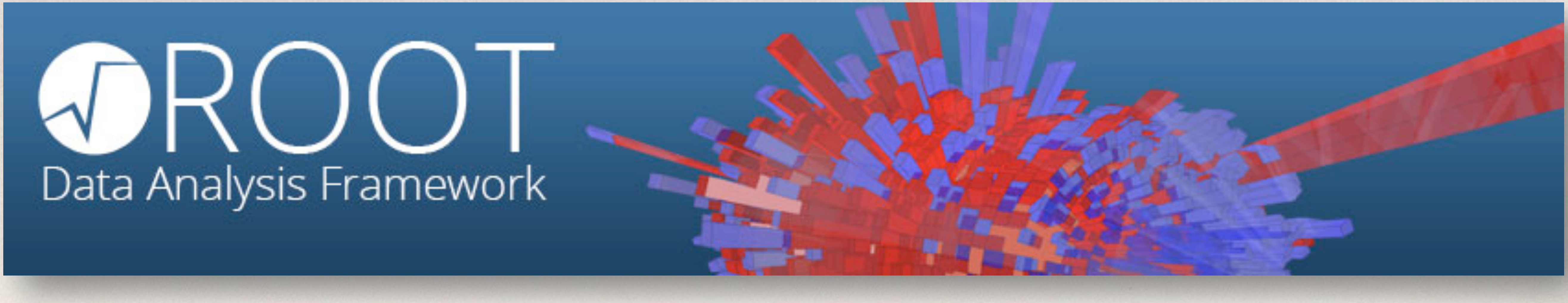


ROOT Plans for 2016 and Beyond

11th Janury 2016, SFT Group Meeting

Outline

- What did we manage from last year’s plans?
- Main areas of work for 2016
  - Development
  - Documentation
  - New Platforms
  - Infrastructure
  - User Support
  - Courses and Tutorials

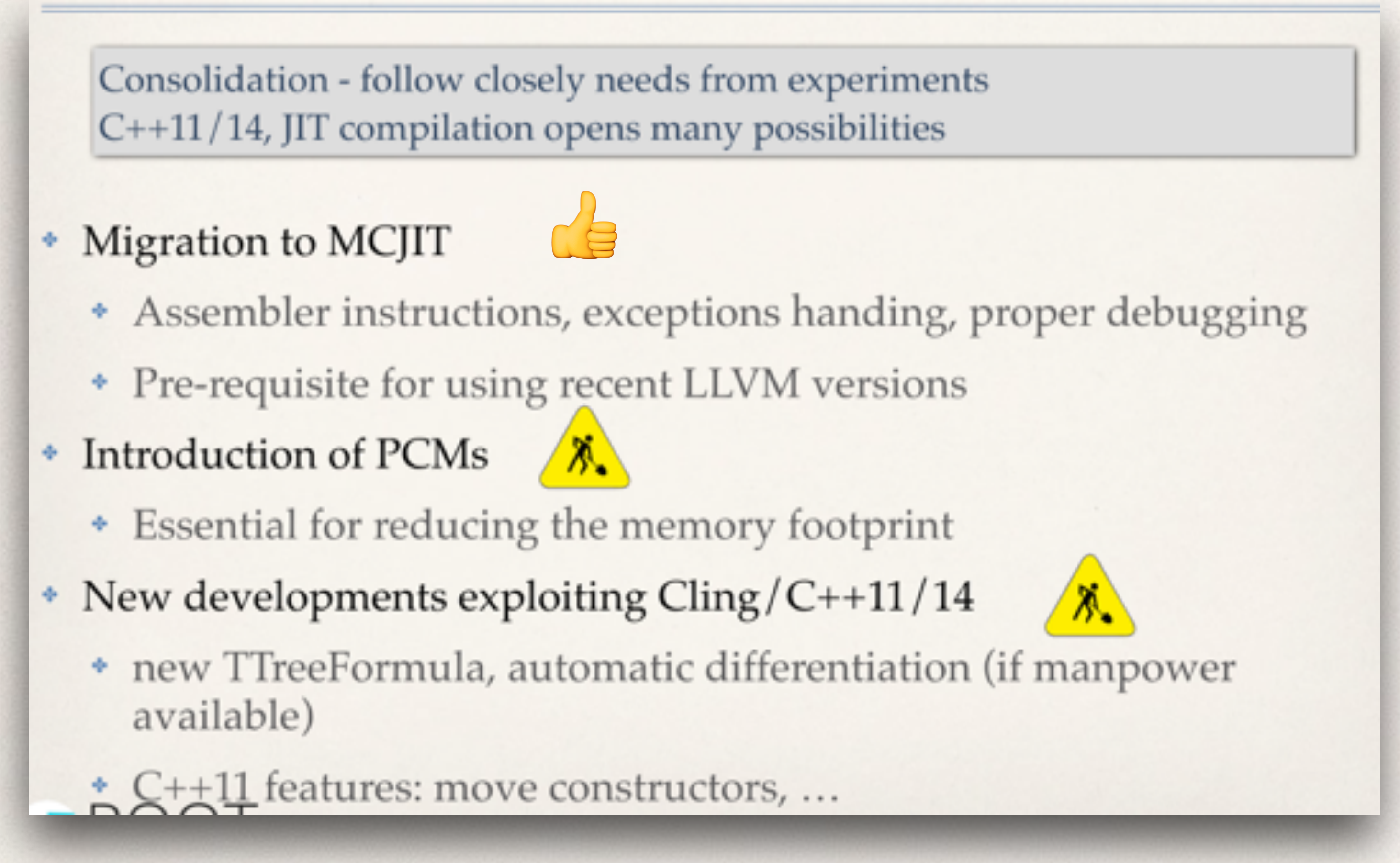
- Note: *LCG Releases* will be covered later in another presentation

The ROOT Core Team

- Bertrand BELLENOT
- Philippe CANAL (FNAL)
- Oliver COUET
- Gerri GANIS
- Benedikt HEGNER
- Pere MATO
- Lorenzo MONETA
- Axel NAUMANN
- Danilo PIPARO
- Elissavet SAKELLARI (TECH)
- David SMITH (IT)
- Enric TEJEDOR (FELL)
- Xavi VALLS (DOCT)
- Vassil VASSILEV (FNAL)

Review Last Year Plan

Cling

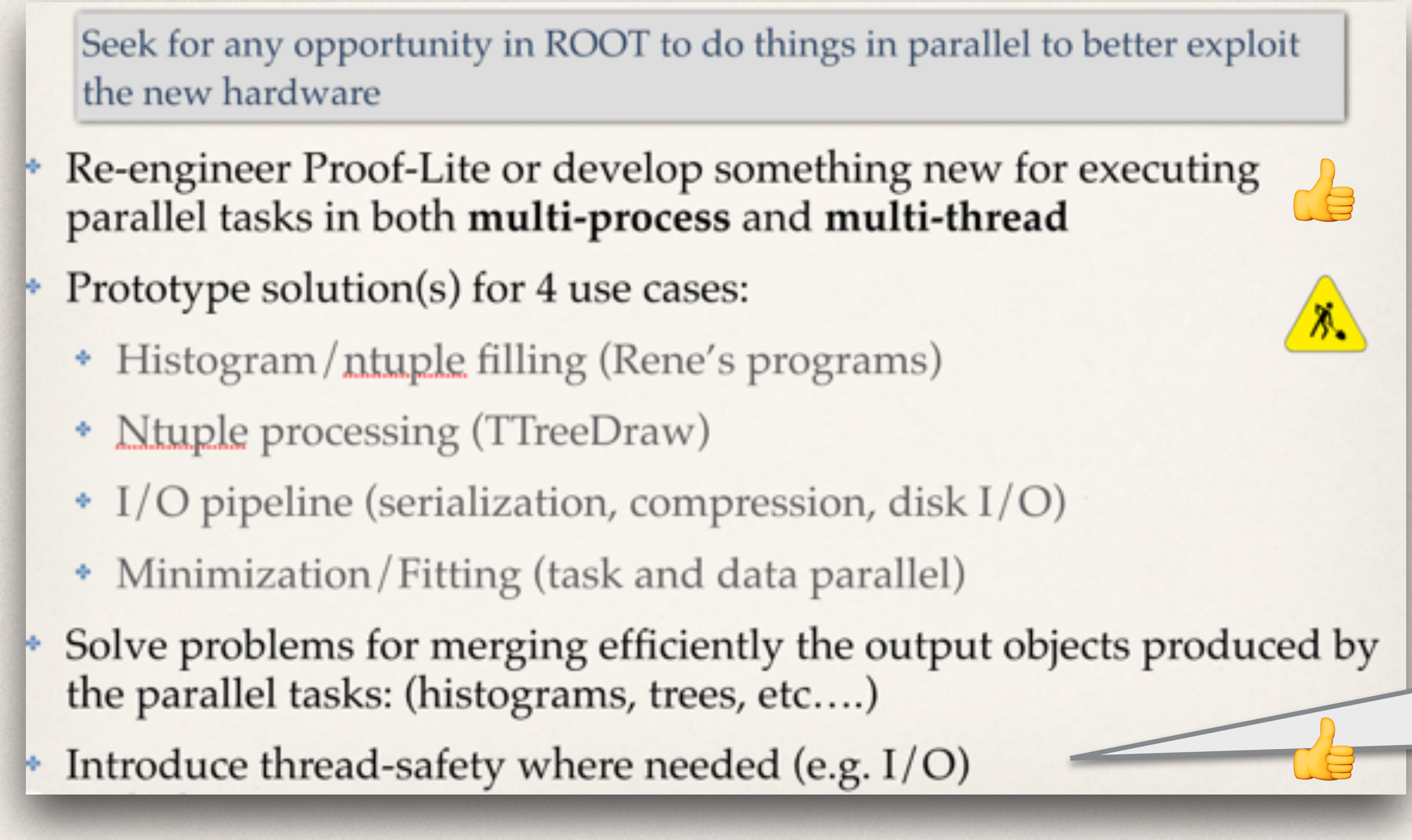


Consolidation - follow closely needs from experiments
C++11/14, JIT compilation opens many possibilities

- Migration to MCJIT 👍
  - Assembler instructions, exceptions handing, proper debugging
  - Pre-requisite for using recent LLVM versions
- Introduction of PCMs 🚧
  - Essential for reducing the memory footprint
- New developments exploiting Cling/C++11/14 🚧
  - new TTreeFormula, automatic differentiation (if manpower available)
  - C++11 features: move constructors, …

- old JIT replaced by OrcJIT
- New TFormula in production
- Not yet PCMs (see later)

