
Subject: Meeting Minutes

Posted by [gaede](#) on Tue, 16 Mar 2004 09:17:57 GMT

[View Forum Message](#) <> [Reply to Message](#)

minutes of meeting 03/15/04

(participants: NG,RC,GM,HV,TB,SA, FG)

1) Status reports/ New Development

NG: new version of Lelaps fast simulation by Willy Langeveld that writes LCIO files to be released by the end of the week

NG: no time yet to check changes proposed by D.Bailey wrt. LCIO version for VC++. FG: would prefer version w/o additional header files NG: should be feasible

NG: will check project files for LCIO/VC++ into CVS for next release

HV: changes/fixes from DB wrt. f77 had already been incorporated in CVS HEAD

NG: proposal to announce bug fixes available in CVS on LCIO homepage

FG: will do that - and also refer to bug report and forum on the LCIO homepage

RC: work on MCParticle-Track-Link, has preliminary definition of status bits for MCParticle.simulatorStatus

RC: will send email to FG with bit/stati

RC: what is the status of publicly available LCIO files ?

-> need to ask TJ

-> Desy-IT grid group interested in providing grid infrastructure for data files (should contact TJ)

FG: has little stand alone java tool to display MCParticle-Tree, used Yappi to display particle names, the JTtree subclass could be used for LCIO plugin (need comment from TJ).

Bug fixes in Yappi needed (will create bug report).

FG: will send around tool soon.

GM: Status of Mokka/LCIO still work to do, details from Paulo who is working on that.

2) Reconstrucion data model

FG: proposal for not inheriting from HepLorentzVector but have standalone subclass that holds original LCIO particle-object.

Advantages: no need to modify original data when applying transforms, easy to use, no conflicts with READ_ONLY/UPDATE mode of LCEvent (impossible to secure HepLorentz 4 vector data against modification).

Will post detailed proposal with examples to the forum.

FG: Started to implement the Track class in LCIO.

Proposed changes wrt. to original design:

- don't make charge persistent, can be stored as sign of the momentum, have convenient method getCharge() in EVENT.Track API; general agreement on that

- proposal to not store pointers to hits but list of indices, advantages:

- * difficult to have set of pointers to different types (SimTrackerHit, TPCHit, ...) as they don't have a common base

- * original mechanism to store pointer relationships in SIO is inefficient as it needs to 32 bit words

- * indices could probably (Calorimeter?) be stored as 16 bit

FG: will post detailed proposal to forum including use cases from TB

NG: proposal to remove 'get' from accessor method names for reconstruction objects, e.g. have recoParticle->energy() instead of recoParticle->getEnergy().

FG: problem is already released code that has the 'get' in all accessor methods.

-> need feedback from TJ

3) other points

HV: would like to have web interface to CVS repository, could provide perl scripts for that

-> need feedback from TJ

NG: should we extend the group of participants in regular LCIO meetings ?

all: preference to keep group small to work more efficient on technical details,

encourage people to make use of the forum (and bug report) to discuss LCIO related issues

and/or make proposals/requests;

depending on the level of controversy of the topic we can invite people to dedicated meetings

FG: proposal to restart regular minutes of phone meetings and take terms in writing the minutes -> general agreement

next meeting 03/29/04 09:00/18:00

=====

Subject: Re: minutes of meeting 03/15/04
Posted by [gaede](#) on Tue, 16 Mar 2004 09:37:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

Concerning Normans proposal to drop the 'get' from the simple accessors:
I like the idea of having shorter names for the methods but feel that it would be inconsistent to have a different naming convention for already released classes, e.g. MCParticle and the newer reconstruction objects. Users doing analysis comparing MC properties to reconstructed properties would probably find it odd to write
`double eRatio = mcp->getEnergy() / rcp->e() ;`
My proposal would be to leave the 'get' in the base interface for persistency (namespace DATA and hep.lcio.data respectively)
and add additional methods with shorter names to the EVEN/hep.lcio.event interface. Thus users can choose whatever names they prefer. A similar approach has been realized in CLHEP where the HepLorentzVector has a number of equivalent methods with different names to ease the use and readability of the code.
-Frank.

Subject: Re: minutes of meeting 03/15/04
Posted by [mora](#) on Tue, 16 Mar 2004 15:02:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Frank, very good news to place the minutes on the LCIO forum

Gabriel told me about the Mokka/LCIO interface questions yesterday. Let me say that there are at least two main missing features in the LCIO files written by Mokka in the current official release:

1) CalorimeterHits are written as the usual Mokka style. It means, for each PDG partial contribution and for each primary there is a hit object in the collection hit. So several hits for the same cell, if several particles crossed it. I've just rewrote part of this code in such way to have just an entry by calorimeter cell in the collection hit, with several partial contributions. I'm attaching here the same file I sent you before, for our colleagues also, to see if this new format is quite good to be tagged;

2) The MCParticle list doesn't follow the MCParticle-Hit-Assignment proposal you have

discussed recently. For the moment it's a kind of ad-hoc solution, controlled by a few command line parameters just to have a tree. But except if you set the parameters in such way to keep all secondaries the tree is unbalanced, so it's not good. I'm planning to implement as near as possible the MCParticle-Hit-Assignment proposal for the next release, if possible still before the LCWS04.

Cheers, Paulo.

File Attachments

1) [zqq_newformat.slcio](#), downloaded 807 times

Subject: minutes of meeting 2004-04-05

Posted by [NormanGraf](#) on Wed, 07 Apr 2004 23:30:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

Participants: FG, TB, HV, RC, NG, GM

Agenda Meeting 04/05/04

=====

1) Status and new Developments

2) Simulator Status (RC's proposal)

3) Reconstruction Data model

a) HepLorentzVectors (FG's proposal)

b) Implementation of Reco model

see posting in forum (Tracks,Clusters,ReconstructedParticles)

4) Redesign of LCIO

I would like to drop the namespace DATA (package hep.lcio.data) from LCIO. The original intend for that was that users should be able write their own classes with LCIO by implementing this 'small' interface. As things are developing the interfaces (in particular for the reco model) are not really small and I don't think that people will want to use their own implementations of the LCIO API.

The advantages are:

- clearer design (on level of abstract inheritance less)
- improved performance (less virtual function calls)

- no 'duplicate' methods any more (getParents/getParentsData)

If we then decide to create the complete API. i.e. including the set methods with the AID tool we are left with a much cleaner design: only one abstract API (common in Java and C++) and one concrete implementation.

Instead of having three different levels, e.g.

MCParticleData, MCParticle and MCParticleImpl and having to cast between those frequently, there will only be the MCParticle interface that is used for reading and writing the objects.

Once jdk1.5 is released Java will also have typed collections and then we could probably avoid casts in user code completely, which reduces run time errors and increases readability of the code.

5) time line for new release
(Java,f77 reco model)

6) other issues

Minutes:

decisions and action items are *

9:08 AM 4/5/2004

1.) NG: working with Willy Langeveld to release lclaps with LCIO.
Implementation exists, undergoing QA on the output.
Part of QA involved checking for memory leaks.
valgrind pointed out several, which were reported to the forum, and have been fixed (by FG) in latest head cvs version.

* agreed to run valgrind before any production release.

GM: Paolo is working on MOKKA LCIO output. Aim is for a release by Paris.

GM: working on CALICE TB setup.

* Handling of secondaries should be consistent between G4 programs.

RC: code which he has been developing is available in cvs,

will work on documentation.

2.) discussion of status bit:

FG: would have expected a simple enum of the different possibilities.

RC: hoping to keep it simple, but if people want more info, one can go in to great detail as to what happened. another byte could be reserved for more information

FG: would like to reserve byte or 16bit word to accomodate more detail at a later time.

bit 31 already defined? for endpoint?

*FG will double-check which bits are currently being used.

RC: vertex not being endpoint of parent is very important.

has only implemented creation bits, still working on the destruction bits. close to being able to release

lcs with these things implemented and writing out lcio.

has been doing QA on the output lcio files, looks good so far.

would like to have implementation for the status bits as soon as possible.

*FG will try to implement in next few days.

* should implement methods which insulate user from the specific bits

RC: Proposed only creating MCParticle from albedo particles if they leave energy in a sensitive tracker.

* agreed.

RC: Can get hits in tracker sensitive volumes which are attributed to neutral particles. Happens when secondaries are produced below the threshold for creating a new particle. Proposed creation of MCParticles for those particles which arise from neutrals and cause tracker hits. Provides a common way to handle "spurious" hits in tracker sensitive volumes.

TB: what is overhead for this?

RC: doesn't think this is a big deal.

FG: how many particles would this add to the event?

RC: maybe 20 in a ttbar event.

*sounds reasonable to implement if possible and overhead is indeed low.

* benefit is that tracker hits always point back to something reasonable

3a. FG HepLorentz vector for LCIO.

Main idea is not to inherit from CLHEP, but have a class which is instantiated from the LCParticle, which can be turned into a 4-vector, keeps 4-vector object separated from the LCParticle, but have different ways to access the attributes of the particle.

perhaps drop the operator overload, since not available for java anyway.

TB: fundamental decision whether to inherit from 4vec or not.

need to decide soon.

FG: inheritance from CLHEP is not a good idea, since many of the methods change the content of the LCParticle.

TB: LHC and BaBar both inherit from CLHEP. so could be done if careful.

NG: must be careful then to know exactly whether one is dealing with a transient copy of the data, or is mucking around with the data.

TB: points to <http://pax.home.cern.ch/pax/> as example.

FG: thinks this would be a thin layer on top of lcio implementation.

* we need to think about this in some detail and come to a decision fairly soon.

forum issue:

TB: subscribed to several topics, but not receiving email updates from forum.

* check with Tony to see if there are any setup issues.

3b implementation

FG: implemented tracks, clusters, reconstructedparticle and typeid.

NG: tracker hits and calorimeter hits not yet defined. Do we need to define an interface or abstract base class for these?

FG: thought there would NOT be a common tracker hit handle by storing the collection types, and use integer indices to point to hits and clusters within the collections.

NG: what would a global track return as a list of hits?

e.g. TESLA track could have

CCD pixel hits, intermediate silicon strip hits, tpc hits, straw chamber hits

FG: track would return a list of collection types. one would iterate over this list, then fetch the hits for each separate type collection.

NG: not too happy with this, since it imposes a very tight coupling between

specific types of hits and reconstruction code. i.e. need to know in track finder or fitter that one is dealing with a detector-specific type of hit. Will give some more thought to this.

NG: calorimeter cluster also defers this question of a base calorimeter hit or interface, but it seems clearer that there is probably going to be only one calorimeter hit type on which we can agree.

* this still needs some serious thought, as it addresses a very fundamental issue.

FG: would like to use pointers, but sio requires pointed-to and pointed-at words, which is seen as too costly an overhead, so stick with indices.

4.)

*general consensus that we should follow Frank's proposal.

5.) FG: hoping to get a beta release with recon model by beginning of next week.

TB: Would like to announce at Paris that a reco model is available.

First design exists, first beta implementation hoped for. Expect user input, so would be nice to have the code available.

*Stress that both design and implementation are first iterations and that feedback is requested.

Meet next week Tuesday (April 13).

Subject: minutes of meeting 2004-04-13

Posted by [NormanGraf](#) on Fri, 16 Apr 2004 00:49:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

9:11 AM 4/13/2004

FG, HV, TB, TJ, RC, NG

NG: QA on lelaps LCIO output ongoing, Will aim to put into cvs this week. Plan to run same events through lelaps and lcs for independent check of lcio output.

FG: Implementation of simulator status bit not done.
working on Reco data model. Modifications agreed to in last meeting included removing data namespace and java package data.

cleans up API.

TJ: still has some concerns about doing this.

Reason to have data namespace is to be able to use existing classes.
in existing lcd reco, we generate different instances of clusters,
e.g. don't have to tie the data to the data format as long as one has
an adapter. New model would insist that reco directly creates these objects.

FG: could also implement event interface which writes lcio w/o instantiating
lcio classes.

TJ: would like to write a toy reco model to see if he can still decouple
the reco model from the persistence model.

lcio concept has changed over time. would still like to see if
this can be made to work.

FG: introduced pointers to hits. last week used indices, which noone liked
so implemented to use real pointers, adds some to the file overhead,
but seems reasonable.

will also need to point between simtrackerhits and raw tpc hits, so
it makes sense to introduce these pointers.

in current implementation, for raw tpc and cal hits, made the pointer
optional. if only writing out raw data, don't need these pointers.

once these are used, and pointed at, the size will grow. now much more
consistent.

updated the posting in the forum.

FG: generic tracker hits proposed.

NG: concerned that this is OK for 3D hits, or 2D on a surface, but not
so good for 1D measurements. Important point is that what we need now
is something to get started, i.e. recognize the need for such a generic
hit.

*NG needs to write up specific objections to the proposed trackerhit
and propose what that interface should be.

FG: discussion of overall model at: http://desy.de/~gaede/reco_entities.pdf

*important to stress at LCWS to point to a design and even a BETA implementation.

All agreed that overall design is OK

FG: heplorentzvector issues:

1.) not a good idea to inherit from clhep's lorentzvector
has ~80 methods!

no control over access of data, can't impose read-only access

heplorentz methods, e.g. boosting, changes internal data

if ReconstructedParticle inherits from heplorentz vector, then RP data
gets modified.

- 2.) implemented something which is a class which copies data to the 4vector which can be used in 4vector calculations, but retains data. implemented in C++ for MCParticle, and would need only a few lines of code for RP.
- 3.) do we want to proceed along these lines?
consensus is to adopt this as a working hypothesis and see how far we get

NG: Would one iterate over the RP collection and then get 4Vector for each to feed, e.g. to a jetfinder?

TJ: RP collection could have method such as returnAsHepLorentzvector() to get a collection of 4vectors.

FG: thinking about object handlers and utility classes, but not yet needed.

3.) release plan:

ETA for java beta release of reconstruction objects.

TJ: before end of month.

1st stage is simply implementation of our current static model

2nd stage would be to implement a toy reconstruction program which exercises this.

* important thing for Paris is to present the model, and thoughts for the implementation. Stress that implementation is beta, and subject to change.

FG: started with Java implementation for RP. using Eclipse as IDE.
check in to cvs, we will look at it here.

TB: still experiencing lack of email notification when subscribed to forum.

TJ: found bug in the emailing code. has fixed and patched local version. will upgrade when fixed by author.

prototype forum setup, ready for users.

Meet again May 3, after LCWS and after ITRP visit to SLAC.

Subject: minutes of meeting 2004-05-03
Posted by [NormanGraf](#) on Tue, 04 May 2004 17:09:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

9:04 AM 5/3/2004

Agenda LCIO meeting May 3 2004

- =====
- 1) Status and new developments
 - 2) proposed developments
 - a) LCRelation - generic, weighted relation between LCOBJECTS (e.g. to be used to link to MC truth)
 - b) additional methods for fast MC:
 - Cluster.getECalEnergy()
 - Cluster.getHCalEnergy()
 - Cluster.getNMuonHits()

 - Track.getMinHitRadius()
 - Track.getNumberOfTPCHits()
 - Track.getNumberOfVTXHits()
 - 3) release plans
 - 4) other issues

1.)
HV: No progress on fortran implementation, busy working on SimDet, etc.
TJ: No progress, but this week blocked off.
aim to have Java implementation done for ANL workshop, but hope to have good start this week.

