole, Vincenzo Padulano, Serguei Linev (GSI)
4. **New:** Jack Parolini, QUEST-35, QUEST-37, LD-55 (We are hiring! See **this job AD**)

Plus students, working with us a few months, up to ~1 year

Not everybody is 100% dedicated to one project, but most are.

The Team and the Focus Areas



Team members are encouraged to be involved in more than one ROOT component, as well as to take part to baseline work.

The structure of the project, components and people's focus has been greatly simplified for this slide: to be taken with a grain of salt!

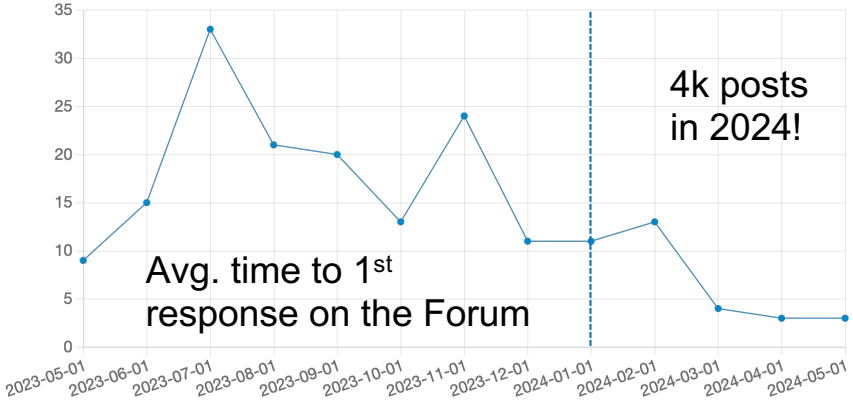
ROOT Community and Development

ROOT is an open source project, supported by a lively community. In 2024:

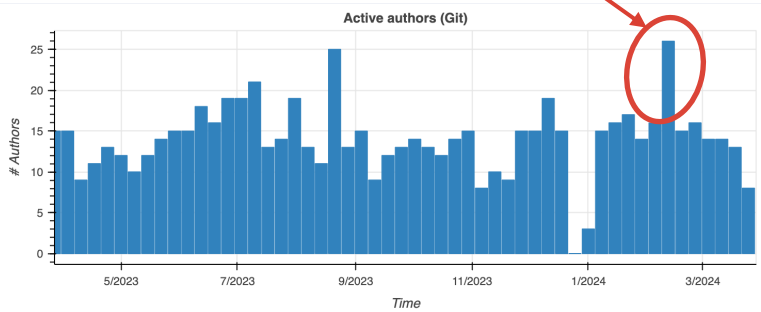
- ROOT Forum (main support channel): **8h avg. to obtain a first response**
- **About 15 active developers/contributors at any point in time**
- **894/897 PR opened/closed** (implies code review!), 1 day median to close a PR
- Solid work done outside of the ROOT team: packaging of ROOT for major linux distros (EL, Arch, Gentoo ...)

ROOT: not only developed by its team, but supported by substantial contributions from the community!

1st Discourse forum at CERN for number of posts (11.4k in 2023)



1st ROOT Hackathon (see next slide)



1st ROOT Hackathon: The Fixathon

- Took place at CERN, Idea2 on February 13-14 (link)
- ~30 people, from early career physicists to senior scientists,
- Onboard early career colleagues: teach how to contribute to a large open-source scientific software
- **A success: main objectives, all achieved**
  - Close as many ROOT issues as possible
  - 78 issues closed (for comparison, 441 were closed in total during 2023), 8 solutions proposed, work started on another 19
  - Connect further with ROOT community, inclusively

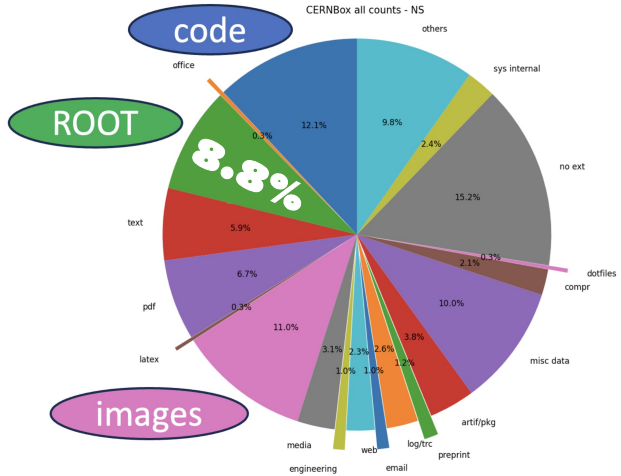
2nd ROOT Hackathon 25-26 November 2024 at Idea2: save the date! More details will be circulated in the next few months.

