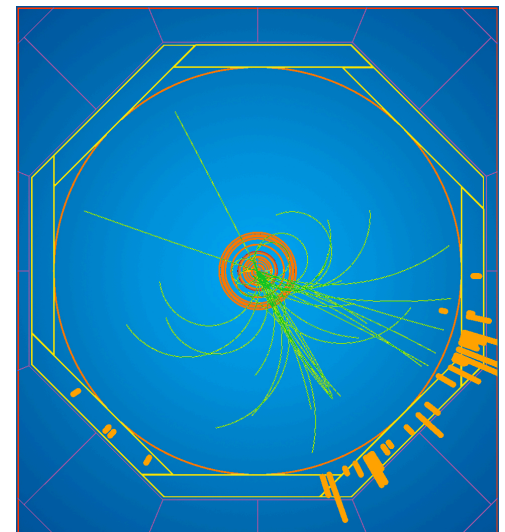


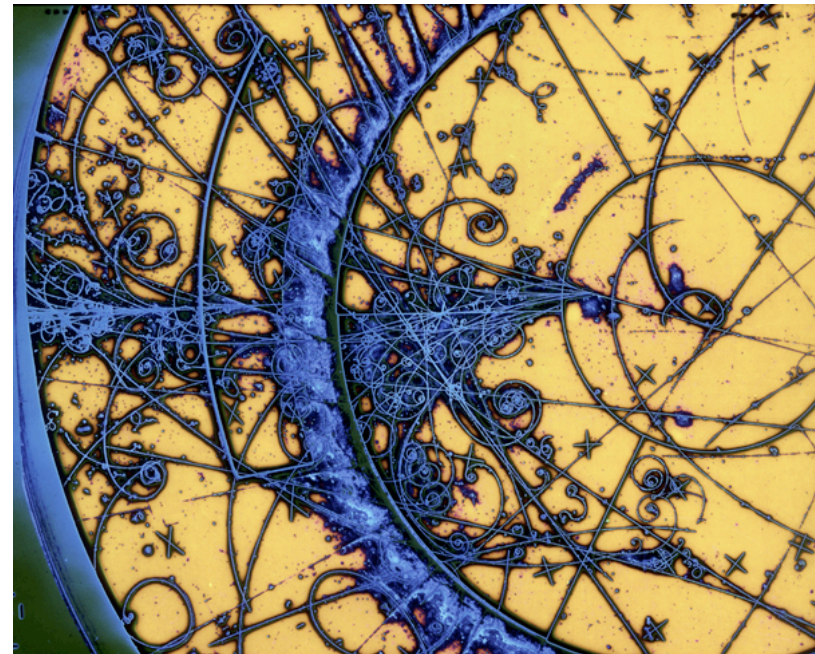
ILD Tracking – Open Issues

Steve Aplin
DESY

ILD Software Pre-meeting 2011 LAL Orsay
22nd May 2011



- Recent Additions
- Issues



IMarlinTrack

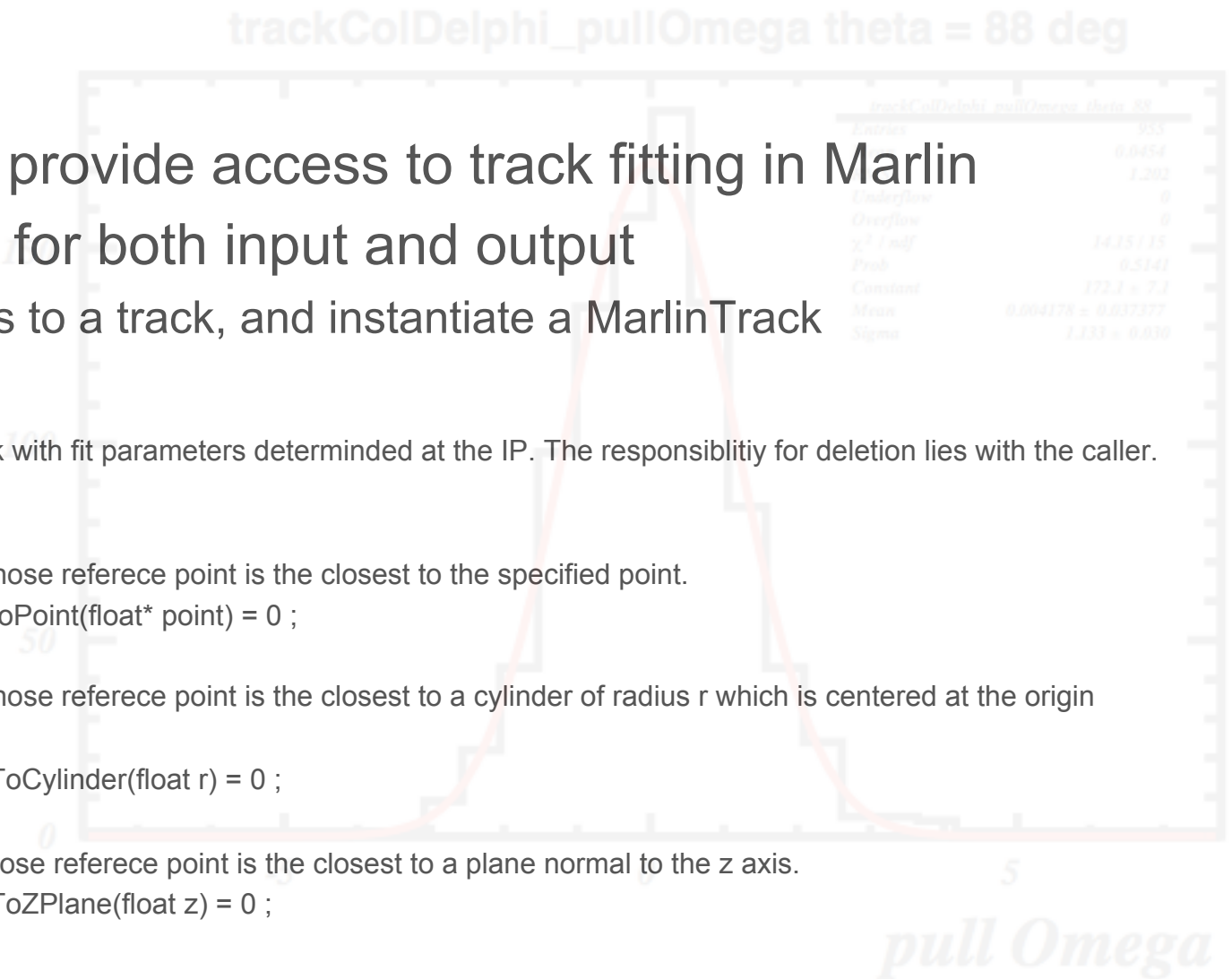
- Interface class to provide access to track fitting in Marlin
- Uses LCIO Track for both input and output
 - assign a set of hits to a track, and instantiate a MarlinTrack

// returns a pointer to a New LCIO Track with fit parameters determined at the IP. The responsibility for deletion lies with the caller.
virtual IMPL::TrackImpl* getIPFit() = 0 ;

// returns a pointer to an LCIO Track whose reference point is the closest to the specified point.
virtual IMPL::TrackImpl* getNearestFitToPoint(float* point) = 0 ;

// returns a pointer to an LCIO Track whose reference point is the closest to a cylinder of radius r which is centered at the origin parallel to the z axis.
virtual IMPL::TrackImpl* getNearestFitToCylinder(float r) = 0 ;

