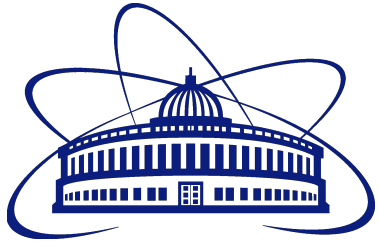
Automatic Build and Distribution of Software for MPD of the NICA Project

Ján Buša Jr.[1,2]
Slavomír Hnatič[1]
Oleg Vasilievich Rogachevsky[1]

[1]Joint Institute for Nuclear Research, Dubna, Russia

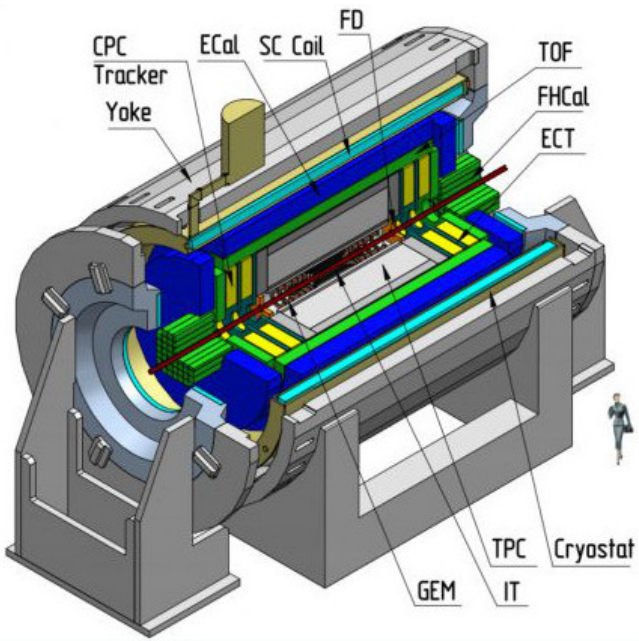
[2]A.I. Alikhanyan National Science Laboratory (Yerevan Physics Institute), Yerevan, Armenia

Yerevan
22. 10. 2024



Multi-Purpose Detector

- part of **N**uclotron-based **I**on **C**ollider f**A**cility
- **M**ulti **P**urpose **D**etector – main detector under costruction
- MPDRoot – library for reconstruction of tracks and physical analysis (`http://mpdroot.jinr.ru/`)



Just Install FairRoot and Build MPDRoot...

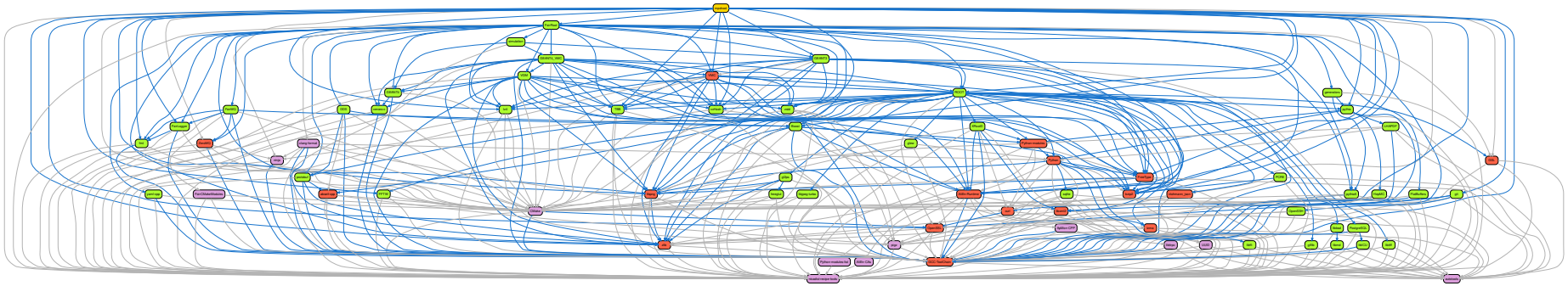
How to install MPDRoot (each user had to build own version):

- install all basic packages (about 70);
- download, patch, and build FairRoot;
- download and build mpdroot from sources. In-source build highly recomended.