Parallelization



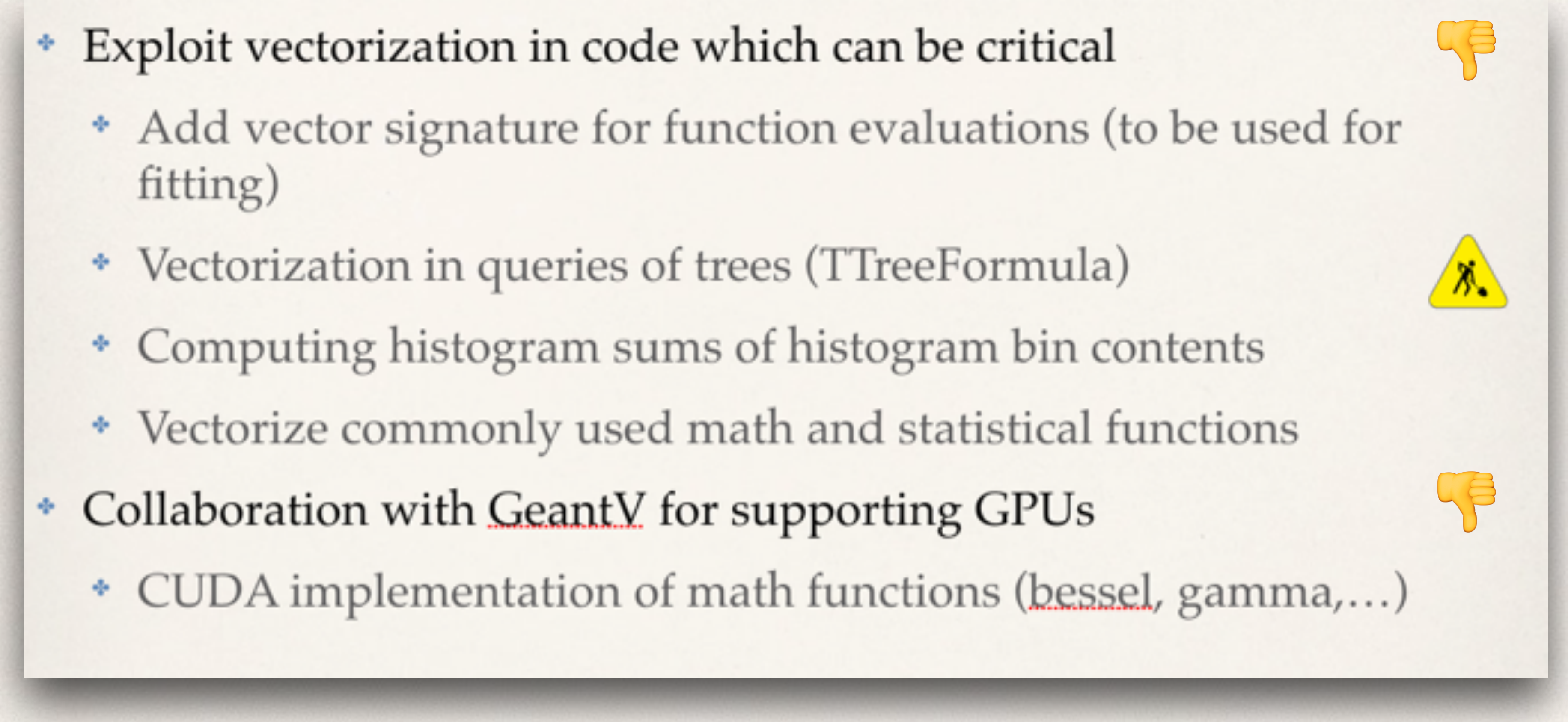
Seek for any opportunity in ROOT to do things in parallel to better exploit the new hardware

- Re-engineer Proof-Lite or develop something new for executing parallel tasks in both **multi-process** and **multi-thread** 👍
- Prototype solution(s) for 4 use cases: 🚧
  - Histogram/ntuple filling (Rene's programs)
  - Ntuple processing (TTreeDraw)
  - I/O pipeline (serialization, compression, disk I/O)
  - Minimization/Fitting (task and data parallel)
- Solve problems for merging efficiently the output objects produced by the parallel tasks: (histograms, trees, etc….)
- Introduce thread-safety where needed (e.g. I/O) 👍

- CMS multithreaded reconstruction relies heavily on ROOT classes
- New multicore tests were added and they revealed some thread-safety issues
- Migration of ROOT threading classes to STL equivalent

- New *multiproc* package released (TProcPool, TMPWorker, TMPClient)
- Implemented task-based TTree reading (I/O pipeline)
  - API to manage IMT (EnableImplicitMT, etc.)
- Prototyped with the *TThreadLocalObject* class (easy merging)
- Just started minimization/fitting use case

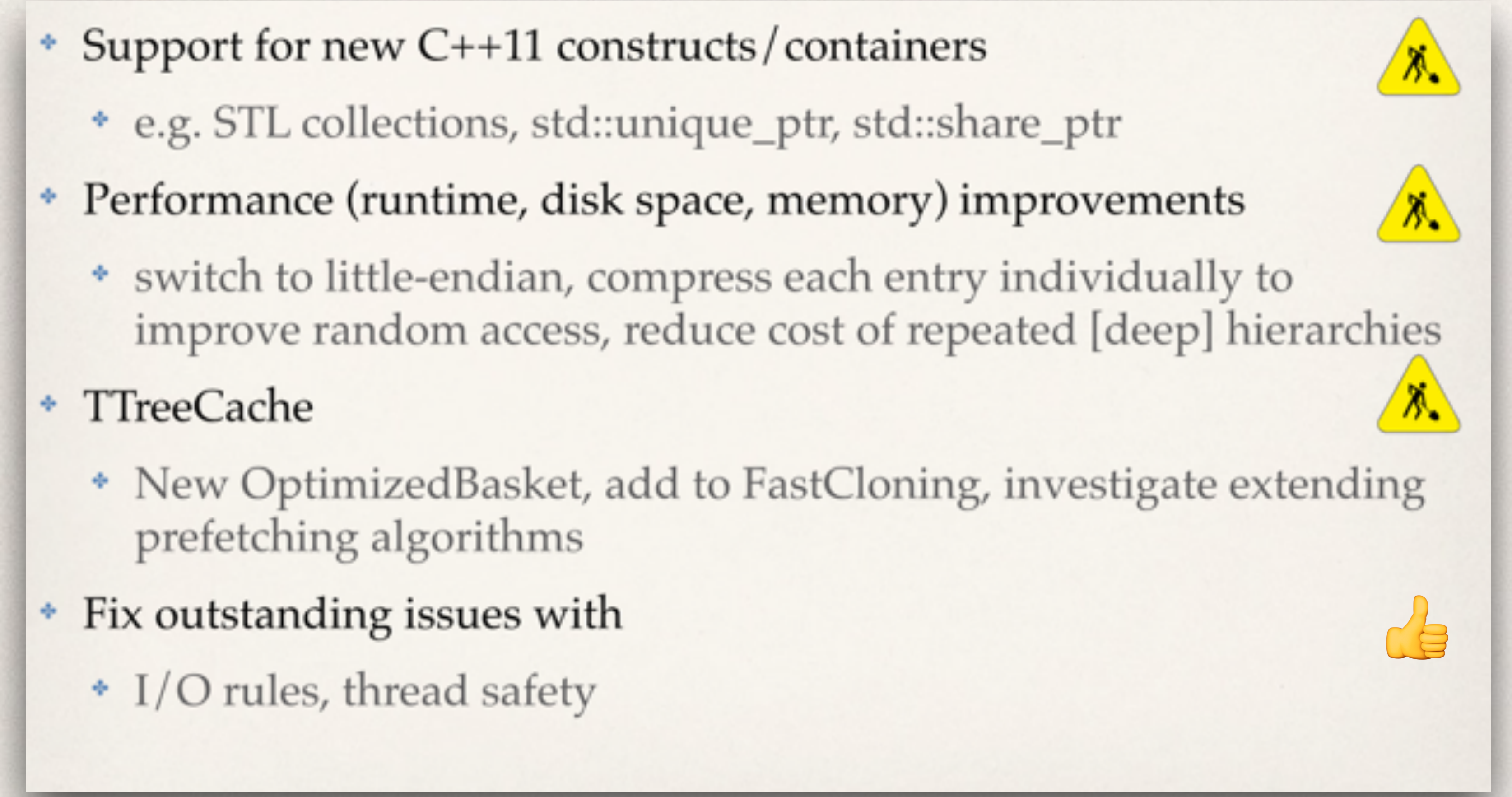
Vectorization



- Exploit vectorization in code which can be critical 👎
  - Add vector signature for function evaluations (to be used for fitting)
  - Vectorization in queries of trees (TTreeFormula) 🚧
  - Computing histogram sums of histogram bin contents
  - Vectorize commonly used math and statistical functions
- Collaboration with GeantV for supporting GPUs 👎
  - CUDA implementation of math functions (bessel, gamma,…)

- Prototype of a vectorized TFormula
- Work really not started in this area
- See plans for this year

