Who didn’t have problems installing ROOT: evolution of ROOT package management

Oksana Shadura, Brian Paul Bockelman, Vassil Vassilev
University of Nebraska–Lincoln, USA
oksana.shadura@cern.ch, bbockelm@cse.unl.edu, vvasilev@cern.ch

ROOT CMake: motivation & current status

Motivation
1. **Build component/package of ROOT on top of pre-configured/build ROOT**
2. Make a ROOT packaging more flexible and less monolithic
3. Develop ROOT-aware dependency manager

Current status & troubles

Typical situation: **user wants to build ROOT foundation libraries and enable TMVA package → enabling TMVA option, user will install other extra** libraries.
ROOT has around 110 CMake options, which are:
1. build features - 10% [cxx11, cxx14, pch, cling.]
2. build options - 90% [gsl_shared, xml.]

CMake options can be unclear with no naming convention: do you know what will be enabled with xml option?

As a solution, we propose to introduce the concept of a ROOT subpackage.

Layering ROOT: design goals

1. Arrange existing ROOT components into layers. For instance, core → mathcore → mathmore.
2. Allow each layer (subpackage) can be enabled/disabled.

This is implemented as an overload of CMake add_subdirectory with iteration loop through ROOTPackageMap.cmake (similar to a package database).

Similar implementation also exists in LLVM project:

- addllvm_subdirectory()
- addclang_subdirectory()
- addcling_subdirectory()

We propose a new way of organisation of ROOT build options → having a possibility to enable one ROOT library with its dependencies

Simplified example of package map, allowing to enable Base, IO, Castor subpackages:

```
>=make ./
>=DROOT_ENABLE_PACKAGES=Base
>=DROOT_ENABLE_PACKAGES=IO
>=DROOT_ENABLE_PACKAGES=Castor
```

User can simply call:

- `>make $ROOT_DIR -DROOT_ENABLE_PACKAGES='Genvector'`:
  Enables only Genvector and dependencies
- `>make $ROOT_DIR -DROOT_ENABLE_PACKAGES='RDataFrame;RVec;VDT'`:
  Enables only RDataFrame, vdt and dependencies and build on top of previous build
- `>make $ROOT_DIR -DROOT_ENABLE_PACKAGES='ALL'`:
  Enables all other packages and dependencies

Benefits for users

This work was supported by the National Science Foundation under Grant ACI-1450323.