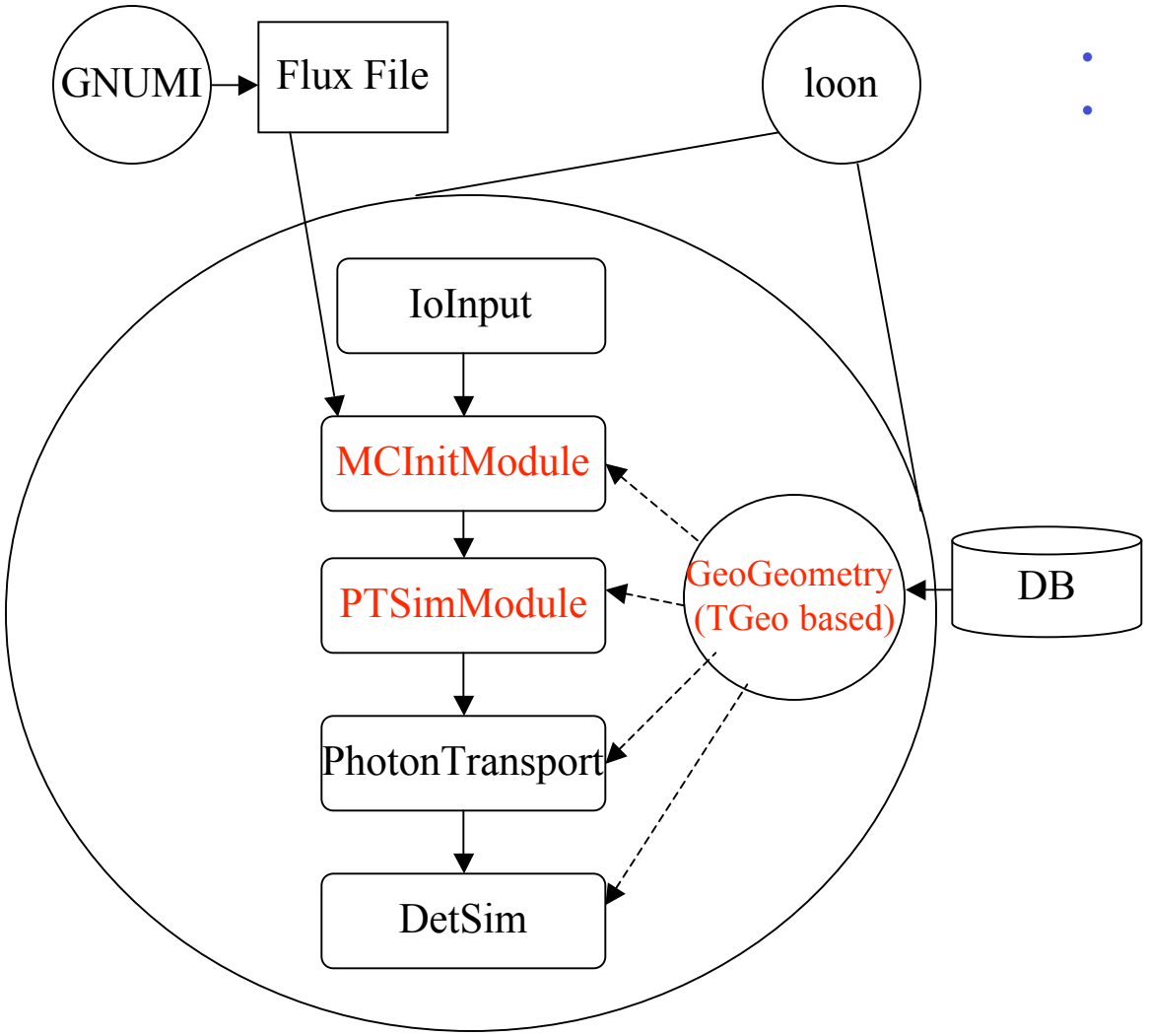
Status of GeoGeometry/PTSim development

Proposal for new MC system:



- GNUMI – Geant3 based. Used to simulate beam resulting in neutrino production.
- Flux File - Primary particle info.
- loon –
  - IoInputModule – essentially acts as dummy in the new MC, with exception of clearing Mom.
  - MCInitModule – new job module generates initial state of event based on data from flux file or algorithm as selected by user.
    - The output of this stage is a SimSnarlRecord containing the initial state information.
  - PTSimModule – new job module performs transport of particles through detector geometry resulting in energy deposition "hits". This Module makes use of the VMC package code. The Module is configured by the user to choose the actual MC implementation (e.g. Geant3).
    - The output of this stage is the SimSnarlRecord with DigiScintHits and stdhep TParticle secondaries.
  - PhotonTransport & DetSim – existing job modules transform "hits" to raw data readout.

Status of GeoGeometry/PTSim development

- New developments
- An update on Performance
- First test of PTSim with Beam Neutrino Events
- Plans

PTSim/GeoGeometry Developments since October

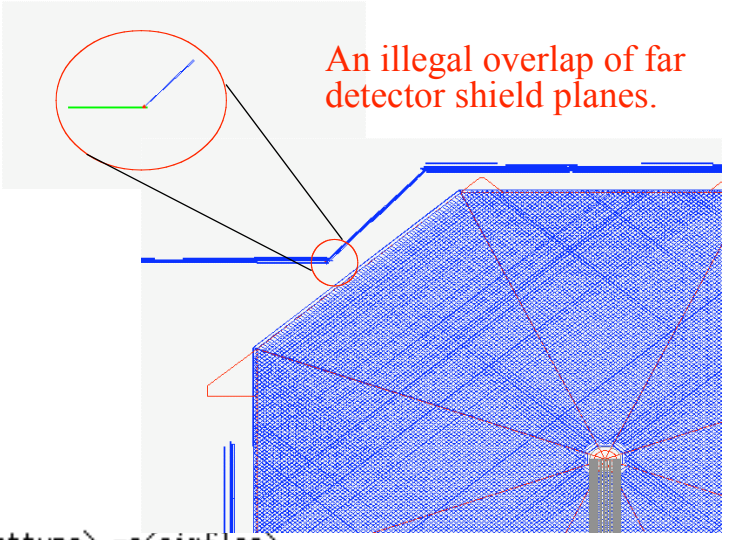
- GeoGeometry
  - Introduced use of new ROOT class TGeoScaledShape for build of scintillator plane modules.
    - One set of "ideal" scint module shapes is built.
    - The ideal shapes are then scaled along the tpos direction to account for plane-to-plane differences in strip tpos in non-perfect detector geometry.

    - Much more efficient than building individual module shapes for every module in detector.
    - Allows a non-perfect (kData) geometry to be built in about the same time as a perfect (kMC) geometry.
    - Requires ROOT version 5.06/00 or newer.

  - Modified far detector coil dimensions to be more like those in GMINOS circa Oct, 2005
    - In response to suggestion by Masaki that difference in coil geometry could account for differences he observed in validation studies.
    - Still need to add a more exacting representation of coil & collar like that of newer GMINOS.

PTSim/GeoGeometry Developments since October

- GeoGeometry
  - Introduced new validation tool to test geometry volume overlaps : TestOverlap
    - Uses TGeoManager::CheckOverlap tool to look for overlaps/extrusions of detector element volumes. Can be used to catch problems in geometry implementation & database position values.
    - Was running CheckOverlap manually, now automated. Run with other validation tools (TestGeo to compare Geo to Ugli, & PTSimValidate) before committing changes to the repository.



    - Configurable:

```
usage: TestOverlap -l<loglevel> -p<precision> -d<dettype> -s<simflag>
 -l: loglevel at which to print messages (default=Info),
     valid arguments are: Verbose, Debug, Synopsis, Info, Warning, Errc
 -p: precision used in check (cm) (def 0.1)
 -d: detector type(s) used in check (def Far,Near,CalDet)
 -s: simflag(s) used in check (def MC,Data)
 -h: print this message
[121]schubert@minos-pc2>
```

    - Only “illegal” overlaps in the current detector geometries are those in the far detector shield planes, none more than 1 cm.