It's really "easy":

- basic packages are often not available in compatible versions
- patching FairRoot can be tedious (during last minor version change more than 100 lines of code)
- building FairRoot takes about 2 – 4 hours on a regular computer
- in-source build guides users to use improper paths when linking own code
- users tend to use same version of mpdroot for long time

MPDRoot and Friends



```
Currently Loaded Modulefiles:
 1) libffi/v3.4.6-1          13) HepMC/v2.6.11-2          25) libjpeg-turbo/v3.0.3-1   37) zstd/v1.5.6-1            49) GEANT4_VMC/v6.6.2-1      61) Catch2/v3.6.0-1
 2) bzip2/v1.0.8-1           14) Python-modules/v2024.06-1 26) giflib/v5.2.2-1         38) protobuf/v27.1-1         50) simulation/v2024.06-1    62) podio/v1.1.0-1
 3) zlib/v1.3.1-1            15) pythia/v8.3.12-1         27) TBB/v2021.12.0-1         39) libxml2/v2.13.0-1        51) Boost/v1.83.0-2          63) HepPDT/v3.04.01-1
 4) GCC-ToolChain/v13.2.0-1  16) pythia6/v6.4.28-2        28) libtiff/v4.6.0-1         40) XRootD/v5.6.9-2          52) ZeroMQ/v4.3.5-2          64) HepMC3/v3.3.0-1
 5) AliEn-Runtime/v2024.03-2 17) generators/v2024.06-1    29) PCRE/v8.45-1             41) glew/v2.3.0-pre-1        53) FairMQ/v1.8.4-2          65) LCIO/v2.22.1-1
 6) OpenSSL/v3.3.1-1         18) fmt/v10.2.1-1            30) nlohmann_json/v3.11.3-2  42) gl2ps/v1.4.1-1           54) OpenSSH/v9.7.1-2         66) EDM4hep/v0.99.1-1
 7) libpng/v1.6.43-2         19) FairLogger/v2.0.0-1      31) freeglut/v3.6.0-1        43) ROOT/v6.32.06-1          55) yaml-cpp/v0.8.0-2        67) DD4hep/v1.30.0-pre-1
 8) FreeType/v2.13.2-2       20) libmd/v1.1.0-1           32) abseil-cpp/v20240116.2-1 44) GEANT3/v4.4-1            56) DDS/v3.10.0-1            68) Eigen3/v3.4.0-2
 9) sqlite/v3.46.0-1         21) curl/v8.8.0-1            33) xxHash/v0.8.2-2          45) xerces-c/v3.2.5-2        57) FlatBuffers/v24.3.25-1   69) ACTS/v37.0.1-1
10) lzma/v5.6.2-1            22) GSL/v2.8-1               34) libICU/v75.1-1           46) GEANT4/v11.2.1-1         58) git/v2.45.2-1
11) Python/v3.12.4-1         23) lz4/v1.9.4-2             35) PostgreSQL/v16.3-1       47) VGM/v5.3-1               59) FairRoot/v18.6.10-2
12) LHAPDF/v6.5.2-2          24) FFTW/v3.3.10-2           36) libbsd/v0.12.2-1         48) VMC/v2.0-2               60) mpdroot/v24.09.24-1
```

59 packages are needed to run MPDRoot, about 70 to build it

Centralized Build of MPDRoot

Requirements:

- we want to deliver latest version of MPDRoot to the users
- all dependencies have to be statisfied and no version clashes are allowed
- results need to be repeatable

Solution (in cooperation with M. Vaľa, UPJŠ Košice, Slovakia):

- CVMFS – robust distribution of software not only between clusters/supercomuters
- aliBuild – set of tools for building software together with its dependencies
- toolbox – rootless wrapper for podman (docker)
- EnvironmentModules – environment enabling loading multiple versions of the same software with its dependencies

CernVM-FS
https://cernvm.cern.ch/fs/

- CMVMFS is a **read-only** file system
- anyone can mount the shared directories to its computer
- extensively uses caching of the files on local computer
- only used (accessed) files are downloaded. That makes first load slow but following are as fast as local usage
- since it is centralized, changes on server are directly available to all users

