Advanced European Infrastructures for Detectors at Accelerators

DD4hep

Starter's Guide

M.Frank
CERN, 1211 Geneva 23, Switzerland

Abstract

This starter's Guide contains the basic information about `DD4hep` . This is the license information, instruction to obtain and build the software as well as suggestions of further reading concerning the various `DD4hep` subpackages.

Document History		
Document version	**Date**	**Author**
1.0	**19/07/2015**	**Markus Frank CERN/LHCb**

Contents

1 `DD4hep` Project Information

1.1 The `DD4hep` Project Web

The project's welcome page can be found under the folling URL:
http://aidasoft.web.cern.ch/DD4hep

Please note:
To access the CERN intranet, to create new bugs in the `DD4hep` bug tracker etc, you require a CERN account, but you can create an external account very easily following the instructions in:
https://account.cern.ch/account/Externals/RegisterAccount.aspx

1.2 The `DD4hep` Developers Mailing List

Mailing list to contact developers is: `dd4hep-developers@cern.ch`.
The mailing list archives can be consulted using the following URL:
`https://groups.cern.ch/group/dd4hep-developers/Lists/Archive/100.aspx`.

1.3 The `DD4hep` Issue Tracker

Bug tracking and issue solving as well as feature requests may be addressed to the developers using the following web-portal:
`http://sft.its.cern.ch/jira/browse/DDFORHEP`.

Please note:
Please stay calm if you do not receive a response within the next 25 nano-seconds. Most of the contributors also have other assignments!

1.4 License

`DD4hep` is made available under the GPL license. For full details see the file LICENSE in the `DD4hep` distribution or:
http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/LICENSE

2 Building and Installing `DD4hep`

The `DD4hep` software is distributed for each release in the form of

- Precompiled tar-balls, where users only untar the downloaded software.

- As source distribution, which may be downloaded from the central repository at Desy.

2.1 Release Versioning

Currently `DD4hep` is released several times a year depending on the importance of new functionality. The versioning schema consists of two digits in the form:

$$vXX-YY \ ,$$

where XX denotes the major release tag and YY the minor release tag. We currently do not support patches, since the project is still quite young and the binary compatibility is broken with each release. Code changes, which require clients to modify existing code are getting rare, since the main package `DD4hep` is maturing.

2.2 Supported Platforms

List of supported platforms (operating system + compiler):

- SLC 6

- MacOS (low priority though)

Declined operating systems:

- Microsoft Windows.

In principle DD4hep does not depend dramatically on specific operating systems. In principle any compiler modern enough to support **C++11** may should be capable of building DD4hep . There are some **C++ABI** issues, which may have to be sorted out for non-gcc platforms, but most of this work was done for MacOS and it should be feasible to solve these problems for other platforms as well.

2.3 README

First glance information may be taken from the readme file at the following location:
http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/ReadMe.txt

2.4 Prerequisites

DD4hep depends on a number of external packages. The user will need to install these in his/her system before building and running the examples. Depending on the usage of DD4hep and it's subpackages software of other projects is used. The configuration options are described later in this document.

- DD4hep uses the $CMake$ build system. A recent version (version 2.8 or higher) of $CMake$ [4] must be installed.

- DD4hep itself depends on the ROOT geometry package tgeo and in turn on the basic ROOT libraries [3, **?**]. Version 5.30 or higher is required.

- DD4hep uses the Boost-Spirit [5] implementation to interprete input and options.

- Optionally DD4hep can use $Xerces - C$ [6] as a parser for XML input. By default it uses $TinyXML$ [**?**], which is included in the DD4hep release.

- The subpackages for alignment DDAlign , conditions data DDCond and event display DDEve do not require additional external libraries.

- The simulation extensions extension DDG4 in addition depends on a valid Geant4 [**?**] installation. Version 9 and 10 of Geant4 is supported.

If DD4hep is not installed in the system name-space (/usr/bin, /usr/lib, etc.) any number of DD4hep versions can exist in parallel.

2.5 Package Contents

The DD4hep project consists of several packages. The idea has been to separate the common parts of the detector description toolkit from concrete detector examples.
DDCore:
Contains the definition of the basic classes of the toolkit: Handle, DetElement, Volume, PlacedVolume, Shapes, Material, etc. Most of these classes are handles to ROOT TGeom classes. The package provides the basic mechanisms for constructing the Generic Detector Description Model in memory from XML compact detector definition files.