PTSim/GeoGeometry Developments since October

- ParticleTransportSim
  - Bug fixes, especially to handling of neutrino initial state.
    - Use of package is best way to root these out.
    - Masaki and Kregg's use of PTSim for validation purposes has dramatically moved things along.
  - New "Demo" MCInitModule to allow PTSim to be run without an input reroot file. (Without MCInitModule, reroot file is required to provide event initial state.)
    - Only works for single particle types for which the user specifies the particle kinematics & type, a la the KINE card in GMINOS.
    - A dummy option was introduced to I/O to allow job control to spin through cycles with no input file.
    - Example job script is given in ParticleTransportSim/test/PTSimValidateInit.C:

```
jc.Input.Set("Format=dummy");                    // no input file, Run(nevt) sets limit
jc.Path.Create("MC", "MCInitModule::Get "     // creates SimSnarlRecord
                     "PTSimModule::Reco "     // particle transport
                     "PhotonTransport::Get  "
                     "DetSim::Get ");
 // set particle kinematics
 jc.Path("MC").Mod("MCInitModule").Cmd("SetKine –13 3. 0. 18. 0. 0. –5."); // pdgId, vtx(x,y,z) ,p(x,y,z) (GeV/c)
 // optional parameters to set detector type (Far,Near,CalDet), validity start date, run, subrun, etc.
```

A loon fully contained MC.
A first!

    - Useful for validating PTSim. Will eventually be replaced by "real" MCInitModule (Robert).

PTSim/GeoGeometry Performance

- GeoGeometry build time is slow
  - On a local pc w/these parameters:
    - AMD Athlon XP 2100+ processor, cpu MHz 1740.729 MHz
  - ~ 1minute build to far detector geometry. (Near detector and caldet are faster.)

- But was not always slow – slowness started when Al skin modules were introduced.
  - Each module is built as an aluminum skin volume with an air volume insert = 2 volumes/module
  - 8 modules/plane x 486 planes x 2 = 7800 additional volumes + shield modules
  - This substantially increased the number of volumes in detector modeled geometry to a total of 12000 volumes.

- Recent search for slow-down cause yielded a primary culprit:

```
Int_t TGeoManager::AddVolume(TGeoVolume* volume) {
    // Called on each new volume construction
    …
    Int_t olduid  = GetUID(volume->GetName());
    …
}
```

Loops through list of previously inserted volumes checking for uniqueness of name. Expensive!

- Andrei Gheata (of the root team) has agreed to work on this.

PTSim/GeoGeometry Performance

- PTSim was also rumored to be slow (20x) relative to GMINOS, but recent test data was lacking, so…

- Kregg Arms performed comparison of PTSim performance to GMINOS
  - Full presentation given at 11/21/05 MC phone meeting in DocDB.

- Generated 1000 each of single μ+/μ- for p_z = 1,2,3,4,5 GeV/c, both near & far:

	GMINOS (near det)	PTSim (near det)	GMINOS (far det)	PTSim (far det)
CPU time/evt (s) μ-, "Test 2"	0.081 (1GeV) 0.296 (5GeV)	0.128 (1GeV) 0.749 (5GeV)	0.155 (1GeV) 0.809 (5GeV)	0.147 (1GeV) 0.870 (5GeV)

- PTSim performance is comparable to GMINOS for FarDet and ~ 2x as slow for NearDet. Not 20x as slow!

- Will continue to whittle away at improvements to PTSim performance…

Validation Test of PTSim/GMINOS for Beam Events

- Event validation for PTSim vs GMINOS has been performed:
  - For Single Muon Tracks of a given momentum/vertex, both Near & Far detector (Masaki Ishi)
    - Reported by Masaki at last collaboration meeting, and in DocDB.
    - These tests helped debug PTSim, and showed that PTSim and GMINOS were in reasonable agreement with each other.
  - For a small sample of Cosmic Ray muons
    - Used only to debug trajectories, no formal study.