1st ROOT Hackaton: a productive experience, high return of results for effort invested

Data Written in ROOT Format

QoS	ALICE	ATLAS	CMS	LHCb	*Total*
Disk [PB]	199	406	304	93	1002
Tape [PB]	283	666	673	250	1875

`24 Pledges: source CRIC



Courtesy of G. Lo Presti and D. Castro (CERN IT)

File types in CERNBox: distribution by number of files. If sizes are considered, ROOT files are 40% of 11 PB

- Disk + tape pledges at T{0,1s,2s}: ~2.9 EB
- Modulo notable exceptions (e.g. ATLAS and LHCb raw), **that space is (going to be) used for data in ROOT format, mainly columnar (TTree): ROOT DOES SCALE.**
- **Storage pledged at Grid sites is not the only one used for ROOT files**: e.g.T3s, university clusters, personal laptops, analysis facilities, cloud…
- RNTuple: new technology being developed to store HL-LHC data – learned from the TTree experience

Exabytes of (HEP) data in ROOT format. Reading them back will be supported in the very long term

Solid Planning, formed inclusively

Link to the PoW

- November 2023: Program of Work (PoW) Jamboree
  - Presentations about plans of different ROOT focus areas, prepared and delivered by team members *not* directly responsible for them: incentive to motivate and pre-discuss future directions.
- 15th January: SFT Special Group Meeting – PoWs 2024
- 23rd January: ROOT Funders Meeting – PoW 2024
- 24th January: ROOT-Experiments Meeting – PoW 2024
- 30th January: Meeting with CMS Common Analysis Tools group (CAT)
  - New: get feedback from teams outside of the traditional area of central Offline and Computing coordination areas. Repeated the success with the CMS Trigger Study Group area, as well as Common Analysis Tools group.
  - 1st Feb: AF Feedback on SFT PoW

Very useful exercise: at each of these steps, the trajectory was refined and the PoW improved

- The items in the PoW are now 69 (they were 61 at the beginning)

Measuring Progress: PoW Completion

- Feedback from ROOT-Experiments meeting: "*it would be useful to provide feedback about the PoW completion every quarter" (*See first Q1 report here*)*
- **Needed to find a good way to address that demand, while keeping Project Management formalism as lightweight as possible. Our solution:**
  - **2 priority levels: 1 and 2** (+ "stretch goals": the items we will address if an injection of effort, e.g. short term students, takes place)
  - **Assign to every item in the PoW a score: 0 not done, 0.5 partially done, 1 done**. The amount of work needed to complete the items in the PoW varies. We assume, for simplicity, that, overall, it averages out.
  - **Introduce the concept of Extra Items**, i.e. the items not foreseen in the first version of the PoW
- Remain open to changes in the PoW and prioritise every new item related to 2024 LHC data taking

2024 PoW Completion

DONE
PARTIALLY DONE
NOT DONE

			Priority	Completion Status: 0, .5 or 1		
Builds and Binaries	1	pip install ROOT for some selected platforms	1	0.5		
	2	Complete transition to GH Actions, adding GPU runners	1	0.5		
	3	Reduce number of services hosted by root.cern with a combination of CERN IT central services	1	0		
	4	Win: Replace Debug builds with ReleaseWithDebInfo in the CI	1	0		
	5	Optimise dictionary dependencies to minimise build real time	2	0		
	6	Win: Add support for Ninja	2	0	16.7	%
I/O and TTree	1	Support std::variant, both in TTree and RNTuple (CMS)	2	0		
	2	Support writing objects larger than 1GB (TBufferFile > 1 GB, ALICE)	1	0		
	3	Complete schema evolution improvements	2	0		
	4	Ensure consistency of std::int types across ROOT I/O	2	0		
	5	Address residual scaling issues with MT writing	2	0.25	5.0	%
RNTuple	1	Support for chains of datasets and merging	1	0		
	2	Complete implementation of datasets chains and merging	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	1	0		
	7	Further develop support for lossy compression with low-precision floats	2	0		
	8	Design compression dictionaries and understand implications for the specification	2	0		
	9	First implementation of highly-scalable parallel writing	2	0.5		
	#	Organise a Design Workshop to discuss intra-link events, metadata, native SoA layout for events	2	0	30.0	%
RooFit	1	Workshop with Experiments: promote features, gather input, speedup integration of RooFit in the existing sw setups	1	1		
	2	Numeric integrals in n-dim with CUDA	1	0		
	3	Evaluation of custom user functions in CUDA	1	0		
	4	Group similar PDFs to speed up evaluation	1	0		
	5	Make the new vectorized CPU likelihood evaluation interface the default	1	1		
	6	Reduce JITting time for AD in RooFit	1	0		
	7	PyROOT: express RooStats configuration with C++-oriented Set* as kwargs	2	0		
	8	Integration of Fumili in RooFit	2	0	25.0	%
RDataFrame	1	Put existing bulk processing in prod	1	0		
	2	DistRDF: reduce memory usage on HTCondor Workers	1	0		
	3	DistRDF: improve user experience when integrated with notebooks and nb services like SWAN	1	0.25		
	4	Make the TTree → RNTuple transition transparent for analysers	1	1		
	5	Further Pythonise the interface	2	0		
	6	Deliver varied snapshots	2	0.25	25.0	%
Math	1	PyROOT: better histos ang graph interoperability with NumPy and UHI protocol	1	0		
	2	Histos: advance current RHist implementation to one testable by experiments	1	0		
	3	Add interface to pass initial error values/cov matrix to Minuit2	1	0		
	4	Release a library for Lorentz vector computations on accelerators in SYCL	1	0.5		
	5	Deliver plan and prototype of algorithmic improvements when dealing with param constraints in ROOT's minimisers	2	0		
	6	PyROOT: Pythonise TF{1,2,3} and numerical algorithms interfaces (e.g. minimisers)	2	0		
	7	Prototype SYCL kernels to be JITted (see Interpreters objectives)	2	0		
	8	Histograms: Model and prototype of pipelining GPU histogram filling	2	0.5	12.5	%
ML/AI	1	Put RBatchGenerator in production	1	0		
	2	Consolidate RBDT	1	1		
	3	Support of integration of SOFIE in experiments Fast Simulation pipelines	1	0		
	4	Add support in SOFIE for NVidia GPUs in CUDA	1	0		
	5	Continue to add support for the ONNX operators requested by experiments	1	0.25		
	6	Make HLS4ML interoperable with SOFIE	2	0		
	7	Streamline ROOT's inference interface, making it able to use models for Python ML frameworks (e.g. Keras/TF) directly	2	0	17.86	%
Visualisation and UI	1	Automated placement/tune of plot elements, "Auto Style"	1	0		
	2	Add missing features of classic graphics to the web-based one	1	0		
	3	Automate web-based graphics test suite	1	0		
	4	Add residual missing TEve features to REve, e.g. digit visualisation and text elements overlay	1	0.5		
	5	Visualization of flat ntuples using predefined visual summary data structures	1	0.5		
	6	Improve REve window manager and browser, polish render engine	2	0	16.7	%

