What RAVE can provide for the ILD

Wolfgang Waltenberger, Fabian Moser
Institute for High Energy Physics of the Austrian Academy of Sciences

Berlin, January 2008

RAVE Status

RAVE supports all major platforms now.

Known to compile and run under:

- Linux, gcc 4.2, Intel architecture.
- Mac OS X, gcc 4.0, PPC architecture.
- Windows, Visual Studio, VC++ 8.0, Intel architecture.
- Windows, MinGW, gcc, Intel architecture.

RAVE can be used from:

- C++
- Java
- Python

What RAVE can do for the ILD

Rave can already perform the following tasks:

- Find and fit primary and secondary vertices, with and without using the beamspot information.
- Refit tracks, exploiting the vertex information.
- Perform flavor tagging, with a rudimentary tagger.
- Perform simple kinematic fits.

What Rave will be able to perform in the future:

- Reconstruct decay-chains kinematically, with more sophisticated constraints.
- Perform b-tagging exploiting more modern, powerful tagging algorithms.

Vertex fitting with beamspot constraints

The beamspot constraint is a source of very detailed information at the ILC. The information in the x coordinate will be dominated by "statistics", not by the machine, while the constraint in z will be dominated by the machine. Which effect will dominate the y-position seems unclear.

We therefore assumed the following values for the beamspot constraint for the analyses:

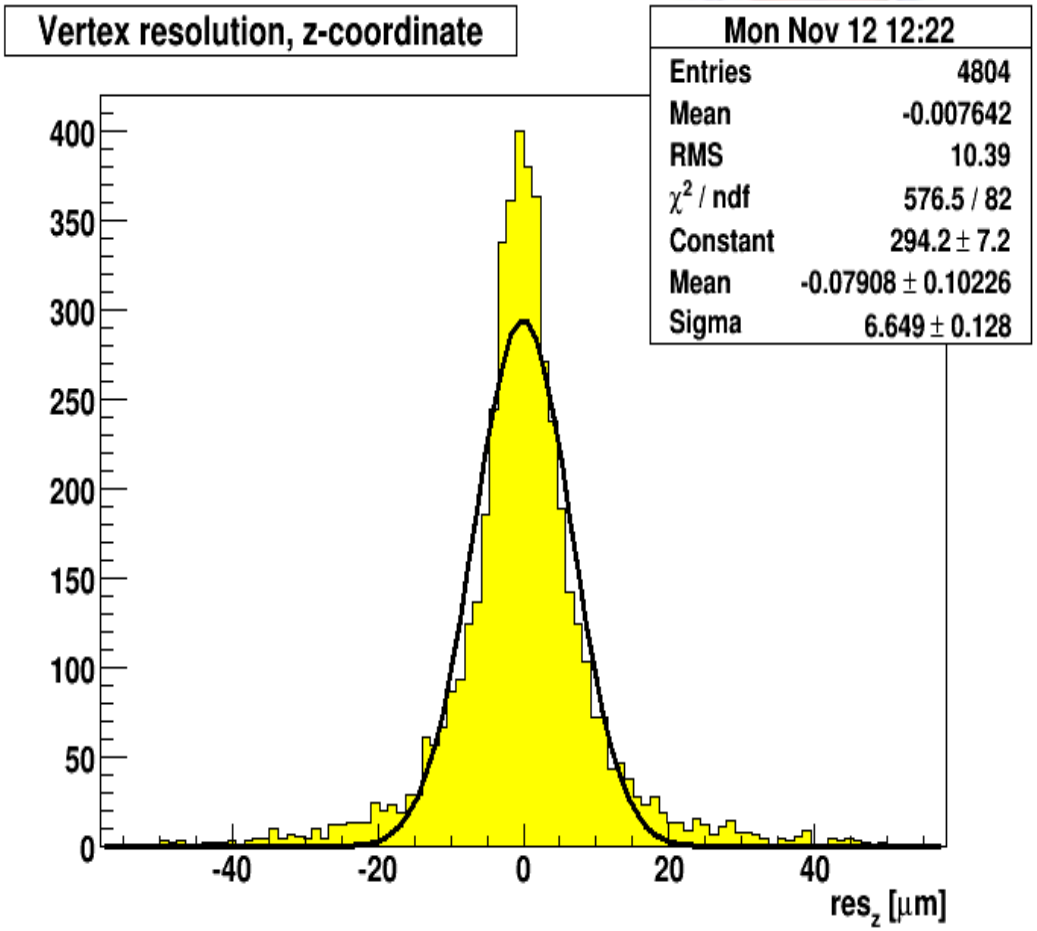
$$\sigma_x = 130 \text{ nm}, \sigma_y = 190 \text{ nm}, \sigma_z = 100 \text{ µm}$$

