
Subject: new iLCSoft release v01-13-04
Posted by [gaede](#) on Fri, 02 Mar 2012 16:43:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

a new developers release of ilcsoft (v01-13-04) is available.
It provides:

- new development simulation models:

- ILD_O1_v02 "ILD simulation reference Model for DBD using Analog HCal"
 - ILD_O2_v02 "ILD simulation reference Model for DBD using SD HCal"
 - ILD_O3_v02 "ILD simulation reference Model for DBD using SciW Ecal and Analog HCal"

- with improvements to the Si-tracking drivers and inclusion of cables and services for the inner detectors

- first version of Si-tracking that deals with strip hits from SIT/SET (not yet FTD)
- considerably improved TPC pattern recognition

Please refer to the Release notes below for details.

Use the ilcinstall tool with the appropriate configuration files in order to install ilcsoft v01-13-04.

Reference installations in afs are available at:

```
/afs/desy.de/project/ilcsoft/sw/i386_gcc41_sl5/v01-13-04 # SL5 32bit  
/afs/desy.de/project/ilcsoft/sw/x86_64_gcc41_sl5/v01-13-04 # SL5 64bit
```

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

The iLCSoft team.

=====

Changes in v01-13-04 w.r.t. iLCSoft v01-13-03

=====

=====

LCIO: v02-01

=====

- minor release including following add-ons:
 - added helper structs to UTIL/ILDCConf:
 - ILDTrkHitTypeBit
 - ILDTrkHitQualityBit
 - added helper class UTIL::BitSet32
 - extension of std::bitset<32> for convenient setting and checking bits in flag words
 - > to replace LCFlagImpl (changed in examples and LCTOOLS)
 - added helper methods make_bitset(), set_bit() and unset_bit()
- bug fixes:
 - in /trunk/src/java/hep/lcio/implementation/sio/SIOTrack.java:
 - out.writeFloat(trackstate.getLocation()); -> writeInt

=====

Gear: v01-02

=====

- Added new functionality for surfaces.
 - Surfaces have a CoordinateSystem and may optionally be bounded.
 - So far only Cartesian Coordinate System are available with rectangular and trapezoid boundaries.
 - gearsurf remains in a separate directory and builds a separate library which is then linked to the gear library.
 - All new functionality is included within the gear namespace.
- TPCParametersXML now allows for multiple default modules (M. Killenberg)

=====

KalDet v01-06

=====

- ild
- ILDMeasurementSurfaceStoreFiller created to fill GEAR MeasurementSurfaceStore.
- Variable number of sensors to the FTD, SIT and SET.
- Added Rotated Strip measurements for ILDParallelStripPlanarMeasLayer.
- Allow offset in z for planes.
- Allow to set the origin of the u coordinate which lies in the plane, necessary for a plane composed of two pieces.
- ILDFTDKalDetector adapted to deal with new gear output.
- Added ILDConeMeasLayer using TCutCone. Still needs correcting in ILDSupportKalDetector

=====
 KalTest: v01-05

- =====
 - Added TCutCone surface with examples of use in hybrid example.
 - ScatterBy function corrected in TVTrack class.

=====
 MarlinUtil: v01-05-01

- bug fixes:
 /MarlinUtil/trunk/source/src/MarlinCED.cc:
- updated for new GEAR attribute getNSensors
 - updated for new definition of FDT sensors (max #2)
 - changed vertex draw method from ced_gebox_r to ced_gebox_r_solid to ensure transparency

- /MarlinUtil/trunk/source/src/CalorimeterHitType.cc:
 - changed order of ring and endcap in order to get HcalEndCapRingsCollection as ring layout (reported by G.Grenier)

=====
 Marlin v01-02-01

- bug fixes:
 - fixed new streamlog levels DEBUG5-DEBUG9, MESSAGE5-MESSAGE9,....ERROR9

- local streamlog level was not used in modifyEvent() and modifyRunHeader()

=====
Mokka: mokka-07-07-p06
=====

I. Implementation of services for inner detectors.

New sub_detector tubeX06 is available, which uses the same driver as tubeX05, but has a new DB, tubeX06_14. This DB contains sections made of Al of different thicknesses to model the services that would go along the tube (only Al, and no Beryllium for those sections).

II. Implementation of services for TPC and Ecal for models using SDHcal.

New driver and sub-detector SServices_02_v00 are available for services (cables, cooling) coming from TPC and Ecal, in the gap between barrel and endcaps, in the model that uses SDHcal.

III. Bug fix in step position of SDHcal endcaps.

Thanks to Gerald Grenier and Ran Han, a fix is available for the calculation of Z-coordinate of step position in SDHcal EndCaps.

IV. Three new detector models were created

Three new models are available, ILD_O(1,2,3)_v02, based on the _v01 versions, where the tubeX05 is replaced by the new tubeX06 (mentioned above), and in the SDHcal model (ILD_O2_v02) the old services sub_detector is replaced with the new one above (SServices_02_v00).