- Validation for Beam Neutrino Events not attempted, until now!
  - Spurred on by Kregg Arm's new MCMonitor tool which can be used to compare GMINOS to GMINOS (different releases), or GMINOS to PTSim

- Plan:
  - Use Carrot06 GMINOS produced reroot files as input
  - Run R1.20 reconstruction on reroot files using standard MC production script – this is GMINOS result.
  - Run R1.20 PTSim + reconstruction on initial neutrino state derived from reroot files – this is PTSim result.
  - Compare results with Event Display for several events – visual inspection.
  - If able to produce data on a large scale, compare with Kregg's new MCMonitor tool.

Validation Test of PTSim/GMINOS for Beam Events

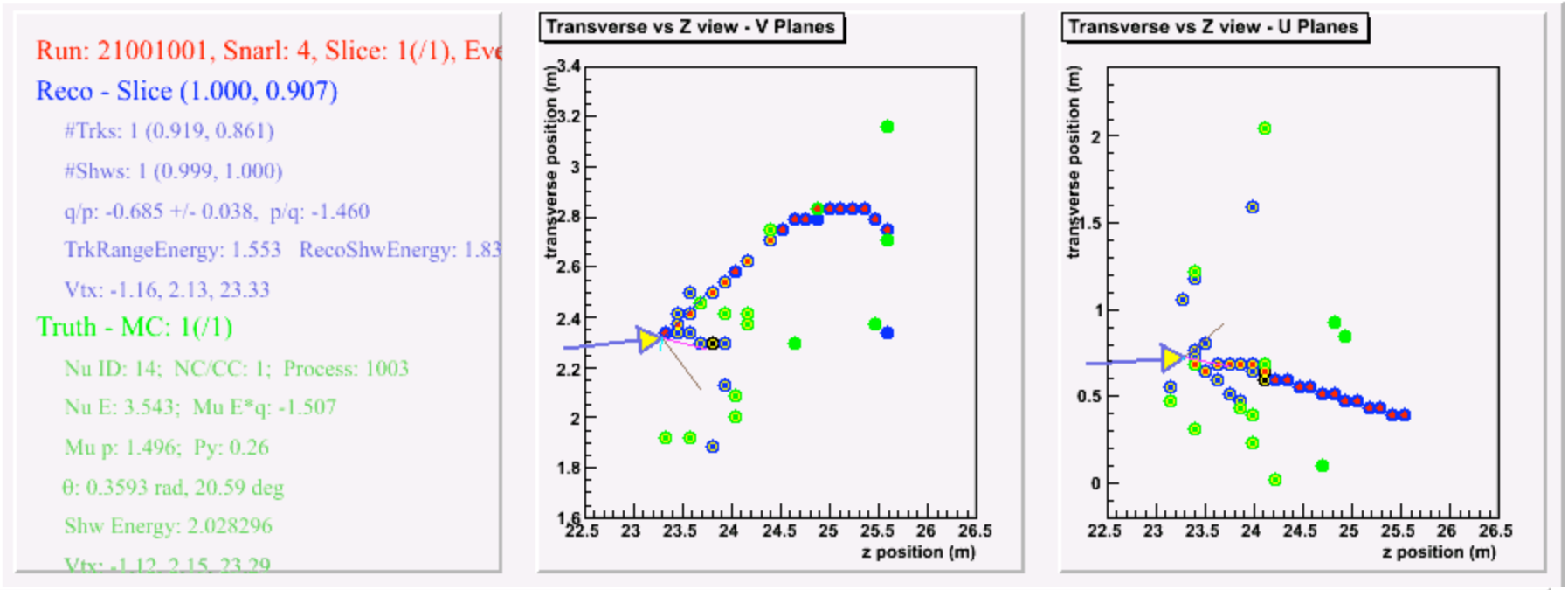
- First attempt failed due to bugs in PTSim handling of initial neutrino event state.
  - Bug fixes now in CVS.

- Second attempt failed due to bug(s) in R1.20/ROOT5.08/00.
  - Reco jobs crash after processing some number of snarls, on both standard gminos & PTSim jobs.
    - Under investigation – possibly a root bug in TSpectrum used by shower reco algorithm?
  - R1.20/ROOT 5.08/00 is largely untested.
    - Dramatic shift in root version (4.02/00 -> 5.08/00) making it likely that R1.20 will require further testing before all bugs are worked out.

