Subject: New iLCSoft developers release v01-17-01
Posted by gaede on Fri, 24 May 2013 12:43:55 GMT
View Forum Message <> Reply to Message

Dear all,

a new developers release of ilcsoft (v01-17-01) is available.

It provides many smaller fixes, a number of new features and some new packages:

DD4hep: Geometry description (M.Frank, P.Mato)
SLIC, XercesC, HepPDT, GDML, LCDD - Slic geant4 simulation (J.McCormick)

Please refer to the Release notes below for details.

Use the ilcinstall tool with the appropriate configuration files in order to install ilcsoft v01-17-01.

Reference installation in afs are available at:

/afs/desy.de/project/ilcsoft/sw/x86_64_gcc41_sl5/v01-17-01 # SL5 64bit
/afs/desy.de/project/ilcsoft/sw/x86_64_gcc44_sl6/v01-17-01 # SL6

Please report any problems and questions regarding this release in this forum.

Frank, for the iLCSoft team.

```
###################################
#
#   iLCSoft release - v01-17-01
#
###################################
```

Developers release that provides many smaller fixes and a
number of new features as well as a number of new packages.

===

Changes in v01-17-01 w.r.t. iLCSoft v01-17

===

New packages added to iLCSoft:

GBL: General Broken Lines (K.Kleinworth, Ch.Rosemann)
DD4hep: Geometry description (M.Frank, P.Mato)
SLIC, XercesC, HepPDT, GDML, LCDD - Slic geant4 simulation (J.McCormick)

===============
LCIO: v02-04
===============

- Ch.Grefe, CERN
  added a new python binding: pyLCIO based on the optional ROOT dictionary
  see: $LCIO/examples/python/README for examples on how to use it

- added new method to limit the collections that are going to be read
  from a file:
    LCReader::setReadCollectionNames(const std::vector<std::string>& cn)
  - C++ only so far
  - anajob and dumpevent now read variable LCIO_READ_COL_NAMES with
    space separated list of colection names to read only these

- pointers to LCObjects that are not in the event are now set to NULL

- an Exception is thrown if a null pointer is found in a subset collection
  (e.g. if LCIO_READ_COL_NAMES is set inconsistently such that
  the collection(s) holding the elements is not read)

  - this behavior can be deactivated with setting the environment variable
    LCIO_IGNORE_NULL_IN_SUBSET_COLLECTIONS=1
    if really needed, e.g. to read inconsistent or corrupt files

- added build option for C++ only builds
  using pre-genetarated header files:

```
cmake -DLCIO_GENERATE_HEADERS=OFF ..
```

  - note: header files will have to be updated manually by developers
    after changing the class layout

- some minor fixes:
  - fix missing include issue in UTIL/LCIterator.h for latest ubuntu
  - update freehep-sio dep to 2.3
  - enforce cmake version >= 2.8

================

Marlin: v01-05

===============

- new optional global parameter: "LCIOReadCollectionNames"
  allows to restrict the collection that are read from the
  LCIO file(s) to the ones given
  (needs LCIO v02-04)

- new command line option (H.Perrey, DESY):
  -u : consistency check and updated version of xml file

- fix problem with ROOT seg fault at program end:
  do not call dlclose on shared libraries anymmore
  (A.Sailer)

=================

CED: v01-09

=================

- turned off C++ name mangling in client library libCED.so/dylib
  to facilitate calling ced from python

===================

CEDViewer: v01-07

===================

- changed calling attributes of ced_hit_ID to newer version
  -> needed for CED v01-09 which has no function overloading

in library anymore (c-style)

- changed the default conventions for the track state to be drawn. Now the following processor options are valid
  -1: do not draw helix
  0: default track state (the 0-th element in the track state vector)
  (the rest options as before)
  1: AtIP
  2: AtFirstHit
  3: AtLastHit
  4: AtCalorimeter
  If no track state were found, exit with an error

- avoid wrong values for the helix parameters for the case of small curvature or zero magnetic field by assigning large value of Pt to the track

- update for the multi-module support via generic GEAR interface

===================
Clupatra: v00-10
===================

- bug fix: remove sqrt from computation of r_phi_res^2 for finding best hit
- included multi-module TPC support via GEAR interface
- ignore hits that are outside of the TPC range (possible in data)
- added new processor FixCellIDs_Error_TestBeam (based on the original FixCellIDs)
  that sets the correct hit CellIDs and hit uncertainties for test beam
data
- added examples for test beam data. Look at examples/README for more details

===================
LCTuple: v01-03
===================
- added support for TrackerHits

================
KalDet: v01-12
================

- added detector geometry for lctpc which is now based on ild classes. Notes:
- it is compatible with Clupatra/MarlinTrk
- the correspondence between row number and module is represented in a different way
compared to the original class GearTPCKalDetector. In the current implementation,
row numbers are global, i.e. rows of different modules but with the same center are
now the same rows! This is needed to unify the measurement layers that are
combinations of several modules.

====================

MarlinTPC: v00-13

====================

- A. Muennich: added preparation processors for Clupatra test beam data usage
added a processor ADC2PrimaryElectronFactorCalculator (needs conditions object transformation)
added a helper tool to calculate intersection between circle and track in global coordinates
Reimplementation of the splitting in the PulseFinder:
bug fix to HitCopier
- F. Mueller: added two more methods to calculate pulse time
correction of hit error calculation
- S. Caiazza: added documentation
- F. Gaede: added extension to CED
- I. Heinze: Extended CutApplicationProcessor so that tracks with more than one hit per row can be cut out.
- O . Volynets introduced new track description with MarlinTrk
- M. Killenberg: Wrote documentation for the IntersectionCalculator
- Ch. Rosemann: added first version of straight line fitting in row based TPC with general broken lines
HitCopier added

+ Bug fixes by I. Heinze, A. Muennich, M. Killenberg, T. Krautscheid, O. Volynets, F. Mueller, F. Gaede, Ch. Rosemann

===================

MarlinTrk: v01-11
===================

- Improved readability of the MarlinTrkUtils functions
- Added LCTPC-specific modifications:
  - geometry building is now based on the detector name from GEAR
  - Magnetic field may be zero -> avoid the value of infinity for kappa
  - Interaction Point (IP) meas. layer may not exist for LCTPC
  - Do not throw exception if the layer is not found, return "no_intersection" code instead. This avoids the problem if some of the state is not or cannot be available, e.g. the IP and CaloFace states are not (yet?) defined for the LCTPC geometry

================================

MarlinTrkProcessors: v01-09-01

================================

- updated calling attrinbutes of ced_hit_ID to new CED v01-09

==============================

MarlinUtil: v01-06-01

==============================

- adopted for multi-module support in MarlinCED for the TPC detector