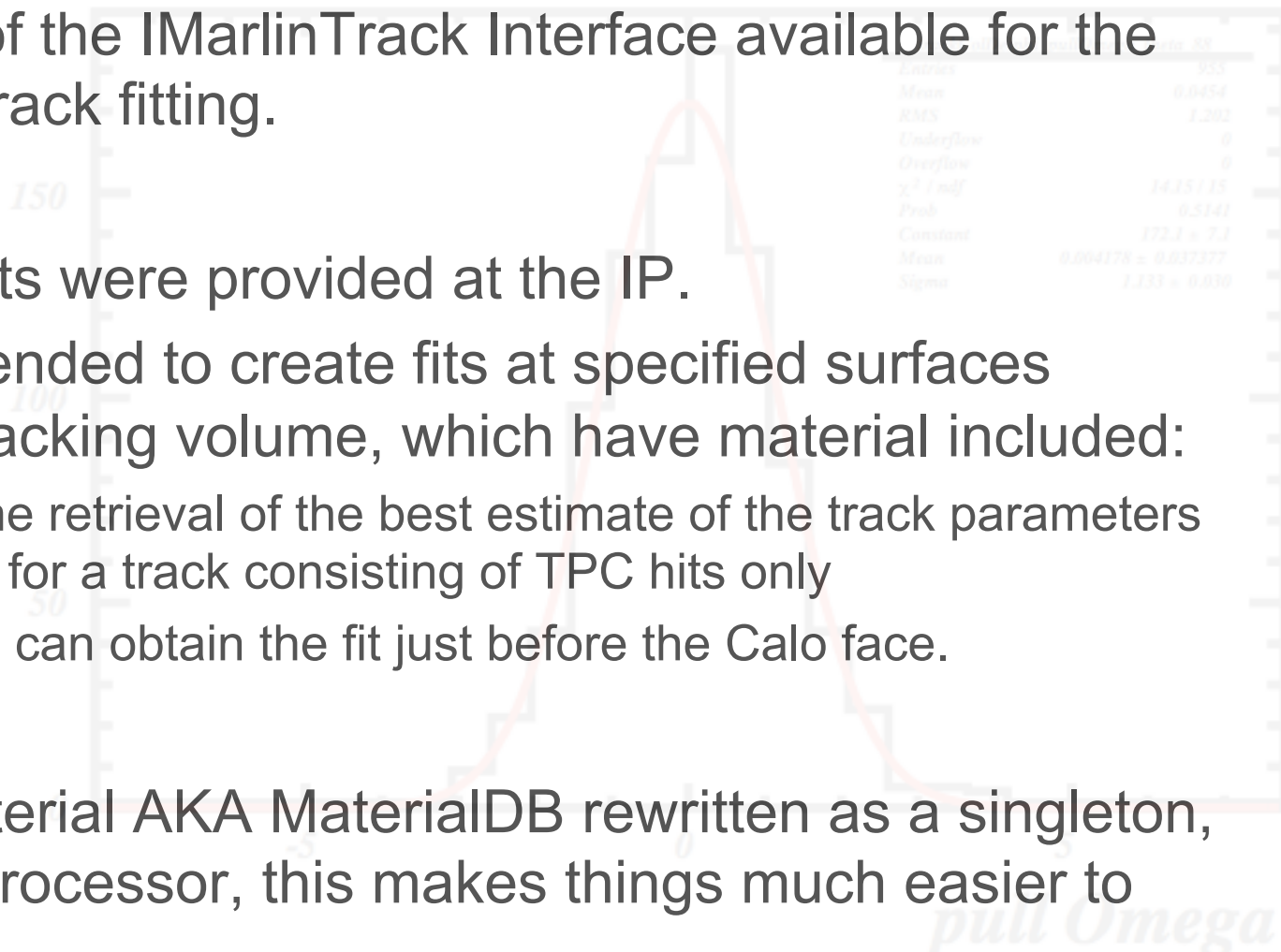
// returns a pointer to an LCIO Track whose reference point is the closest to a plane normal to the z axis.
virtual IMPL::TrackImpl* getNearestFitToZPlane(float z) = 0 ;