DDG4:
Is the package that contains the conversion of DD4hep geometry into Geant4 geometry to be used for simulation.

DDAlign:
The `DDAlign` toolkit implements a modular and flexible approach to introduce and access the alignment parameters in `DD4hep` geometry descruptions.

DDCond:
Support for conditions data for `DD4hep` .

DDEve:
`DDEve` provides an event display for detector geometries implemented using DD4hep. `DDEve` hereby takes advantage of the TEve toolkit naturally provided by the ROOT framework like the ROOT geometry toolkit TGeo.

examples:
Contains different detector examples (CLICSiD, AlignDet, ClientTests) together with example applications to illustrate the their usage for simulation, reconstruction and visualization with an emphasis to minimize software dependencies. For instance, the simulation application example will require to compiler and link against Geant4 but not the reconstruction application.

2.6 Installing `DD4hep` from Precompiled Binaries

The binaries are available for downloading from afs under the following location: `to be done.....` Once downloaded you need to unzip and de-tar the file. For example, if you have downloaded the debug version of DD4hep v00-13 for Linux-SLC6:

```
% gunzip dd4hep-v00-13-slc6-gcc4.8-dbg.tar.gz
% tar xvf dd4hep-v00-13-slc6-gcc4.8-dbg.tar.gz
```

This will create the directory root. Before getting started read the file ReadMe.txt. Also, read the Introduction chapter for an explanation of the directory structure.

TO BE DONE !!!

2.7 Access and Build `DD4hep` from the Repository

The following sections explain how to obtain a usable binary build of the `DD4hep` software.

2.7.1 Downloading the Code from the Repository

You can use the Subversion (svn) source code version control system and check out the most recent source. The compressed file is a one time only choice; every time you would like to upgrade you will need to download the entire new version.

The `DD4hep` code is accessible in anonymous read-only mode by anybody from the svn repository at **Desy**:

```
% svn co https://svnsrv.desy.de/basic/aidasoft/DD4hep/trunk
% mv trunk DD4hep_src
```

2.7.2 Installing the Source from Subversion

This paragraph describes how to checkout and build DD4hep from a checkout using subversion for Unix-like systems.

```
% mkdir DD4hep_install
% mkdir DD4hep_build
% export DD4hepINSTALL=`pwd`/DD4hep_install
% cd DD4hep_build
% cmake -DCMAKE_BUILD_TYPE=Debug \
```

```
        -DCMAKE_INSTALL_PREFIX=\${DD4hepINSTALL} \
        [option [option]] ...
% make install
```

Possible additional options for the cmake build system are described in the following section.

Note:

You might have to update your environment beforehand to have all needed libraries in the shared lib search path (this will vary with OS, shell, etc.) e.g:

```
% . /data/ilcsoft/geant4/9.5/bin/geant4.sh
% export CLHEP_BASE_DIR="/data/ilcsoft/HEAD/CLHEP/2.1.0.1"
% export LD_LIBRARY_PATH="$CLHEP_BASE_DIR/lib:$LD_LIBRARY_PATH"
```

2.7.3   CMake Configuration Options

We describe in the following only options, which are unique to the build of `DD4hep` . Boolean values are either `ON` or `OFF` to enable or disable a feature.

DD4hep_ROOT=<directory>
  Funtion: Location of the `DD4hep` installation directory.

DD4HEP_USE_GEANT4=<boolean>
  Default: OFF
  Function: Steer the building of the Geant4 interface `DDG4` .
  Optional: By default no simulation support is enabled.
  Remark: If enabled the variable **Geant4_ROOT**=<directory> must point to a valid Geant4 installation directory.
  Remark: if Geant4 was built with external CLHEP, add: -DCLHEP_DIR=_path_to_CLHEP

DD4HEP_USE_BOOST=<boolean>
  Default: OFF
  Function: Steer the usage of **boost::spirit** as option parser for `DDG4` .
  Optional: Optional setting with the following caveat.
               De-facto mandatory for the python interface to `DDG4` as well as Geant4 interactive use of the of the `DDG4` components.
  Remark: If enabled, boost software must be installed and found within the CMAKE_MODULE_PATH.

