Physics Reconstruction Documentation

Overview

The reconstruction is implemented within the HPS Java project as a chain of org.lcsim Drivers (event processors) described by lcsim xml and run by the lcsim job manager. The EvioToLcio command line tool is used to convert the EVIO to LCIO events using an LCSimEventBuilder such as LCSimEngRunEventBuilder. The LCIO events are accessed in Java code through the EventHeader API. The Drivers add output collections to the event such as tracks, calorimeter clusters, reconstructed particles and vertices. The combined data/physics LCIO events are written out to an LCIO file, which can subsequently be converted to ROOT DST Data Format. The LCIO events may also be loaded back into the HPS Java environment for analysis.

Basic Steps

These are the steps performed in reconstructing the data:

1. EvioToLcio command line tool is run with command line parameters like the EVIO file(s) and the path to the XML steering file.
2. The JobControlManager loads the steering file which defines the chain of reconstruction Drivers and their parameters.
3. Detector Conditions including per channel calibrations are read and applied in Driver *detectorChanged* methods.
4. Each EvioEvent is read with EvioReader and converted to an LCIO raw data event using an appropriate LCSimEventBuilder.
5. The HPS Java reconstruction runs on the LCIO event, adding additional reconstruction collections to the events.
6. The events are written to an LCIO file containing the results of the recon.
7. The output LCIO file is converted to ROOT DST Data Format for analysis.

Steps 1-6 are typically performed in the same job using the EvioToLcio command line utility.

User analysis can be performed on the LCIO files using the job manager command line tool or within the ROOT environment using the DSTs.

Reconstruction Drivers

The reconstruction Driver chain is defined in production steering files such as EngineeringRun2015FullRecon.lcsim, which are kept in this SVN folder and typically accessed as a class resource from a jar file.

Order	Driver Name	Driver Class	Description
1	RfFitter	RfFitterDriver	converts accelerator's RF wave form to time and inserts into event
2	EcalRunningPedestal	EcalRunningPedestalDriver	calculates per channel running averages for ECal signal pedestals
3	EcalRawConverter	EcalRawConverterDriver	converts ECal digits to CalorimeterHit collection with energy and time measurements
4	ReconClusterer	ReconClusterDriver	performs calorimeter clustering algorithm on ECal hits
5	CopyCollection	CopyClusterCollectionDriver	copies calorimeter clusters to new collection to preserve uncorrected energy measurements
6	RawTrackerHitSensorSetup	RawTrackerHitSensorSetup	assigns RawTrackerHits to their sensors for use by track recon
7	RawTrackerHitFitterDriver	RawTrackerHitFitterDriver	fits ADC vs time signal and stores the results, associated to each raw hit
8	TrackerHitDriver	DataTrackerHitDriver	creates stereo pairs from SVT strip hits
9	HelicalTrackHitDriver	HelicalTrackHitDriver	creates 3D hit clusters from stereo pairs
10	TrackReconSeed345Conf2Extd16	TrackerReconDriver	track finding using layers 3, 4 & 5 as a seed, layer 2 to confirm, and layers 1 and 6 to extend
11	TrackReconSeed456Conf3Extd21	TrackerReconDriver	track finding using layers 4, 5 & 6 as a seed, layer 3 to confirm, and layers 2 and 1 to extend
12	TrackReconSeed123Conf4Extd56	TrackerReconDriver	track finding using layers 1, 2 & 3 as a seed, layer 4 to confirm, and layers 5 and 6 to extend

13	TrackReconSeed123Conf5Extd46	TrackerReconDriver	track finding using layers 1, 2 & 3 as a seed, layer 5 to confirm, and layers 4 and 6 to extend
14	MergeTrackCollections	MergeTrackCollections	merges collections from track finding into a single output collection
15	GBLRefitterDriver	GBLRefitterDriver	performs GBL track refit
16	TrackDataDriver	TrackDataDriver	adds additional collections containing track information to the output event
17	ReconParticleDriver	HpsReconParticleDriver	creates output reconstructed particle collections, associating tracks with clusters also performs vertex reconstruction and creates vertex collection
18	LCIOWriter	LCIODriver	writes output LCIO file
19	CleanupDriver	ReadoutCleanupDriver	cleans up readout state for next event (clears assignments of SVT raw hits to sensors)