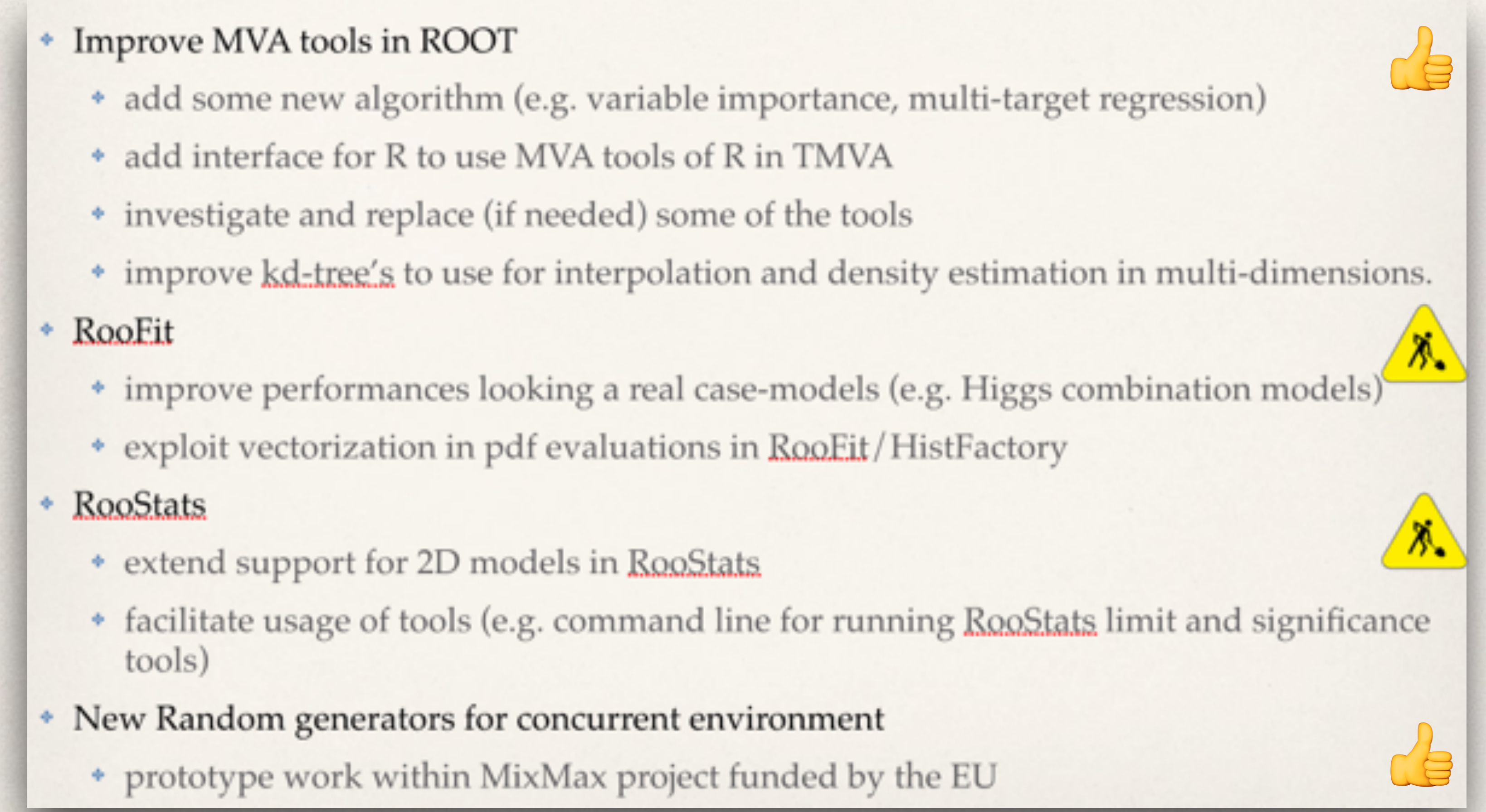
I/O Improvements



- Support for new C++11 constructs/containers ⚠
  - e.g. STL collections, std::unique_ptr, std::share_ptr
- Performance (runtime, disk space, memory) improvements ⚠
  - switch to little-endian, compress each entry individually to improve random access, reduce cost of repeated [deep] hierarchies
- TTreeCache ⚠
  - New OptimizedBasket, add to FastCloning, investigate extending prefetching algorithms
- Fix outstanding issues with 👍
  - I/O rules, thread safety

- Support of C++11 STL containers
- Improvements of the TTreeCache user interface and internals
- Support of std::complex
- User friendly and expressive set of warnings and errors prompted when analysing selection files

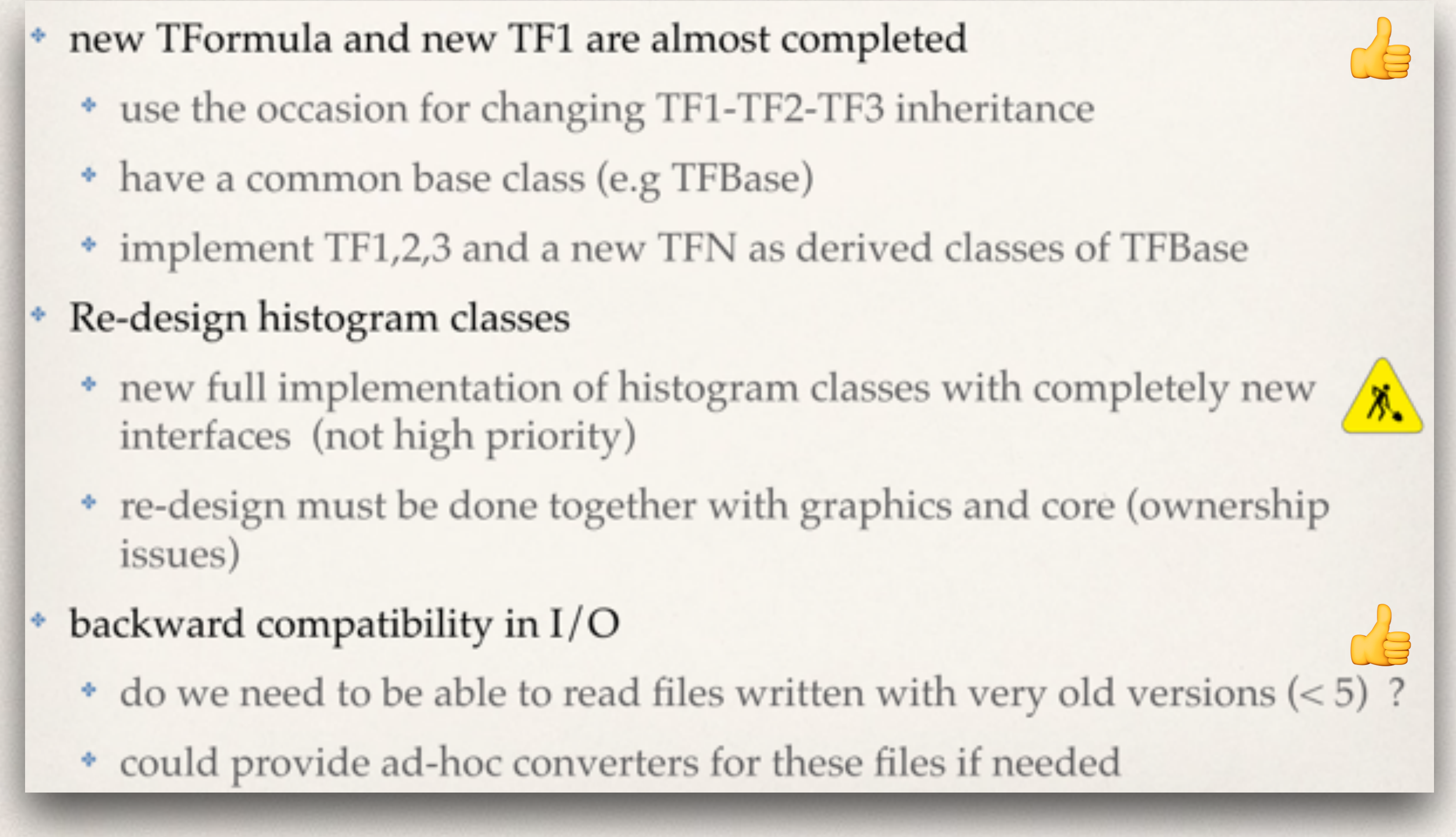
Math Libraries



- Improve MVA tools in ROOT 👍
  - add some new algorithm (e.g. variable importance, multi-target regression)
  - add interface for R to use MVA tools of R in TMVA
  - investigate and replace (if needed) some of the tools
  - improve kd-tree's to use for interpolation and density estimation in multi-dimensions.
- RooFit 🚧
  - improve performances looking a real case-models (e.g. Higgs combination models)
  - exploit vectorization in pdf evaluations in RooFit/HistFactory
- RooStats 🚧
  - extend support for 2D models in RooStats
  - facilitate usage of tools (e.g. command line for running RooStats limit and significance tools)
- New Random generators for concurrent environment
  - prototype work within MixMax project funded by the EU 👍

- TMVA: interface to R and Python, variable importance, ongoing re-design
- kd-tree's to use for interpolation and density estimation
- New Random number classes, MixMax integrated in ROOT
- Support for pdf addition and convolutions in ROOT fitting
- New 2D interpolation and triangulation for TGraph2D

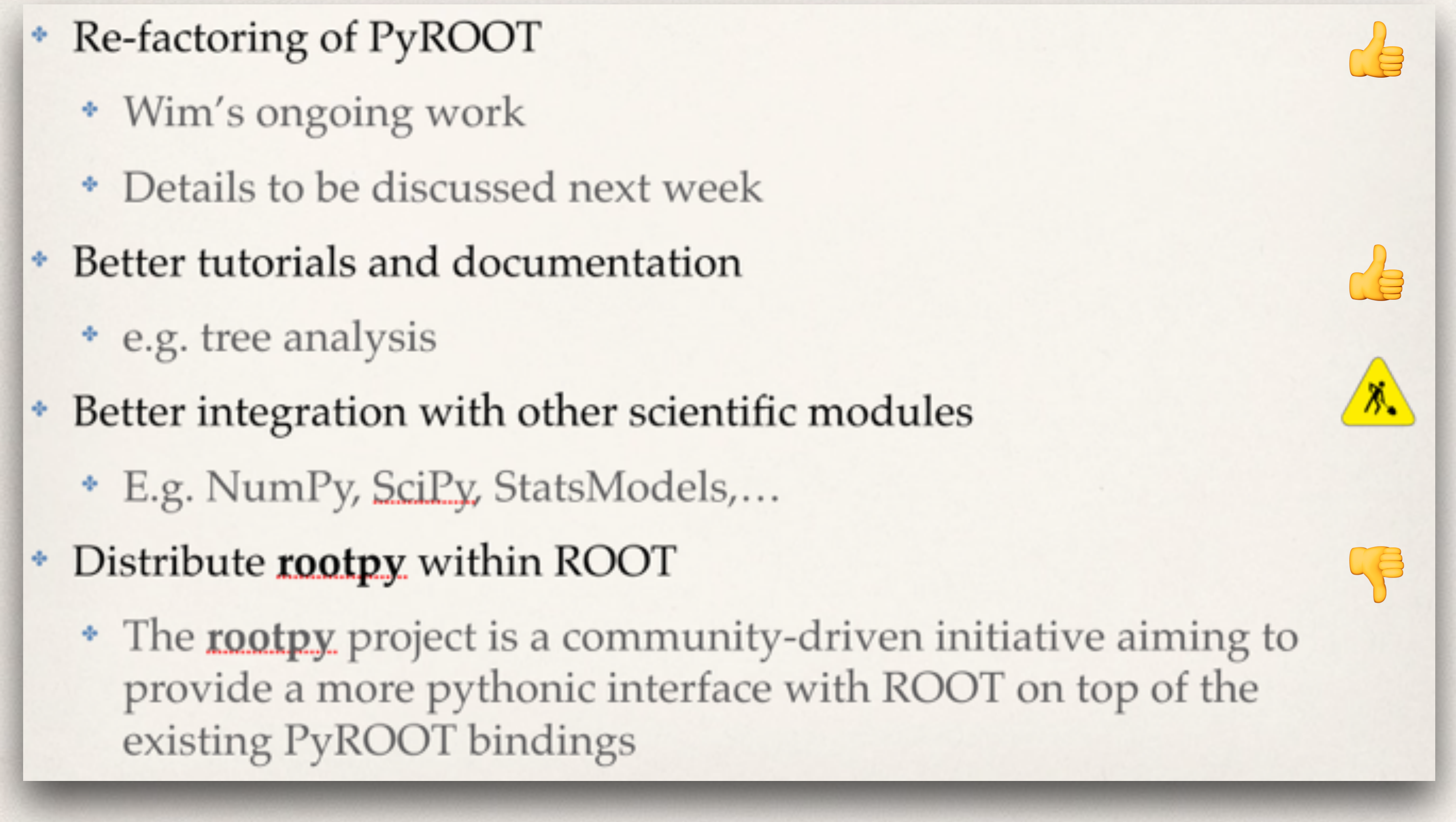
New Histograms



- new TFormula and new TF1 are almost completed 👍
  - use the occasion for changing TF1-TF2-TF3 inheritance
  - have a common base class (e.g TFBase)
  - implement TF1,2,3 and a new TFN as derived classes of TFBase
- Re-design histogram classes
  - new full implementation of histogram classes with completely new interfaces (not high priority) ⚠
  - re-design must be done together with graphics and core (ownership issues)
- backward compatibility in I/O 👍
  - do we need to be able to read files written with very old versions (< 5) ?
  - could provide ad-hoc converters for these files if needed

