

Topics and Work Items for Linear Collider Software Hands-On Week

F.Gaede, DESY

LC-SW Meeting, CERN, Mar 19-24, 2017

- transition of iLCSoft to Github
- restructuring of iLCSoft packages
- implement missing features (e.g. TrackingCellIDEncoding singleton)
- work on iLCDirac

- packages already on <https://github.com/iLCSoft>:
 - to be migrated Mon/Tue

- CED
- CEDViewer
- CondDBMySQL
- ConformalTracking
- DDKalTest
- FastJetClustering
- ForwardTracking
- GEAR
- iLCInstall
- iLCUtil
- ILDConfig
- ILDPerformance
- KalDet
- KalTest
- KiTrack
- KiTrackMarlin
- LCCD
- LCFIVertex
- LCTuple
- MarlinKinfite
- MarlinKinfiteProcessors
- MarlinReco
- MarlinTrk
- MarlinUtil
- MemoryMonitor
- RAIDA

- packages not yet on <https://github.com/iLCSoft>:
 - leave it to authors/package maintainers if they want join iLCSoft
 - or create their own GitHub repository
- BBQ
- Druid
- Eutelescope
- Garlic
- LICH
- MarlinTPC
- PathFinder
- Physsim

Technicalities of transition to Github

- all packages need to have a LICENCE file:
 - if they use *LCIO* or *Marlin*: **GPLv3**
- all packages ideally should have:
- README (README.md)
 - basic introduction to package
 - **copyright** statement, e.g.
 - copyright: the *package_name* authors
- AUTHORS file
 - listing authors that have made *significant* contributions
- releases.notes

should use transition to add missing files as much as possible

- start to fix warnings

- managing ~40 packages is not trivial
- could potentially simplify maintenance and release cycles by combining into larger **meta packages**
- need to account for:
 - main authors/librarians
 - list of main developers
 - package dependencies
 - **leave generic packages standalone**

start this week with investigating potential package structure

- LCTrackingCore
 - core tracking libraries - no Marlin processors
 - aidaTT, MarlinTrk, KiTrack, KiTrackMarlin (rename ?)
- MarlinTracking
 - marlin packages with tracking processors
 - Clupatra, ForwardTracking, MarlinTrkProcessors, ConformalTracking
- merge FastJetClustering into MarlinFastJet and then to MarlinReco
- potentially move more marlin packages to MarlinReco
 - phase out packages that are not used and/or not maintained from MarlinReco
- *details to be worked out as we go along*

- ideally would like to have coverity code checker for all packages
- CERN instance of Coverity or use service on GitHub ?
- what are the requirements for Github ?
 - dependencies to other packages
 - ...
- can this be done for all packages ?

- need proper selection cuts for TPC and all-Silicon tracking
 - complete set of good tracks
 - no double counting
 - ...
- Shaojun (ILD)
- Matthias (CLIC)

- main production sites:
 - CLICdp - CERN
 - ILD - DESY, KEK
 - SiD - PNNL
- should probably use these primarily for storage
- how about CPU ?
 - use all available sites ?
- can we have default configuration scripts in iLCDirac ?

- TrackingCellIDEncoding singleton
 - move to DDRec or LCIO
 - changes in LCIO, DD4hep/lcgeo and tracking packages
- clean up ilcutil: remove un-needed Find-modules (ROOT, LCIO, CLHEP,...)
 - will potentially affect CMake files in many packages
- ...

- created in LCIO: LCTrackerCellID
- now need to make following replacements:
ILDCellID0 -> LCTrackerCellID
LCTrackerCellID::encoder_string -> LCTrackerCellID::encoding_string()
- optionally (LCTrackerConf.h is included in ILDConf.h)
#include "ILDConf.h" -> #include "LCTrackerConf.h"
- eventually remove lcio::ILDCellID and DDKalTest::CellIDEncoding

- main goal for ILD:
 - get production scripts running for large scale MC production with new software chain
- could we have more commonality in production scripts for ILD (SiD?) and CLIC ?
- work planned for end of week

- DESY: afs and cvfms installations of releases and HEAD builds (as needed):

```
/afs/desy.de/project/ilcsoft/sw/x86_64_gcc49_s16
```

```
/cvmfs/ilc.desy.de/sw/x86_64_gcc49_s16
```

- uses standard ilcinstall scripts in ilcinstall/releases
- CERN: nightly cvmfs HEAD builds - also used for CI with Travis

```
/cvmfs/clicdp.cern.ch/software ??
```

```
clang and gcc6.2 ??
```

- uses scripts in ilcinstall/builds

should try and unify the build and public installation schemes again !

- can we have regular HEAD builds to be used on the Grid for test productions ?
- can we agree on a standard compiler ?

- more frequent releases would be beneficial to all groups
- need to streamline the release procedure, e.g.

package maintainer

- increase the release version directly after release and start new entry in release notes

developers

- add an entry to the release notes (under new version tag) for **every** PR
 - don't merge if it does not
- potential issue: expect conflicts in release notes in concurrent PRs

- do we have other issues or topics that we should address this week ?