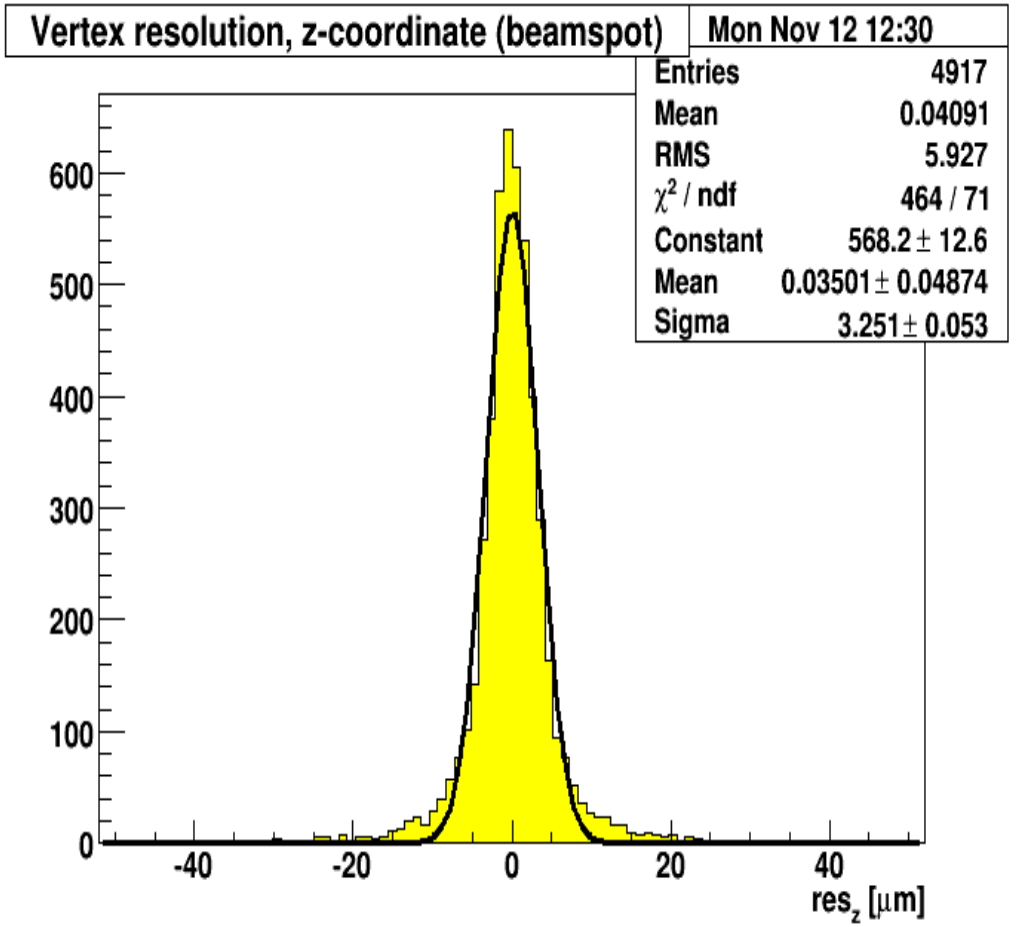
σ_x, σ_y seem like reasonable values after reconstructing O(10,000) primary vertices (assuming a stable beam in these 10,000 events).