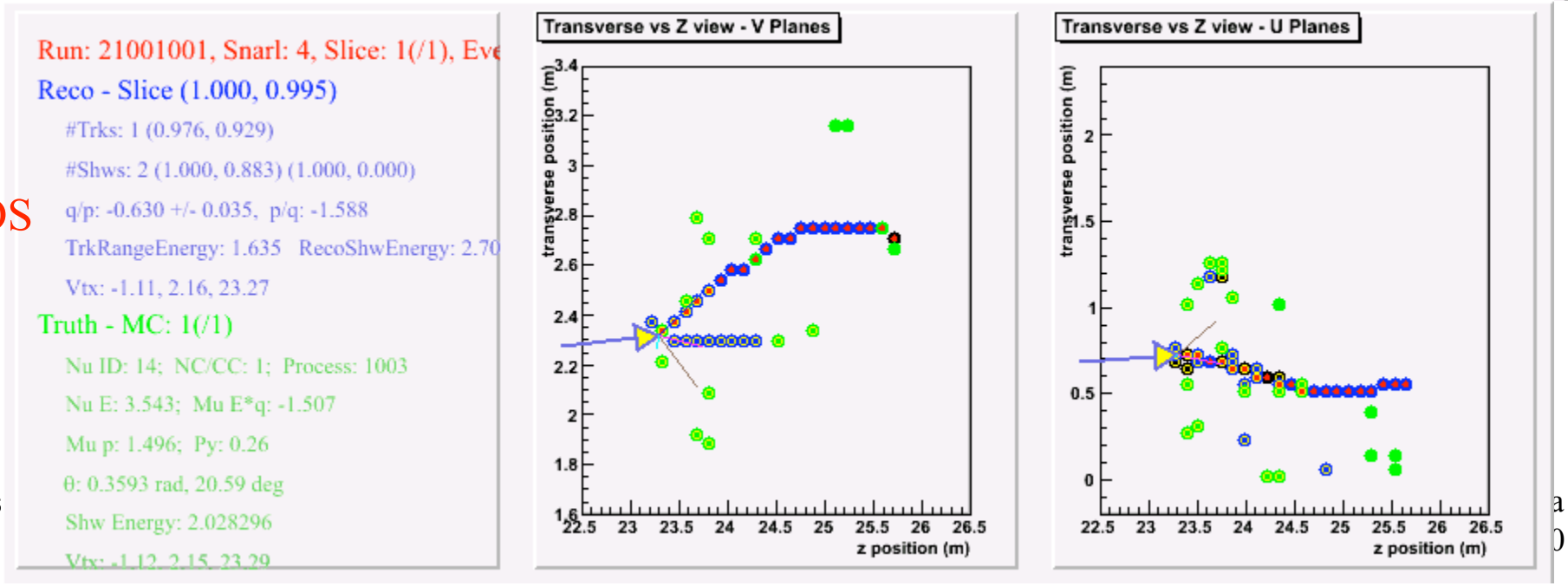
- Also took some time to work out Geant3 config parameters to use with PTSim.
  - Change from:
    - 100 keV -> 10 keV energy cutoff (same as GMINOS)
    - 10,000 -> 800,000 max steps (same as GMINOS)
    - GHEISHA -> FLUKA/MICAP hadronic interaction model (GMINOS uses GCALOR)
      - Will install GCALOR for use with PTSim for next set of tests.
  - to be more compatible with GMINOS.
- For now, only examine small set of events with event display => Next slides.
- After bug fix to R1.20/Root, Kregg will run large sample so that can be examined with MCMonitor package.