Toolbox
https://github.com/containers/toolbox

- wrapper for podman (rootless docker alternative) – ordinary user can use it
- allows user to "work" inside virtual machine – no connection to the users system
- we build primarly for Alma 9 version but still support CentOS7 due to our supercomputer Govorun requirements
- basic system image is fixed without any updates and therefore all results are reproducible
- supported on most linux-based systems. On Windows and MacOS one can use VMware (Fusion)

Environment Modules
https://modules.readthedocs.io/

- started as a project to separate various versions of same software
- simple "module files" contain paths to the software together with list of dependencies
- to use some software one has to load it first via command `module add mpdroot/latest`. Paths to executables as well as libraries are then added to the environment. All necessary dependencies are loaded as well.

Example of module file (excerpt):

```
module load GEANT4/$GEANT4_VERSION-$GEANT4_REVISION
module load ROOT/$ROOT_VERSION-$ROOT_REVISION
setenv MPDROOT \$PKG_ROOT
prepend-path ROOT_INCLUDE_PATH \$PKG_ROOT/include
prepend-path LD_LIBRARY_PATH \$PKG_ROOT/lib
```

aliBuild
https://alisw.github.io/alibuild/

- created as a solution to the ALICE experiment software distribution problem
- automatic build of software and all its dependencies
- reusing already built packages
- if some package was changed, all dependants are rebuild
- apropriate modules are created and numbered automatically
- can get messy if packages change frequently and are not cleared
- based on recipes – guides how to build some package

The MPDRoot Recipe

```
package: mpdroot
version: "%(commit_hash)s"
tag: "dev"
source: https://git.jinr.ru/nica/mpdroot.git
requires:
  - FairRoot
  - eigen3
  - nlohmann_json
  - VMC
  - GSL
---
unset SIMPATH
export MPDROOT=$INSTALLROOT

cmake $SOURCEDIR                                                              \
      ${BOOST_ROOT:+-DBOOST_ROOT=$BOOST_ROOT}                                 \
      ...
      ${LIBXML2_ROOT:+-DPC_LIBXML_LIBDIR=$LIBXML2_ROOT/lib}                   \
      ${NLOHMANN_JSON_ROOT:+-DNLOHMANN_JSON_ROOT=$NLOHMANN_JSON_ROOT}         \
      -DCMAKE_INSTALL_PREFIX=$MPDROOT
cmake --build .  -j$JOBS install

# Modulefile
MODULEDIR="$INSTALLROOT/etc/modulefiles"
MODULEFILE="$MODULEDIR/$PKGNAME"
mkdir -p "$MODULEDIR"
generate-module --project=NICA --bin --lib --path > $MODULEFILE
cat >> "$MODULEFILE" <<EoF
setenv MPDROOT \$PKG_ROOT
setenv MPDROOT_MACROS \$PKG_ROOT/macros
...
prepend-path ROOT_INCLUDE_PATH \$PKG_ROOT/include
EoF
```

Wrapping-Up

User's view:

- `toolbox enter a9-nica`
- `source /cvmfs/nica.jinr.ru/sw/login/login.sh`
- `module add mpdroot/v24.09.24-1`

Developer's view:

- `toolbox enter a9-nica`
- `source /cvmfs/nica.jinr.ru/sw/login/login.sh`
- `module add mpddev/latest`
- `export MPDROOT=/path/to/install`
- `git clone path_to_mpdroot && mkdir build && cd build && cmake .. && make install`

Conclusions

- for ordinary users build of mpdroot takes no time
- latest version of software is instantly available to everyone
- it is easy to return to previous versions (including dependencies)
- all results are repeatable since we store basic system image as well as all recipes
- most of the packages we build from scratch (including low level, s.a., CMake, GCC, OpenSSL)
- with recipes new dependencies are distributed among developers without necessity for them to install anything (it is done automatically)

Future Work

- follow updates of our dependencies and modify build accordingly during the run of experiment and after (30+ years)

- allow users to download "offline" version of the software – rpm
- support build for LTS versions Ubuntu (+ deb for offline)
- support MacOS

- find a way how to replace a broken package without rebuilding all