Vertex fitting with beamspot constraints

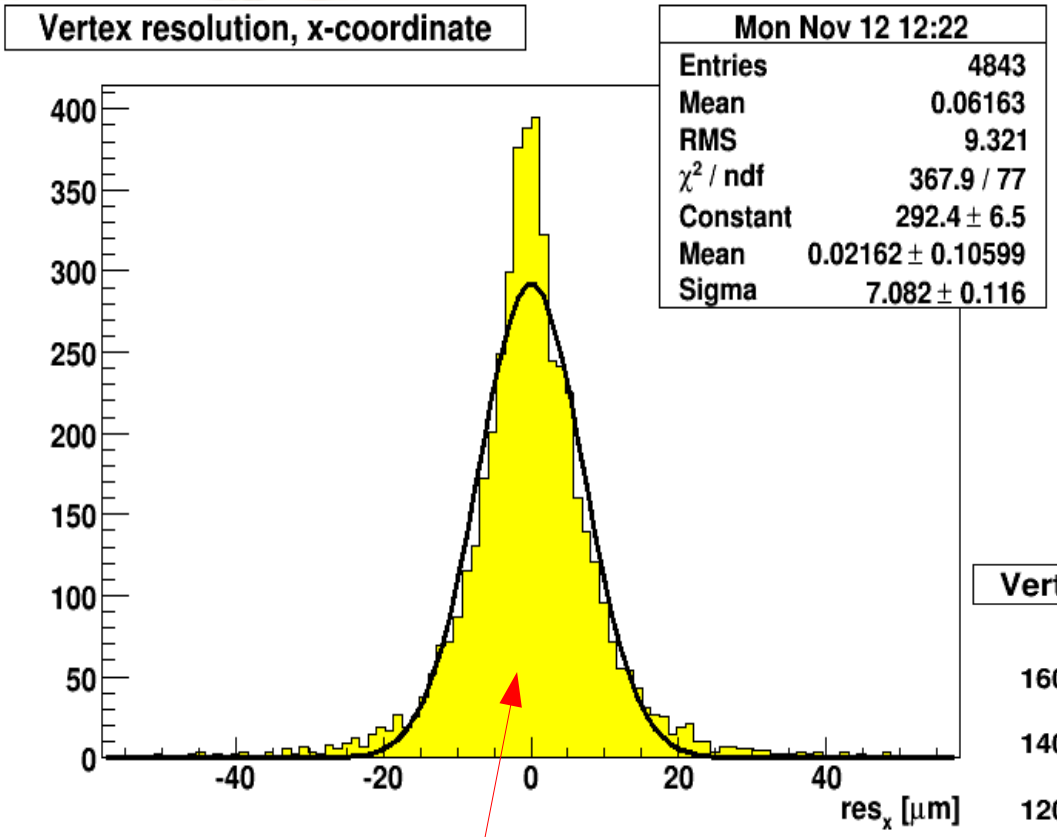
Vertex finding and fitting can also exploit the knowledge of the beamspot constraint now.

z-coordinates of all reconstructed primary vertices, with and without beamspot constraint.