V. Improved Si-Tracking drivers for ILD (Aplin, Glattauer)

- SET_Simple_Planar, SIT_Simple_Planar and FTD_Simple_Staggered now write some additional gear parameters needed for reconstruction (strip angle, pitch, etc)
- added separate hit collections for pixel and strip discs in the FTD_Simple_Staggered
- added simplified sensor volumes to SIT and SET

=====
MarlinReco: v01-01
=====

- RecoMCTruthLinker

- added mcTruthTrackLink: inverse relation from MCParticles to Tracks
the weight will be the fraction of all sim-hits from the MCParticle that contributed to this track
- changed weight definition for trackMCTruthLink:

- weight is the fraction of all hits on the the track that have contributions from this MCParticle
- computed as $\text{sumSimHits_from_this_MCParticle} / \text{total_number_of_hits_on_track}$
 - > weight can be large than 1.0 (if more than one hit from an MCParticle would be used in a given layer (delta ray))
 - > this definition of weights is more accurate for defining the 'fake hit rate' as (1.-weight) in the case of many merged sim hits for a given track
- changed steering logic, such that it can be more easily called for certain sub-tasks, eg. Track-MCTruth-Link only:
- made the following output collections optional:
trackMCTruthLink, clusterMCTruthLink, RecoMCTruthLink, calo-hit MCTruthLink, skimmed

MCParticle

- > collections won't be created if empty name specified (default)
- removed options:
OutputClusterRelation, OutputCalohitRelation, OutputTrackRelation
they are now also based on name empty/not empty
- changed default for UseTrackerHitRelations to 'true'
 - print WARNING of set to false
- F.Gaede

- LDCCaloDigi

Implement use of LCIO calorimeter Hit step position for SDHCAL digitizer (multiplicity simulation) and add parameters to the processor to better control the digitization -- Gerald Grenier IPNL

- TPCDigi

Bug fix: added smearing of merged TPC hits
(using rather large errors - which may have to be iterated on.)

- BCalReconstruction

Improved calibration (A.Rosca)

=====

MarlinTrk v01-04

=====

- Changed fit method to optionally respect a maxDeltaChi2 in IMarlinTrack interface:
fit(double maxChi2Increment=DBL_MAX) ;
- Added new constructor to HelixTrack that generates helix from three point.
- Corrected sign of d0 when initialising from an LCIO track state in MarlinKalTestTrack.
- Fill GEAR MeasurementSurfaceStore using ILDMeasurementSurfaceStoreFiller from KalDet.
- Cleaned up dependencies

=====
MarlinTrkProcessors v01-03
=====

General: Make use of MeasurementSurface classes from GEAR, as well as new UTIL::BitSet32 and UTIL::ILDTrkHitTypeBit from ILDCConf (LCIO).

SiliconTracking_MarlinTrk: Adapted to use composite Space-points composed from strip hits. Added new parameter NHitsChi2 which controls the maximal number of hits for which a track with n hits is always better than one with n-1 hits. General Cleanup and merger from changes from CLIC CDR.

FullLDCTracking_MarlinTrk: Adapted to use composite Space-points composed from strip hits.

SimplePlanarDigiProcessor: Adapted to have option to produce strip hits.

TruthTracker: Flexible collection input and able to construct track including composite Space-points.

SpacePointBuilder: New - Builds 3D composite Space-points from strip hits.

=====
Clupatra v00-05
=====

- reduce memory footprint: deal with one KalTest track at a time wherever possible (saves >2MByte/track)
- properly set the bit UTIL::ILDTrkHitQualityBit::USED_IN_FIT for all trackerhits used in fit
- set correct subdetectorHitNumbers() for used in fit
(trk->subdetectorHitNumbers()[2*Icio::ILDDetID::TPC - 1] = usedInFit ;
trk->subdetectorHitNumbers()[2*Icio::ILDDetID::TPC - 2] = nHit ;)
- algorithmic improvements:
 - introduce a loop over increasing distance cuts for finding the tracks seeds
 - > fixes some of the problems seen @ 3 TeV with extremely boosted jets

- use function split_multiplicity() to split up merged cluster seeds if possible
(improves efficiencies in forward region)
- added create_n_clusters and use for splitting up clusters with multiplicities 4 and 5
- added gear and steering file for CLIC detector
- added some debugging functionality
 - new track debugging processor - under developmet: TrackCheckMCTruth
 - added debug method printTrackerHit() for picking
 - added optional debug collections:
ClupatraPoorQualityTracks, ClupatraOuterSegments, ClupatraInnerSegments, ClupatraMiddleSegments
- improved TrackEfficiencyProcessor
 - extended pt range to 500 GeV
 - introduce cut on good hit fraction (96%)
(needs MarlinReco v01-01)

```
=====
MarlinKinfit v00-01-01
=====
```

- patch release which includes new cmake configuration files for allowing other packages to compile/link against MarlinKinfit

```
=====
CEDViewer v01-05
=====
```

CEDViewer:

added parameter WaitForKeyboard

ced2go:

modified collections names to be the same as in bbudsc_stdreco.xml

allow to specify steering file template on the command line with -t