- New TFormula completed and solved I/O backward compatibility problems
- Started to prototype new histogram classes as part of the new C++ interfaces for ROOT 7 (see later)

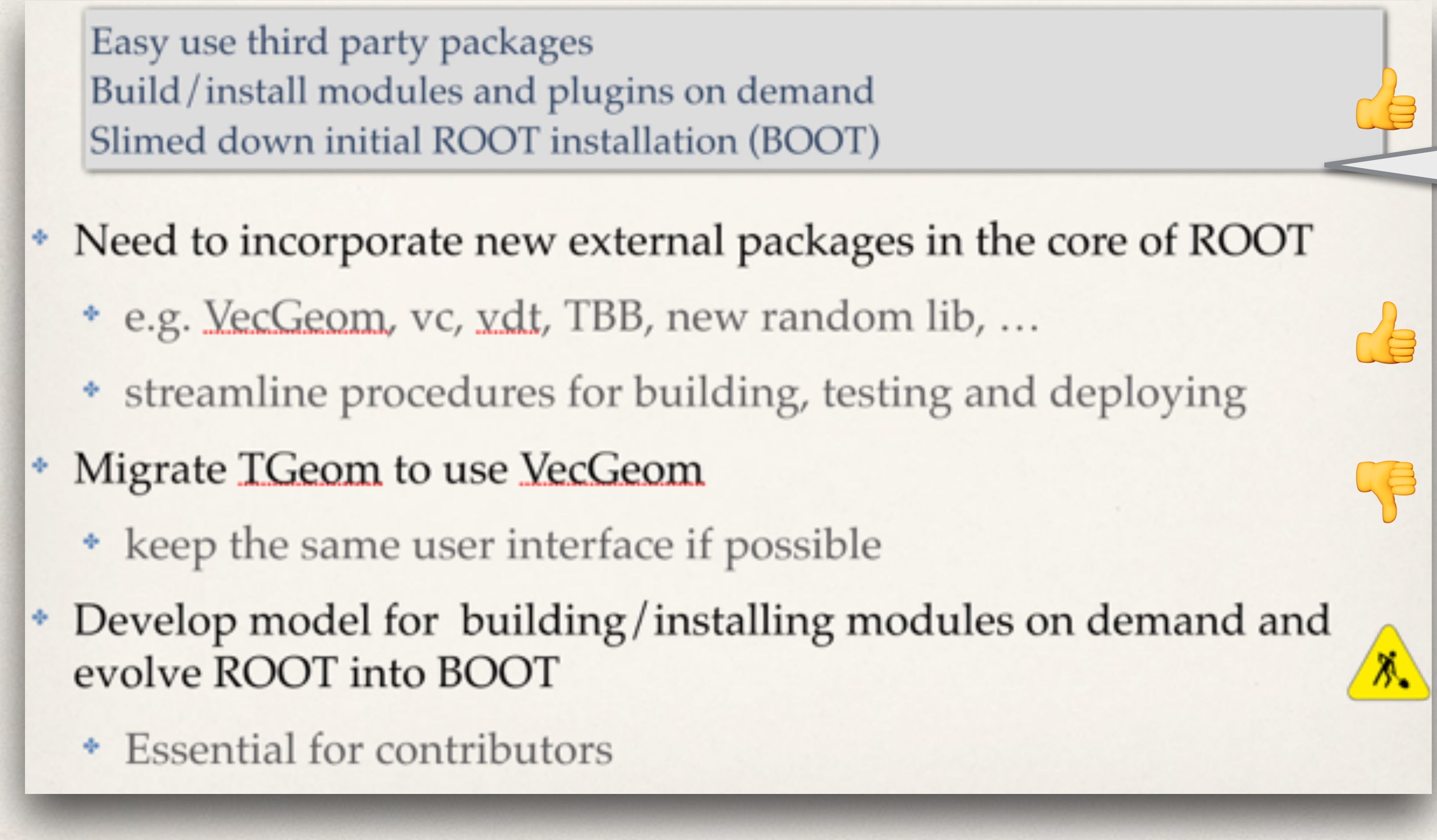
Python Reloaded



- Re-factoring of PyROOT 👍
  - Wim's ongoing work
  - Details to be discussed next week
- Better tutorials and documentation 👍
  - e.g. tree analysis
- Better integration with other scientific modules ⚠
  - E.g. NumPy, SciPy, StatsModels,…
- Distribute **rootpy** within ROOT 👎
  - The **rootpy** project is a community-driven initiative aiming to provide a more pythonic interface with ROOT on top of the existing PyROOT bindings

- Revised tutorials and documentation based on the new ROOTbooks (see later)
- On going work to integrate with other scientific modules (PyMVA)
- **Major problem**: ensure the maintenance of PyROOT after Wim

Packaging



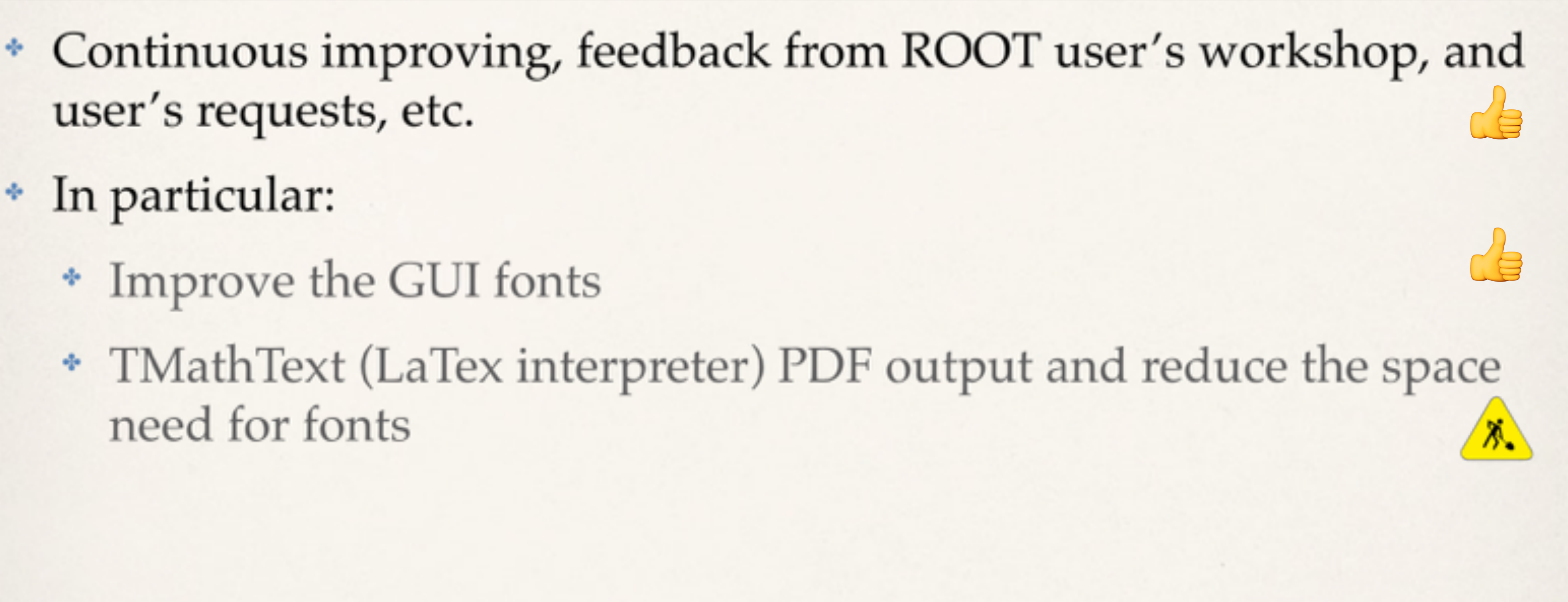
Easy use third party packages
Build/install modules and plugins on demand
Slimed down initial ROOT installation (BOOT) 👍

- Need to incorporate new external packages in the core of ROOT
  - e.g. VecGeom, vc, vdt, TBB, new random lib, … 👍
  - streamline procedures for building, testing and deploying
- Migrate TGeom to use VecGeom 👎
  - keep the same user interface if possible
- Develop model for building/installing modules on demand and evolve ROOT into BOOT ⚠
  - Essential for contributors

- When building ROOT we may decide to build all the external packages as part of the build

- ROOT build can now automatically download and build external packages such as TBB, GSL, FFTW3, CFITSIO, Davix,…
- Established an standardized way to handle them