DD4HEP_USE_XERCESC=<boolean>
  Default: OFF
  Function: Steer the usage of **Xerces-C** as xml file parser.
  Optional: Default is set to use **TinyXML**.
  Remark: If enabled, the variable XERCESC_ROOT_DIR=<directory> must point to a valid **Xerces-C** installation directory.

DD4HEP_USE_LCIO=<boolean>
  Default: OFF
  Function: Steer the usage of **lcio** I/O engine used by the linear collider community.
  Optional: By default no **lcio** support is enabled.
  Remark: If enabled, the variable LCIO_DIR=<directory> must point to a valid **lcio** installation directory.

BUILD_TESTING=<boolean>
  Default: OFF
  Function: Build `DD4hep` tests.

DD4HEP_USE_CXX11=<boolean>
  Default: OFF
  Function: Steers the compilation to use the **C++11** standard.
               Otherwise the **C++98** standard is enabled by default.

DD4HEP_USE_PYROOT=<boolean>

Default:	OFF
Function:	Steer the usage of **PyROOT** `DD4hep` detector construction.
Remark:	**DEPRECATED**. Purely historical and unsupported.

2.7.4   Run-time Environment

During the build a shell script is generated and copied to the installation area called "thisdd4hep.(c)sh". To setup the runtime and build environment for layered software using `DD4hep` this script musty be source'd.

2.7.5   Using `DD4hep` in Layered Software as a Client

It is necessary to setup the `DD4hep` runtime environment before the configuration of the client software. In the installation directory the cmake build instructions `DD4hepConfig.cmake` are generated, to be used for the cmake configuration of the client software. Cmake should pick-up the build instructions automatically.

2.7.6   Relocation of the Installation directory

Once build, `DD4hep` can be tar'ed and/or moved to any other directory. The instructions for the usage by layered client software stay intact.
   TO BE DONE !!!!!

2.7.7 Example Build Script

The following commands may be used to build `DD4hep` from the svn repository:
http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/example_build_script.sh.

```
 1 #!/bin/bash
 2 #
 3 # ====  User configuration =======================================
 4 # Set here the proper installation directories of the pre-requisites
 5 # ================================================================
 6 SW=~/SW
 7 MY_ROOT=${SW}/root_v5.34.25_dbg
 8 MY_GEANT4=${SW}/g4_10.01.p02_dbg
 9 MY_LCIO=${SW}/lcio/v02-04-03
10 MY_XERCES=${SW}/xerces-c-3.1.1
11 MY_DD4HEP=`pwd`/DD4hep_tmp2
12 # ================================================================
13 # From here on all should be box-standard
14 # ================================================================
15 . ${MY_ROOT}/bin/thisroot.sh
16 . ${MY_GEANT4}/bin/geant4.sh          # Remove line if no DDG4
17
18 mkdir -p ${MY_DD4HEP}/build
19 mkdir -p ${MY_DD4HEP}/install
20 cd ${MY_DD4HEP}
21 svn co https://svnsrv.desy.de/basic/aidasoft/DD4hep/trunk
22
23 cd    build
24 cmake -DCMAKE_BUILD_TYPE=Debug                                \
25       -DCMAKE_INSTALL_PREFIX=${MY_DD4HEP}/install             \
26       -DCMAKE_MODULE_PATH=${MY_DD4HEP}/install                \
27       -DDD4HEP_USE_XERCESC=ON -DXERCESC_ROOT_DIR=${MY_XERCES}  \
28       -DDD4HEP_USE_BOOST=ON -DDD4HEP_USE_GEANT4=ON -DGeant4_ROOT=${MY_GEANT4} \
29       -DDD4HEP_USE_LCIO=ON -DLCIO_DIR=${MY_LCIO}              \
30       ../trunk
31
32 # Build and install
33 make -j 6 install
34
35 # Move to installation and setup runtime environment
36 cd ${MY_DD4HEP}/install
37 . bin/thisdd4hep.sh
38
39 export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:${MY_LCIO}/lib
40
41 # Run Examples....
42 python examples/DDG4/examples/CLICSidSimu.py
```

Notes:

- If Geant4 is not available or undesired remove lines 8 and 28.

- If you want to use **TinyXML** instead of **Xerces-C** remove lines 10, and 27.