Interpreters	1	Cling: identify potential Cling codebase reductions through the reuse of parts of clang-repl	1	0		
	2	Cling: cppyy rebase on top of cling/clang-repl	1	0		
	3	Migrate PyROOT to the latest Cppyy	1	1		
	4	Cling: Prototype SYCL support	2	0	25.0	%
Doc and education	1	Re-evaluate, update, and improve course material, making it more visible and better organised on the website	1	0		
	2	(Re-)evaluate tuts, eliminating what's outdated, what newer features would benefit from a (better) tutorial, improve visibility	1	0.5	25.0	%
Extra Items	1	Copyless reading in RNTuple - ALICE	1	1		
	2	Physics objects representations out of NanoAOD in RDataFrame - CMS CAT	1	0		
	3	Bulk Processing + GPU offloading for distRDF - CMS CAT	1	0		
	4	include the open source Tex Gyre Heros clone of Helvetica in root fonts - CMS CAT	2	1		
	5	Friend trees and RNtuples - CMS CAT	2	0		
	6	Support for joins in RNtuples - CMS CAT	2	0		
	7	Multithreading-friendly interfaces to the histogram types - CMS CAT/TSG	1	0		
	8	A library of matrix operations that can run on GPUs - CMS TSG	1	0	25.0	%
				overall:	20.2	%
				overall + Extra Items:	20.7	%

ROOT planning: solid, formed involving all stakeholders. Progress monitored continuously.

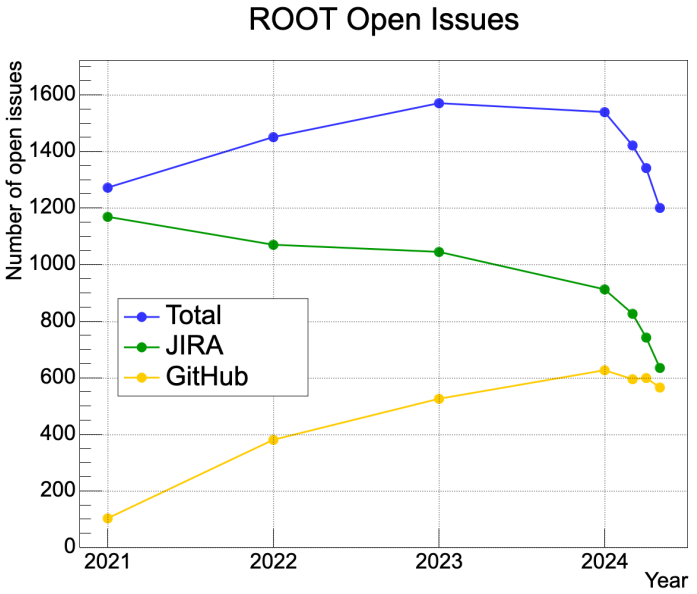
- **4 new arrivals foreseen during the next 3 months; final completion percentage will depend a lot on these new colleagues**
- External help, i.e. ROOT community (e.g. experiments, but not only) can make the difference, too

PoW is 20 % complete

Measuring Progress: Backlog

- Strong focus on reducing number of open issues
- Backlog reduction is implicitly *part of the PoW*
- **641 issues closed this year so far (441 during 2023)**
- **22% reduction of # open issues so far wrt Jan 1st**

	JIRA	GitHub	Total	Notes
Dec-20	1169	104	1273	
Dec-21	1071	380	1451	
Dec-22	1045	525	1570	
Dec-23	912	627	1539	
Feb-24	826	596	1422	54 JIRA issues migrated to GH
Mar-24	739	601	1340	10 JIRA issues migrated to GH
Apr-24	635	566	1201	