PTSim vs GMINOS - Beam Neutrinos/Far Detector

PTSim

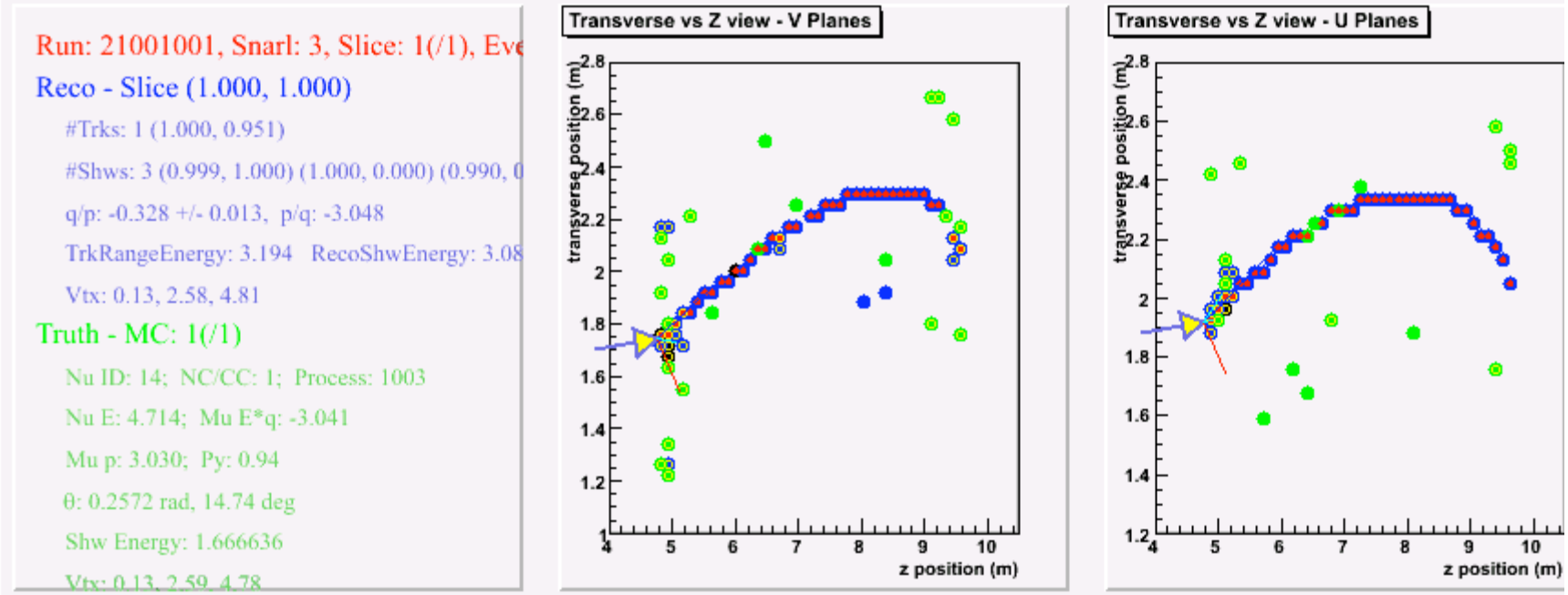


GMINOS

