



RAVE for the ILC community

Wolfgang Waltenberger, Herbert Valerio Riedel,
Winni Mitaroff, Bernhard Pflugfelder,
Fabian Moser

Institute for High Energy Physics of the Austrian
Academy of Sciences

Paris, May 2007



RAVE – Reconstruction in a Versatile Environment

Main goal is the creation of a reconstruction library that **creates reconstructed vertices, given a set of reconstructed tracks** (independent of the underlying detector hardware).

News – technical issues



Build system: scons has been kicked in favor of **autotools**

Rave successfully compiled under a greater variety of platforms:

- **intel** and **ppc**.
- **linux, win32 (cygwin)** (minor issues need still to be resolved – currently static library only)
- targetting Mac OS X, also.

Packaging is straightforward with autotools.

- Three SLC4 **rpm packages**
- Three packages at the **debian apt server** (deb <http://debian.teilchen.at> ./)

Project now hosted at <http://www.hepforge.org>

Rave still depends only on boost and clhep.

New features



- Functional **java bindings** via SWIG (also in cygwin environment) (the java bindings go into separate distribution packages)
- Optional general-purpose **Btagger** added.
- A few minor fixes/additional features from CMSSW developers keep coming in.

Rave and Zvres



- The **Zvres** algorithm is a **promising vertex finding algorithm** (as opposed to a vertex fitter)
- The e.g. **adaptive MultiVertexFitter** is **statistically meaningful** but **needs seeding** from a vertex finding algorithm.
- A **combination** of **zvres and** the adaptive **MultiVertexFitter** will be an interesting combination.
- Trying to interface Oxford's C++ implementation of Zvres with rave, so that we can try for CMS, as well.
- Oxford implementation unfortunately not yet available.
- Rave should eventually also make it possible to compare C++ Zvres with Java Zvres implementation!

Vertexing, Performance



Performance for a **vertex fitter** can be measured **easily** (resolution, pulls, etc)

Performance for a **vertex finder** / vertex reconstructor **depends on use case**: btagging, kinematic, etc.

So vertex reconstruction strategies must be compared for each use case separately.

b-tagging



Optionally, a **general-purpose** CMSSW **btagger** has been **added** to Rave. Currently works with a likelihood-ratio method, but the method will be pluggable in the future. Some plugins will depend on different technologies/"foreign" packages. Btagger exploits vertexing information. The Btagger is a compile-time option.

kinematic fitting



It is our plan to **add** the option of **kinematic fitting** to rave.
CMSSW code exists.
It will be a compile-time option.

It is **not intended to** let rave **grow beyond** these three
tasks:
vertexing, b-tagging, kinematic finding/fitting.

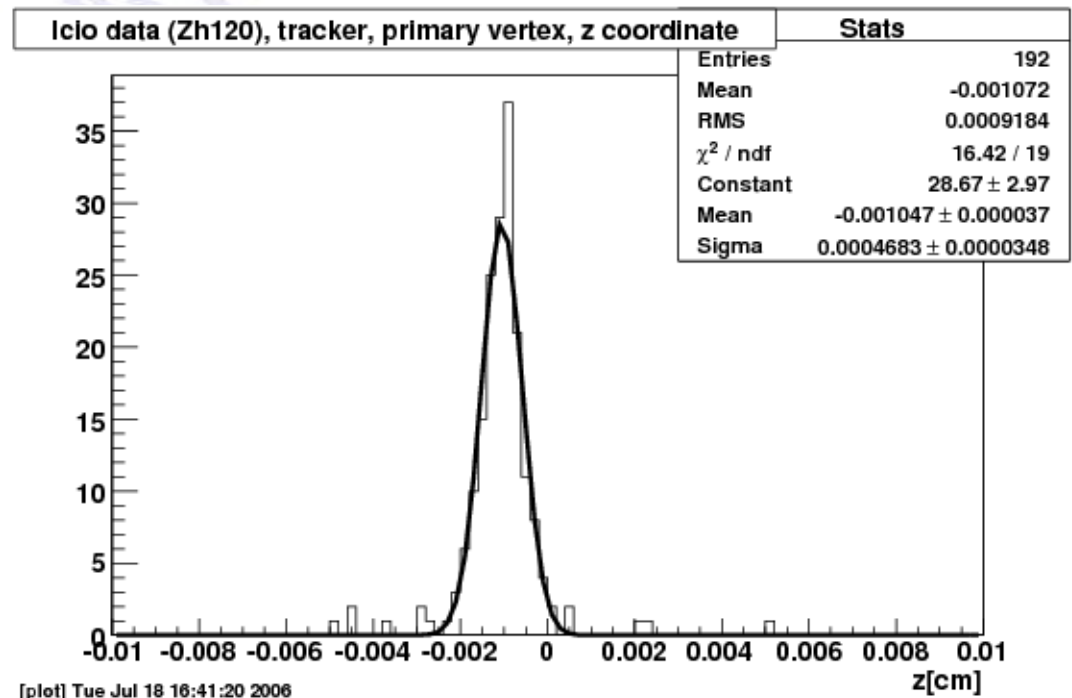
Particularly, tracking will not be part of rave.

Rave and Marlin



As already mentioned previously,
Rave has already been run as a Marlin processor.
In the very first attempt it found a bug in the Marlin
track reconstruction – bug is fixed now. So Rave can
indeed already be used in Marlin.

No new results have
been produced:
waiting for new event
samples with properly
reconstructed tracks
(containing track
parameters +
covariance matrix)



Off-topic



A small off-topic issue:

Question: which file format to use for algorithm debugging / analysis? Root? Aida? Hdf (i.e. pytables)?

We wrote a small tool (“dataharvester”) for transparent read/write access to a few different file formats – a concept that is shared with Aida. Except that the dataharvester is much less ambitious.

It all centers around the concept of heterogeneous hierarchic tuples.

It tries to make creating tuples as easy as possible.

Currently supports: root, hbook, xml, sqlite, hdf, txt.

Languages: C++, Python.

It's not suited for production, but people might find it useful for debugging / analysis task.

It is also hosted at [hepforge](http://www.hepforge.org) and comes with “finished” packages for ubuntu/debian, slc4, archlinux (all backends for the different file formats are packaged separately).

<http://www.hepforge.org>.