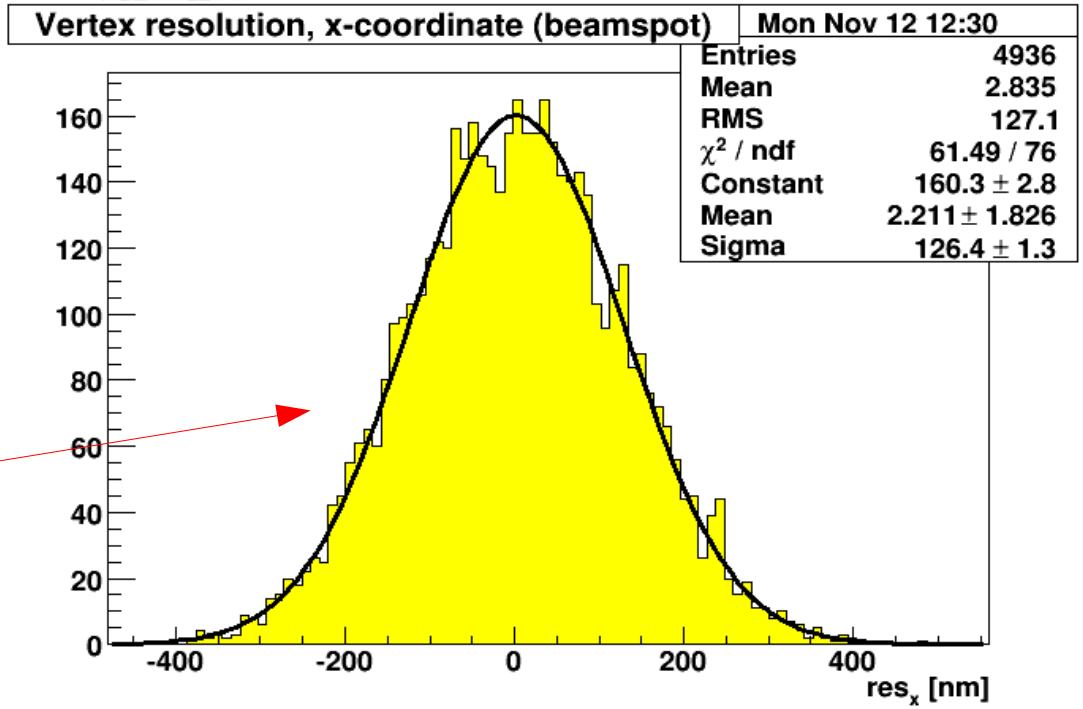


Vertex fitting with beamspot constraints



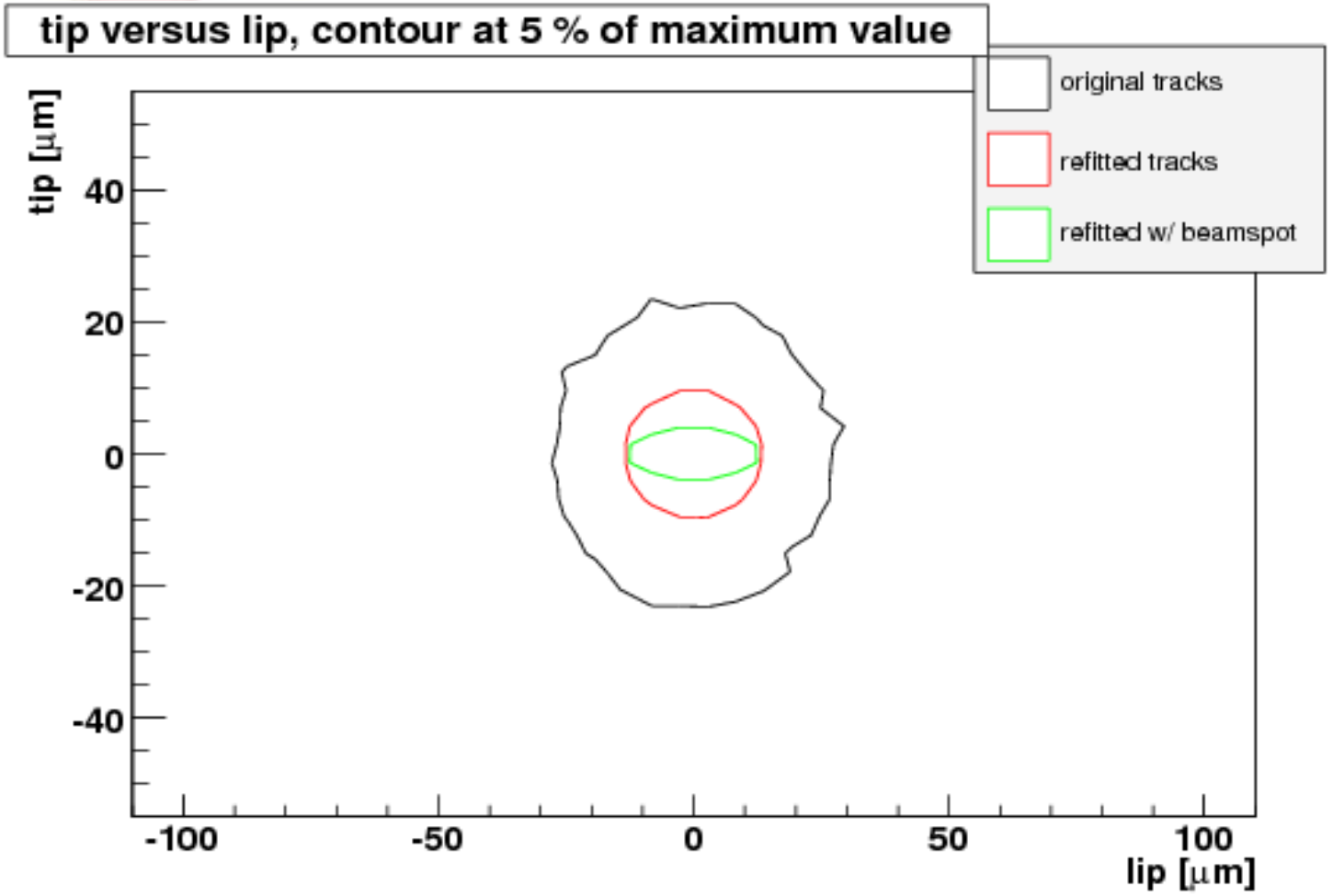
Beamspot information dominates x- and y-coordinates.

