



## *art news*

Marc Paterno

*art* Stakeholder Meeting

29 Sept 2016

# Breaking news in ROOT 6.x

- As of 10:10 CDT today, Chris Backhouse's implementation for parsing arithmetic expressions in GDML is in the ROOT repository.

```
* f8233c0 - (origin/master, origin/HEAD) Mention R_DEPRECATED, root-config and -Wno-deprecated-declarations. (7 minutes ago) <Axel Naumann>
* 7de677a - More auto_ptr => unique_ptr replacements. (14 minutes ago) <Axel Naumann>
* d0d8975 - Unused param. (14 minutes ago) <Axel Naumann>
* 145097d - Fix spelling of ROOT_VERSION_CODE. (15 minutes ago) <Axel Naumann>
* e1bfd91 - Use JsMVA repository in root.cern.ch (2 hours ago) <Enric Tejedor Saavedra>
* ba60aeb - Replace more auto_ptrs, silence a few -Wno-deprecated-declarations. (3 hours ago) <Axel Naumann>
* 89490b4 - TMVA: small fix in DataLoader to print message in logger (3 hours ago) <Omar Zapata>
* 5a8c67d - Merge branch 'cjbackhouse_gdml_consts' (4 hours ago) <Andrei Gheata>
* 1aea6f2 - Parse arithmetic expressions in GDML files again. (4 hours ago) <Christopher Backhouse>
* 99c55ea - More TSpectrum examples (5 hours ago) <Olivier Couet>
* 236071c - (HEAD, master) Make sure IMT tutorials are vetoed if imt=OFF (5 hours ago) <Enric Tejedor Saavedra>
* 9b2b4e3 - Mention that Draw() will return the Drawable&. (5 hours ago) <Axel Naumann>
* 97fc150 - Fix ownership doc. (5 hours ago) <Axel Naumann>
* de004f3 - Move TThreadPool.h. (6 hours ago) <Vassil Vassilev>
* 11378ff - Remove -Wno-deprecated-declarations from top-most flags. (8 hours ago) <Axel Naumann>
* 936e64b - Remove -Wno-deprecated-declarations from root-config --cflags: we want that to work. (8 hours ago) <Axel Naumann>
* f04ea5b - Deprecate functions that already Error() since years. (8 hours ago) <Axel Naumann>
* 0ca2229 - Add version number to R_DEPRECATED(6,10,"why"). (8 hours ago) <Axel Naumann>
* 4975d39 - Rest in peace, System/390. (8 hours ago) <Axel Naumann>
```

- We expect this to go into the 6.08 release.
- As of 13:45 CDT, the tag v6-08-00-rc1 was created.

## More ROOT 6.x news

- We are continuing to work on a build of *art* based on the ROOT 6.08 release candidate. The biggest change is an updating of the internal clang version used by ROOT.
- Several problems have been identified and overcome
  - name mangling problem in clang; bug report filed, *gallery* code modified to work around failure in interactive use of ROOT
  - use of PyROOT fails unless ROOT's clang is told
    - about explicit template instantiations
    - to explicitly autoparse (!) the *gallery* Event header
  - Some exception catching in ROOT macros fails (possibly because of the name mangling bug); workaround is to catch *cet::Exception* rather than *art::Exception* when you encounter this problem. This may have been fixed in the ROOT master.

# ROOT 6.x on Apple operating systems

- The only compiler officially supported for building ROOT 6 on Apple operating systems is Apple's version of LLVM/Clang (as shipped with XCode).
  - this differs from ROOT 5, for which support of GCC was official
- XCode versions depend on OS version:
  - Sierra (not yet available to FNAL), El Capitan (only supported with SIP disabled) both use XCode 8.x.
  - Yosemite uses v7.x, will not move to 8.x.
- We have been lucky so far in building with GCC; unclear how long this will last.
- *Should we move to using only native LLVM/Clang on Apple OSes, or should we stay with GCC until it really does fail?*

# Proposal for support of Docker

- Hosted at <https://hub.docker.com/u/fnalart/dashboard/> is a sample of Docker images containing *distribution bundles* as found on SciSoft. Each image contains an already-setup environment.
- We propose to make creating such images part of our regular release cycle. Note: some tuning of what we deliver is needed.
- Users can install these images on their own machines (one must install Docker itself on one's own).
  - for use as a base image in a personalized Dockerfile (e.g., to add additional system utilities like *emacs*)
  - for direct use, with the user managing containers directly
  - by using throw-away containers, and sharing files with the host filesystem
  - or by any other means that Docker provides and the user can think of.
- We will create and make public the images.
  - We can't provide additional tooling on top of the images.