Reduction of existing backlog incorporated in the PoW

Communication with Users and Stakeholders

- **ROOT-Experiments meeting**: at least quarterly for discussing planning, such as releases, as well as PoW (Q1 meeting)
- **ROOT-Funders meeting**: ~twice per year (January meeting)
- **Experiments' core sw teams: fluid communication channels** (in person, email, mattermost, participation to experiments' meetings), ad-hoc meetings (e.g. about RNTuple integration or RooFit related topics)
- Involving collaborators as guests or recurrent participants in the ROOT Team, I/O, Parallelism Performance and Programming model (PPP) meetings
- GitHub: issues and Pull Requests
- ROOT Mattermost Team
- ROOT Forum

Clear communication lines, adapted to all categories of stakeholders

Releases and Some 6.32.00 Highlights

Release Schedule for 2024

2 releases: May (long term support) and November (short term support). Big change with respect to the past! May *was* the devel release, November the production one!

May W4 (4th week of May), long term support:

- Target integration in experiments' stacks for 2025 data taking
- v6.32.00-rc1 Release Candidate available since April for experiments and users to test

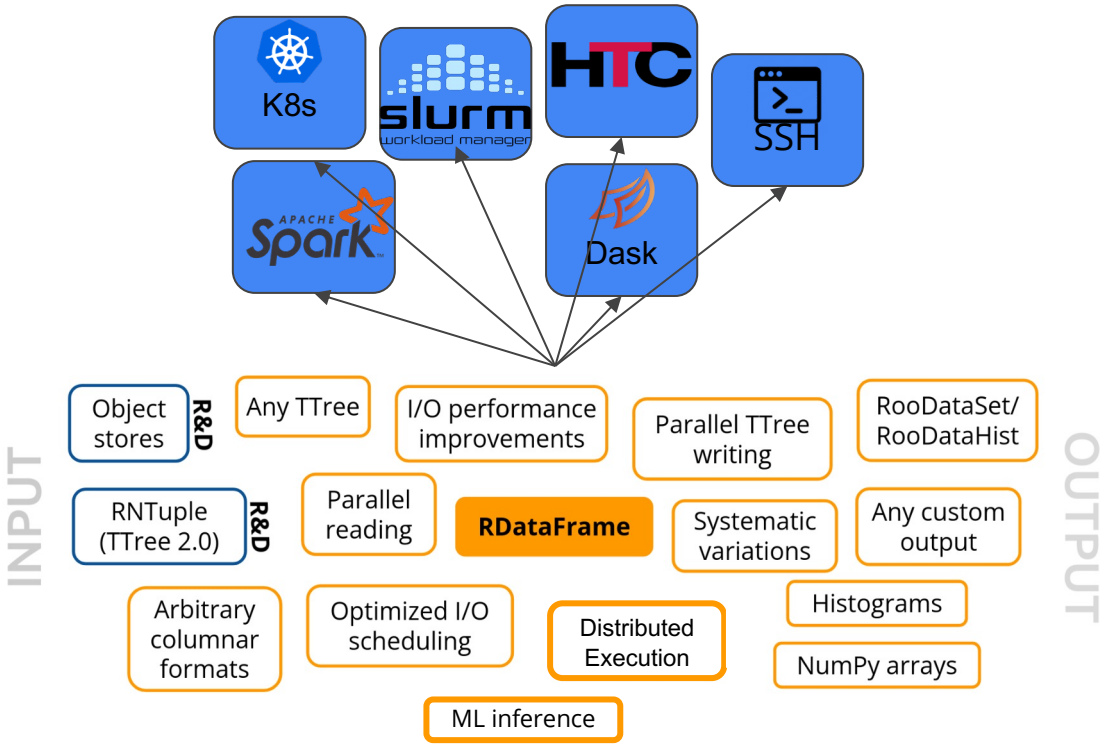
November W1, development release, short term support:

- Fit all the features we have at that time
- Release Candidate 2 weeks before: October W3
- The LLVM version is not yet decided, clear plan:
  - We will continue with the endeavour of upgrading ROOT to LLVM 18.
  - 2 decision points before October: May and late August.
  - We'll decide if it's worth continuing the effort or if we have hard bugs that prevent us from adopting the version. Three possible outcomes: we stay with llvm16, we move to llvm17, we adopt llvm18.

This schedule incorporates requests by experiments to prepare for 2025.

RDataFrame: high level API for data analysis

- Tailored to HENP data analysis requirements
- **Widely used and battle-tested**
  - Users in the wild
  - Basis for more specialised frameworks (e.g. bamboo, CROWN)
- Processing of columnar datasets, natively parallel
  - Not only TTree and RNTuple
  - Single node with multitheading
  - Seamless distribution to computing clusters via DistRDF



RDataFrame and Analysis Facilities

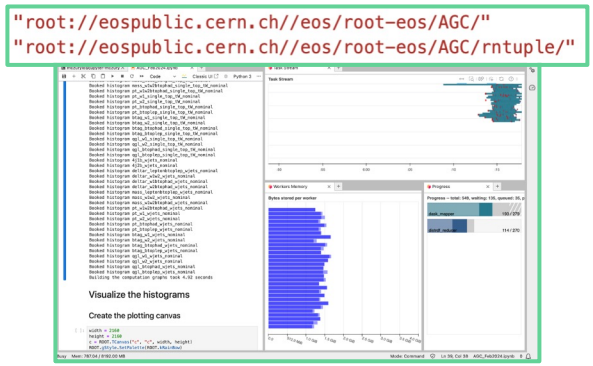
RDF runs on distributed systems, already today: DistRDF