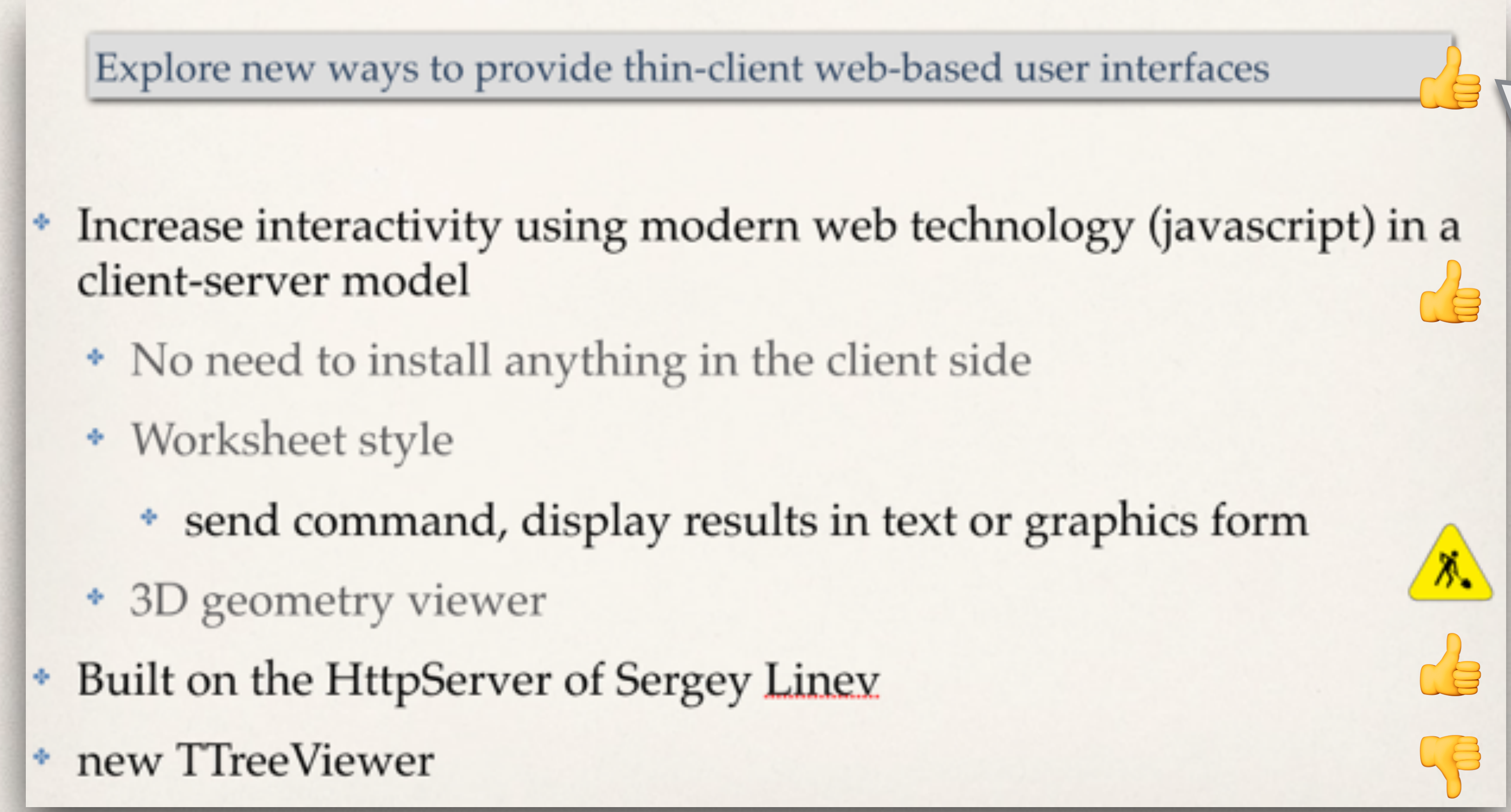
Graphics



- Continuous improving, feedback from ROOT user's workshop, and user's requests, etc. 👍
- In particular:
  - Improve the GUI fonts 👍
  - TMathText (LaTex interpreter) PDF output and reduce the space need for fonts 🚧

- Didn't manage to attract developer to do the work for TMathText
  - Still part of the plan of work (addressing first the failures)

Rethinking UI



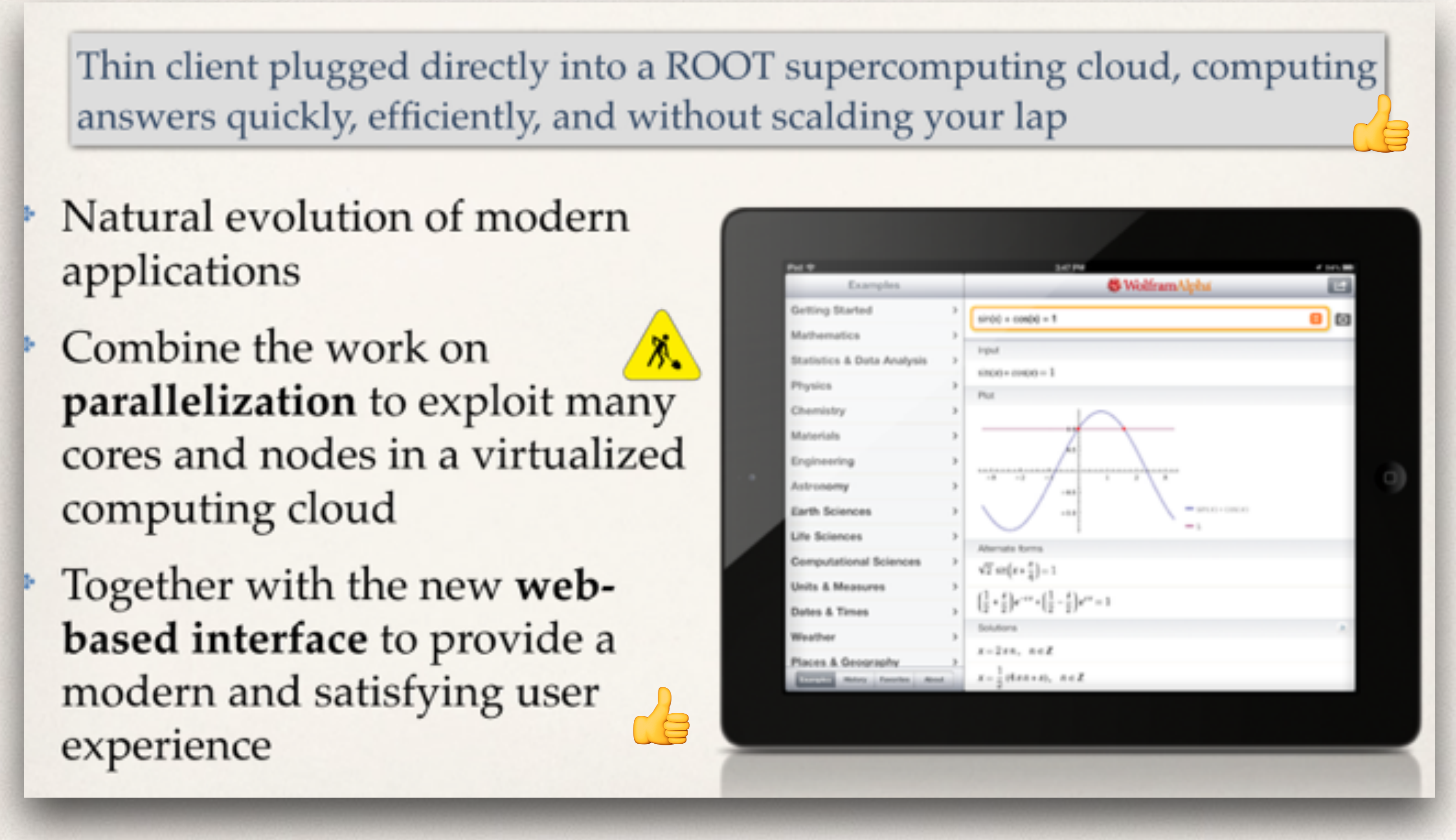
Explore new ways to provide thin-client web-based user interfaces 👍

- Increase interactivity using modern web technology (javascript) in a client-server model 👍
  - No need to install anything in the client side
  - Worksheet style
    - send command, display results in text or graphics form ⚠
  - 3D geometry viewer
- Built on the HttpServer of Sergey Linev 👍
- new TTreeViewer 👎

- Exploration phase completed !!
- We are convinced that **Jupyter notebooks** are the right technology
  - Used successfully in several courses
  - Very positive feedback

- Delivered a full integration of ROOT with the Jupyter notebook technology (**ROOTBook**) (C++, Python, JSROOT)
- Many improvements in JSROOT (3D, new graphs)
- Much better HttpServer

ROOT as-a-Service



Thin client plugged directly into a ROOT supercomputing cloud, computing answers quickly, efficiently, and without scalding your lap 👍

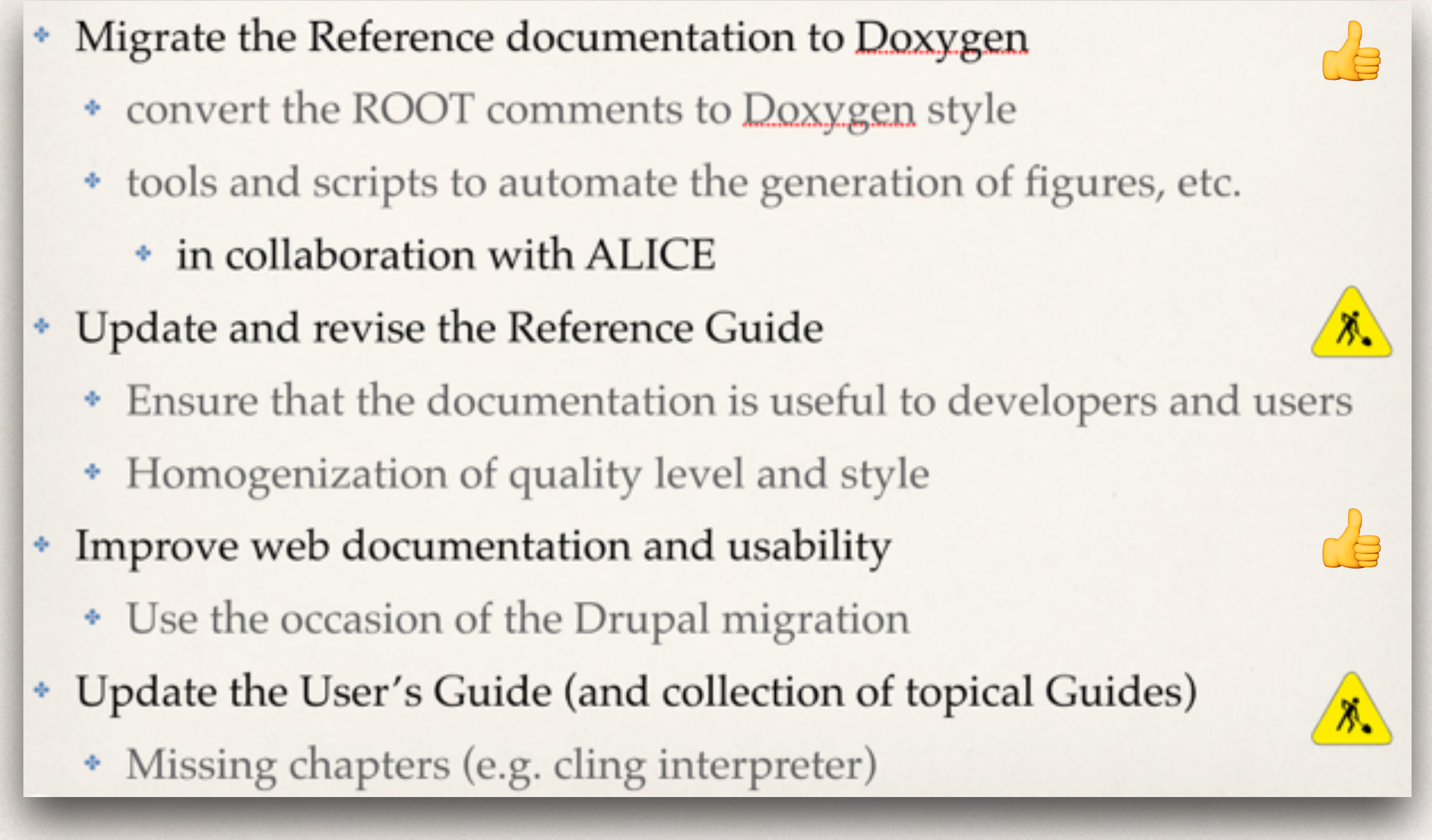
- Natural evolution of modern applications
- Combine the work on **parallelization** to exploit many cores and nodes in a virtualized computing cloud
- Together with the new **web-based interface** to provide a modern and satisfying user experience 👍