FG: News from LCWS04 is good, many groups using or planning to use LCIO.
No concrete feedback on reconstruction model.
Emphasize that model is beta.
Mokka still missing MC particle link in LCIO.
RC: reminder that MC status word still needs to be implemented for LCIO.
FG: will work on it this week.

2a.)

FG: introduce concept of LCRelation.

used to link between objects, e.g. raw to processed hit
or raw to MC particle.

BaBar uses something similar. (MCTruthAssociation?)

TJ: SLD had something like this as well.

FG: the relation object is untyped, currently uses std::multimap.

TJ: WOULD like to hide specific implementation from the final user, who
would like to be able to get these links in a
transparent way.

see http://www-it.desy.de/physics/projects/simsoft/lcio/v01-01beta/doc/doxygen_api/html/LCRelation_8h-source.html

TJ: would like to look at this in more detail. is there an example yet?

FG: see recjob.cc.

TJ: SLD used something like this to associate tracks to vertices,
so concept is useful.

FG: fastMC has no explicit hits or clusters, so need some way to
associate tracks to MC particles.

TJ: will look at and consider Java implementation.

2b.)

FG: Torsten Kuhl, doing some b-tagging jet analysis using fastMC, would
like some convenience methods along the lines of those proposed.

NG: Seems too specific. Let's think about how to abstract this
and make it more general.

3.) targetting ANL workshop.

TJ: as we get close, we will have to decide whether the workshop will
be used to decide on these issues, or whether we will have something
finished by then. In any case, we should have a release candidate
which represents our group's thinking.

4.)

TJ: setting up new enterprise version of bug checking system.
propose moving lcio bug database from freehep to this new system
this week. History should be preserved.

HV: can stop work on SimDet and Brahms, and will concentrate on Fortran
implementation of reconstructed partucle. Should be straightforward.

Meet again Monday May 10, 9AM PDT.

Subject: minutes of meeting 2004-05-10
Posted by [gaede](#) on Wed, 12 May 2004 07:51:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Minutes LCIO meeting May 10 2004

=====

Agenda:

- 1) Status and new developments
 - 2) proposed developments - a) and b) continued from last week
 - a) LCRelation - generic, weighted relation between
LCObjects (e.g. to be used to link to MC truth)
-> API/ implementation
 - b) additional methods for fast MC:
Cluster.getECalEnergy()
Cluster.getHCalEnergy()
Cluster.getNMuonHits()

Track.getMinHitRadius()
Track.getNumberOfTypeHits()
Track.getNumberOfTypeHits()
- > need more generic method names (not Tesla specific)

c) proposal: have all user methods in the abstract interface, i.e. also the set methods. The advantage is that users only have to use one interface for reading and writing (the implementation classes are only needed for object instantiation and even that could be replaced by factory methods in the future).

-> to be discussed

3) other issues

Present: Frank Gaede, Harald Vogt, Tony Johnson, Ron Cassell, Norman Graf

1) TJ has started to implement the reconstruction data model in Java. Until now implementation of Cluster, Track and ReconstructedParticle to follow soon.

HV has started to implement the reco model in f77. Working on a fortran test/example code analogous to recojob.cc.

FG has implemented the simulation status in MCParticle according to RC's proposal. Modification to endpointIsVertexOfDaughters not correct according to RC. Logic needs to be: vertexIsEndpointOfParent and independently endpointIsSet (i.e. the endpoint is not the daughter's vertex)

FG -> will implement that logic.

2a) TJ: how do users know which relation object in the event holds which relationships ?

FG: situation is completely analogous to the collections in the event. Users have to know the names of the collections they are interested in, either by definition within the community/working group or by a general naming convention. A while ago we started to think about such a general naming convention for collection in LCEvent but did not come to a conclusion as it turned out to be not so straightforward.

The current proposal for LCRelation allows the users to store more than one hypothesis for the relations between two concrete types, say ReconstructedParticle and MCParticle under two different names.

TJ would like to implement the LCRelation in Java to see whether the proposed API fits our needs and is reasonably convenient for the user. TJ also wants to compare to existing relationship handling in SLD

software, though this is not OO.

FG: not so easy finding a common API for Java and C++ that handles relations. For the user it would be more convenient to get all relations for one object with one method call and the same for the weights.

C++

doesn't have an abstract definition of collections/iterators equivalent to Java's collection framework. A possible implementation would be to return an `std::vector<LCObject*>` in C++ and a vector (collection) in Java. (When switching to JSDK1.5 Java could use a `vector<LCObject>` closer to the C++ implementation). On the other hand if mostly one-to-one relations are modelled then it is less convenient for the users as they have to access the one and only relation as the first element of a vector.

[FG comment: the current implementation would allow us to use covariant returns once we switch to JSDK1.5 for the relations, saving the users from having to apply a cast, e.g.

```
LCRelation rel = new LCRelationImpl<CalorimeterHit,SimCalorimeterHit>;
....
int nRel = rel.numberOfRelations() ;
for(int i=0 ; i<nRel ; i++){
    SimCalorimeterHit* simHit = rel.getRelation( calHit, i ) ;
    ...
}
```

whereas now users would have to write:

```
SimCalorimeterHit* simHit = dynamic_cast< SimCalorimeterHit*> (
    rel.getRelation( calHit, i ) ) ;
```

I think it would probably be worthwhile to have some discussion of the merits of switching to JSDK1.5 and using the new features of Java. In particular having parameterized types (C++ templates) and covariant returns. If we will make use of those features this would involve some changes in the API and should probably be done before our next major release or at least be foreseen for the future.

]

all -> need clarification of the issues soon (before the Argonne

workshop)

2b) It has been agreed to add more generic extensions to Track and Cluster API than proposed originally by FG and Thorsten Kuhl:

```
Cluster.getEnergyFraction( string calorimeterName ) ;  
and  
Track.getNumberOfHits( string trackerName ) ;
```

this makes the API more flexible wrt. different detector designs.

FG -> has to check with Th. Kuhl about the necessity for storing the radius of the innermost hit of a track.

TJ: there are other constant defined in the current implementation that are not generic enough, e.g. Cluster::ECAL,HCAL,COMBINED,LAT,LCAL.
TJ -> will think of a way to code the subdetector type information in a more generic way without the need of having to add additional constants for new subdetectors.

2c) TJ: doesn't like the idea of extending the abstract interface to also

include the set methods. Most users (in particular users that are less experienced in sw development) will use the read only interface anyhow. Expert users (e.g. writing reconstruction code) can easily use the implementation classes and apply downcasts if needed. Also this would make it easier to have additional implementations of LCIO data entities. For example one could have different Track implementation with a different parameter set (where the get methods apply the needed parameter transformation which would be harder for the set methods).

FG: advantage of having everything in the abstract API is the possibility to change the implementation without breaking user code, e.g. we could introduce some reference counting for memory management in C++. This would require that users do not instantiate the implementation classes at all, but only use corresponding factory methods.

TJ: wants to have a closer look while implementing the reco model in Java.

Next meeting: May 17, 2004 09:00/18:00 PDT/CET

=====

Subject: Agenda Meeting 26-05-2004
Posted by [gaede](#) on Tue, 25 May 2004 12:40:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

as discussed in our last meeting I post the proposed agenda for tomorrows meeting in the forum. (We can probably simply edit this message to include the minutes after the meeting in order to restrict the number of postings to one per meeting).

Agenda Meeting 26-05-2004

1) Status f77, java, cpp

2) Reconstruction data model:

- a) TJs comments on Tracks and LCRelations
- b) user extensions (see message #94 in this forum)
- c) extension to Track: getRadiusOfInnermostHit()

3) Transient LCCollections and multiple containment of one object

Discussing with users from DESY it turned out that they would need the possibility to create subcollections of existing collections, e.g. to analyze $e+e \rightarrow HH \rightarrow l+l-$, jets one would pass a list of all tracks except the two identified leptons to a jet algorithm. Copying the tracks is prohibitive in terms of memory and cpu, simply creating a collection with pointers to existing tracks will cause a segmentation violation (in C++, f77) when the event is deleted, as the assumption is made that every object is contained uniquely in one collection. One way around that is to keep an additional reference to every object in an event and delete objects to wrt. that list. Another way would be to

tag the collection as 'transient' and then not delete 'objects' in transient collections. To my mind these are independent concepts and we should implement both: allow one object to be contained in more than one collection and allow for users to flag collections as transient, i.e. they are only used as input to some subsequent analysis/reconstruction module.

4) Other issues

Frank.

Subject: Minutes of Meeting 2004-05-17
Posted by [NormanGraf](#) on Wed, 26 May 2004 06:15:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

9:06 AM 5/17/2004

Frank Gaede, Ties Behnke, Harald Vogt, Tony Johnson, Ron Cassell, Norman Graf

HV: fortran reconstruction package has been implemented. See note on LC forum.

TJ: has made good progress on java reconstruction code.

starting from existing LCD fastmc, to at least create smeared tracks, and clusters. Was sidetracked by development of the event display, which was necessary to be able to debug the reconstructed objects. will release code to cvs hopefully today.

RC: has looked at modifications made by FG to the MC status, but has not implemented code yet. Was sidetracked by bug in cvs head. Now fixed.

FG: cvs head version of lciio doesn't build also due to new java virtual methods being introduced but not implemented. will try to do so by tomorrow.

TJ: new version of lciio plugin for jas working with wired event display.

uses hit positions from track and cal hits, but not useful without some idea of where detector is.
using heprep for wireframe skeleton, and using magnetic field to swim tracks.
heprep file is quite simple at this point. Default is SDJan03,
Volunteer to help implement other detector geometries until we devise

a way to automatically create these from the geometry system.

FG: created heprep file from Mokka for hcal prototype. for the full tesla detector, this was 50MB, causing out-of-memory problem in JAS3.

TJ: will probably not want to use the default G4 heprep output. can also reset default java memory allocation to a higher value.

FG: would be nice to have some way to turn off parts of the heirarchy. e.g. load detailed G4 heprep, then turn off various items.

TJ: wired4 has a save-as feature, not sure whether this writes everything or just the visible items.

FG: Mokka has hard-coded drivers with visualization, etc. so painful to change.

TJ: should be possible to turn off visibility within G4. will send suggestions to local wired/g4 team.

FG: has hardcoded visibility attributes in G4, elements with visibility set to false don't show up in the heprep file, so this might be possible from within geant4 using interactive visualization commands.

TJ: picking doesn't seem to work yet, will follow up.
can use both wired3 and wired4 plugins to compare functionality.

TJ: lcio plugin produces hepreps (if wired plugin is available), which are then used by the wired plugin to display.

TJ: track parameters need discussing. lcio proposal is different from old lcd

- lcd: d0 (xy impact parameter)
- phi0
- omega (signed 1/radius of curvature)
- z0
- s (tan lambda)

lcio proposal seems to differ even from what we thought it was.

FG: presented proposal at ECFA meeting, but got no feedback. Maybe no one is paying attention?

NG: should review these parameters and decide what we want.
this proposal is beta, so we should not hesitate to fix this.

TB: tan lambda and 1/pt or curvature should be used.

TJ: some other points: need to get number of degrees of freedom.
reference point is origin, in lcio reference point is explicit.
would like to provide the user with a common reference point, so they get a commensurate set.

FG: should we have a boolean flag to indicate whether reference point is the pca?

TJ: getType has a bunch of known track types. Would like to either add a few more or generalize this.

FG: should we use generic strings as was proposed for collections?

NG: Use a DetectorManager class to handle this, using name to discover the types of collections, detectors, and types of hits in detectors.

FG: will look through code to find where integer constants are being used

TJ: has rearranged the meeting minutes to sit under a common thread in the forum making them easier to find and follow.

would be more open to post not only the minutes, but also the proposed agendas and meeting times.

FG: has been discussing with TB how timing is being saved in simcalorimeterhit. for every simcalhit, time was being stored individually for each MCparticle type contribution.

TB: MOKka has data saved by pdg type, not each distinct mcparticle

NG: this is somewhat odd, since one would only get one time for all photons or neutrons, for example, when what one presumably really wants is the time of deposition from the original, shower-inducing mcparticle.

FG: has updated mokka to store cal information by original mcparticle, but also can store every secondary deposition as well for detailed depositions by pdg type.

NG: This should only be used for really detailed studies of shower evolution.

Code documentation should note that use of this should be restricted.

FG: CALICE group has also asked for timing information to be added to Cal Hit.

NG: had also come across this missing functionality.

Action items agreed upon:

- use LCD track parameters, i.e. d0, z0, phi, omega, tan lambda
- add getNDF() and isReferencePointPCA() to Track
- add getTime() to CalorimeterHit

Meet again next week, Wednesday, May 26.

Subject: Minutes of Meeting 2004-05-26

Posted by [NormanGraf](#) on Mon, 28 Jun 2004 16:14:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

9:08 AM 5/26/2004

TJ, NG, RC, FG, HV

Agenda Meeting 26-05-2004

1) Status f77, java, cpp

2) Reconstruction data model:

a) TJs comments on Tracks and LCRelations

b) user extensions (see message #94 in this forum)

c) extension to Track: getRadiusOfInnermostHit()

3) Transient LCCollections and multiple containment of one object
Discussing with users from DESY it turned out that they would need the possibility to create subcollections of existing collections, e.g. to analyze $e+e \rightarrow HH \rightarrow l+l-, jets$ one would pass a list of all tracks except the two identified leptons to a jet algorithm. Copying the tracks is prohibitive in terms of memory and cpu, simply creating a collection with pointers to existing tracks will cause a segmentation violation (in C++, f77) when the event is deleted, as the assumption is made that every object is contained uniquely in one collection. One way around that is to keep an additional reference to every object in an event and delete objects to wrt. that list. Another way would be to tag the collection as 'transient' and then not delete 'objects' in transient collections. To my mind these are independent concepts and we should implement both: allow one object to be contained in more than one collection and allow for users to flag collections as transient, i.e. they are only used as input to some subsequent analysis/reconstruction module.

4) Other issues

Minutes:

TJ: updated java version of RECO, cluster, track and some parts of Relation (not IO)
also working on bporting fastMC and reco to use LCIO and newer JAVA version.

FG: was not able to read a file written in C++ reading with Java.

TJ still under development.

HV no updates on fortran version. will update documentation.

try to write lcio output from brahms after reconstruction. eta few weeks

FG: C++ started to implement changes for the track class. not yet checked in.

wanted to use strings for collection types, seems wasteful of space.

could just use an integer type, where user has defined bits.

TJ: no way to interpret bits in an arbitrary file, which is limiting.

would prefer to have something in header to define what the bits mean.

FG: can set only two bits for a track containing VTX and TPC hits.

TJ: would like to be able to use these files even if we don't necessarily know what the content is or who used it.

TJ: header provides a detector (string), use this name to find out what the detector is
lookup is easier if it's defined by string and not some bitmap.