Data Collections

Collection	Java Class	Created By	Description
BeamspotConstrainedMollerCandidates	ReconstructedParticle	HpsReconParticleDriver	Møller candidate particles required to point back to beamspot at the target
BeamspotConstrainedMollerVertices	Vertex	HpsReconParticleDriver	Møller vertices required to point back to beamspot at the target
BeamspotConstraintedV0Candidates	ReconstructedParticle	HpsReconParticleDriver	electron-positron candidate particles required to point back to beamspot at the target
BeamspotConstraintedV0Vertices	Vertex	HpsReconParticleDriver	electron-positron vertices required to point back to beamspot at the target
EcalCalHits	CalorimeterHit	EcalRawConverterDriver	calibrated ECal Hits
EcalClusters	Cluster	ReconClusterDriver	reconstructed ECal clusters with uncorrected energies
EcalClustersCorr	Cluster	CopyClusterCollectionDriver	reconstructed ECal clusters with corrected energies
EcalReadoutHits	RawTrackerHit	EcalEvioReader	ECal Hits in ADC counts
EpicsData	GenericObject	LCSimEngRunEventBuilder	EPICS data banks
FADCGenericHits	GenericObject	EcalEvioReader	additional FADC readout information
FinalStateParticles	ReconstructedParticle	HpsReconParticleDriver	final state particles (electrons, positrons, photons) with 4-momenta
GBLKinkData	GenericObject	GBLRefitterDriver	extra GBL track kink data
GBLKinkDataRelations	LCRelation	GBLRefitterDriver	relation from GBLTracks to GBLKinkData
GBLTracks	Track	GBLRefitterDriver	tracks created from GBL refit
HelicalTrackHitRelations	LCRelation	HelicalTrackHitDriver	relation from HelicalTrackHits to StripClusterer_SiTrackerHitStrip1D
HelicalTrackHits	TrackerHit	HelicalTrackHitDriver	3D hits combining StripClusterer_SiTrackerHitStrip1D hits in axial/stereo layers
MatchedToGBLTrackRelations	LCRelation	GBLRefitterDriver	relation from MatchedTracks to GBLTracks
MatchedTracks	Track	TrackerReconDriver	primary collection of reconstructed tracks merged from collections with tracks generated from different strategies
PartialTracks	Track	MergeTrackCollections	collection of tracks which have a set of hits that are a strict subset of another track
RotatedHelicalTrackHitRelations	LCRelation	HelicalTrackHitDriver	relation from RotatedHelicalTrackHit to HelicalTrackHit
RotatedHelicalTrackHits	TrackerHit	HelicalTrackHitDriver	HelicalTrackHits rotated into SeedTracker tracking frame: xy, yz, zx
SVTFittedRawTrackerHits	LCRelation	RawTrackerHitFitterDriver	relation from SVTRawTrackerHits to SVTShapeFitParameters
SVTRawTrackerHits	RawTrackerHit	SvtEvioReader	Si sensor single strip hits
SVTShapeFitParameters	GenericObject	RawTrackerHitFitterDriver	results of the ADC vs sample number fits for SVT data
StripClusterer_SiTrackerHitStrip1D	TrackerHit	DataTrackerHitDriver	1D Si strip clusters
TargetConstrainedMollerCandidates	ReconstructedParticle	HpsReconParticleDriver	Møller candidate particles with the vertex z fixed to the target position and (x,y) constrained to beamspot

TargetConstrainedMollerVertices	Vertex	HpsReconParticleDriver	Møller vertices with the vertex z fixed to the target position and (x,y) constrained to beamspot
TargetConstrainedV0Candidates	ReconstructedParticle	HpsReconParticleDriver	electron-positron pairs with the vertex z fixed to the target position and (x,y) constrained to beamspot
TargetConstrainedV0Vertices	Vertex	HpsReconParticleDriver	electron-positron vertices with the vertex z fixed to the target position and (x,y) constrained to beamspot
TrackData	GenericObject	TrackDataDriver	additional track information
TrackDataRelations	LCRelation	TrackDataDriver	relation from TrackData to a Track
TrackResiduals	GenericObject	TrackDataDriver	X & Y track residuals calculated at the stereo hit position
TrackResidualsRelations	LCRelation	TrackDataDriver	relation from TrackResiduals to a Track
TriggerBank	GenericObject	LCSimEngRunEventBuilder	trigger information for the event
UnconstrainedMollerCandidates	ReconstructedParticle	HpsReconParticleDriver	Møller particle candidates with unconstrained vertex
UnconstrainedMollerVertices	Vertex	HpsReconParticleDriver	unconstrained Møller vertices
UnconstrainedV0Candidates	ReconstructedParticle	HpsReconParticleDriver	electron-positron pairs with unconstrained vertex
UnconstrainedV0Vertices	Vertex	HpsReconParticleDriver	unconstrained electron-positron vertices

Algorithm Details

Data Conversion

The LCSimEventBuilder defines an interface for converting from EVIO to LCIO events, with the LCSimEngRunEventBuilder providing the current implementation of this conversion process. EVIO collections are processed by a reader which gets raw bank data and converts it into a typed LCIO collection.