Take a tour with Binder:
(no account needed)
http://mybinder.org/../rootbinder

ROOTaaS prototype:
https://dmaas-proto-cc7.cern.ch

- Have set up a **single-node prototype** of ROOTaaS that users can access with just a web browser. Integrated in CERN ecosystem (SSO, EOS, CERNBox, CVMFS, OpenStack, …)
  - Collaboration with IT-DSS and IT-CDA

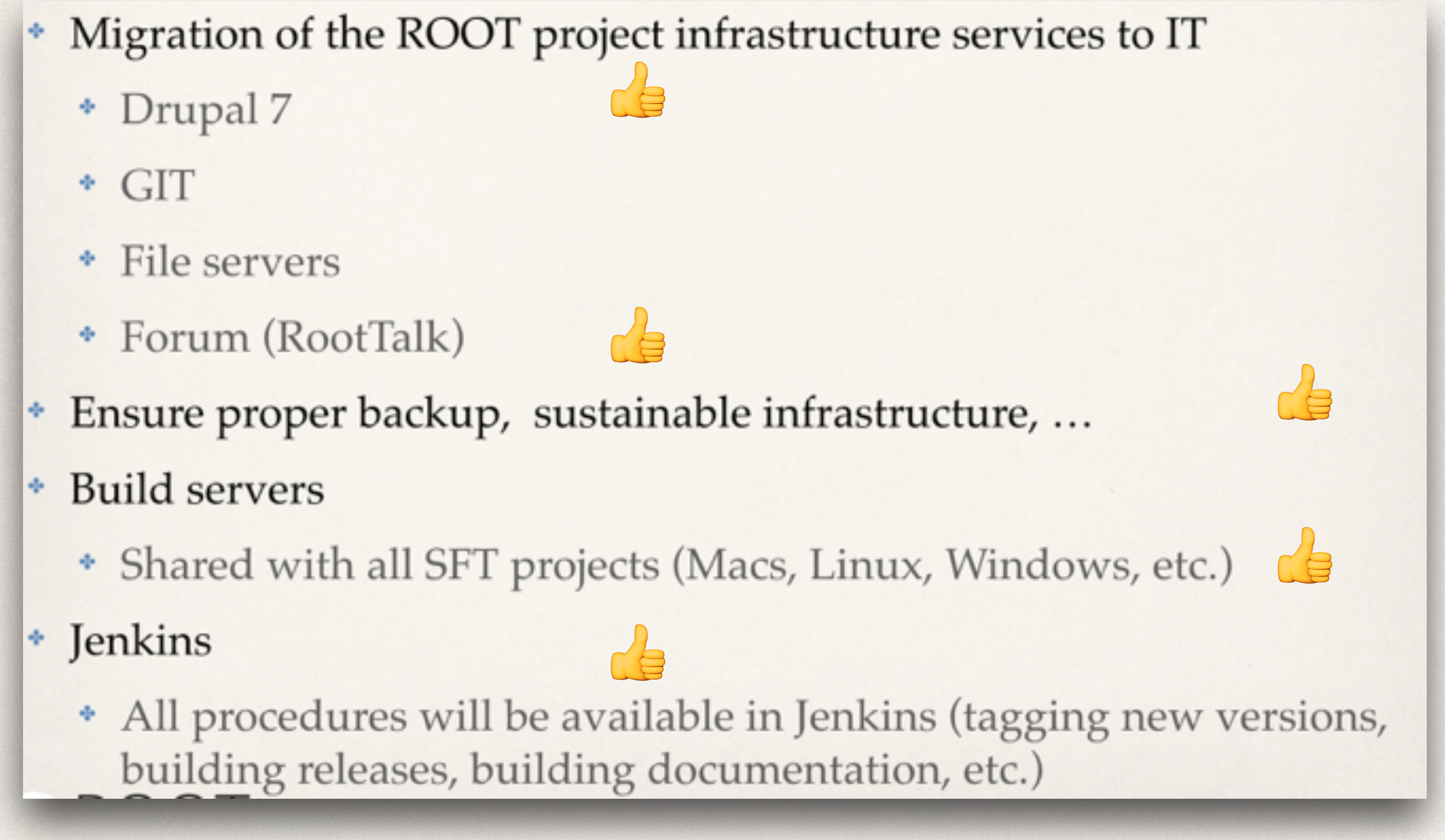
Documentation



- Migrate the Reference documentation to Doxygen 👍
  - convert the ROOT comments to Doxygen style
  - tools and scripts to automate the generation of figures, etc.
    - in collaboration with ALICE
- Update and revise the Reference Guide 🚧
  - Ensure that the documentation is useful to developers and users
  - Homogenization of quality level and style
- Improve web documentation and usability 👍
  - Use the occasion of the Drupal migration
- Update the User's Guide (and collection of topical Guides) 🚧
  - Missing chapters (e.g. cling interpreter)

- Doxygen migration almost finished (generation of figures)
- Drupal migration and content re-organization done (thanks to Nefeli!)
- Migration of tutorials to ROOTbooks started

Infrastructure

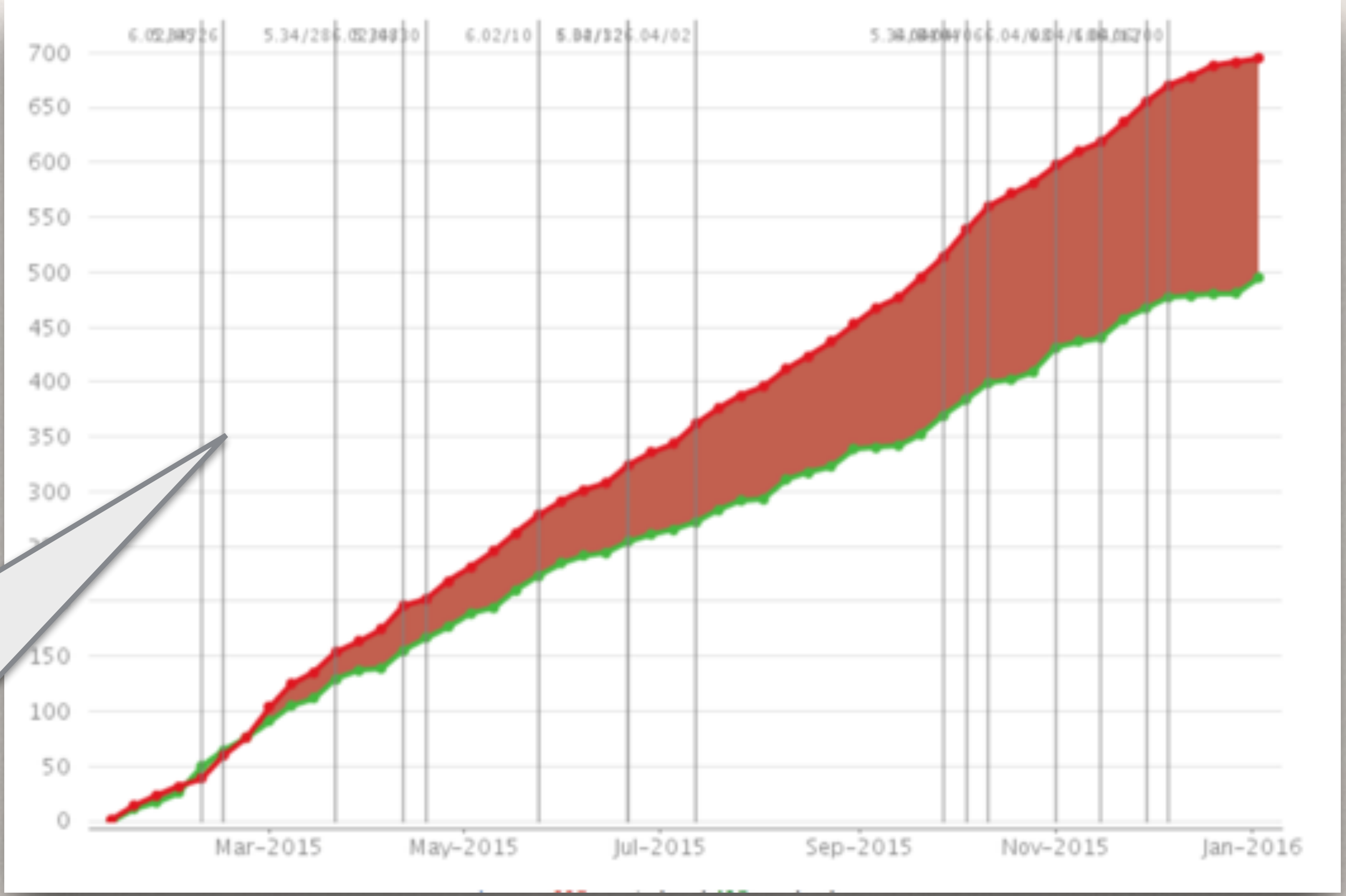


- Migration of the ROOT project infrastructure services to IT
  - Drupal 7 👍
  - GIT
  - File servers
  - Forum (RootTalk) 👍
- Ensure proper backup, sustainable infrastructure, … 👍
- Build servers
  - Shared with all SFT projects (Macs, Linux, Windows, etc.) 👍
- Jenkins 👍
  - All procedures will be available in Jenkins (tagging new versions, building releases, building documentation, etc.)