RC: put bits in for each track, use header to map bits onto names.

FG: had talked about putting some structue into run header.

TJ at a minimum, map bits onto detector names

FG: concerned about disk space, so would like to replace the doca boolean with another bit.

currently proposed to allow user to dynamically define the number of hits per detector.

e.g. getNumHits(string name of subdetector.

NG: should this be in run or event header. probably event, since we ocassionally will strip out events.

TJ names needn't be defined in a generic way, since we need to look up the detector geometry anyway.

how to predefine usable bits, e.g. tracker bits 0-15. so user needn't dump the run header to decode.

FG; doesn't solve problem of how to add a list of number of hits to a track.

currently we would need a string for the detector plus an integer defining the number of hits.

this is expensive.

TJ: comes up primarily for the fastmc or full reco where we drop the hits.

Requires more thought.

2a.) TJ had posted comments on relations.

should we have method to remove relationships?

writing into file relationships between types of class a and b with some weight.

sld found it useful to go in boh directions. current interface sees it as a one-way relationship. proposes making this two-way. could implement using two maps, one for each direction, with some additional overhead. can perhaps allow user to define if relationship is two-way, or use lazy instantiation.

FG;what happens if we relate objects of the same type? be explicit about what direction we are going in.

TJ; make method names somewhat more explicit.

some discussion about api

TJ,FG: will review, propose more meaningfull methods names and functionality

TJ in constructor, not clear on what arguments are: interface name

2b.)

TJ: request #94 wasn't really generic, so prototype wouldn't be a solution.

FG: prototype + lcrelation would. should we try to implement this simple solution?

introducing generic object, leaving out macros

keeping in mind we want a better more flexible solution in the future..

TJ: sounds sensible. so user still needs to some coding, but targets the generic object, not sio explicitly.

FG: could introduce mapping to define the data layout of the object.

TJ if string is purely descriptive would be a lot happier.

3.)

two concepts ,

a.) having collections point to subsets of other collections.

b.) flag collection as transient

FG; think some more about the C++ implementation

TJ in implementing java reco package on top of lcio, started by exposing lcio collections to user. not convenient, so have now put a thin layer above the lcio collections. tightly couple those to the IO, but will want to have some layer between the IO and the user.

problems, e.g lcio returns float[] since this makes sense for io, but transient event shouldn't have to be that tightly bound to the io.

downside is that there is a layer between user and lcio.

FG: would prefer to expose the lcio to the final user.

tj: low-level data interfaces tied to io file format, and higher level event interfaces

both defined by aid files to specify java and C++ implementations. would like to make user interfaces easier to allow them to use natural features of the languages.

leads to divergence, but this gets around coding to the lowest common denominator.

RC: looked at status bits, looks fine. how soon can we make a release?

TJ, FG: propose tagging a version which can be used for lcs developers.

TJ: updated java MCParticle.

Subject: Minutes of Meeting 2004-06-21

Posted by [gaede](#) on Mon, 05 Jul 2004 07:54:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

Agenda

- 1) Status f77, Java, C++
- 2) New developments
 - a) user extensions
 - b) transient collections and multiple containment
 - c) parameters for collections events and runs
- 3) Release plans
- 4) Other points

- 1) HV: extended and modified fortran examples to be (almost fully) compatible with the C++ examples.
added documentation for reconstruction data model .
modified array indices to be consistently (1,n)
added LCStrVec for user extensions in C++ and f77

TJ: experimented with Java code for the LCRelations to see how the interface can be made easy to use and efficient to access relations in both directions.
also experimenting with user interfaces for other classes, e.g. stdhep, is there a need for an additional interface that is exposed to the users or not ?

FG: implemented subdetectorHitNumbers[] and subdetectorEnergies[] for tracks and clusters respectively and radiusOfInnermostHit for tracks.
changed int type to flag word for tracks, clusters and reconstructed particles.
all changes for java, c++, f77.

- 2) a) Agreement that we would like to have a simple version of user extension objects, provided that they can be described with our current XML-dtd.

b) TJ: having multiple containment of an element in several collections, is possible in Java anyhow; another way of passing

subcollections between modules would be to add predecate classes similar to what is available in the LCD-software.

FG: wants to have a look at predecate approach to understand the benefits.

TJ: wants the possibility to add 'anything' to the event and LCIO should simply ignore unknown data types during writing.

FG: flagging sth. as transient is more general, i.e. users can also flag collections of existing LCIO classes as transient.

for C++ 'anything' has to be limited to 'anything inheriting from LCOBJECT'.

need to make sure that LCIO doesn't crash when sth. unknown is in the event !

all: agreement to add some transient flag to LCCollection.

c) FG: proposes to have simply LCPParameters class, holding named attributes of the base types int, float, string, that can be added to the LCRun, LCEvent and LCCollection. this can then be used for example to define the meaning of bits in the type word of tracks, clusters and reconstructed particles.

TJ: need a way to assign/map attributes like 'bit0' to a particular type of data, i.e. tracks.

FG: would have to be done via meaningful names, e.g.

'TrackTypeBitTPC = 0'

TJ: would like a way to decode this information automatically and transparent to the user.

FG: don't know whether this is possible, will probably always need some additional information from somewhere else.

TJ: could maybe use XML description for that

FG: will implement something so that all can see how things might work in practise.

3) all: agreement to have another beta release before the Victoria workshop (July 28) that has all the main new features and data classes in it

4) plans to have an LCIO meeting at SLAC in the second week of August (9-13)

Minutes LCIO Meeting 07/20/2001

=====

Agenda

- 1) Status
 - a) f77
 - b) java
 - c) cpp
- 2) Discussion on LCRelation
- 3) ParticleID in Track and Cluster
- 4) Schedule for next (beta) release
- 5) AOB

- 1a)
HV: some minor bugs fixed. Working in Mokka/Brahms LCIO interface.
- 1b)
TJ: nothing new. What is the default value for MCParticle::simulatorStatus, e.g. in fast simulation ? General agreement that 0 should be used.
- 1c)
FG: added LCPParameters to LCEvent and LCCollection in java and cpp. Implemented new proposal for relation handling.
- 2) FG: proposal for LCRelation: have simple class for the relation itself as proposed by TJ, called LCWgtRelation (to avoid name clashes) and store those in normal LCCollections. Use additional navigator classes to lookup relations conveniently and efficiently. For efficiency reasons have two implementations for navigating the two directions (from-to/to-from) of the relation.
TJ: This is not clear to the user, when to use which implementation. Prefers to have one symmetric implementation for the navigator/relation.

FG: Thinks this causes performance costs.

TJ: Can be implemented in a way that maps are created only when needed.

FG: Could implement symmetric navigator in current C++ proposal.

TJ: wants to go back and see how relations might be used and how the proposed designs would work in practice.

3) RC: In principle ParticleID used for ReconstructedParticle could be used for Tracks and Clusters as well. Would need way to tell whether typeID is PDG code or something else.

FG: Currently not possible, would need additional attribute or could use collection parameters to encode this.

4) have beta release by Aug 20th (one week after FGs visit at SLAC)

5) TJ: Need convention on how to use parameters in Run, Event and Collection to encode type bits etc. -> will make proposal.

TJ: Wants to replace LCOBJECT in Java version with java.lang.Object. This would enable users to store arbitrary data objects in the event/collections.

FG: This removes the possibility to implement common code for LCOBJECTS, e.g. reference counting or uniqueID, not sure whether this is needed at all for Java.

TJ: needs to go ahead and implement it in order to see the impacts.

next meeting not scheduled - sometime after 08/13/2004

Subject: Minutes of Meeting 2004-09-14

Posted by [NormanGraf](#) on Thu, 23 Sep 2004 15:45:55 GMT

[View Forum Message](#) <> [Reply to Message](#)

Agenda LCIO Meeting 09/14/2004

=====

1) Status

- a) f77
- b) java
- c) cpp

2) Stdhep2LCIO interface

3) Link between Hits and raw Hits

- a) TrackerHit->TPCHit: need 1 to many relation, cannot use LcRelation, because of different types (VTXHit,SITHit,...)
- b) CalorimeterHit->RawHit - need only 1 to 1 relation, could in principle use LcRelation - but should be consistent with TrackerHit

4) ReconstructedParticle

TB,FG would like to have collection parameter flag "isUnique" telling the user that the collection at hand doesn't have double counting (and is complete) or "isBaseList"

5) Release schedule

FG, TJ: before CHEP if possible

6) AOB

FG, HV, TJ, RC, NG

1a.)

HV has added relations and generic user objects to the f77 implementation, parameters are not yet implemented.

FG points out that the bug tracking forum (with assignments) is available at <http://bugs.freehep.org/secure/BrowseProject.jsps?id=10050> and should be kept up-to-date.

HV expects to be done coding by the end of the week, hopes to have documentation finished by end of next week.

NG asks if intention is to have output of Brahms reconstruction available in lcio.

HV yes. currently waiting on resolution of event model and specifics of some of the objects. e.g. reconstructed particle currently has (x,y,z) and covariance matrix, but these aren't gaussian distributed, so questions validity. will need to check on this and decide whether this is what we want. will not happen before CHEP.

1b.)

TJ went through list of missing Java stuff in the bug reports and put in essentially everything except relations.

FG checked out cvs head snapshot, run C++ sim and reco jobs, write out results, try to read using Java version. This failed since relations are not yet implemented.

TJ points out that trackerhit positions are doubles, but cov. matrix is float. Is this OK? (probably)

TJ getgoodnessPID is in ParticleID; should this be in ReconstructedParticle? (yes)

TJ algorithm in PID is currently a String. should this be an integer which points to a string listed in the event header (yes)
change String getIdentifier into int getAlgorithmID

1c.)

FG believes C++ version is currently up-to-date with all features.
except leave long method names in getters of LCCollection.
except linking raw and generic hits.

Next release will be 1.3. Aim to support 1.0, but not 1.1 and 1.2 for backwards compatibility. plan is to finish basics this week, and to discuss advanced topics/usage/utils next week at DESY when TJ visits.

FG urges all to look at updated manual and provide feedback.

2)

RC has started on this, not done yet.
package should go into utils.
discussion of whether stdhep lite classes from Willy Langeveld should simply be copied or externally referenced. All agreed that having these classes in multiple locations (lclaps, Mokka, LCIO) was not clean, but will proceed this way anyway.
need to add float time to MCParticle to allow simulators such as G4 to properly handle decays.
will aim to finish package by end of next week.

3)

FG points out that the use of LCRelations for TrackerHits probably will not work since it will have to point back to different types of hits, e.g. TPCHits, VXDHits, etc. and we do not want to use LCRelations for mixed types. We also imagine that TrackerHits may point to many raw data hits, e.g. a TrackerHit built of many waveforms (TPCHits)

proposal is that CalHit.getRawHit() returns a single LCOBJECT
TrackerHit.getRawHits() returns a vector of LCOBJECTS.

4)

There is currently no way to restrict how users construct collections of ReconstructedParticles. It could be that various algorithms create custom collections representing the output of their reconstruction. The end user will then not know what constitutes either a complete or unique set of ReconstructedParticles which characterizes the event. The proposal is to implement two flags. Although there was no strong objection to this it was felt that it warranted further discussion at DESY next week and perhaps also on the forum.

5) we are still aiming to release by CHEP.

Subject: Re: Minutes of Meeting 2004-09-14
Posted by [lima](#) on Tue, 16 Nov 2004 19:51:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Regarding item 3 of these minutes, I would like to clarify what the "1 to 1 relationship" for raw calorimeter hits implies.

Some excerpts from the minutes follow:

- > 3) Link between Hits and raw Hits
- > a) TrackerHit->TPCHit: need 1 to many relation, cannot
- > use LcRelation, because of different types (VTXHit,SITHit,...)
- > b) CalorimeterHit->RawHit - need only 1 to 1 relation, could
- > in principle use LCRelation - but should be consistent
- > with TrackerHit

Subject: Re: Minutes of Meeting 2004-09-14
Posted by [lima](#) on Wed, 17 Nov 2004 10:28:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank,

Thanks for your answer. Now I would like to understand how to use/store the LCRRelation objects. Could you point me to some examples?

Thanks,
Guilherme

Subject: Re: Minutes of Meeting 2004-09-14
Posted by [gaede](#) on Thu, 18 Nov 2004 11:10:21 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Guilherme,

LCRelations are stored in ordinary LCCollections as any other LCOBJECT. Some example code is in src/cpp/src/EXAMPLE/recjob.cc.

For your case you'd do sth. like:

```
LCCollectionVec* calHits = new LCCollectionVec( LCIO::RAWCALORIMETERHIT ) ;

LCCollectionVec* scRel = new LCCollectionVec(LCIO::LCRELATION ) ;
scRel->parameters().setValue( "RelationFromType" , LCIO::RAWCALORIMETERHIT ) ;
scRel->parameters().setValue( "RelationToType" , LCIO::SIMCALORIMETERHIT ) ;

// use collection of existing SimCalorimeterHits
int nSimHits = simcalHits->getNumberOfElements() ;
for(int j=0;j<nSimHits;j++){

    RawCalorimeterHitImpl* calHit = new RawCalorimeterHitImpl ;

    SimCalorimeterHit* simcalHit = dynamic_cast<SimCalorimeterHit*> (
simcalHits->getElementAt(j) ) ;
```



```

// the cast is only needed to access the SimCalorimeterHit's attributes
// for the relation you can treat it as LCOBJECT:
// LCOBJECT* simcalHit = simcalHits->getElementAt(j) ;

// set calHit attributes
// ....

// add relation
scRel->addElement( new LCRelationImpl( calHit , simcalHit , 1.0 ) ) ;

// this assumes 1-1 relation
// if you have n-m (e.g. due to crosstalk)
// you can add as many relations as needed
// possibly using the weight as (relative) energy
// contribution

calHits->addElement( calHit ) ;
}
evt->addCollection( calHits , "CalorimeterHits" ) ;

LCFlagImpl relFlag(0) ;
relFlag.setBit( LCIO::LCREL_WEIGHTED ) ;
scRel->setFlag( relFlag.getFlag() ) ;

evt->addCollection( scRel , "CalorimeterHitsSimRel" ) ;

```

For reading back the relations there is a convenience class LCRelationNavigator that makes it easier to access the information, example code is also in recjob.cc.

I hope this helps.

Cheers, Frank.

Subject: LCCD without MySQL?

Posted by [hooberman](#) on Wed, 18 Jul 2007 22:38:32 GMT

[View Forum Message](#) <> [Reply to Message](#)

Is it possible to use LCCD without MySQL? If so how is this done?

Ben Hooberman
Berkeley Lab

Subject: Re: LCCD without MySQL?
Posted by [gaede](#) on Wed, 08 Aug 2007 15:55:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yes, you can use LCCD w/o MySQL by simply not specifying that you want to build with ConDBMySQL. In the latest version v00-03-06 this is the default if you build with cmake, e.g.

```
mkdir build
cd build
cmake -C ../BuildSetup.cmake ..
```

ConDBMySQL will not be used unless you comment out the following line from the BuildSetup.cmake:

```
SET( BUILD_WITH "ConDBMySQL" CACHE STRING "Build LCCD with these optional packages" FORCE )
```

-Frank.