- Same analysis code of local execution
- **Shell and notebooks**
- Runs everywhere
  - Bare-metal clusters
  - HPC environments
  - **Analysis Facilities!**

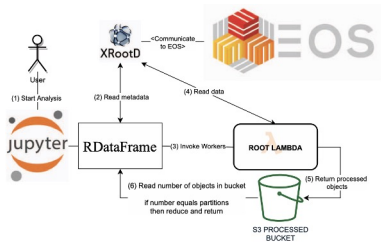
See linked papers for more information

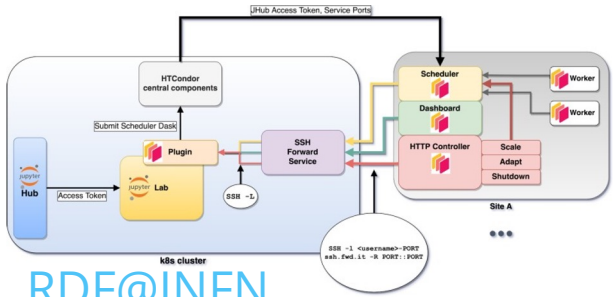
ROOT is ready to run at your favourite analysis facility, today

RDF+RNTuple@SWAN

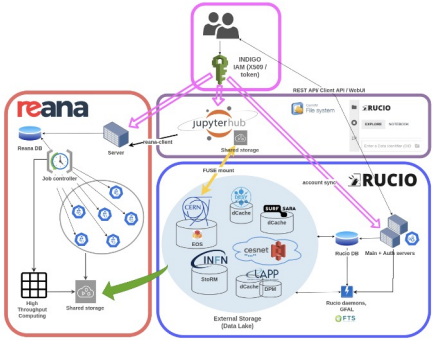


RDF@AWS





RDF@INFN



RDF@VRE

Python Interface: PyROOT

ROOT has been part of the Python ecosystem since years. Clear direction: reinforce this position, prioritizing ROOT experience for Python users. Example actions taken:

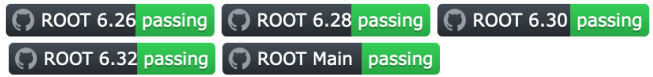
1. **ROOT's C++-Python "interop engine" (cppyy) was updated to its latest version**
   - Faster, less custom patches to maintain (~20), much improved C++ support (e.g. templates)
   - Tested promptly by CMS in the production software stack
   - Made available to all users + ATLAS/LHCb in *dev3 LCG stack on CVMFS* to be tested from day 1
2. For the first time, a demo infrastructure to install ROOT with *pip* was set up
   - Plan: continue to work on the demonstrator, to assess whether it can be used as one of the main ROOT binaries distribution channels – **Try it and give us feedback!**

```
pip install ROOT --index-url https://root-experimental-python-wheels.web.cern.ch
```

1 FTE to join the ROOT team in Summer to work on PyROOT

A New Approach to CI

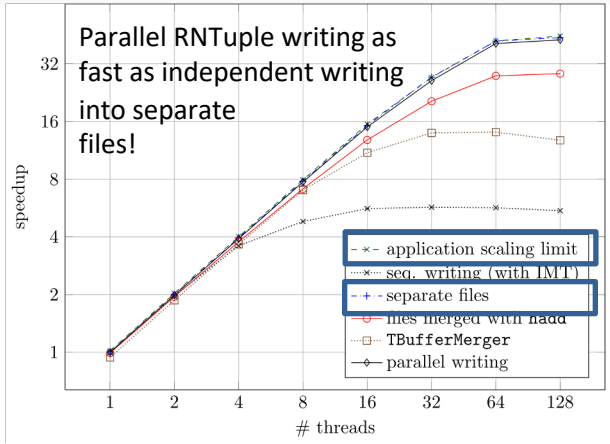
- **ROOT is transitioning its Jenkins-based CI to GitHub Actions**
- **Fits well Open Source philosophy of ROOT**
  - E.g. test results (and failures) visible to anybody, further lowers the barrier for external contributors
- **Integration and testing of the 5 active branches (6.26, 6.28, 6.30, 6.32 and Main) is now steered with GitHub actions**

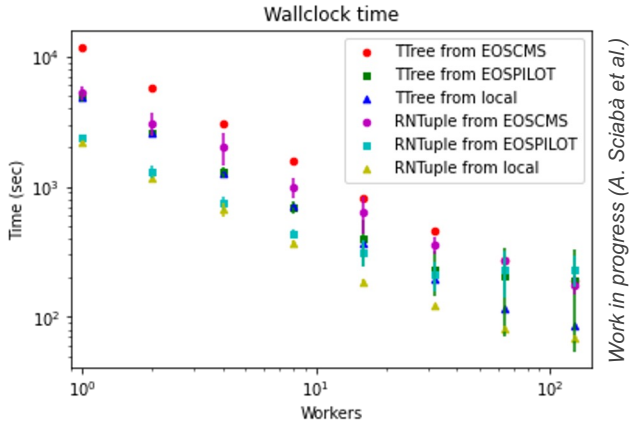


- Technical detail: expanded the pool of builder nodes.
  - Immediate feedback to developers submitting changes to ROOT: all tests run ~immediately
  - Mac nodes that are part of the Apple Beta program: catch early problems that could be triggered by the future MacOS updates