- Big step forward in sharing and rationalizing services (many thanks to Patricia!)
- 3 clicks procedure for creating new releases (fully automated, jenkins driven steps to generate binaries, transfer them on AFS and create pages in the website)

User Support: JIRA

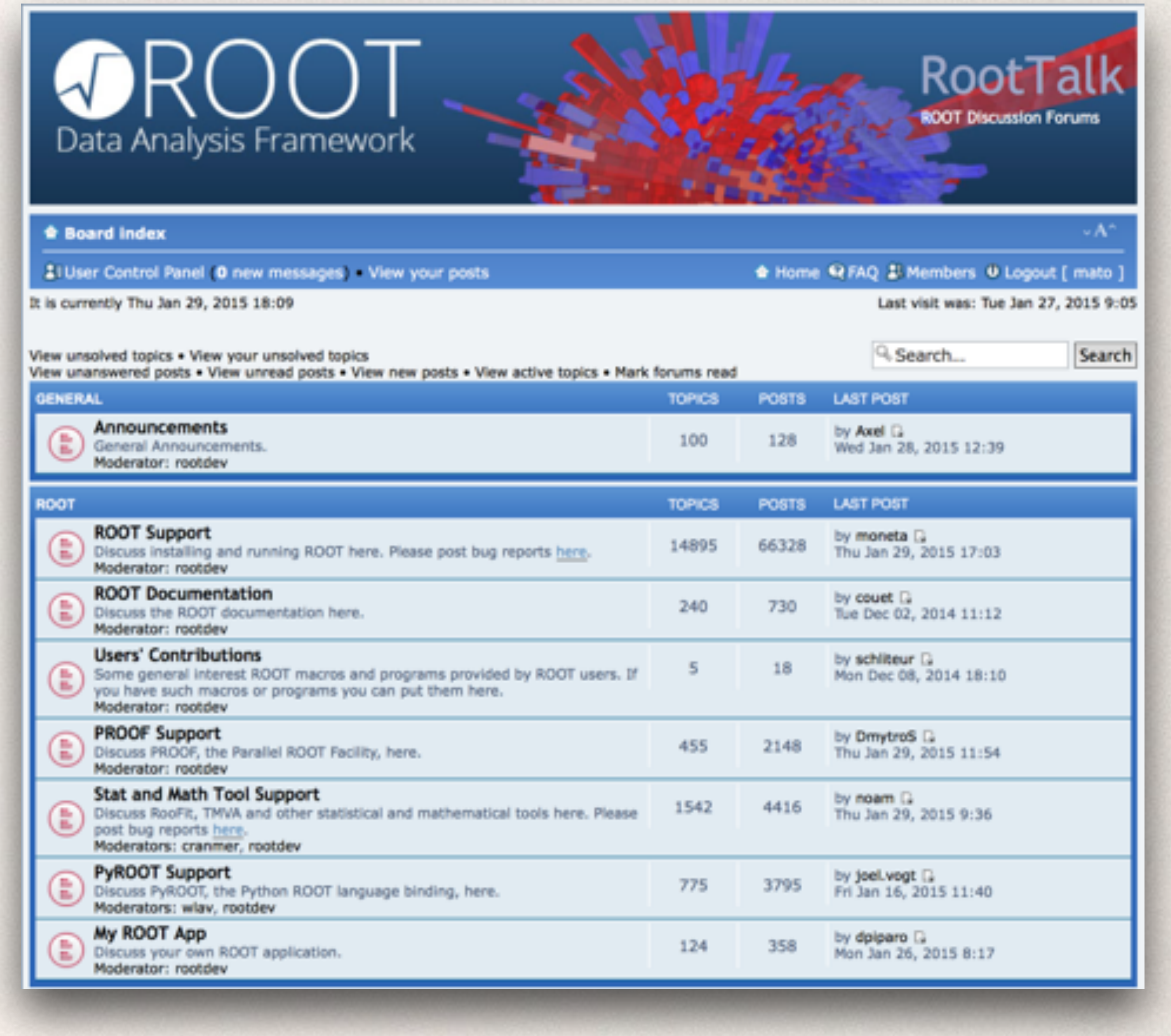
- Bugs: 695 created and 495 resolved
  - backlog increased by 200 issues (of total about 735)
  - ~4 new bugs/working day
- Definitively needed a campaign to triage and eventually close 'obsolete' bugs and tasks.



15 releases in 2015 !

User Support: RootTalk

- In addition to JIRA we have the RootTalk Forum:
  - Total posts 89456
  - Total topics 20513
  - ~ 20 posts/day
  - ~ 5 new topics/day
- Setup weekly shifts to ensure that no post gets unanswered
  - A lot of effort ~1 FTE
  - Latency reduced dramatically!



ROOT
Data Analysis Framework
RootTalk
ROOT Discussion Forums
Board index
User Control Panel (0 new messages) • View your posts
Home FAQ Members Logout [mato]
Search...
Search

It is currently Thu Jan 29, 2015 18:09

Last visit was: Tue Jan 27, 2015 9:05

View unsolved topics • View your unsolved topics
View unanswered posts • View unread posts • View new posts • View active topics • Mark forums read

GENERAL	TOPICS	POSTS	LAST POST
Announcements General Announcements. Moderator: rootdev	100	128	by Axel Wed Jan 28, 2015 12:39

ROOT	TOPICS	POSTS	LAST POST
ROOT Support Discuss installing and running ROOT here. Please post bug reports here. Moderator: rootdev	14895	66328	by moneta Thu Jan 29, 2015 17:03
ROOT Documentation Discuss the ROOT documentation here. Moderator: rootdev	240	730	by couet Tue Dec 02, 2014 11:12
Users' Contributions Some general interest ROOT macros and programs provided by ROOT users. If you have such macros or programs you can put them here. Moderator: rootdev	5	18	by schliteur Mon Dec 08, 2014 18:10
PROOF Support Discuss PROOF, the Parallel ROOT Facility, here. Moderator: rootdev	455	2148	by DmytroS Thu Jan 29, 2015 11:54
Stat and Math Tool Support Discuss RooFit, TMVA and other statistical and mathematical tools here. Please post bug reports here. Moderators: cranmer, rootdev	1542	4416	by noam Thu Jan 29, 2015 9:36
PyROOT Support Discuss PyROOT, the Python ROOT language binding, here. Moderators: wlav, rootdev	775	3795	by joel.vogt Fri Jan 16, 2015 11:40
My ROOT App Discuss your own ROOT application. Moderator: rootdev	124	358	by dpiparo Mon Jan 26, 2015 8:17

Conferences/Workshops 2015

- XrootD workshop, 26-29 January, UCSD
  - Philippe
- SAAGAS , 23-25 February 2015, in Aachen
  - Keynote by Axel
- CHEP, 13-17 April 2015, Okinawa
  - XXX parallel presentations, 1 track summary Danilo
- ACCU, 21-25 April 2015, Bristol
  - Plenary by Axel
- ROOT Users' Workshop, 15-18 September 2015, Saas-Fee
  - Many presentations
- ATLAS and LHCb workshops, November 2015

Courses and Tutorials

- 2015 CSC (Danilo, Benedikt)
- 2015 tCSC (Danilo)
- 2015 Summer Students' ROOT Lectures (Olivier, Danilo)
- E-Planet secondment at Universidade do Estado de Rio de Janeiro (Lorenzo, Sergei)

Main areas of work for 2016

Cling Interpreter and PCMs

- Upgrade to latest LLVM/CLANG
  - Support new ABI (GCC 5)
- Optimizations and improvements
  - reduce and CPU and memory consumption
  - transparently be able to create functions and the ROOT prompt
  - tab-completion based on multi-interpreters
- PCMs for ROOT’s dictionaries
  - be able to replace the today unique PCH by a number of PCMs

Parallelization: Multi-Threading

- Continue tackling the parallelization use cases
- In particular, the next steps are:
  - Investigate the *TTree::Draw* parallelization and its combination with the *TThreadLocalObject* class for the parallel histogram filling
  - *TTree::Process(lambda)*
  - Investigate how to read multiple events in parallel
  - Investigate how to write multiple events/branches in parallel
- Benchmark different prototypes
  - Speedup, understand the CPU overheads, memory increase for the different cases
- Investigate how to interface to an externally provided **scheduler**

Parallelization: Multi-Process

- Ensure the same (compatible) interface between the multi-threading and multi-processor solutions
- Improve the new **core/multiproc** package
  - Complete support for TSelector and provide integration with TTree::Draw
  - Provide detailed documentation and more complex examples
  - Investigate the extension of the multiproc interfaces to a cluster of machines (à la iPython parallel)
- PROOF in maintenance mode

Parallelization: Math Use Cases

- Deploy parallelization for ROOT fitting using both multi-threads and multi-processes
- Optimize the low-level implementation of **multiproc** to be useful for replacing the RooFit specific solution
- Use multi-process solution for RooStats calculators (e.g. frequentist studies which requires large generation of pseudo-experiments)

Vectorization

- Use types in the VecCore library, which embeds low-level support for vectorization, in the ROOT Liner Algebra classes.
- VecGeom could be an external dependency, but it is very likely to be directly used by core ROOT components, in which case it would need to be included as a base package.
- Interface to Vector Function evaluation to be used mainly for fitting.