RMS ~ 10 μm without,
~ 130 nm with beamspot constraint (in x).



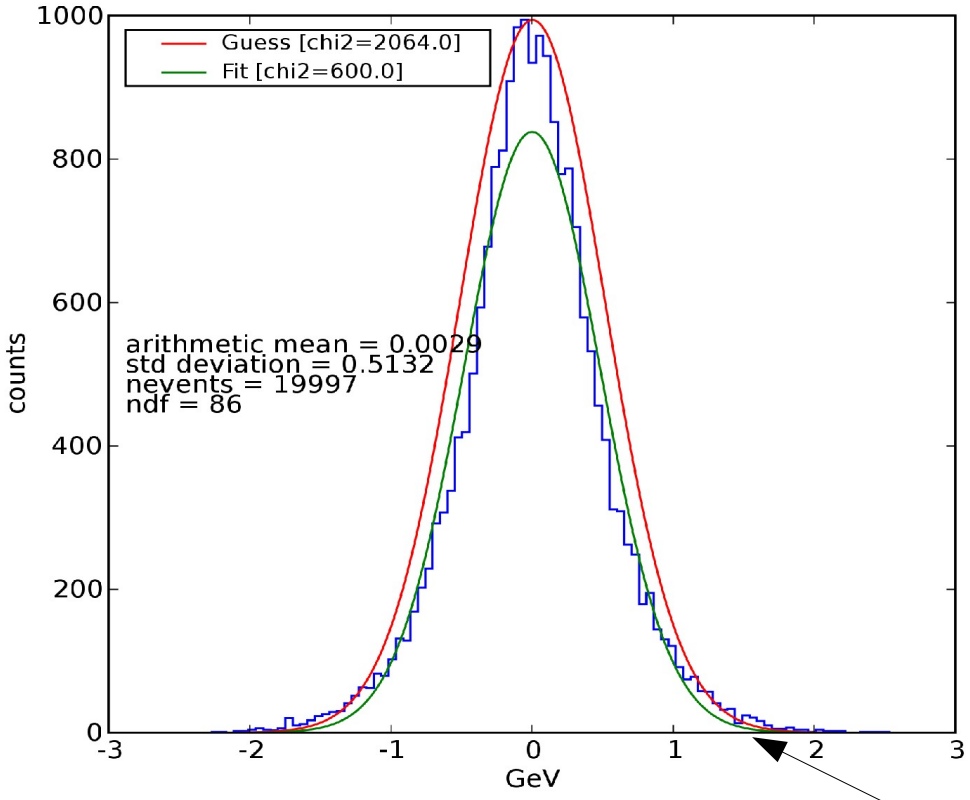
Track refitting

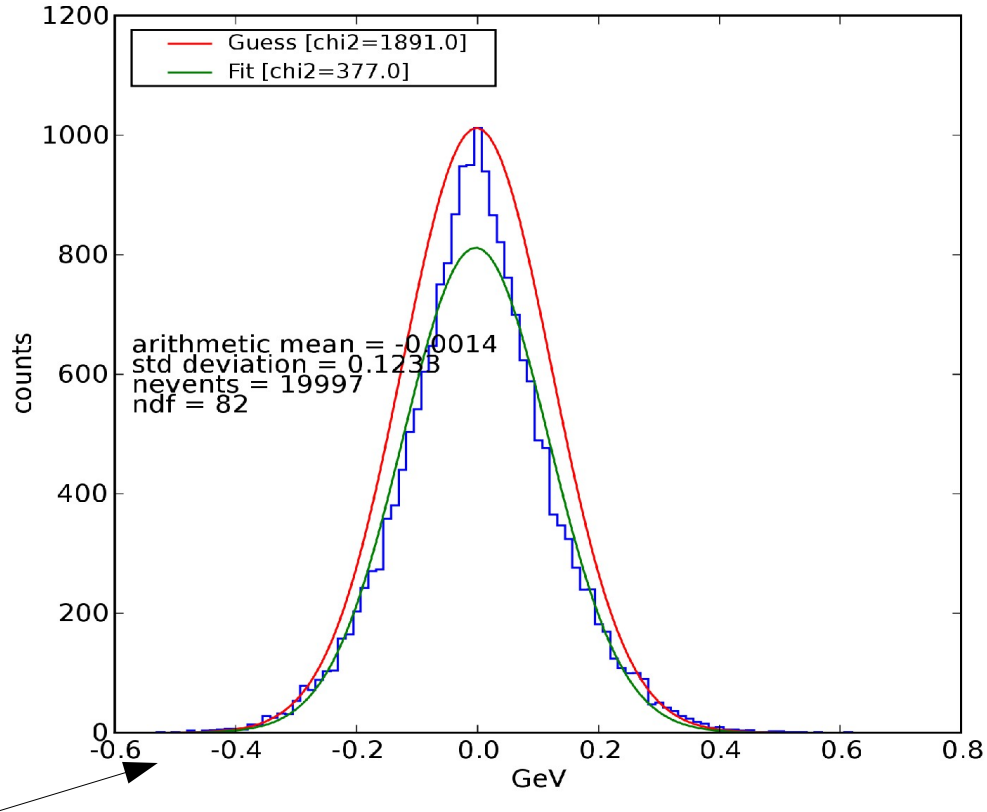
Track refitting can be performed, also. It has a major effect on the resolutions of the two "spatial" coordinates d_0 (tip) and z_0 (lip), while it has little to no effect on the other track parameters.



Kinematic fitting

First version of kinematic fitter works. So far tested only with Z -> μμ topology, constrained with a "hard" Z mass.





First tests with "artificial" data, reconstruction of momentum of z with and without Z mass constraint.

Rave and Marlin

Using RAVE within a Marlin Processor ("MarlinRave") is an ongoing process. MarlinRave can already perform vertex finding and fitting, without beamspot constraints, with track refitting. First prototype of a simple kinematic fit within Marlin is available.

Marlin's steering files do not provide nested structures. Marlin's interface to the steering file seems to be too simple for more sophisticated user input (decay trees with multiple constraints). Kinematic fitters should be definable at run time.

It is our ultimate goal to have MarlinRave appear as a standard Marlin processor.

Missing LCIO features

- Position-to-energy correlations cannot be saved in an LCIO ReconstructedParticle.
- Ndf (number of degrees of freedom) is still an integral value.
- Track-to-track correlations cannot be saved in an LCIO ReconstructedParticle.

Availability

RAVE is hosted at hepforge:

http://projects.hepforge.org/rave

MarlinRave:

http://websvn.teilchen.at/listing.php?repname=marlinrave