- If LCIO I/O is not available or undesired remove lines 9, 29 and 39.

3 DD4hep Documentation

General build information is provided by this document:
http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DD4hepStarter.pdf.
In addition all sub-packages of the DD4hep toolkit provide documentation in the form of manuals.
Suggestions (or better improvements) of user documentation of packages being validated are highly welcome. Please contact the developers using the DD4hep mailing list. If a package is under development and not offered to the public, please ignore the package and it's errors unless you are eager to contribute.

3.1 DD4hep Users Guide

The DD4hep guide is part of the distribution and the svn repository.
Please see http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DD4hepManual.pdf.
Author: M.Frank

3.2 DDAlign Users Guide

The DDAlign guide is part of the distribution and the svn repository.
Please see http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DDAlignManual.pdf.
Note: This package is under development. The guide is very rudimentary.

3.3 DDCond Users Guide

The DDCond guide is part of the distribution and the svn repository.
Please see http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DDCondManual.pdf.
Note: This package is under development. The guide is very rudimentary.

3.4 DDEve Users Guide

The DDEve guide is part of the distribution and the svn repository.
Please see http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DDEveManual.pdf.
Note: This package is not officially part of DD4hep and in addition under development. The guide is very rudimentary.

3.5 DDG4 Users Guide

The DDG4 guide is part of the distribution and the svn repository.
Please see http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DDG4Manual.pdf.
Author: M.Frank

3.6 DDRec Users Guide

The DDRec guide is part of the distribution and the svn repository.
Please see http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/DDRecManual.pdf.
Author: F.Gaede

3.7 Reference Manual

An up to date build of the code documentation compiled using Doxygen [**?**] can be found at the following URL:

```
http://test-dd4hep.web.cern.ch/test-dd4hep/doxygen/html/index.html
```

3.8 Testing and Nightly builds

A **CDash** installation provides daily information about the the current status of the software in the svn trunk.

4 How to contribute

`DD4hep` was started by me in the year 2012 in response of the needs of the linear collider community (ILD, CLIC) for a flexible detector geometry usable to develop tracking code. This development relatively quickly emerged into desire to actually obtain a complete description of the entire detector spanning not only the geometrical part, but "everything" required to process the data from particle collisions including provision to support all sorts of conditions data and alignment data. Obviously the simulation of particle collisions should more or less come with it for free and - of course - once simulated, the results want to be visualized.

These initial requirement led to a substantial development, which however is not complete and always wants to be improved - and if only to be adapted to new compiler versions. For this reason additional effort from the emerging user community is highly welcome. If you are convinced by the concept, but lack some functionality and are willing to add this as a contribution you are more then welcome. If your changes affect the basic design I would appreciate to be contacted in advance for a brief discussion of your plans.

`DD4hep` is supposed to quickly enable users to develop test and simulate the physics response of new detector models and concepts. Although a certain palette of such subdetectors concepts is provided in the **DDDetectors** package, the palette is not complete. I would like to encourage users to contribute missing subdetector designs such e.g. RICH detectors provided the design is modular and flexible so it can be picked up by the community.

The same argument obvious applies to the development of new or more detailed sensitive detector concepts for the `DDG4` package.

4.1 Contributors

In the file http://svnsrv.desy.de/viewvc/aidasoft/DD4hep/trunk/doc/CREDITS the most important contributors to the DD4hep project and a short description of their work can be found. If you think you should be part of this list, but somehow got forgotten, please contact us.

5 Conclusions

No a lot to be said: **Good luck! ... and have fun!**.

References

[1] DD4Hep web page, http://aidasoft.web.cern.ch/DD4hep.

[2] R.Brun, A.Gheata, M.Gheata, "The ROOT geometry package",
Nuclear Instruments and Methods **A** 502 (2003) 676-680.

[3] R.Brun et al., "Root - An object oriented data analysis framework",
Nuclear Instruments and Methods **A** 389 (1997) 81-86.

[4] Welcome to CMake, the cross-platform, open-source build system, www.cmake.org

[5] Boost C++ Libraries www.boost.org

[6] The Apache Xerces Project - xerces.apache.org

[7] S. Agostinelli et al., "Geant4 - A Simulation Toolkit",
Nuclear Instruments and Methods **A** 506 (2003) 250-303.