New C++ Interfaces (ROOT 7)

- Many interfaces can be improved with C++14,17
  - Ownership, type safe containers, string options
  - Improved user productivity, by dramatically reducing memory errors, wrong results, etc.
- Bi-weekly meetings with users from experiments
  - started discussing iteration and ownership issues
- Starting with histograms + visualization, and TFile
- Investigation of TTree and POD storage interface

I/O Performance

- Better support concurrent I/O operations
- Code (algorithm) optimization
  - New version of Optimize Basket
  - Prefetching in fast cloning
  - I/O implementation improvements
- Optimization via change in (low level) file format
  - Endianess leading to R&D on “memcpy to disk”, i.e. persistent memory layout (POD)
  - compression windows, eliminating metadata redundancies
- Provide help for benchmarking I/O performance in presence of automatically generated POD data models (e.g. HepMC3, PODIO)

I/O New Features

- Support for C++11 std::unique and std::array
- Investigate how to support C++11 shared and weak pointers
- Streamlining of I/O interface (mostly in the context of the v7 new interfaces)
- Open up the interface of TFile to be able to make use of key-value storage
  - Investigate the Kinetic key-value store technology
- Support for JIT-ted collection proxies
  - Enable transparently the serialization of collections
  - Stream collection without dictionary with MultiProc

Geometry

- Provide the users of **TGeo package** and its derived navigation interfaces (VMC) automatic support for vectorized navigation based on the VecGeom package
    - Add VecGeom library as CMake-configurable external module
    - Phase-1: Implementation of a TGeoShape-derived bridge class (TGeoVGShape) delegating the navigation interface to the VecGeom solid
    - Phase-2: Implementation of a VecGeom-aware navigation interface that can redirect the current navigation API of TGeoManager/ TGeoNavigator to native VecGeom navigators

Math Libraries

- Treat VC and VDT libraries like all the other external libraries
  - Look for them externally, otherwise build them internally (builtin_XXX option)
- Integrate new minimizers based on **libcmaes** (new external)
- Re-implement TRandom classes using ROOT::Math::Random
- Complete **MixMax** testing and make it the default random generator
- Extend **RooStats** asymptotic calculator to 2 dimensions
- Integrate in **RooFit** new class for multi-dimensional kernel density estimator

Machine Learning

- Active participation to the Inter-Experimental LHC Machine Learning Working Group IML launched recently
- Foreseen tasks and activities:
  - Complete TMVA re-design
  - Add support for cross-validation in TMVA
  - Add possibility for parallel execution in TMVA methods
  - Evaluate and Integrate a new deep learning neural network in TMVA
  - New SVM (support vector machines) models in TMVA

Graphics

- Explore integration with JSROOT, i.e. avoid parallel developments
- Development of new plots
  - Helper class for Ratio Plots (e.g. Data/MC, MC/MC)
  - Spider plot not based on TTrees (e.g. Data Frames)
- Consolidation of the present functionalities, e.g. fixed size fonts.
- Style object containing all style elements, e.g. legend placement applicable to other graphics objects
- The graphics UI needs some rethinking and/or simplification
  - leaner interface, helper tools (classes/functions) for plotting, automatic color/line/marker/legend schemes
- Many requests in JIRA

JSROOT

- Complete 2D drawing options
- Add missing 3D drawing & options
- Add simple GUI (overlay buttons) to access some functionality (e.g. saving as .png)
- Finalize geometry
  - Add (many) missing shapes and constructive solid geometry (CSG)
- Add GUI for interactivity with THttpServer

Python bindings and ecosystem

- Python3 fully supported
  - Few pending issues to be fixed
- Release PyROOT as a genuine Python package
  - Co-existing within Python2 and Python3
  - build/install based on *pip* package management?
- Prototype tree/ntuple analysis which adopts a Data Frame-like interface
  - Learn from interfaces used by crowds of data scientists

Packaging and Modularization

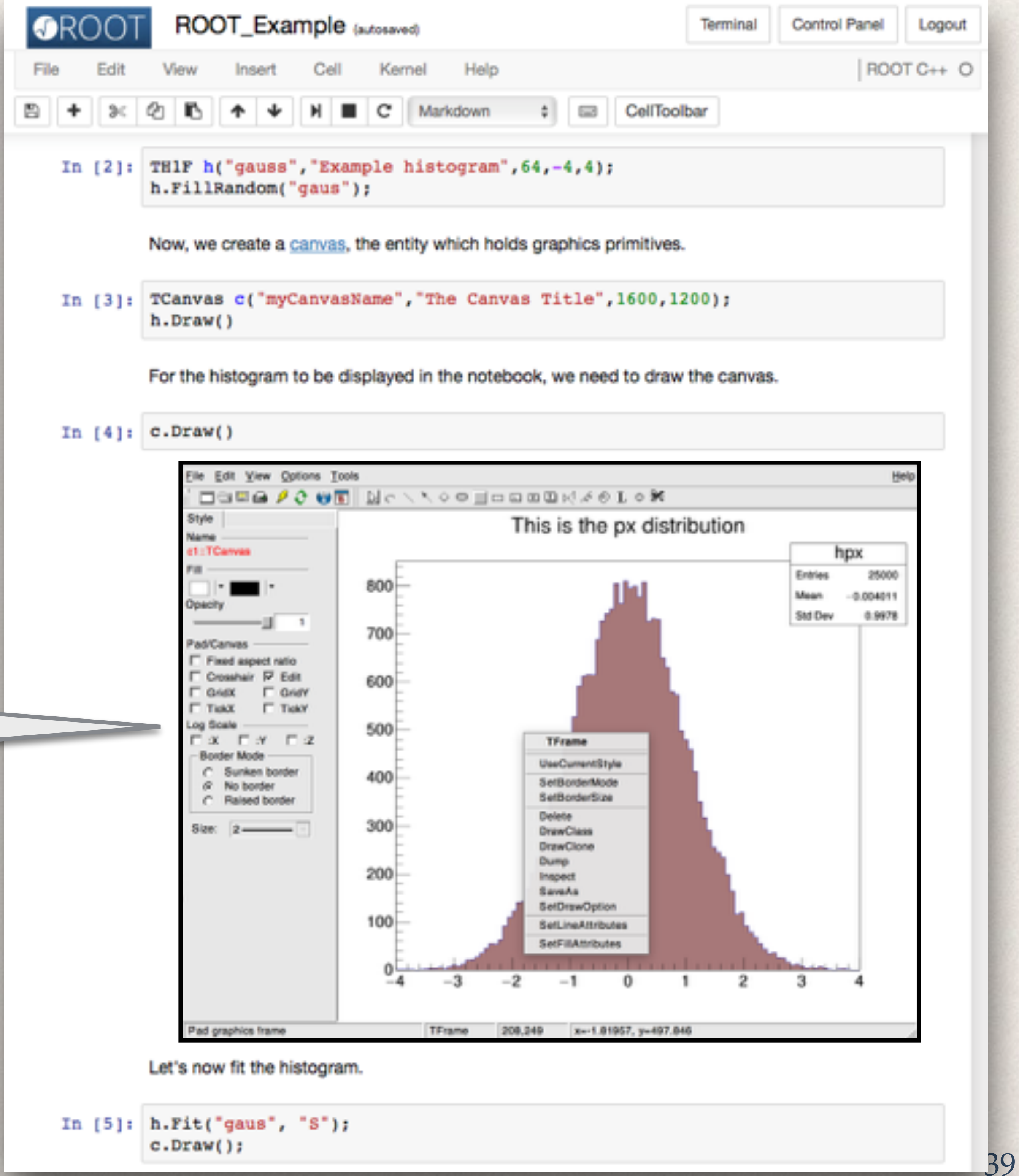
- Converge on a factorization of the logical components of ROOT and produce an analysis of the current interdependencies among them
  - Estimate the effort needed to reduce the module coupling
  - Demonstrate that the coarse grained factorisation is possible in our build system
- Develop model for building/installing modules on demand and evolve ROOT into BOOT (à la R)
- Decide what distribution mechanism for the single packages is adequate for our needs
  - Collaborate with the HSF packaging working group

ROOTbooks

- Integration of ROOT and Jupyter notebooks is well advanced, although there is still some work to do in the following areas:
  - Disseminate and advertise the use of ROOTbooks
    - produce ROOTbook tutorials, examples
    - use of Binder as a try-out demo
  - Make the JS visualization the default and improve it
    - save as button, back to initial state button, save 3D geometry as image
  - Add the R magic to use ROOT-R and combine it with Python and C++
  - Rollback feature for cells (specially for C++)
  - Cling kernel integrated in JupyROOT
- Demonstrate the possibility to integrate with the **Akhet** VNC system

ROOTbooks: Akhet VNC

Embedding in the notebook a fully interactive 'window' to a graphics objects running in the sever.



```
In [2]: TH1F h("gauss","Example histogram",64,-4,4);
        h.FillRandom("gaus");
```

Now, we create a canvas, the entity which holds graphics primitives.

```
In [3]: TCanvas c("myCanvasName","The Canvas Title",1600,1200);
        h.Draw()
```

For the histogram to be displayed in the notebook, we need to draw the canvas.

```
In [4]: c.Draw()
```

Let's now fit the histogram.

```
In [5]: h.Fit("gaus", "S");
        c.Draw();
```

ROOT as-a-service

- Deliver in 2016 a **pilot service** that is able to serve requests for a medium number of users (O(100)). These could be the milestones to achieve:
  - Consolidate the single-node prototype
  - Crash test it: feedback from early users
  - Puppetise it for replication
  - Allow the user to customize their container through JupyterHub forms, e.g. with respect to the LCG release to be used.
  - Security check
  - Evolve to a distributed setup where containers are created in multiple VMs.
  - Service monitoring and management (e.g. clean dead sessions)

Documentation

- Website
  - “Try me” Binder button for ROOTbooks in front page
- Doxygen
  - Finish it for good and improve the current graphical look and feel
- Tutorials
  - Entirely review the full set of tutorials, modernize them, eliminate non-didactical ones, add new from common questions
- ROOTbooks
  - Migrate tutorials to ROOTbooks when possible/useful
  - Transform ROOT primer into a ROOTbook - with input data
- User Guides
  - Update the revise the User Guide and Topical Manuals
- O’Reilly book
  - If additional effort is found, write a O’Reilly book for ROOT

New Platforms

- Make ARM64 and PPCle 1st class citizens
  - Broken tests have to be fixed and kept green
- Support for new ABI coming with GCC 5
- Windows (Visual Studio 2015)
  - New (latest) version of llvm/clang
  - Have ROOT 6 fully ported and running
  - Help porting DAVIX to Windows

Infrastructure

- Jenkins
  - new Jenkins build nodes (e.g. 32 bit nodes, …)
  - proper Jenkins report emails
- Migrate RootTalk forum to the *discourse* platform
- Coding guidelines checker
  - Define new coding guidelines
  - Implement automatic checker with Jenkins

Conferences/Workshops

- ACAT, 18-22 January 2016, Valparaíso
  - 1 plenary, 2 parallel contributions
- CHEP, October 8-9 2016, San Francisco
  - Expected several contributions

Tutorials and Courses

- Continue to guarantee the Summer Student lectures
  - Factorize the introduction out and create a second lecture made of advanced topics
- Deliver advanced training to Universities
  - Hamburg
  - Milan grad course
- CERN Technical Training
  - Preparation of the proposed courses

External Contributions

- ROOT has currently many external contributors:
  - in 2015 we had commits from 57 different authors
- Active Collaborate with us and Ideas web page
- Interest of other projects to collaborate with ROOT
  - DIANA
    - Data Intensive ANAlysis, 4-year NSF funded
    - Focus on analysis software, including ROOT and its ecosystem
    - Three primary goals: performance, interoperability, support for collaborative analysis
  - Other initiatives in the pipeline
- Need to integrate these collaborations and achieve a coherent program of work