Subject: How to merge two LCIO events
Posted by [lima](#) on Wed, 05 Sep 2007 18:17:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I would like to know whether there is something out there already for merging hits from two independent LCIO events. I would like to know of either Java and C++ implementations.

Thanks,
Guilherme

Subject: Re: How to merge two LCIO events
Posted by [tonyj](#) on Wed, 05 Sep 2007 18:20:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

Something exists for this in the Java implementation -- it is pure LCIO so the resulting merged files can be used by anyone. Jeremy would know the details.

Tony

Subject: Re: How to merge two LCIO events
Posted by [NormanGraf](#) on Wed, 05 Sep 2007 21:33:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Guilherme,
As Tony pointed out, this functionality does exist in the lcio distribution. For details see:
<http://confluence.slac.stanford.edu/display/ilc/LCIO+Command+Line+Tool#LCIOCommandLineTool-merge>

Norman

Subject: Re: How to merge two LCIO events
Posted by [gaede](#) on Thu, 06 Sep 2007 15:04:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Guilherme.

there is a new package for Marlin called Overlay that provides this functionality for Marlin. You can specify the number of events that are to be overlaid per 'physics' event either fixed or from a poisson distribution.

You can download the package at:

http://www-zeuthen.desy.de/lc-cgi-bin/cvsweb.cgi/Overlay/?cv_sroot=marlin

Frank.

Subject: Re: How to merge two LCIO events
Posted by [killenberg](#) on Thu, 06 Sep 2007 19:42:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

I would like to point out that merging events is not that simple with a TPC. This is due to the fact that the third coordinate which is measured is time. So one has to take into account the drift velocity and (temporal) bunch spacing to calculate the correct time where the TPC sees the hit. One also has to keep in mind that the TPC only takes one large picture of the whole bunch train and one has to disentangle later which track belongs to which event.

To calculate the occupancy (which I suppose is the reason why you want to overlay events) one further has to take into account the readout frequency and the diffusion in the gas to calculate the correct number of occupied voxels.

In my opinion it does not make much sense to simply merge SimTrackerHits from different events for a TPC, since one can only obtain reasonable tracks if one bypasses major parts of the digitisation and reconstruction and more or less directly fits the track to the SimTrackerHits (maybe with a little smearing). But I doubt that such a simplified way is suited for an occupancy study.

Track reconstruction efficiency etc. can only be tested if the tracks have the correct 'position' in time, anyway.

The MarlinTPC group has just started working on a pileup processor for the TPC to get a realistic digitisation. Is there a similar solution available in Java?

Greetings

Martin

Subject: Re: How to merge two LCIO events
Posted by [NormanGraf](#) on Thu, 06 Sep 2007 20:53:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello All,

I don't know why, but I was assuming that Guilherme was asking about merging the output from the full detector simulations, i.e. SimCalorimeterHits and SimTrackerHits. For this, the lcio merge package correctly accounts for time offsets, so that further digitization can do the right thing. I believe it is impossible to correctly merge hits further down the line (e.g. TrackerHit and CalorimeterHit). One can come up with technical ways of doing so, but they all have inherent problems. Martin has pointed out some of them for the TPC, but there are similar problems for

other subdetectors. I would strongly encourage people to restrict themselves to overlaying the raw output and doing the digitization correctly.
Norman

Subject: Evolving the Track Class
Posted by [kutschke](#) on Tue, 11 Sep 2007 20:07:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Colleagues,

I would like to draw your attention to a new topic, "Evolving the Track Class", that I just created within the Tracking and Vertexing forum. As we learn how we want to evolve the track class, we will need to include LCIO people in this discussion.

Regards,
Rob Kutschke

Subject: patch release v01-08-04
Posted by [gaede](#) on Wed, 19 Sep 2007 19:04:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

A new patch release is available from the CVS repository that fixes a few bugs. No changes in the file format or API. If you want to download this release please use the cvs tag v01-08-04. Refer to the documentation of v01-08.

Release notes:

=====

v01-08-04

=====

- bug fix release
- no changes in file format and API
- minor fixes in Java code (JMC)
- minor fixes for building with cmake

- changed order of modifyEvent and processEvent in SIOLCReader
(needed for Marlin Overlay processor)

- simple shell like tool for browsing lcio files in C++: .../EXAMPLES/lsh.cc
(use -DBUILD_LCIO_SHELL=true - requires ncurses and readline)
[experimental code]

Subject: Re: How to merge two LCIO events
Posted by [antonio.bulgheroni](#) on Mon, 24 Sep 2007 04:49:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

I was looking for a smart way of merging the content of LCEvent coming from different files, when I found this post.

I had a look and install the Overlay Marlin package, but I don't think it fulfills my needs.

What I would like to do is in the following:

During a test beam effort, it is very lucky that the tracking hodoscope (our EUDET-JRA1 telescope) is saving its output into a LCIO file, while the Device Under Test is steered by its own DAQ and it is saving its output into another file.

At a certain point we need to "overlay" the two files (say that are both LCIO) but in a very precise way keeping the run number and the event number perfectly synchronized.

Moreover, the merged event should contain all the collections available in the two input files.

Is it possible to do something like that? I have the impression that Overlay is not the right tool to do it, but can it be tailored? Should we write something from scratch?

Thanks for your help !!!

Antonio

Subject: lcio for whizard
Posted by [garren](#) on Wed, 07 Nov 2007 23:03:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

I understand that a fix to allow lcio to read whizard files was just put into the lcio cvs repository. My question is - what do I check out to get this fix???

I have tried building the head, but that doesn't seem to work:

```
$LCIO/bin/lcio stdhep -i /ilc/detector/graham/w09724_01.stdhep -o t.slcio
```

```
Exception in thread "main" java.lang.NoClassDefFoundError: hep/lcio/util/CommandLineTool
```

```
echo $PATH
```

```
/fnal/ups/prd/java/v1.5.0/Linux-2/bin:/fnal/ups/prd/ups/v4_7_2/Linux-2/bin:/usr/krb5/bin:/usr/kerberos/bin:/usr/local/bin:/usr/bin:/usr/sbin:/etc:/usr/etc:/bin:/sbin:/usr/bin/X11:/usr/X11R6/bin:./fnal/ups/prd/lcio/lcio/bin:/fnal/ups/prd/lcio/lcio/tools:/fnal/ups/prd/lcio/lcio.try/lcio/bin :/fnal/ups/prd/lcio/lcio.try/lcio/tools
```

Thanks,
Lynn

Subject: Re: lcio for whizard
Posted by [jeremy](#) on Thu, 08 Nov 2007 00:21:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi, Lynn.

Please update to LCIO CVS head. Earlier today I updated all the jars in tools to the latest versions from freehep, but I introduced a mistype in the classpath string that I've now corrected.

To rebuild the LCIO Java binding (Linux)...

```
cd lcio
. setup.sh
ant aid.generate
ant aid
```

Now you should be able to run the command line tool...

```
./bin/lcio stdhep -i [stdhepFile]
```

Unfortunately, we have a new problem, because the Stdhep converter runs out of memory due to

the files being too large. I will try to fix this.

Subject: Re: Lcio for whizard
Posted by [garren](#) on Thu, 08 Nov 2007 00:48:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks Jeremy - it works now.

Subject: Re: Lcio for whizard
Posted by [jeremy](#) on Thu, 08 Nov 2007 01:30:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I just checked-in some modifications to the StdHep converter for the 'Lcio stdhep' command.

- 1) Added -m switch to specify maximum number of events to read from the Stdhep file.
- 2) Added -s switch to specify number of events to be skipped.
- 3) Added code to run the garbage collector every 1000th event processed. Otherwise, I get an OutOfMemoryError when processing the million events in the background files. With the addition of manual GC, it ran fine for me and produced a correct LCIO file. Otherwise, it fails around the halfway point.

Please update to Lcio cvs head to get these fixes.

--Jeremy

Subject: Event weight in LCEvent
Posted by [tonyj](#) on Thu, 08 Nov 2007 17:18:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank, some of the stdhep events being generated for benchmarking studies include weighted events. There is no "getEventWeight" method in the LCEvent object. Do you remember if we agreed on a standard location associated with the event to store the event weight? If not should we define one?

Tony

Subject: copy constructor for LCFixedObject
Posted by [krautscheid](#) on Mon, 19 Nov 2007 14:50:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

for MarlinTPC we use TPCVoxel a class derived from LCFixedObject. Errors occurred when instanciating this object with the copy constructor because no copy constructor is defined for LCFixedObject. We fixed the problem by adding a copy constructor for TPCVoxel:

```
[TPCVoxel2(const TPCVoxel2 &original)
 : UTIL::LCFixedObject<TPCVoxel2NINTVals,
   TPCVoxel2NFLOATVals,
   TPCVoxel2NDOUBLEVals>(original)
{
  if( _createdObject ) _obj = new LCGenericObjectImpl(*(original._obj));
}
```

Now my question: Is this by intention, or should a copy constructor be added to LCFixedObject in a similar way?

Cheers, Thorsten

Subject: Re: copy constructor for LCFixedObject
Posted by [gaede](#) on Tue, 20 Nov 2007 11:07:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

no LCIO class has a copy c'tor as an implementation is not straight forward (deep vs. shallow) copy. We plan to hav some safe 'clone()' methods in a future release. The LCFixedObject and subclasses thereof are essentially handles to the underlying LCGenericObjectImpl and depending on how it was created it takes ownership of the underlying object or not. So you should not copy LCFixedObjects but rather implement a clone() method in your subclass that creates a new

LCGenericObject and copies the data.

Frank.

Subject: Re: copy constructor for LCFixedObject
Posted by [killenberg](#) on Tue, 20 Nov 2007 12:27:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Frank,

we want to store the object in a vector, and this uses the copy constructor. So the clone solution does not work.

In this case it is very obvious how the copy constructor should look like. The LCFixedObject is not even derived from LCOBJECT, so I think it can be treated differently. It can create it's own instance of a LCGenericObject, and deletes it in the constructor. It knows about the ownership because of the `_createdObject` flag. So the copy constructor knows when to copy the object and when not to.

As there is no clone-method we currently have to use the copy constructor. But in my opinion the copy constructor of the FixedObject would be the place to use the clone-method of the GenericObject.

Cheers

Martin

Subject: Re: copy constructor for LCFixedObject
Posted by [killenberg](#) on Tue, 20 Nov 2007 13:06:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

BTW:

It was exactly the fact that the object is deleted in the destructor which caused the segfault. If an object takes over the ownership and deletes the object it points to, then it should have a copy constructor which duplicates this object.

This only holds for the LCFixedObject, as it does not derive from LCOBJECT and is not a container (like a Track containing pointers to Hits), but a handle which can have the ownership.

Martin

Subject: Re: copy constructor for LCFixedObject
Posted by [gaede](#) on Tue, 20 Nov 2007 13:09:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Martin,

one of the design principles in LCIO is that objects are created on the heap - either by the user or be the LCReader - and that all collections only store pointers. This is true for the LCCollection and also for all user defined STL containers. So you should never ever use a vector of LCIO objects or the like as this is asking for trouble with memory management and might result in severe runtime overhead due to object relocation.

Please do not use copy c'tors with LCIO objects, we will probably make them protected in a future release when we implenent the clone method.

If you could let me know what exactly you would like to do I can probably suggest a more LCIO complaiant way of achieving this.

Cheers, Frank.

Subject: Re: copy constructor for LCFixedObject
Posted by [killenberg](#) on Tue, 20 Nov 2007 13:33:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Frank,

yes, you are right. Creating an LCGenericObject from the heap, like our copy constructor does, creates a huge overhead.

How about throwing a "dontUseCopyConstructorsException" in the copy constructor of the LCObject? So the user knows that he/she should not do it. I think we might not be the only ones who tried it.

Greetings

Martin

Subject: new LCIO release v01-09
Posted by [gaede](#) on Fri, 23 Nov 2007 17:19:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

A new release of LCIO is available for cvs checkout.
Please check the online documentation at <http://lcio.desy.de>.

=====
v01-09

=====
no changes in file format

- added event weight: `LCEvent.getWeight()` (Java/C++)
- added `LCWriter::setCompressionLevel(int level)` (C++)
- `lcio` (java tool) and `stdhepjob` (C++) improved
 - made compatible with new `stdhep` files (Java/C++)
 - fill `stdhep` event weight (and `_idrup` / user process id)
 - `stdhepjob` now build by default (can be used to convert `stdhep` to LCIO)
- `LCStdHepRdr` now fill `MCParticle::charge` properly (C++)
- Java uses new `sio` classes from `hep.io.sio`
- LCIO (C++) no longer has optional direct dependency on CLHEP
(only file is `UTIL/LCFourVector.h` that can be used in
programs built with LCIO and CLHEP)

Subject: Re: copy constructor for `LCFixedObject`
Posted by [vogel](#) on Mon, 26 Nov 2007 23:59:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

gaede wrote on Tue, 20 November 2007 05:09: Please do not use copy c'tors with LCIO objects, we will probably make them protected in a future release when we implenent the clone method.

Just a short question for my understanding: Wouldn't a private copy constructor (and copy assignment operator) in the base class be just the right thing for this purpose?

Adrian

Subject: Re: copy constructor for LCFixedObject
Posted by [killenberg](#) on Tue, 27 Nov 2007 08:32:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Adrian,

I would use it as a starting point. The default copy constructor calls the copy constructor of its parent, so one would prevent using it in all derived classes. But I don't know if this would compile.

One can always write a copy constructor for your own class by explicitly making it call the default constructor of the parent and implementing the copy functionality manually. But this has to be done deliberately and is what should be in the Clone() function, if I understand it correctly.

I don't know if a "per class" clone function makes sense. Suppose I have a collection with track-hypotheses, each of them containing pointers to hits, but all tracks are pointing to the same hits. Then a clone of the track has to point to copied hits (because one wants to keep them in memory while loading the next event from the file), but one does not want n copies of the hits collection.

How about a clone() method for the LCEvent, which copies everything once and adapts the pointers in the classes to points to the copied objects?

Cheers

Martin

Subject: getEventMap(), memory consumption
Posted by [srichter](#) on Mon, 11 Feb 2008 10:08:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Icio users!

I use Marlin and Icio in the versions comming along ilcsoft v01-03.
I noticed that the new direct access feature of Icio, where the message

SIORReader::getEventMap() recreating event map for direct access ...

comes from(?),

leads to a huge memory consumption. I run over raw data of the calice hcal testbeam. These have a volume of 2GB. It takes about a minute until

SIOReader::getEventMap() : done

and 1GB of memory.

Is there a way to turn the direct access feature off or do I have a problem somewhere else?
Beside the ProgressHandler I turned all processors off.

Thanks,

Sebastian

Subject: Re: getEventMap(), memory consumption
Posted by [gaede](#) on Mon, 11 Feb 2008 16:56:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

The latest version of Icio provides some initial version of direct access which has been requested by a number of users and is needed for example for overlaying events. In the current version a 'directory' of the events in the file is created - this in fact causes some delay when opening the file and some memory for the directory. However the numbers that you report seem to be rather extreme. Can you point me to an example data file?