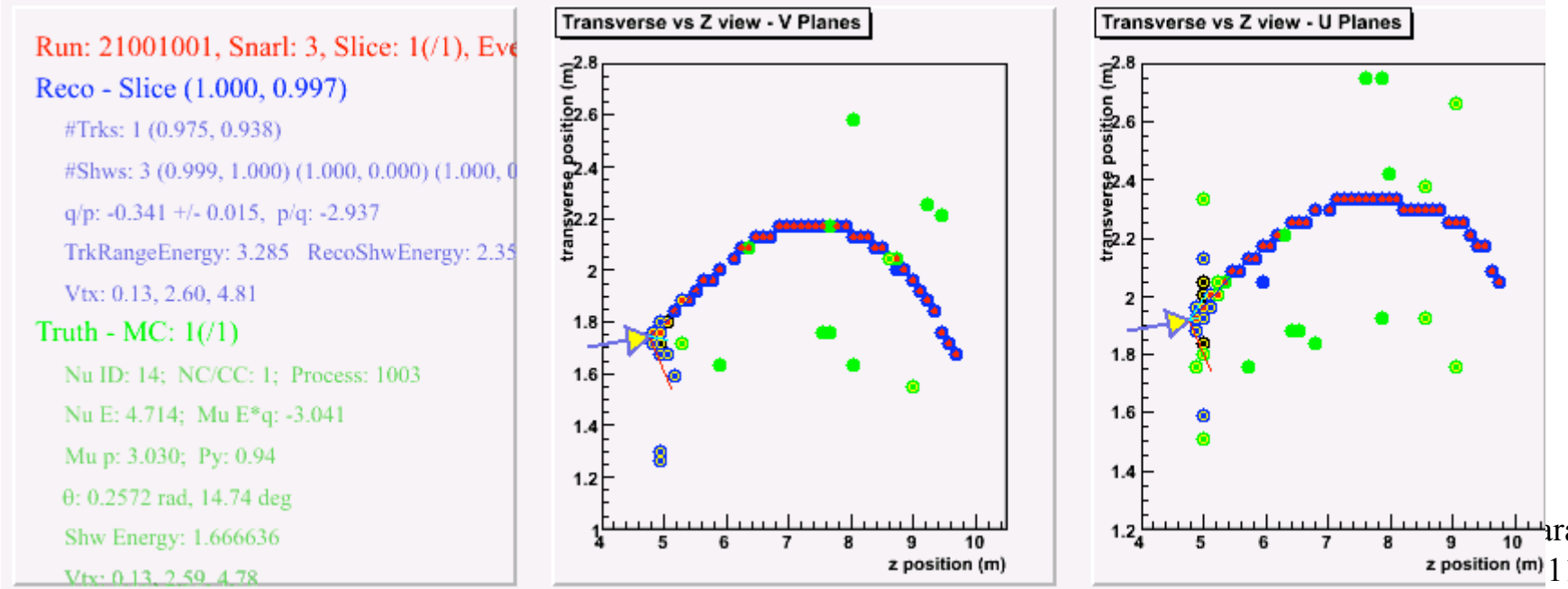


PTSim vs GMINOS - Beam Neutrinos/Far Detector

PTSim

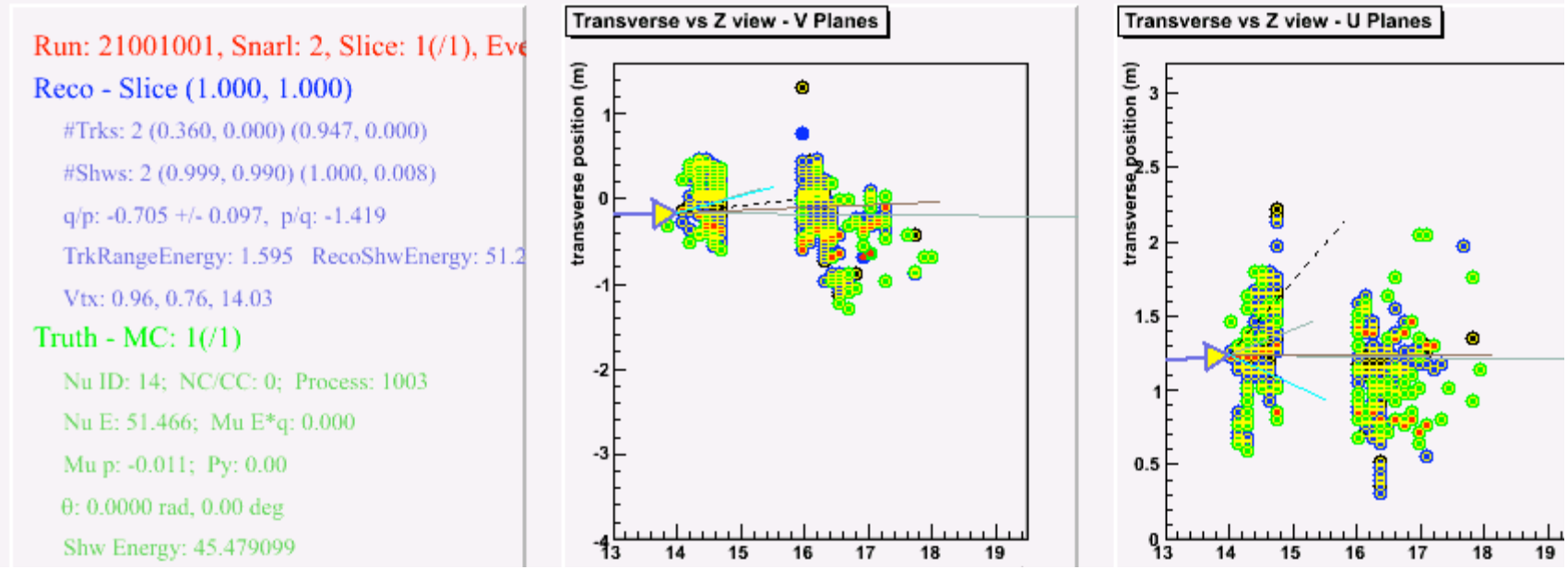