RNTuple

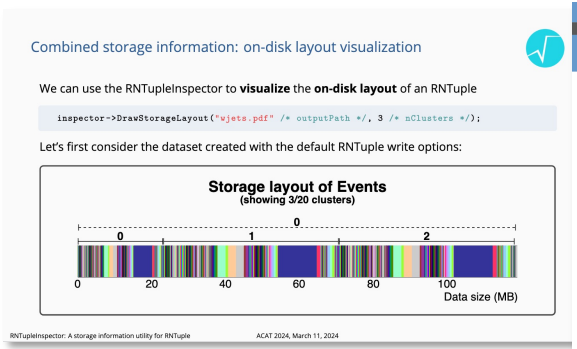
- Skimming of the "Analysis Grand Challenge" (AGC) dataset
  - Drop unused columns
  - Filter events based on coarse cuts and entries in nested collections
- Compare multiple implementations of parallel writing
  - Using ROOT's implicit multithreading (IMT)
  - Separate files + merging with hadd
  - TBufferMerger (in-memory merging)
  - Parallel RNTuple writing
- Common CERN IT - ROOT effort
  - Quick cycles of benchmarking and software improvements on ROOT and XRootD
  - First numbers with the AGC confirm the speed improvements of RNTuple when reading from EOS with high core counts
  - Ongoing work on exploring full cluster scale and full spectrum of workload variations

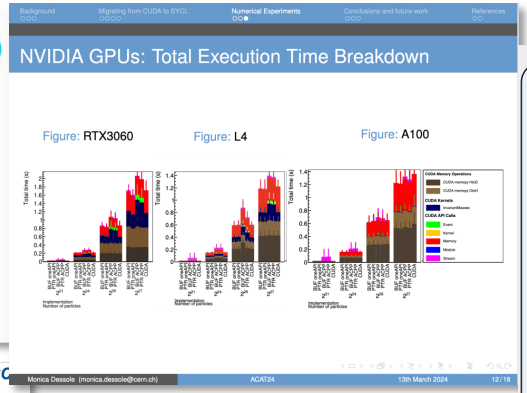


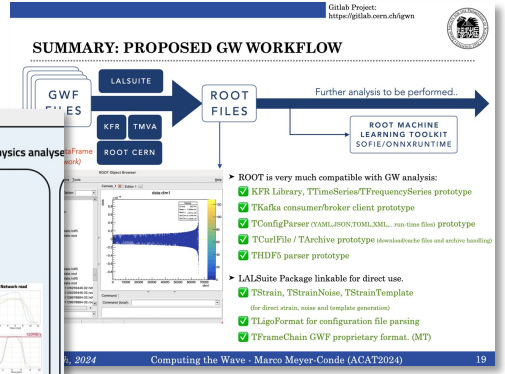


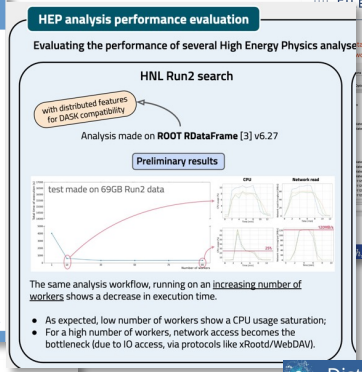
Please see **Development Status of RNtuple, the future HEP Columnar Storage Software Technology** on Wednesday!

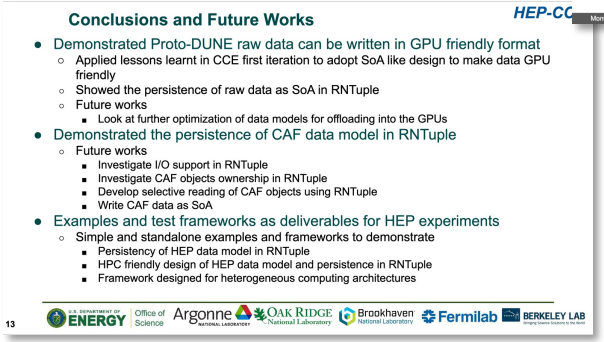
Excellent contributions by ROOT Team members and several ROOT related talks. An (incomplete) selection in this slide. Next appointment: CHEP!