Frank.

Subject: Re: getEventMap(), memory consumption
Posted by [srichter](#) on Mon, 11 Feb 2008 17:14:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Frank!

You can take for example:

```
/pnfs/desy.de/calice/tb-cern/raw/conv_v0402/Run321071.000.sl cio
```

I monitored the memory with

```
watch -n1 free -m
```

```
total    used    free    shared  buffers    cached
```

Mem: 2024 2001 23 0 12 1749
-/+ buffers/cache: 239 1785
Swap: 2000 10 1989

It started swapping...

Thanks in advance,

Sebastian

Subject: concatenating lcio files.
Posted by [wenzel](#) on Mon, 25 Feb 2008 16:21:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi

When trying to concatenate slcio files produced by slic in parallel I every now and then run into problem below. Any idea how I get around this.

Hans

```
lcio concat -i files.txt -o dual_e-_20.0_3917.slcio
Exception in thread "main" java.io.IOException: Framing error
    at hep.lcio.util.WritableSIORecord.read(WritableSIORecord.java: 29)
    at hep.lcio.util.Concatenate.concatenate(Concatenate.java:55)
    at hep.lcio.util.ConcatenateCommandHandler.execute(ConcatenateC
ommandHandler.java:65)
    at hep.lcio.util.CommandLineTool.parse(CommandLineTool.java:217 )
    at hep.lcio.util.CommandLineTool.main(CommandLineTool.java:124)
```

Subject: Re: concatenating lcio files.
Posted by [tonyj](#) on Tue, 26 Feb 2008 03:26:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Hans, in general a framing error means that the file is corrupted, perhaps because the job which created it crashed or didn't close it properly for some other reason?

It would be useful if the concatenate function told you which file it was reading when such an error occurs.

Subject: Re: concatenating lcio files.
Posted by [wenzel](#) on Tue, 26 Feb 2008 04:02:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Tony

Thanks for the reply. What I did in the end was to run lcio validate on all the files. Any file that showed any error I would just delete. then concatenating worked just fine. Didn't have a chance yet to figure out why some of the files were corrupted from the the logs nothing strange pops up.

Hans

Subject: apology: getEventManager() does not cause memory consumption
Posted by [srichter](#) on Wed, 27 Feb 2008 18:28:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear lcio-forum-readers!

In a former thread I claimed, that the direct access feature of lcio causes a huge memory consumption. I want to straighten out, that this is not the case. I used the wrong tools, the wrong way to come to wrong conclusions.

Sorry for the possible confusion.

- Sebastian

Subject: CellIDDecoder usage without knowledge of hit type
Posted by [srichter](#) on Mon, 21 Apr 2008 15:09:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear lcio users!

I would like to have a marlin processors which depends only on the information stored in the cellID0 and cellID1 of a hit. It is obviously easy if you use the CellIDDecoder, the hit type and the hit.

But the hit type is what bothers me. Is there any possibility to use the same code for example for SimCalorimeterHits, CalorimeterHits and RawCalorimeterHits? They all have getCellID0/1(). At the moment I have unattractive if(_hitType == etc.) statements...

Thanks in advance,

Sebastian

Subject: Quality Flag for EVENT::TrackerHit
Posted by [simone](#) on Thu, 24 Apr 2008 11:28:43 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

for the TPC reconstruction it would be very helpful to be able to store quality information about each TrackerHit.
Would it be possible to implement a new member function in LCIO's Event::TrackerHit that allows to set an integer which contains quality bits?

Thanks a lot,

Simone

Subject: new release v01-10
Posted by [gaede](#) on Fri, 06 Jun 2008 08:07:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

A new release of LCIO is available v01-10.
Please check the online documentation at <http://lcio.desy.de>.

=====
v01-10
=====

- added optional attribute CalorimeterHit::getEnergyError()
 - > set/check flag bit RCH_BIT_ENERGY_ERRORrequest from calice: [http://bugs.freehep.org/browse/LCIO-56]
- optional linkage againsts libdcap (dCache) for C++
 - > cmake -DBUILD_WITH_DCAP=1 -D DCAP_HOME=dcap_home
- added PIDHandler class to UTIL (C++) for convenient access to ParticleID properties of ReconstructedParticle (needed for LCFI flavour tag quantities)
- restructured test programs for C++ (make tests ; make test ; when built with cmake)
- made compatible w/ gcc4.3 C++ (A.Bulgheroni)

Subject: Re: CellIDDecoder usage without knowledge of hit type
Posted by [gaede](#) on Fri, 06 Jun 2008 08:15:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Sebastian,

in general a Marlin processor should know the type of objects it needs, ie. SimCalorimeterHits vs RawCalorimeterHits. For example this is used when checking the consistency of a steering file wrt to the availability of a collection of the requested type.

In order to avoid duplication of code, however you could try and write a templated processor with typedefs for SimCaloHit, CaloHit and RawCaloHit...

Don't forget to have a global instance of each in your library.

-Frank.

Subject: stdehepjob fails on 64 bit
Posted by [poeschl](#) on Fri, 06 Jun 2008 09:07:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Experts,

executing the stdehepjob on a 64 bit machine leads to a segmentation

fault while it runs fine if one compiles on a 64 bit machine using the -m32 option, i.e. creating a 32bit version.

I have traced the error down to

```
long IStdHep::Event::read(IStdHep &Is)
396 {
397 //
398 // Read event header
399 //
400 long len;
401
402 cleanup();
403
404 blockid = Is.readLong();
405 ntot = Is.readLong();
406 version = Is.readString(len);
407 if (blockid != LSH_EVENTHEADER) Is.setError(LSH_NOEVENT);
408
409 evtnum = Is.readLong();
410 storenum = Is.readLong();
```

where the 32 and 64 bit output for 'evtnum' and 'storenum' deviate during the second passage of this method.

The program fails finally at another point but still the point above would be the first where agreement has to be assured.

The method in turn calls methods from the IXDR class which indeed doesn't look very portable, see e.g.

```
114 long IXDR::checkRead(long *l)
115 {
116 if (_openForWrite) return(_error = LXDR_READONLY);
117 if (_fp == 0) return(_error = LXDR_NOFILE);
118 if (!l) {
119 long nr;
120 if ((nr = fread(l, 4, 1, _fp)) != 1) return(_error = LXDR_READERROR);
121 *l = ntohl(*l);
122 }
123 return(LXDR_SUCCESS);
```

124 }

which assumes a long to be 4 byte while in fact a long has 8 byte on a 64 bit machine. Also the application of fread and fwrite leads to non-portable methods, i.e.,

```
[poeschl@lx2 UTIL]$ man fread
```

```
...
```

APPLICATION USAGE

The `ferror()` or `feof()` functions must be used to distinguish between an error condition and an end-of-file condition.

Because of possible differences in element length and byte ordering, files written using `fwrite()` are application-dependent, and possibly cannot be read using `fread()` by a different application or by the same application on a different processor.

Does anybody know how to fix the bug? In any case I believe that we need to revise this class completely as 64 bit machines are getting more and more common.

I observe the same behaviour for Lcio v01-08-02, v01-09 and v01-10, always compiled following the instructions on the Lcio website, for both cmake or classical building.

Infos about my architecture

```
[poeschl@lx2 UTIL]$ uname -a
```

```
Linux lx2.lal.in2p3.fr 2.6.9-42.0.3.ELlargesmp #1 SMP Thu Oct 5 16:46:10 CDT 2006 x86_64  
x86_64 x86_64 GNU/Linux
```

with a standard SLC4 installation.

Cheers,

Roman

Subject: Counting Events in an LCIO File
Posted by [vogel](#) on Mon, 28 Jul 2008 21:00:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

I have no idea whether the following will be useful for anybody else, but at least it was useful for myself: in order to find out quickly how many events are contained in an LCIO file (in the slcio format, to be precise), you can run
`strings foo.slcio | grep '^LCEvent$' | wc -l`However, this might only work with LCIO version 1 files, and it will only find the top-level objects LCRunHeader, LCEventHeader, and LCEvent, but no collections or any kind of contained LCOjects.

Cheers,
Adrian

Update on 2008-08-05: For some reason, strings doesn't seem to work with DCAP. If you want to scan LCIO files on the dCache, prepend the pipe with an ordinary cat:
`LD_PRELOAD=/opt/products/lib/libpdcap.so cat foo.slcio | strings | grep '^LCEvent$' | wc -l`

Subject: Icio cellID1 flag set accidentally
Posted by [srichter](#) on Mon, 04 Aug 2008 13:25:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Icio users!

I have the problem that the flag to store the cellID1 is set even if my encoding string is exactly 32 bits long. dumpevent shows:

```
flag: 0x20000000
parameter CellIDEncoding [string]: module:0:6,chip:6:5,channel:11:5,SiPM:16:16,
  LCIO::RCHBIT_ID1   : 1
  LCIO::RCHBIT_TIME  : 0
  LCIO::RCHBIT_NO_PTR : 0
```

I set the string via:

```
LCCollectionVec* pOutputCol =
  new LCCollectionVec(LCIO::RAWCALORIMETERHIT);

std::string
  encodingString("module:0:6,chip:6:5,channel:11:5,SiPM:16:16 ");
```

Icio::CellIDEncoder<Icio::RawCalorimeterHitImpl>
 outgoingCellIDEncoder(encodingString, pOutputCol);

Where is my error?

Thanks in advance,

Sebastian

Subject: Re: Counting Events in an LCIO File
Posted by [srichter](#) on Mon, 04 Aug 2008 15:11:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Adrian!

Thanks for this nice hack!

Sebastian

Subject: Re: Counting Events in an LCIO File
Posted by [jeremy](#) on Thu, 04 Sep 2008 21:05:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

The Icio command line tool has a count command to do just this (no hacks required!).

<http://confluence.slac.stanford.edu/display/ilc/LCIO+Command+Line+Tool>

Example should be pretty self-explanatory. Let me know of any questions.

Subject: Re: Counting Events in an LCIO File
Posted by [antonio.bulgheroni](#) on Fri, 05 Sep 2008 08:07:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Jeremy,

I'm very interested in the Icio command line tool, especially the compare command. But when I try to execute Icio I got the following error:

Exception in thread "main" java.lang.NoClassDefFoundError: hep/lcio/util/CommandLineTool

Since I'm not a Java guy I don't really know what should I do to fix it. Can you please give me some hints?

thanks

toto

Subject: Re: Counting Events in an LCIO File
Posted by [jeremy](#) on Mon, 08 Sep 2008 21:04:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

You have probably not built the Java binding.

In the LCIO directory, the script build.sh will create the Java library using ant.

```
> ./build.sh
```

You should see the jar in the lib/ directory.

I'm not sure about cmake builds but I know there is a way to create the Java library.

Subject: Re: Counting Events in an LCIO File
Posted by [antonio.bulgheroni](#) on Tue, 09 Sep 2008 06:36:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

The jar file was there, but a rebuild fixed the issue.

thanks!

antonio

Subject: Re: Counting Events in an LCIO File

Posted by [vogel](#) on Fri, 12 Sep 2008 17:06:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Jeremy,

thanks for providing the hint to the Java LCIO command line tool! However, I thought an advantage of the "hack" is that you'll need neither an LCIO installation nor Java, but only a few system tools ... (For example in order to have a quick look at files residing on a machine that is used only for mass storage.) The plain-text scan is also marginally faster, but I guess that's not a relevant argument.

Cheers,
Adrian

Subject: Re: stdehepjob fails on 64 bit
Posted by [engels](#) on Mon, 29 Sep 2008 09:49:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Roman,

I've checked in the fix in cvs for this. If you get to test it, please let me know if it works.

Cheers,
Jan

Subject: Re: stdehepjob fails on 64 bit
Posted by [doublet](#) on Thu, 02 Oct 2008 16:30:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Jan,

The new version has been tested and it works correctly. Thanks.
Other 64-bit issues (if some occur) will be pointed out.

Regards,
Philippe

Subject: Problem using files with Cluster and CalorimeterHit collections

Posted by [phmag](#) on Thu, 06 Nov 2008 12:57:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

If I perform particle flow with PandoraPFA and then save the file without dropping the CalorimeterHit collections then I can't use the file again. Marlin, anajob or whatever program crashes with an error like:

```
*** glibc detected *** free(): invalid pointer: 0x0000000000b629e0 ***
```

Both anajob and Marlin are okay when they're running over the run headers, but not the events. I've found that if I comment out any lines like `cluster->addHit(...)` in `PandoraPFAProcessor.cc` the problem goes away. I suspect this is a problem with LCIO rather than Pandora though.

My guess is that after finishing with the first event, LCIO is trying to delete the CalorimeterHits twice - once for the actual hit collections and then again for each hit associated to a cluster. However, I've had a look at the ClusterImpl destructor and it only deletes the ParticleID objects and not the CalorimeterHits. Does anybody else know what it could be?

I'm using HEAD versions of LCIO and PandoraPFA. It's a 64bit machine but I don't think that matters.

Subject: Strange behaviour of `LCEvent::takeCollection()`
Posted by [killenberg](#) on Sun, 07 Dec 2008 20:18:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

the behaviour of `LCEvent::takeCollection()` is not as I expected.
The ownership is transferred correctly, but

the collection stays in the event I took it from.
the collection is set to transient.

If I take the collection away from the event I would expect that the collection is removed from the event. And I can't call `removeCollection()` since this deletes the collection from memory. This definitely is a bug since an event which gave away the ownership must not delete the collection.

That fact that it is set transient is also not very helpful, since I want to add it to another event (compose an event from several subevents) and write it to disc. So I remove the transient flag which means it will also be non-transient in the subevent I took it from. Currently this is not a problem since I delete the subevent, which correctly does not delete the collections that don't belong to it.

Here are my proposals for improvement:

If it is desired that the taken collection stays in the source object, the `removeCollection()` should be modified so the collection which does no longer belong to the event is not deleted.

In any case the transientness of the collection should not be modified. One could modify the streamer so only collections belonging to the event are written to disc.

Greetings

Martin

Subject: Re: Strange behaviour of `LCEvent::takeCollection()`
Posted by [gaede](#) on Mon, 08 Dec 2008 08:53:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Martin,

from the documentation of `LCEvent::takeCollection()`:

Returns the collection for the given name and transfers the ownership of the collection to the caller.

The caller is responsible for deleting the collection `_after_` the Event is deleted. The collection is still in the event but flagged as transient after the call returns.

This is usefull when you want to keep the collection for the next events.

Use with care!

which is what it does.

This behaviour is needed foe example for conditions data collections that should be kept for the following events, as long as they are valid.

Can you not use it that way, i.e. not remove it from the original event ?

Cheers, Frank.

Subject: new release v01-11
Posted by [gaede](#) on Sat, 13 Dec 2008 13:45:21 GMT
[View Forum Message](#) <> [Reply to Message](#)

A new release of LCIO is available v01-11.
Please check the online documentation at <http://lcio.desy.de>.