MarlinDelphiTrack

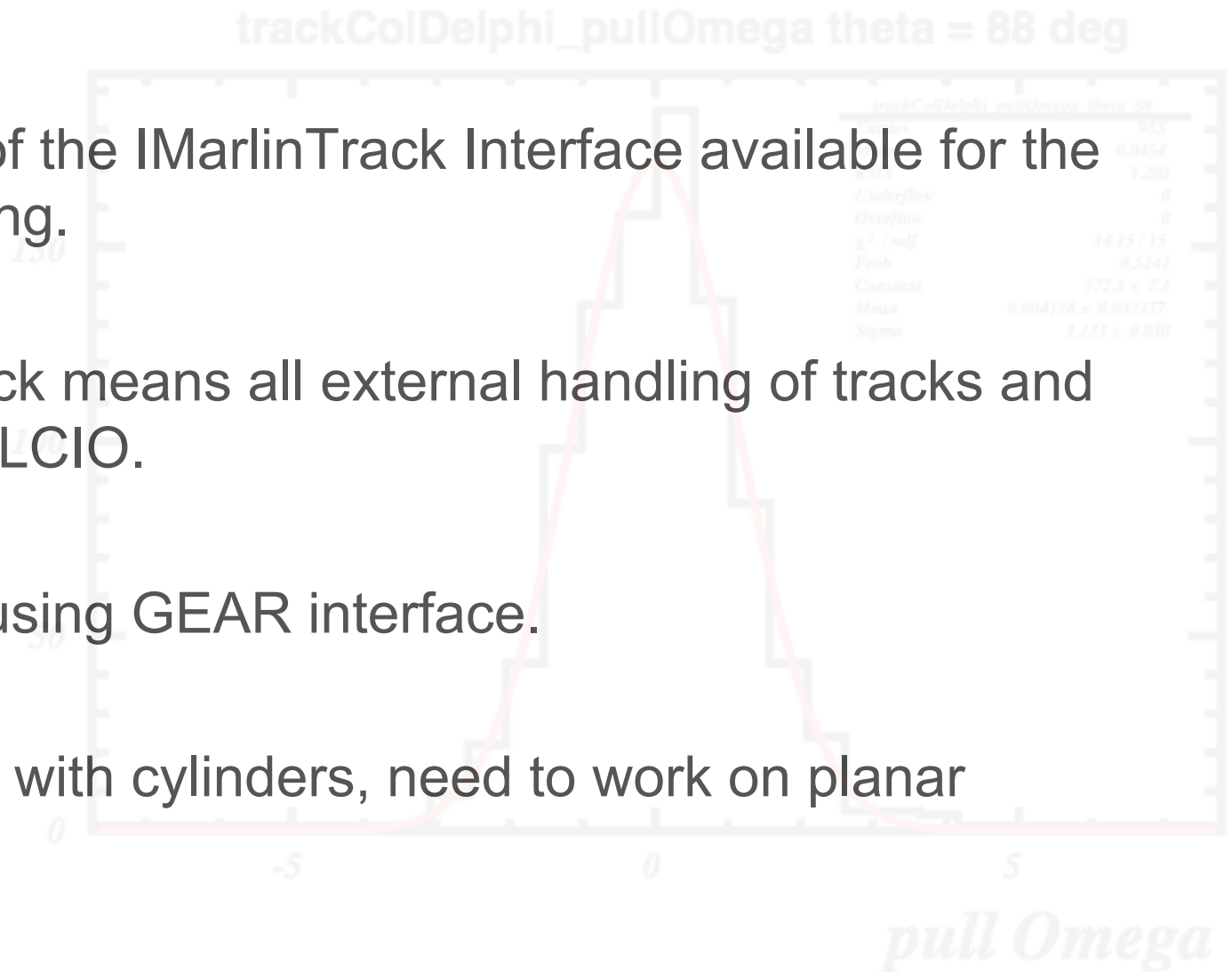
- Implementation of the IMarlinTrack Interface available for the existing Delphi Track fitting.
- Previously only fits were provided at the IP.
- Functionality extended to create fits at specified surfaces throughout the tracking volume, which have material included:
 - This allows for the retrieval of the best estimate of the track parameters at the SIT layers for a track consisting of TPC hits only
 - Alternatively one can obtain the fit just before the Calo face.
- Treatment of Material AKA MaterialDB rewritten as a singleton, i.e. no longer a Processor, this makes things much easier to control.