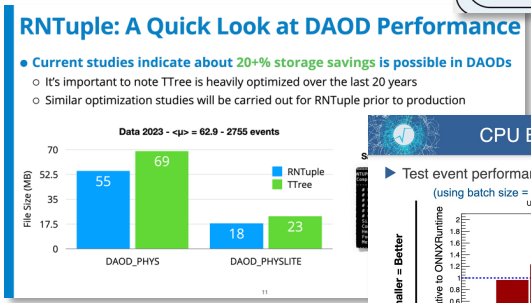


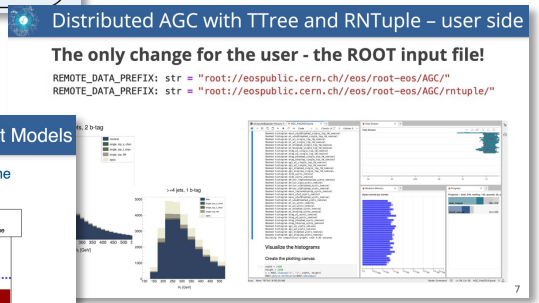


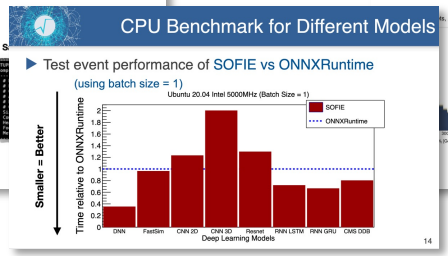








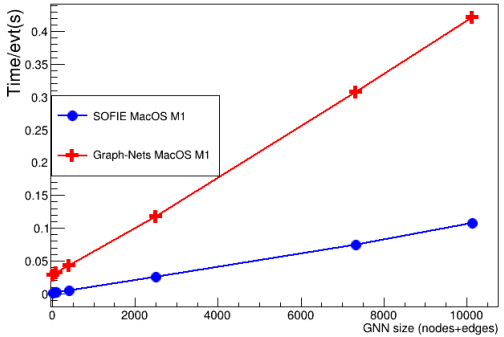




Machine Learning

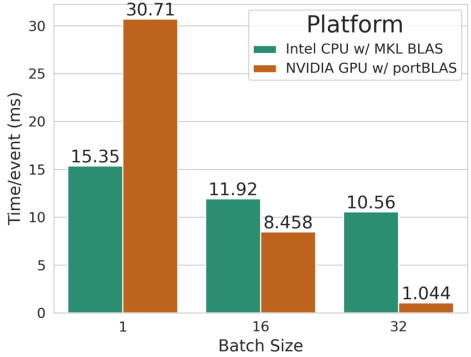
ML4EP project: initiative aimed at grouping all ML-related efforts within EP-SFT, the software "Support Group" of CERN's Experimental Physics Department. That includes ROOT's ML components.

ROOT's philosophy: integrate with existing ML frameworks, do not compete with them



Latest achievements

- Extended SOFIE's ONNX support, and adding message-passing GNN (upon LHCb request, needed for full-event interpretation)
- Batch Geneator for training. Full integration with RDataFrame is in progress
- Fast evaluation of BDT using the FastForest package
- Prototype a SYCL-based implementation of SOFIE



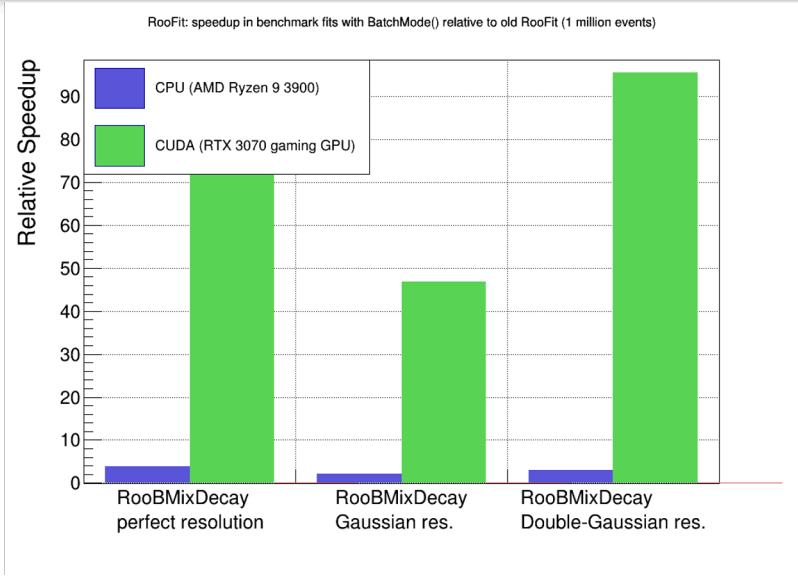
Future Plans

- Integration of SOFIE with the experiments' fast simulation (for GAN and VAE models)
- Further extend ONNX operator support for complex models (GNN, transformers)
- Add GPU support (ALPAKA) for integration in high-level trigger setups

SOFIE *generates C++ functions invokable for the inference of trained neural network models. It takes ONNX model files as inputs and produces C++ header. Available in the 6.32.00 release.*

RooFit: Developments in the Last Few Years

- RooFit: a framework for statistical modelling at LHC analysis scale, which is:
  - **Fast -** optimized likelihood evaluation with aggressive caching, vectorization, **GPU acceleration** and automatic differentiation
  - **Robust -** numeric techniques in likelihood definition to make minimization converge better
  - **Flexible -** it's in C++, so no performance overhead for any creative user code
- The recent years focused on **performance improvements,pythonic interfaces and interoperability** with other tools