=====

v01-11

=====

- made direct (random) access mode in C++ optional:
use `LCFactory::getInstance()->createLCReader(LCReader::directAccess)` if
direct access is needed (default is off, i.e. 'no recreating of event map'
when opening a file)
- bug fixes and improvements in C++
(CMake build procedure, compatibility w/ gcc4.x, 32bit compatibility mode)

[file format not changed since v01-08]

Frank.

Subject: confusion about memory issues.
Posted by [fengy](#) on Fri, 19 Dec 2008 05:14:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi I am beginning with LCIO.

In the manual I saw that every reader is created by the singlet & factory. What I am curious is how these reader can be free'd.

There is no destroy/dispose method in the factory.

Is it OK if I use delete to free them?

If I delete them, will the stream be flushed to the storage automatically?

If I close them, are they automatically free'd or do I need to explicitly delete them?

Any clarify will be appreciated. I haven't been coding in C++ for ages.

Regards,

- Yu

Subject: Re: confusion about memory issues.
Posted by [gaede](#) on Fri, 19 Dec 2008 15:06:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Yu,

the lcio manual explains the memory handling:

Quote:

LCIO Memory management in C++

The philosophy of memory management in LCIO is: " If you created the object (using new), you are responsible for deleting it when it is no longer needed !".

In order to make this easier the LCEvent will delete everything that has been attached to the event. If you are only reading, LCIO will create the objects and thus delete them (when reading the next object of the same type) - thus no need for deleting on your side.

If you are reading and adding sth. to the event it is LCIO that deletes the event (as it created it) and thus also everything attached to it - again no need for deleting on your side.

As a rule of thumb: Use delete evt/runhdr at the end of every event/run loop where you created the event/runHeader and don't use delete in all other cases.

Of course if you use your own implementation of the EVENT interface you are also responsible for the memory management.

it is really simple for the user - you can also have a look at the examples (simjob.cc, anajob.cc ,...)
in \$LCIO/src/cpp/src/EXAMPLE.

The LCReader/LCWriter objects should also be deleted at the end of the job (after calling close()
).

Frank.

Subject: Re: confusion about memory issues.
Posted by [fengy](#) on Fri, 19 Dec 2008 15:35:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks for the clarify, Frank.

So it basically means LCEvent holds the ownership of the LCCollections, and the LCCollections hold the ownership of LCOjects.

I also noticed LCOject::clone is not overridden in MCParticle subclass.

therefore, I can not preserve a LCOject after the event is removed, right?

BTW: did you know GObject system? GObject features dynamical types at runtime. It seems like a lot of physics projects also need to do this heavily(ROOT!?), but C++ actually doesn't support it well in a fundamental level.

AKA in gobject you can do a LCIO_IS_MCPARTICLE_COLLECTION(mc_particles), or (mc_particles is MCParticleCollection), where in C++ we have to define mc_particles.getName() == LCIO::MCPARTICLE: basically implementing minimal dynamical types.

I know it is impossible to switch the fundamental of the frameworks. I am just trying to let more people know this excellent toolkit.

Yu

Subject: Re: new release v01-11
Posted by [fengy](#) on Fri, 19 Dec 2008 18:09:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

LCIO 1.11 also fixes a segment fault of dumpevent on the DST slcios. I've just dicovered this.

Subject: easier way to test if an object is in an collection?
Posted by [fengy](#) on Sun, 18 Jan 2009 19:03:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Suppose there is a reconstructed particle found from MCTruthLink;
and there is a collection FTJet1_Selected;

One way to test if the particle is in FTJet1_selected is to iterate over all objects in the collection and see if any of them is the particle;

Are there any better alternatives?

Regards,

Yu

Subject: Daughter Collections

Posted by [oschaefer](#) on Thu, 19 Mar 2009 11:04:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

Somebody told me about this mechanism as the one to use in Lcio whenever I wished to simply split up a collection regarding certain criteria but not to modify it.

Now this case appeared: I want to make two collections of hits from the original one, sorted by whether they have contact to the padplane's edge or not. So all the contents of the collection stays the same. The very same "somebody" at that time told me that the new daughter collections only contain pointers to the original objects or so.

But how do I use this mechanism? Does it have something to do with the LCRTRelations? I searched forum and documentation but I can't figure out ...

Assume I have

```
LCCollectionVec AllHitsColl;
```

How to obtain the daughter collections?

Subject: Re: Daughter Collections

Posted by [killenberg](#) on Thu, 19 Mar 2009 13:12:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Oliver,

I think the LCRelation is not what you want. The Idea behind this is to have a link e.g. from a TrackerHit to the SimTrackerHit it originated from (if I am not mistaken).

Let me start with your second remark:

Quote:that the new daughter collections only contain pointers to the original objects

LCCollections always contain pointers to objects, because Lcio objects are always created on the heap using new. The special thing about LCCollections is that they own the objects they are

pointing to, which means the destructor of LCCollection deletes all objects from memory when the collections itself is deleted.

If you now make a copy of these pointers you have two collections pointing to the same objects, each of them holding the ownership. If you delete one of them, the other one is pointing to objects which no longer exist.

For this purpose there is the subset flag. A collection declared as subset does not take over the ownership, so it does not delete the objects if the collection is deleted. So you want to make the two split collections a subset. The original, unsplit collection just stays intact and own the objects.

Example:

```
// A collection of tracks from the event.
LCCollection *trackCollection = event->getCollection("Tracks");

// We want to split into long an short tracks.
// These collections are subsets.
LCCollectionVec *longTracksCollection = new LCCollectionVec(LCIO::TRACK);
longTracksCollection->setSubset();

LCCollectionVec *shortTracks = new LCCollectionVec(LCIO::TRACK);
shortTracksCollection->setSubset();

for (int i = 0; i < trackCollection->getNumberOfElements() ; i++)
{
    Track * mytrack = dynamic_cast<Track *>(trackCollection->getElementAt(i));

    if (mytrack->getTrackerHits().size() > 5)
    {
        longTracksCollection->addElement(mytrack);
    }
    else
    {
        shortTracksCollection->addElement(mytrack);
    }
}

// Add the collections to the event.
// Make sure the original collection is not set transient, otherwise
// the track objects are not in the data file.
event->addCollection( longTracksCollection );
```

```
event->addCollection( shortTracksCollection );
```

Cheers

Martin

Subject: Re: Daughter Collections
Posted by [gaede](#) on Thu, 19 Mar 2009 13:21:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Oliver,

as Martin has pointed out correctly (missed his posting when writing mine..) :

the mechanism you can use are the subset collections. You simple create two new collections and set the subset flag:

```
LCCollectionVec* col1= new LCCollectionVec ;  
LCCollectionVec* col2= new LCCollectionVec ;  
col1.setSubset(true) ;  
col2.setSubset(true) ;
```

```
//... loop over original collection ....
```

```
if( cond1 )  
    col1->addElement( e ) ;  
else  
    col2->addElement( e ) ;
```

These collections will just hold pointers, i.e. no unnecessary copying involved.

-Frank.

Subject: Re: Daughter Collections
Posted by [oschaefer](#) on Thu, 19 Mar 2009 13:46:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks a lot, this worked out very well and is surprisingly easy to use.
Oliver.

Subject: lsh: this event does not exist
Posted by [antonio.bulgheroni](#) on Wed, 08 Apr 2009 14:08:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

I'm using the lsh utility to browse the content of a LCIO file.

The problem is that when I cd into the run folder and a type ls I get a list of events (number in square brackets) with the corresponding collection. But when I try to cd in one event I always get the message

this event does not exist

Do you have any idea?

cheers,
toto

Subject: LCWriter error when inputfile and outputfile only differ in "_" changed to "."
Posted by [ralf](#) on Fri, 24 Apr 2009 08:09:23 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi.

When using a self written program, that changes LCIO run header information, I experienced a strange error:

I read in a file named "cosmic_0091-01_P5_stagg-2_3_5T.001.slcio" and try to write to a file named "cosmic_0091-01_P5_stagg-2_3.5T.001.slcio". So the names only differ in "3_5T" changed to "3.5T".

When running the program I get the following error message:
"lciio::IOException: [SIOWriter::open()] Bad or duplicate stream name:

cosmic_009101_P5_stagg2_3_5T_001_slcio".

This happens in the line where I open the output file (the input file is opened without problem and I can read from it earlier during the execution):

```
LCWriter* fLCWriter = LCFactory::getInstance()->createLCWriter();
    try
    {
        fLCWriter->open(outputfile);
    }
```

When I change the "." in "3.5T" in the output file name to another character or number or use a completely other name it works fine. When I use the above given output file name but a different directory for the output it works fine.

It seems like the open command interprets the "." like in an regular expression (where it stands for any character) when the checking if the file already exists is done.

I did my tests under Scientific Linux 4 with different LCIO versions (v01-08, v01-09 and v01-10).

It is not a big problem, since it is easy to avoid, but I thought this might be interesting for bug fixing ... and save others from spending time and wondering why the writing doesn't work

CU, Ralf.

Subject: Re: LCWriter error when inputfile and outputfile only differ in "_" changed to "."

Posted by [samson](#) on Fri, 24 Apr 2009 10:00:25 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Ralf,

the problem you are describing is related to this bug

<http://bugs.freehep.org/browse/LCIO-60> which I reported one year ago. (Actually it is the same bug). The reason for this bug is that in LCIO every (SIO-)data-stream needs an unique identifier (without '.' characters) which (in the current LCIO implementation) only is derived from the file-name. In your case these identifiers for both of your files are the same and SIO (the low level data format for slcio) complains about the "duplicate stream name".

As described in the bug report. The same problem occurs if you want to open two different LCIO streams from the same file. In this case the problem is not that ease to avoid.

Cheers, J.

Subject: Re: LCWriter error when inputfile and outputfile only differ in "_" changed to "."

Posted by [ralf](#) on Fri, 24 Apr 2009 11:09:06 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thanks for the info. This explains also, why the name in the error message contains only underscores and no points.

And sorry for posting a known bug, but I didn't know that this bug list exists

Subject: Re: Problem using files with Cluster and CalorimeterHit collections

Posted by [sailer](#) on Wed, 27 May 2009 16:41:34 GMT

[View Forum Message](#) <> [Reply to Message](#)

Has this problem been resolved?

I run the Standard Reconstruction with ILCSoft 1.06 and LCIO 01-11 and cannot access the produced REC files using, for example "dumpevent", but also get the same error as in the other post

If I don't drop the CalorimeterHits from the DST file, this file becomes inaccessible due to that error as well.

Any idea what is causing this or how to avoid it?

Thanks and best Regards,
André

Subject: Re: Problem using files with Cluster and CalorimeterHit collections

Posted by [samson](#) on Fri, 29 May 2009 15:57:57 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Andre,

as far as I can see, there is no obvious solution to this problem.
So the short answer probably is the following: If you have an collection of clusters with links to

calorimeter hits you cannot drop the collections containing the calorimeter hits without corrupting the output file.

So for the time being the only way to allow users to drop the calorimeter hits would be to ask Mark Thomson to add an steering parameter to PandoraPFA. If one could remove all the XXX.addHit() calls for the calorimeter clusters with a PandoraPFA steering parameter, one could safely drop the calorimeter hit collection.

The long answer to this problem, would state that fixing this problem on the basis of (the c++ implementation) of LCIO is quite difficult. SIO (the current low level file i/o format of LCIO) relies on the fact that all objects that are pointed at by other objects are written to the file. In this case the cluster objects point to some calorimeter hits. If you drop (i.e. don't write) these hits the file gets corrupted. The main problem is due to the fact, that LCIO cannot easily check whether the hits, which the clusters point to are written to the output file, because the clusters don't have the information in which collections the corresponding hits live.

The only rather quick workaround on LCIO level for this problem would be an option to remove all calorimeter hits of the clusters before writing the clusters to the file.

Usually, I would suggest to post this issue to the LCIO bug tracker (<http://bugs.freehep.org/secure/Dashboard.jspa>), however even critical bugs in the c++ part of the bug tracker are currently ignored, which sheds quite a bad light on the current support situation of LCIO.

Maybe this requires some discussion on a "non-software" level.

Cheers, J.

Subject: Re: Problem using files with Cluster and CalorimeterHit collections
Posted by [gaede](#) on Tue, 02 Jun 2009 16:53:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Mark and Andre,

I am not sure I understand the problem with dropping the CalorimeterHits. Of course this is possible with the LCIOOutputProcessor and we are doing this in our Monte Carlo mass production. See the standard steering file:

stdreco.xml for example where REC files have the hits and DST files don't.

If you want to do this in your own code and not in Marlin (why would you?) you need to set the collection flags of the Cluster collection as described in the Doxygen documentation - note that the LCIOOutputProcessor of Marlin does this automatically for you, if you drop the CalorimeterHits.

Joergen's comments about this not being possible in LCIO/Marlin are incorrect - there is no need to change Pandora or LCIO to do this !

Also incorrect of course is the statement that we are currently ignoring critical bugs that are reported. As you know we are all very busy with the LOI and IDAG's followup to the LOI, thus we have focused in the last month on fixing all bugs/issues that were critical to the LOI Monte Carlo mass production and thereby left some minor feature requests or uncritical issues to be addressed after the LOI.

So, LCIO is of course fully supported and we will shortly also address all of the remaining unresolved issues.

Please let me know of any problem that you still have with dropping CalorimeterHits.
Cheers, Frank.

Subject: Re: Problem using files with Cluster and CalorimeterHit collections
Posted by [sailer](#) on Tue, 02 Jun 2009 17:49:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank and Mark,

I don't want to drop the CalorimeterHits collection. I just noticed that I am not able to use the slcio files coming from the stdreco.xml for example if they contain the CalorimeterHits collections.

I was trying to look at some events using "dumpevent", which crashed if the file was containing CalorimeterHits.

I don't know if this is a problem of my installation or "global".

Cheers,
André

Subject: Re: LCWriter error when inputfile and outputfile only differ in "_" changed to "&".

Posted by [engels](#) on Thu, 04 Jun 2009 09:43:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Ralf and Juergen,

i just checked in cvs the patch for fixing this.

please let me know if you still encounter any problem with this.

Cheers,
Jan

Subject: schema evolution with LCGenericObject
Posted by [antonio.bulgheroni](#) on Sat, 11 Jul 2009 14:26:24 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

almost two years ago we decided to derive a class from LCGenericObject to store a set of information. This is quite a simple structure made by N double values.

Now we realized that, in some case, it would be better to have at least one additional information, say one more integer value.

This is done very simply by replacing this code

```
MyStructure(int n) : LCGenericObject( 0, 0, n) {  
  
...  
  
}
```

with

```
MyStructure(int n) : LCGenericObject( 1, 0, n) {
```

```
...
```

```
}
```

The question is: will we be able to read old object with new definition? We don't need to be able to write the old object, but at least to read back the ones we have already created.

Thanks for your help,

Subject: Re: schema evolution with LCGenericObject
Posted by [gaede](#) on Sat, 11 Jul 2009 15:22:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Toto,

in principle one can read arbitrary LCGenericObjects from LCIO files, so of course one can also read older files where the number of LCGenericObject data members is different.