SVT data banks are handled by an SvtEvioReader and converted into RawTrackerHit and GenericObject collections.

Various modes of EVIO ECal data are converted using the EcalEvioReader.

The default builder will also convert and write DAQ config information, EPICS data, and scaler bank data into the output LCSim events, if these banks are present in the EVIO data.

Track Reconstruction

These are the primary steps involved in the HPS Java track reconstruction:

1. RawTrackerHitFitterDriver is used to fit the ADC vs time signals from the raw data and writes a new collection with the fit result.
2. DataTrackerHitDriver creates stereo hit pairs from the strip hits along with the fit results.
3. HelicalTrackHitDriver creates 3D hits (clusters) from input stereo hits.
4. TrackerReconDriver runs track finding on the 3D hit collection.
    a. Track finding runs multiple times with different tracking strategy files, creating a track collection for each strategy used.
5. The MergeTrackCollections Driver is used to merge the multiple track collections into a single output collection.
6. The GBLRefitterDriver refits the tracks using GBL and writes a number of additional output collections with this information.
7. TrackDataDriver adds a Generic Object collection containing additional information about the track for persistency.

The tracking packages in lcsim form the basis for HPS's tracking algorithms through usage and extension. Seed Tracker is used for track finding using a set of input tracking strategies.

The track fit from lcsim is further refined using a Java implementation (port) of the GBL C++ algorithm.

Each track has a TrackType assigned which indicates the SeedTracker algorithm used, in a bitwise fashion, and sets bit 6 (2^5=32) if the track was refined by GBL. The TrackType is inherited by any particle and is obtained with the getType() method. See: TrackType and StrategyType for details.

Additional References

This paper describes the LCIO track parameters.

These slides provide some details about how tracking strategies are used (see pages 4 & 8).

Cluster Reconstruction

These are the basic steps of the ECal reconstruction:

1. EcalRawConverterDriver converts RawTrackerHit input collection into CalorimeterHit collection using the EcalRawConverter.
2. ReconClusterDriver uses the ReconClusterer to create calorimeter Cluster collection from input hits collection.

3. CopyClusterCollectionDriver copies the clusters (with raw energies) to a different collection.

The copied collection will be updated with corrected energies in the step which creates recon particles.

Additional References

This CLAS Note describes the basic clustering algorithm.

This HPS Note covers position corrections and other analysis.

Reconstructed Particles

The ReconParticleDriver creates ReconstructedParticle objects representing the final state particles from the event reconstruction. These are tracks with matching clusters (when applicable). It also performs vertex reconstruction and creates a number of candidate particle collections.

The ReconParticleDriver is sub-classed by the actual HpsReconParticleDriver from the steering which adds Møller candidate collections.

Java Packages

HPS Java Reconstruction Packages

The HPS Java Documentation can be used to browse the packages and classes used for physics reconstruction.

Java Package	Description	Notes	Module
org.hps.evio	converts EVIO raw data to LCIO	evio readers for converting EVIO raw data to LCIO events	evio
org.hps.recon.ecal	ECal reconstruction utilities	primarily for converting from raw data to CalorimeterHits	ecal-recon
org.hps.recon.ecal.cluster	ECal hit clustering framework	includes recon clustering and GTP/CTP hardware emulation clusterers	ecal-recon
org.hps.recon.tracking	track reconstruction from SVT hits	based on Seed Tracker from lcsim	tracking
org.hps.recon.tracking.gbl	GBLtrack refit	ported from C++ to Java; actual Java package now outside HPS Java	tracking
org.hps.recon.particle	builds ReconstructedParticles from tracks and clusters	builds reconstructed particles from input event collections	recon
org.hps.recon.vertexing	vertex reconstruction	based on Billoir vertexing algorithm	recon
org.hps.recon.filtering	event skimming utilities		recon

LCSim Packages

These lcsim packages are used extensively within the HPS Java reconstruction code.

Java Package	Description	Notes
org.lcsim.event	physics event interfaces (implemented by LCIO)	interfaces used extensively in HPS Java Drivers
org.lcsim.util.loop	event processing loop	extends Freehep loop classes for lcsim usage
org.lcsim.job	lcsim job manager which reads lcsim xml steering files	
org.lcsim.lcio	Java implementation of LCIO file format	implements event interfaces
org.lcsim.recon.tracking.seedtracker	Seed Tracker track reconstruction algorithm	basis for HPS Java tracking algorithm
org.lcsim.util	Driver class for event data processing	
org.lcsim.conditions	detector conditions system backend	
org.lcsim.geometry, org.lcsim.detector	detector description and geometry classes	