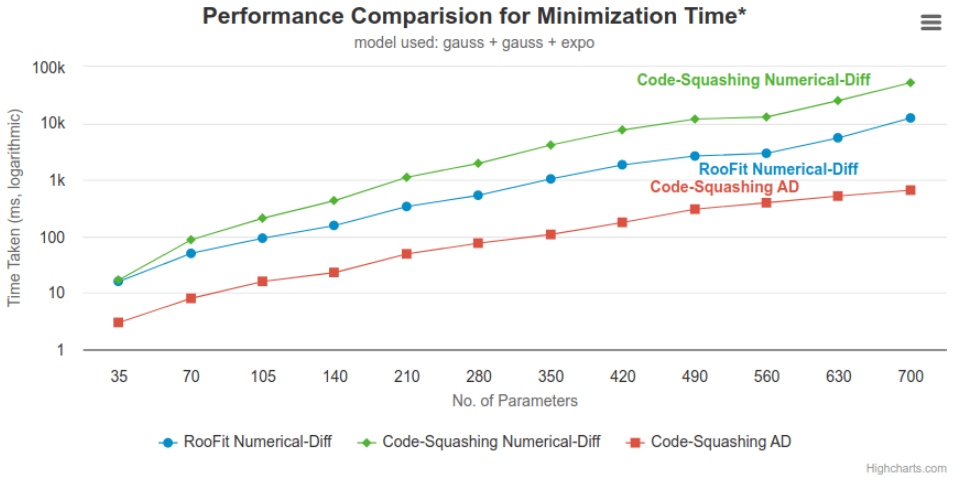
Relative speedup compared to old RooFit of new CPU and CUDA backends (from CHEP 2023) for benchRooFitBackends and RooFitUnBinnedBenchmarks

NVidia GPUs can be transparently used with RooFit, today.

RooFit: The Future

- RooFit PoW also steered through the recent `24 RooFit Workshop
  - Increase R&D investment on minimizers like **Minuit 2,** including better use of Automatic Differentiation (AD), + More focus on documentation
- RooFit developers will continue to be actively engaged in experiments improving RooFit-based frameworks and likelihood publishing.
  - CMS is moving from ROOT 6.22 to 6.30 for statistical analysis with our help
  - Bringing together experts from CMS and ATLAS together to work on common tools, using ROOT as a platform

Automatic Differentiation (AD): a set of techniques to evaluate the partial derivative of a function specified by a computer program. *In some benchmarks, the speedup in minimization from using AD is up to 10x (from CHEP 2023).*



Path to ROOT 7 – PTR7

- Goal: reach consensus on what should be the content of the ROOT 7.00.00 release
  - At first, a process within the ROOT team
  - Some elements already clear now, e.g. Cling, RDataFrame, RNTuple, new histograms
  - Example question: what interfaces can be broken, at what cost, for what benefit?
- 1 Chair and 4 Focus Area Chairs:
  1. Obtaining ROOT, user experience, user interfaces and languages
  2. Statistics, histograms, Machine Learning and mathematical tools
  3. Graphics and GUIs
  4. Core libraries, type system and interpreters

The PTR7 process will start during H2 2024

A Path To ROOT 7

v0.1 15-12-23

At the start of LHC Run 2, ROOT 6 represented a major modernisation of ROOT. The most visible element of this modernisation was Cling, the new LLVM based C++ interpreter, which replaced CINT and much of the type system that came with it. The new release cycle of ROOT, ROOT 7, started with the creation of RDataFrame, before ROOT 7 was released. A major modernisation of RooFit also took place, complementing the rich modelling capabilities provided by RooFit with seamless offloading of calculations on accelerating hardware devices - making RooFit the first accelerated component of ROOT. Those are not the only innovations planned for the new release cycle, the most prominent component of ROOT 7 will be RNTuple: the new column-wise storage of ROOT, which replaces TTree.

ROOT, and its 7th release cycle, is not just RDataFrame, the new RooFit and RNTuple, but also a great opportunity to discuss further modernisations, (backward incompatible ?) changes and new interfaces, improvements that ameliorate ROOT and address even better the needs of our community. **There is an opportunity to seize: thinking of ROOT 7 as ROOT 6 with RDataFrame and RNTuple is just the start.**

An effective way to converge on good solutions and improvements for ROOT 7 is through a process made of well scoped and result oriented blue printing discussions, involving at first ROOT team members, and later, our user community, including LHC experiments.

Conclusions

Conclusions

- **ROOT is at the heart of the (HL-)LHC project, fully committed to support its users in the very long term**
  - 8h to get a first answer on the ROOT Forum
  - Fluent communication with experiments and users
- **Innovation always accompanies stability and solid support model**
  - New PyROOT, RNTuple, web graphics, seamless **Analysis Facility integration**
  - 4 new team members joining this year (not counting students)
- Actively developed (894 PRs opened in 2024), by its team and lively community
  - **Embrace open-source practices to maximise community contributions**
- Solid planning, formed listening to all stakeholders
  - 2024 PoW 20% complete, issues backlog reduced by 22% since Jan 1st
- Process in place to reach consensus about v7.00.00 release targeting HL-LHC