So it is up to your wrapper class to handle the 'schema evolution' when reading, e.g. by checking the number of ints before accessing them.

A small problem arises if you are using the LCFixedObject<I,J,K> template as this really requires the number of elements to be fixed. In this case you would have to modify your wrapper to use LCGenericObjectImpl, which should be somewhat straight forward.

-Frank.

Subject: Re: schema evolution with LCGenericObject
Posted by [antonio.bulgheroni](#) on Sat, 11 Jul 2009 15:34:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks!

Subject: Re: schema evolution with LCGenericObject
Posted by [killenberg](#) on Sat, 11 Jul 2009 17:28:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Antonio,

we did the same thing some time ago in MarlinTPC, we increased the number of ints from 4 to 5. The new function which accesses the new int just checks if there are 4 or 5 ints and returns 0 if there are only 4:

```
int ADCChannelMapping::getReadoutGroup() const {
    // This implementation should be backward compatible to the version
    // without readoutGroup, so the generic object might only contain 4 integers.
    // Return moduleGroup = 0 in this case.

    if ( _myObj->getNInt() >=5)
    {
        return _myObj->getIntVal( 4 ) ;
    }
    else
    {
        return 0;
    }
}
```

If you have svn and a graphical diff tool like kdiff3 installed you can see the changes in revision 1480 by issuing

```
svn diff -c1480 --diff-cmd kdiff3
svn://pi.physik.uni-bonn.de/MarlinTPC/branches/killenb/tpcconddata
```

If you want to see the change from LCFixedObject to LCGenericObject have a look at the changes in revision 1479, where we extended from 3 to 4 ints.

Cheers

Martin

Subject: Re: schema evolution with LCGenericObject
Posted by [antonio.bulgheroni](#) on Sat, 11 Jul 2009 18:39:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thanks Martin,

at the end, I'm doing something very similar.

cheers,
toto

Subject: Segmentation violation when copying ParticleIDs of ReconstructedParticle
Posted by [doublet](#) on Thu, 21 Jan 2010 15:35:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

Dear LCIO users,

I am facing a segmentation problem when trying to copy ReconstructedParticles. I am naively copying every information of one RecoParticle to another (see code below) and I found out that in several cases, the ParticleIDs fail to do so and give me this error. All of the RecoParticles only have 1 PID associated. Some fail, some don't. I am sorry I have made no other observation that could help understanding this. Anyone already had this problem or has an idea how to avoid this ?

Thanks in advance.

Best regards,
Philippe.

```
// CODE
for ( int i = 0 ; i < recobj->getNumberOfElements() ; i++ )
{
// The one to be read
ReconstructedParticle* aPFO = dynamic_cast<ReconstructedParticle*>( recobj->getElementAt(i) )
;
// The new to be created
IMPL::ReconstructedParticleImpl* recoPFO = new IMPL::ReconstructedParticleImpl() ;
// Naive copy
    recoPFO->setType( aPFO->getType() );
    recoPFO->setMomentum( aPFO->getMomentum() );
    recoPFO->setEnergy( aPFO->getEnergy() );
    recoPFO->setCovMatrix( aPFO->getCovMatrix() );
    recoPFO->setMass( aPFO->getMass() );
    recoPFO->setCharge( aPFO->getCharge() );
}
```

```

    recoPFO->setReferencePoint( aPFO->getReferencePoint() );
// The following commented lines make *** Break ***
    /*for ( unsigned int k = 0 ; k < aPFO->getParticleIDs().size() ; k++ )

        { recoPFO->addParticleID( aPFO->getParticleIDs()[k] ) ; }*/
// The next lines are OK
    recoPFO->setParticleIDUsed( aPFO->getParticleIDUsed() );
    recoPFO->setGoodnessOfPID( aPFO->getGoodnessOfPID() );
    for ( unsigned int k = 0 ; k < aPFO->getParticles().size() ; k++ )
        { recoPFO->addParticle( aPFO->getParticles()[k] ); }
    for ( unsigned int k = 0 ; k < aPFO->getClusters().size() ; k++ )
        { recoPFO->addCluster( aPFO->getClusters()[k] ); }
    for ( unsigned int k = 0 ; k < aPFO->getTracks().size() ; k++ )
        { recoPFO->addTrack( aPFO->getTracks()[k] ); }
    recoPFO->setStartVertex( aPFO->getStartVertex() );

myNewPFOsCollection->addElement( recoPFO ) ;

}

evt->addCollection( myNewPFOsCollection , _NewPFOsColName ) ;
// END

```

Subject: Re: Segmentation violation when copying ParticleIDs of ReconstructedParticle
 Posted by [gaede](#) on Thu, 21 Jan 2010 16:53:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Phillipe,

the ReconstructedParticle owns its ParticleIDs, thus when copying a ReconstructedObject one has to also copy the ParticleIDs.

Could you try the following:

```

// replace
// recoPFO->addParticleID( aPFO->getParticleIDs()[k] ) ;
// with

recoPFO->addParticleID( new ParticleIDImpl( aPFO->getParticleIDs()[k] ) ) ;

```

Frank.

Subject: Re: Segmentation violation when copying ParticleIDs of ReconstructedParticle
Posted by [doublet](#) on Thu, 21 Jan 2010 19:16:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Frank,

Thank you very much for your quick answer. However, it did not work as expected, probably due to a constructor problem.

Nevertheless, following your comment, I decided to again copy naively all the information of the ParticleID (code follows). And it now works.

Thanks again for the help.

I would suggest, since this may often happen to LCIO users, that each LObject class should have a sort of "copy" function that would do such things automatically. It is often useful to copy a collection if one wants to work on it and do some modifications or manipulations.

Best regards,
Philippe.

```
//CODE
for ( unsigned int k = 0 ; k < aPFO->getParticleIDs().size() ; k++ )
{
  ParticleID *pid = aPFO->getParticleIDs()[k] ;
  IMPL::ParticleIDImpl* implPID = new IMPL::ParticleIDImpl() ;
  implPID->setType( pid->getType() ) ;
  implPID->setPDG( pid->getPDG() ) ;
  implPID->setLikelihood( pid->getLikelihood() ) ;
  implPID->setAlgorithmType( pid->getAlgorithmType() ) ;
  for ( unsigned int l = 0 ; l < pid->getParameters().size() ; l++ )
    { implPID->addParameter( pid->getParameters()[l] ) ; }
  recoPFO->addParticleID( implPID ) ;
}
```

Subject: LCIO install error

Posted by [Ryan_Page](#) on Thu, 01 Apr 2010 13:40:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

I am trying to install lcio and have hit a problem. It started because I did not have zlib installed but after i installed it I tried to install lcio again and it gave this error:

```
/exports/gpfs/phrfp/products_32/zlib-1.2.4/lib/libz.so: could not read symbols: File in wrong format
```

The version of zlib is new, from what I can tell it was released this month. Has anyone else run into this problem, any suggestions would be very welcome!

I am running on this system

Linux - Scientific Linux SL release 4.3 (Beryllium) , gcc (GCC) 3.4.6

Thanks

Ryan

Subject: Error Using LCIOWriter

Posted by [alex.bogert](#) on Fri, 02 Apr 2010 17:06:32 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

I am trying to create a .lcio file from a .stdhep file using org.lcsim.util.lc.LCIOWriter. I am adding a few sub-collections of MCParticles to the EventHeader then using LCIOWriter to write out the event. When I look at the resulting .lcio file in JAS3 a large number of events give a NullPointerException. I have recreated the .lcio file several times and the null events seemed to be randomly distributed amongst the 1000 events in the file. I have also tried flushing the LCIOWriter after each write, this causes no change in behavior. Any suggestions how I could resolve this issue? On average I'm losing about 30% of the events to NullPointerExceptions. Also, I've printed out event information just before Writing and the events are not null.

Edit: I removed the sub-collection of MCParticles from the list of collection for LCIOWriter to write out. This seems to fix the Null events. I would still like to be able to write out sub-collections.

Thanks,
Alex

Subject: Re: Error Using LCIOWriter
Posted by [NormanGraf](#) on Mon, 05 Apr 2010 22:04:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Alex,

Can you provide more details on how you are writing subcollections and also why you need this functionality?

Correctly handling particle hierarchies is a tricky business and having this extra information would make it easier to assist you.

Thanks,
Norman

Subject: Re: Error Using LCIOWriter
Posted by [alex.bogert](#) on Mon, 05 Apr 2010 23:45:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

I am organizing the MCParticles in each event into BHadrons and their primary decay products. Then using a covariance matrix to calculate the Impact Parameter Significance of each BHadron. Recently I was informed the vertexing code I will be using is written in C++. For convenience I thought I could save the organisation I had already accomplished in a lcio formatted file. I was trying to add the list of BHadrons to the EventHeader:

```
event.put("BHadron", bhMCPList);  
lcWriter.addWriteOnly("BHadron");  
lcWriter.write(event);
```

Again this worked for majority of the events. Except random events are being reported as NullPointerExceptions when I read the lcio file in jas3. I was also planning to save the Significance of each BHadron in MetaData but haven't got around to trying that. Hope this is clear.

Alex

Subject: Re: LCIO install error
Posted by [engels](#) on Tue, 06 Apr 2010 10:01:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Ryan,

the problem seems to be a 32/64 bit incompatibility between LCIO and the libz.so installed on your system.

LCIO is built per default in 32 bit (even on 64 bit platforms)

There is a cmake flag called BUILD_32BIT_COMPATIBLE (set to ON per default) which you can turn off to build LCIO in native 64 bit, i.e.:

```
cmake -DBUILD_32BIT_COMPATIBLE=OFF ..
```

you can easily check if your libz.so is 32 or 64 bit with:

```
file $(readlink -f /exports/gpfs/phrfp/products_32/zlib-1.2.4/lib/libz.so)
```

Cheers,
Jan

Subject: Re: LCIO install error

Posted by [Ryan_Page](#) on Tue, 06 Apr 2010 10:38:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thanks, the version I have is 64bit. I have changed env variables CMAKE_LIBRARY_PATH and CMAKE_INCLUDE_PATH to the dir that contains the 32 bit version of zlib, but now cmake can not find it, are there any other variable that I should be setting?

Thanks

Ryan

Subject: Re: LCIO install error

Posted by [engels](#) on Tue, 06 Apr 2010 12:11:40 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Ryan,

this may happen because cmake stores configurations in a file called CMakeCache.txt inside your build directory.

Please erase that file, set the env vars CMAKE_LIBRARY_PATH and CMAKE_INCLUDE_PATH and re-run cmake, it should then find your 32 bit zlib.

Cheers,
Jan

Subject: Re: LCIO install error
Posted by [Ryan_Page](#) on Tue, 06 Apr 2010 12:52:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Thanks, I tried that sadly to know affect. In the configuration file i have set the following

```
ilcsoft.env["CMAKE_LIBRARY_PATH"]="/usr/lib"  
ilcsoft.env["CMAKE_INCLUDE_PATH"]="/usr/include"  
ilcsoft.env["ZLIB_LIBRARY"]="/usr/lib/libz.so.1.2.1.2"  
ilcsoft.env["ZLIB_INCLUDE_PATH"]="/usr/include"
```

I have also tried setting these using export, but I still have the same problem.

Is there another cache somewhere or another variable that I can play with?

Subject: Re: LCIO install error
Posted by [engels](#) on Tue, 06 Apr 2010 13:08:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

ok, could you please send me your ilcinstall configuration file?

Thanks,
Jan

Subject: Re: LCIO install error
Posted by [Ryan_Page](#) on Tue, 06 Apr 2010 13:41:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Here it is.

Thanks

File Attachments

1) [release_v01-08-base.cfg](#), downloaded 460 times

Subject: Re: LCIO install error

Posted by [engels](#) on Tue, 06 Apr 2010 15:23:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Ryan,

if I understand correctly you have a 32bit version of zlib installed in
/exports/gpfs/phrfp/products_32/zlib-1.2.4

if this is true than what you want in your ilcinstall configuration file is:

```
# find 32bit zlib in /exports/gpfs/phrfp/products_32/zlib-1.2.4
zlib_32bit_path="/exports/gpfs/phrfp/products_32/zlib-1.2.4"
ilcsoft.env["CMAKE_LIBRARY_PATH"]=zlib_32bit_path+"/lib"
ilcsoft.env["CMAKE_INCLUDE_PATH"]=zlib_32bit_path+"/include"
please find the modified configuration file attached in this message.
```

since cmake stores the paths in the CMakeCache.txt you should first remove the file:

```
rm /exports/gpfs/phrfp/ilcsoft_32/base/lcio/v01-12-02/build/CMakeCache.txt
```

and then re-run ilcinstall with the attached cfg file.

P.S. there is already a newer patch release of ilcsoft (v01-08-01) which contains important bug fixes.

Please find more details here and the newer ilcinstall version here

Cheers,

Jan

File Attachments

1) [release_v01-08-base-32bit.cfg](#), downloaded 497 times

Subject: Re: LCIO install error

Posted by [Ryan_Page](#) on Wed, 07 Apr 2010 15:07:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

I have made some changes to the sym links for zlib and that seems to have fixed the problem.

Thanks very much for your help

Ryan

Subject: TrackerHitImpl const-ness of setPosition

Posted by [jabernathy](#) on Fri, 27 Aug 2010 19:45:30 GMT

[View Forum Message](#) <> [Reply to Message](#)

Just wondering if changing the setPosition signature from:

```
void setPosition (double pos[3])
```

to

```
void setPosition (const double pos[3])
```

would make sense. This would allow:

```
new_hit->setPosition ( old_hit->getPosition() );
```

Subject: Re: TrackerHitImpl const-ness of setPosition

Posted by [engels](#) on Mon, 06 Sep 2010 15:10:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Jason,

yes indeed, this (and probably other) method arguments should be const. I will look into fixing this soon.

Thanks again for your feedback,
Jan

Subject: problems reading lcio files in direct access mode
Posted by [daniel_jeans](#) on Wed, 10 Nov 2010 14:32:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear LCIO experts,

We are having problems reading lcio files (using the readNextEvent method of LCReader) written with v01-51 using code compiled against lcio v01-51, when using the DirectAccess mode.

I attach a small test code to illustrate the problem. It tries to read the first 10 events of a file using the readNextEvent method. (we compile as follows: "g++ -I\${LCIO}/src/cpp/include -L\${LCIO}/lib -llcio -o testlcio.exe testlcio.cc")

When we compile against lcio v1-11, and run on a file made with v1-11, it runs fine both in direct and not-direct modes.

When we compile against v01-51, and run on a file made with v01-11, it also works fine in both modes.

When we compile against v01-51, and run on a file made with v01-51, indirect mode works fine, but using directAccess, readNextEvent (and nextRunHeader) never gives any valid events.

We notice that if we first do a readEvent (with valid run and event numbers), then readNextEvent does function properly, also in this last case.

Is this expected behaviour? It is inconvenient for us...

Thanks,
Daniel and Manqi.