trackColDelphi_pullOmega theta = 88 deg



MarlinKalTestTrack

- Implementation of the IMarlinTrack Interface available for the new KalTest Fitting.
- Using IMarlinTrack means all external handling of tracks and hits provided via LCIO.
- Material read in using GEAR interface.
- Validated for use with cylinders, need to work on planar surfaces



Propagators

- Track propagation functions previously buried deep inside the F77 tracking code, not available for Lcio track class.
- New set of track propagators added to MarlinUtil:

// Propagate track to a new reference point

```
IMPL::TrackImpl* PropagateLCIOToNewRef( EVENT::Track* trk, double xref, double yref, double zref ) ;
```

// Propagate track to a new reference point taken as its crossing point with a cylinder of infinite length centered at x0,y0, parallel to the z axis.

```
IMPL::TrackImpl* PropagateLCIOToCylinder( EVENT::Track* trk, float r, float x0, float y0, int direction=0, double epsilon=1.0e-8) ;
```

// Propagate track to a new reference point taken as its crossing point with an infinite plane located at z, perpendicular to the z axis

```
IMPL::TrackImpl* PropagateLCIOToZPlane( EVENT::Track* trk, float z) ;
```

// Propagate track to a new reference point taken as its crossing point with a plane parallel to the z axis, containing points x1,x2 and y1,y2.

```
IMPL::TrackImpl* PropagateLCIOToPlaneParallelToZ( EVENT::Track* trk, float x1, float y1, float x2, float y2, int direction=0, double epsilon=1.0e-8) ;
```

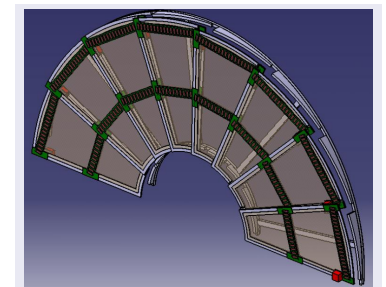
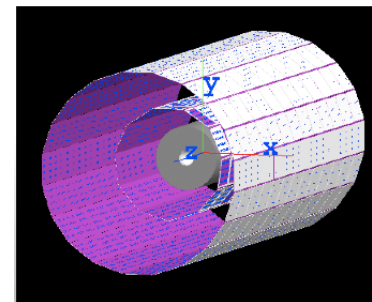
LCIO Tracks can now be propagated to an arbitrary reference point with Cov Matrix

Digitisation

- Currently there are no digitisers available in MarlinReco for the new SIT, SET, ETD and FTD designs present ILD_01.
- We need these very soon if we are to move ahead with the modifications of the tracking code for use with ILD_01.
- How can we achieve this best in a timely manner ?
- Do we need to find extra effort, or are there technical limitations in the software which are holding back this issue?

Navigation

- Both the Delphi fitting code and KalTest have limitations concerning navigation:
 - Delphi code is only able to cope with cylinders, and planar surfaces which are either parallel or perpendicular to the z axis.
 - Whilst KalTest has numerical methods to work with planes of arbitrary inclination, it current lacks the functionality for bounded planes with overlapping regions.
- This has implication concerning the current designs of the SIT, SET, ETD and FTD.



Summary

- Welcome progress in pattern recognition, though much still to do.
- Newly added functionality for fitting and track propagation will help enable users to investigate tracking related issues themselves.
- Digitisers need to be provided for silicon strip detectors.
- Handling planes of arbitrary rotation, and overlapping geometry may prove difficult to realise.