File Attachments

1) [testlcio.cc](#), downloaded 599 times

Subject: Re: problems reading lcio files in direct access mode
Posted by [gaede](#) on Wed, 10 Nov 2010 15:14:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Daniel and Manqi,

this is a bug that is fixed in v01-51-01 - so please switch to that version for your use case.

Cheers, Frank.

Subject: Re: problems reading lcio files in direct access mode
Posted by [daniel_jeans](#) on Wed, 10 Nov 2010 15:44:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

thanks, that fixed it.

Subject: GeomConverter build failedh
Posted by [poojasaxena](#) on Tue, 18 Jan 2011 10:01:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

I am working on UBUNTU (version 10.04) & have installed the following version of JAVA & maven:

Apache Maven 3.0.2

Java version: 1.6.0_20

I have build several projects(like lcsim, lcsim-cotrib, lcsim parent) using mvn (mvn -DskipTests=true clean install) but while building GeomConverter project, build failed error message is coming. Please find here the full error message:

Error Message:

[INFO] Building GeomConverter

[INFO] task-segment: [clean, install]

[INFO] -----

[INFO] [clean:clean {execution: default-clean}]

[INFO] [resources:resources {execution: default-resources}]

[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!

[INFO] Copying 21 resources

```
[INFO] [compiler:compile {execution: default-compile}]
[INFO] Compiling 359 source files to /home/poojasaxena/CLIC/GeomConverter/target/classes
[INFO] [antrun:run {execution: default}]
[INFO] Executing tasks
[mkdir] Created dir: /home/poojasaxena/CLIC/GeomConverter/target/test-output
[INFO] Executed tasks
[INFO] [resources:testResources {execution: default-testResources}]
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is
platform dependent!
[INFO] Copying 38 resources
[INFO] [compiler:testCompile {execution: default-testCompile}]
[INFO] Compiling 75 source files to /home/poojasaxena/CLIC/GeomConverter/target/test-classes
[INFO] [surefire:test {execution: default-test}]
[INFO] Tests are skipped.
[INFO] [jar:jar {execution: default-jar}]
[INFO] Building jar: /home/poojasaxena/CLIC/GeomConverter/target/GeomConverter-1.
11-SNAPSHOT.jar
[INFO] [assembly:attached {execution: default}]
[INFO] Reading assembly descriptor:
/home/poojasaxena/CLIC/GeomConverter/src/assembly/bin.xml
[INFO] -----
[ERROR] BUILD ERROR
[INFO] -----
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-assembly-plugin:2.2-beta-5:at
tached (default) on project GeomConverter: Failed to create assembly: Error adding file-set for
'hep.testdata.aida:lizard:aida:1.0' to archive: Error adding archived file-set.
PlexusResourceCollection not found for:
/home/poojasaxena/.m2/repository/hep/testdata/aida/lizard/1.0/lizard-1.0.aida: No such archiver:
'aida'. -> [Help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following
articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/MojoExecutionException
*****
```

please reply.
Thanks in advance

Subject: Re: GeomConverter build failedh
Posted by [gaede](#) on Tue, 18 Jan 2011 10:40:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

this seems to be a problem that with the org.lcsim build system not related to LCIO. Please report this error in the org.lcsim forum.

Frank.

Subject: Re: GeomConverter build failedh
Posted by [poojasaxena](#) on Tue, 18 Jan 2011 11:55:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

sorry for reporting it at wrong place. Now, I reported at the org.lcsim forum.

thanks
pooja

Subject: Merging reconstructed particles
Posted by [doublet](#) on Fri, 18 Mar 2011 09:28:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear LCIO users,

1) I would like to merge ReconstructedParticles which are present in a collection "myParticles" but appear in a collection of composites ReconstructedParticles "myJets" where I have added them via the method addParticles().

My concrete case is for bremsstrahlung photons that I want to add to their electron :
- If I do electron.addParticle(photon), then remove photon from "myParticles", am I sure that the changes are also made in "myJets" collection ?
Is this the best way to do so ?
Or do I have to create new collections and use new IMPLs ?

2) And by the way, when I am using addParticles(), is there a way to recalculate the energy-momentum without using an IMPL ?

Thank you in advance for you help.

Regards,
Philippe.

Subject: const correctness of LCTime
Posted by [caiazza](#) on Fri, 22 Jul 2011 16:46:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear developers.
Would it be possible to declare const the get functions of the LCTime (and whatever other class IMHO) so that they can be used in a const function?

Subject: LCIO Wiki
Posted by [caiazza](#) on Thu, 13 Oct 2011 15:18:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

I don't know if anybody noticed that the LCIO WIKI has been hacked and filled with junk. Maybe the manager of those pages of ILC SOFT should clean up that page or remove it completely

Subject: Re: LCIO Wiki
Posted by [engels](#) on Mon, 17 Oct 2011 13:17:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Stefano,

thanks for the notice. The pages have been cleaned up.

Cheers, Jan.

Subject: Seg Fault when adding collections
Posted by [tonyprice1877](#) on Wed, 15 Feb 2012 12:29:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi All,

I hope somebody can help me I am fairly new to the iLCSoft/LCIO stuff. I am trying to figure out how to add Collections to an .slcio file and I keep getting a seg fault for some reason! I have stripped my code back to filling a ReconstructedParticleVec with all the ReconstructedParticles and then trying to write them back as a new collection the code is simply as follows

```
void MyTestJetFinderProcessor::processEvent( LCEvent * evt ) {

    // this gets called for every event
    // usually the working horse ...

    //////////////////////////////////////
    ////////// Look at the PandoraPFO //////////
    //////////////////////////////////////

    //get the reconstucted particles collection
    LCCollection* col_rp = evt->getCollection( _colName_ReconstructedParticle ) ;
    if( col_rp != 0 )
    {
        _vParticles.clear();
        _vJets.clear();

        int n = col_rp->getNumberOfElements();

        //loop through and get the particles
        for(int i=0; i<n; i++)
        {
            ReconstructedParticle* rp = dynamic_cast<ReconstructedParticle*>( col_rp->getElementAt(i) );
            //std::cout << rp->getParticles().size() << std::endl;
            _vParticles.push_back(rp);
        }

        IMPL::LCCollectionVec* lccParticlesOut = new
        IMPL::LCCollectionVec(LCIO::RECONSTRUCTEDPARTICLE);
        for(unsigned i=0; i<_vParticles.size(); i++) lccParticlesOut->addElement( _vParticles[i] );
        evt->addCollection(lccParticlesOut, _lcJetOutName);
        } // end of if( col_rp != 0 )
    std::cout << "Event " << _nEvt << " analysed" << std::endl;

    //increment the event counter
    _nEvt ++ ;
}
```

```
}
```

if I remove the line

```
lccParticlesOut->addElement( _vParticles[i] );
```

then there is no seg fault. Can anybody see what I am doing wrong?

Tony

Subject: Re: Seg Fault when adding collections
Posted by [gaede](#) on Wed, 15 Feb 2012 12:38:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Tony,

LCIO collections take ownership of their elements.

If you copy elements (pointers to elements, really) from an existing collection to a new one, you need to flag the new collection as a 'subset collection' (i.e. only pointers are stored and elements not deleted at end of scope):

```
lccParticlesOut->setSubset(true);
```

Cheers, Frank.

Subject: Re: Seg Fault when adding collections
Posted by [tonyprice1877](#) on Wed, 15 Feb 2012 13:05:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank,

You are a superstar I have been trying to figure this out all morning

Thank you very much

Tony

Subject: Re: Seg Fault when adding collections
Posted by [tonyprice1877](#) on Wed, 15 Feb 2012 14:07:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank,

Unfortunately not all my problems were fixed as I first thought... When I just copy the collection with the code I supplied earlier then Marlin runs fine and anajob shows the collection and dumpevent prints all the particles.

What I am actually doing however is forming some jets from the particles and then storing the jets in the lcio.

The `lccMyJetsOut->setSubset(true)`; flag is once again required or Marlin seg faults and in anajob my jet collection appears but when doing dumpevent it gets to my collection and seg faults without printing any particles. Is there another flag I need since I am reading the particles from "PandoraPFOs" combining some into a new particle and storing again?

Tony

Subject: Re: Seg Fault when adding collections
Posted by [gaede](#) on Wed, 15 Feb 2012 14:59:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Tony,

can you post the code ?

Normally `isSubset` should not be used with objects that you created yourself - also when pointing back to other reconstructed particles (PandoraPFO).

-Frank.

Subject: Re: Seg Fault when adding collections
Posted by [tonyprice1877](#) on Wed, 15 Feb 2012 15:04:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank,

I have attached my .cc file I cannot really see much difference between the basics of

MarlinFastJet which does work for me....

I basically get all PandoraPFOs, pass them through my jet finder where I combine particles and then try to write these to file when a certain number of jets remain

Tony

File Attachments

1) [MyTestJetFinderProcessor.cc](#), downloaded 378 times

Subject: Re: Seg Fault when adding collections
Posted by [gaede](#) on Wed, 15 Feb 2012 17:15:23 GMT
[View Forum Message](#) <> [Reply to Message](#)

hi tony,

as far as I can see there are a few issues with your code:

- you should not use setsubset with collections of elements that you created -> this will result in corrupt LCIO files (l.278)

- a memory leak in ee_Durham_Jet_Finder_Exclusive (l. 158):

you create a new particle on the heap every time you combine two particles but in the end you only add n jet particles to the collection

-> you need to delete the intermediate ReconstructParticles (possibly in CombineParticles)

- the following will result in an undefined pointer in the LCIO file (if the vector is empty the first element is undefined):

```
if(p1_particles.size()==0)
{
  jet->addParticle( p1_particles[0] );
}
```

hope this helps to find your problem.

Cheers, Frank.

Subject: Re: Seg Fault when adding collections
Posted by [tonyprice1877](#) on Thu, 16 Feb 2012 13:08:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Frank,

I went through my code tried changing multiple things with no luck. I then rewrote it where I use the ReconstructedParticles in a different way and all seems fine. As you suggested it was probably something to do with the CombineParticles function

Thank you for your help

Tony

Subject: Request: addition of "quality" word to TrackerData
Posted by [rosemann](#) on Fri, 20 Apr 2012 15:14:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

To all LCIO users: are there any objections to the request below?

I'd like to request the creation of another data member with accessors and mutators for EVENT::TrackerData ; similar to EVENT::TrackerPulse or EVENT::TrackerHit

(e.g. confer

http://lcio.desy.de/v02-00/doc/doxygen_api/html/classEVENT_1_1TrackerData.html

http://lcio.desy.de/v02-00/doc/doxygen_api/html/classEVENT_1_1TrackerPulse.html

http://lcio.desy.de/v02-00/doc/doxygen_api/html/classEVENT_1_1TrackerHit.html)

We (LCTPC/testbeam data reco & analysis) would like to have a data field, where we can describe the quality of a channel in the spirit of the description of TrackerData : "TrackerData contains the corrected (calibrated) raw tracker data."

Since it is an addition with no change to the existing interfaces I wouldn't expect any issues with downward compatibility.

Best regards,

Christoph

Subject: Re: Request: addition of "quality" word to TrackerData
Posted by [rosemann](#) on Fri, 04 May 2012 15:13:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

two weeks later no one expressed any concern (or: no one noticed the request?), so I'd kindly ask Jan/Frank to implement the addition of this data field.

Cheers,
Christoph

Subject: SIOWriter couldn't write event header
Posted by [grenier](#) on Wed, 01 Aug 2012 16:50:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

Trying to run 2 processors in Marlin, the second one being LCIOOutputProcessor, I got a crash at the first event with the following message :

```
*****
```

A runtime error occured - (uncaught exception):

lcio::IOException: [SIOWriter::writeEvent] couldn't write event header to stream:
_scratch_RawHits_slcio0

Marlin will have to be terminated, sorry.

```
*****
```

How could I debug such a thing ? Is there a way to tell SIO to be more informative ?
Is it a usual message if I've not set properly something ?

In case you want to try to reproduce it :

I'm using ilcsoft release v01-14,

I'm running processors from the Trivent package release 8 (svn co

<https://svnsrv.desy.de/public/trivent/Trivent/trunk> Trivent)

Look into the README file for instructions on how to compile the package.

I'm using the Marlin xml file which is under <Trivent>/steer/streamout.xml where I've

uncommented the LCIOOutputProcessor.
The input file I'm using is on the calice grid at
/grid/calice/SDHCAL/TB/CERN/PS_April2012/RAW/DHCAL_713775_I0_0.slcio

Thanks for any advices.
GÃ©rald

Subject: segfaulting PIDHandler in python
Posted by [jfstrube](#) on Sat, 22 Nov 2014 09:47:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

I have a DST + LCFIPlus Jet collections.
The xml file that was used to create the file from the DST is attached.
I am reading this file via the python bindings.
from pyLCIO.io.LcioReader import LcioReader
from ROOT import UTIL
import sys
for dst in sys.argv[1:]:
 reader = LcioReader(dst)
 for event in reader:
 refinedJetCollection = event.getCollection("RefinedJets")
 pidh = UTIL.PIDHandler(refinedJetCollection)

Instantiating the PIDHandler leads to a seg fault.
However, not in every event.
Please let me know if you can reproduce this bug.
There might be processors in the xml which are not distributed with ILCSOFT. Please comment them out.

File Attachments

1) [ww6j.xml](#), downloaded 162 times

Subject: Re: segfaulting PIDHandler in python
Posted by [jfstrube](#) on Sat, 22 Nov 2014 10:29:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

I was able to get a stack trace:

#5 0x00000001108c2820 in UTIL::PIDHandler::~~PIDHandler (this=0x7fe0d57d4880) at /Users/jstrube/WorkDir/lcio/v02-04-03/src/cpp/src/UTIL/PIDHandler.cc:120
#6 0x0000000110a5ba47 in ROOT::delete_UTILcLcLPIDHandler (p=0x7fe0d57d4880) at /Users/jstrube/WorkDir/lcio/v02-04-03/build/rootdict/UTIL.cxx:7063
#7 0x000000010eedbe70 in TClass::~Destructor ()
#8 0x000000010e7a0094 in PyROOT::(anonymous namespace)::op_dealloc ()
#9 0x000000010e347083 in subtype_dealloc ()
#10 0x000000010e322a66 in insertdict_by_entry ()
#11 0x000000010e31f1be in PyDict_SetItem ()
#12 0x000000010e381d05 in PyEval_EvalFrameEx ()
#13 0x000000010e38009d in PyEval_EvalCodeEx ()
#14 0x000000010e37f9d6 in PyEval_EvalCode ()
#15 0x000000010e3a9544 in PyRun_FileExFlags ()
#16 0x000000010e3a90c1 in PyRun_SimpleFileExFlags ()
#17 0x000000010e3bed1e in Py_Main ()
#18 0x00007fff8b7f55fd in start ()

Note that this is a bit suspicious, since I'm seemingly getting the segfault when calling the PIDHandler constructor, not the destructor, but it might be related to ROOT's interaction with the python garbage collector.

Subject: Re: segfaulting PIDHandler in python
Posted by [jstrube](#) on Sat, 22 Nov 2014 13:00:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

It's indeed the destructor.
adding a "del pidh" in each iteration gets rid of the segfault.