GMINOS

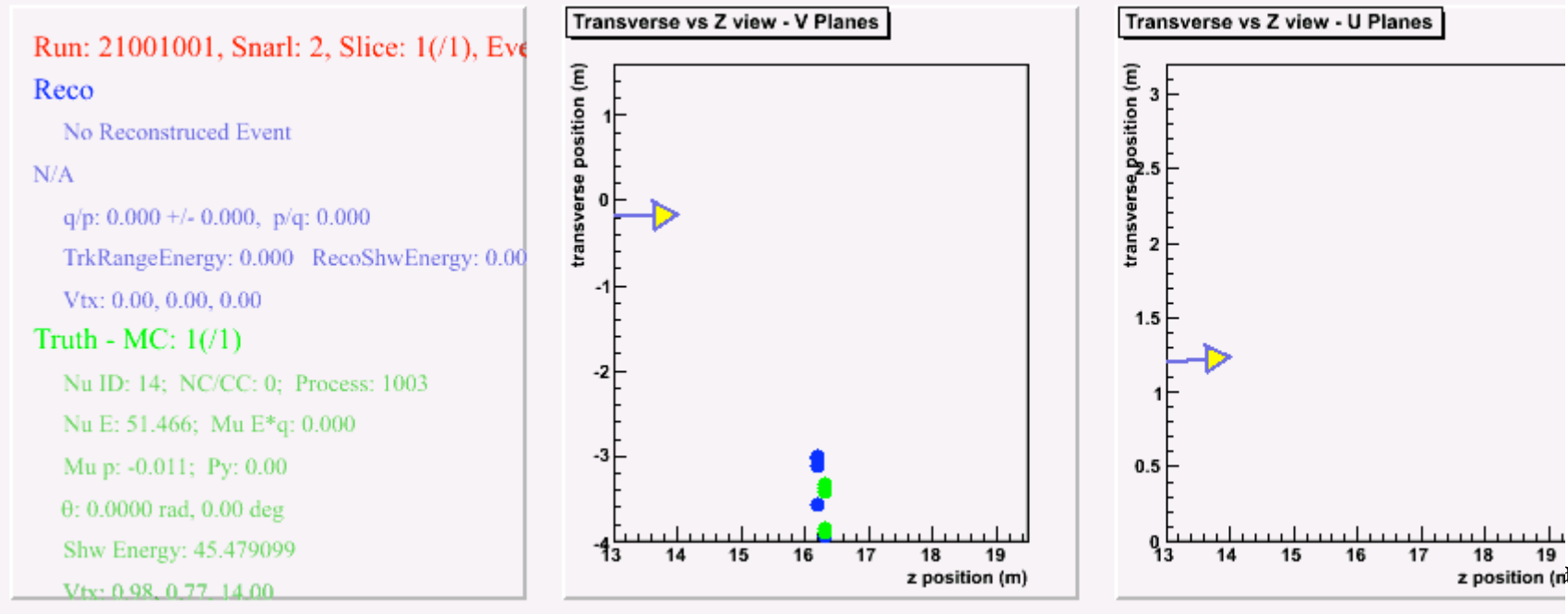


PTSim vs GMINOS - Beam Neutrinos/Far Detector

PTSim

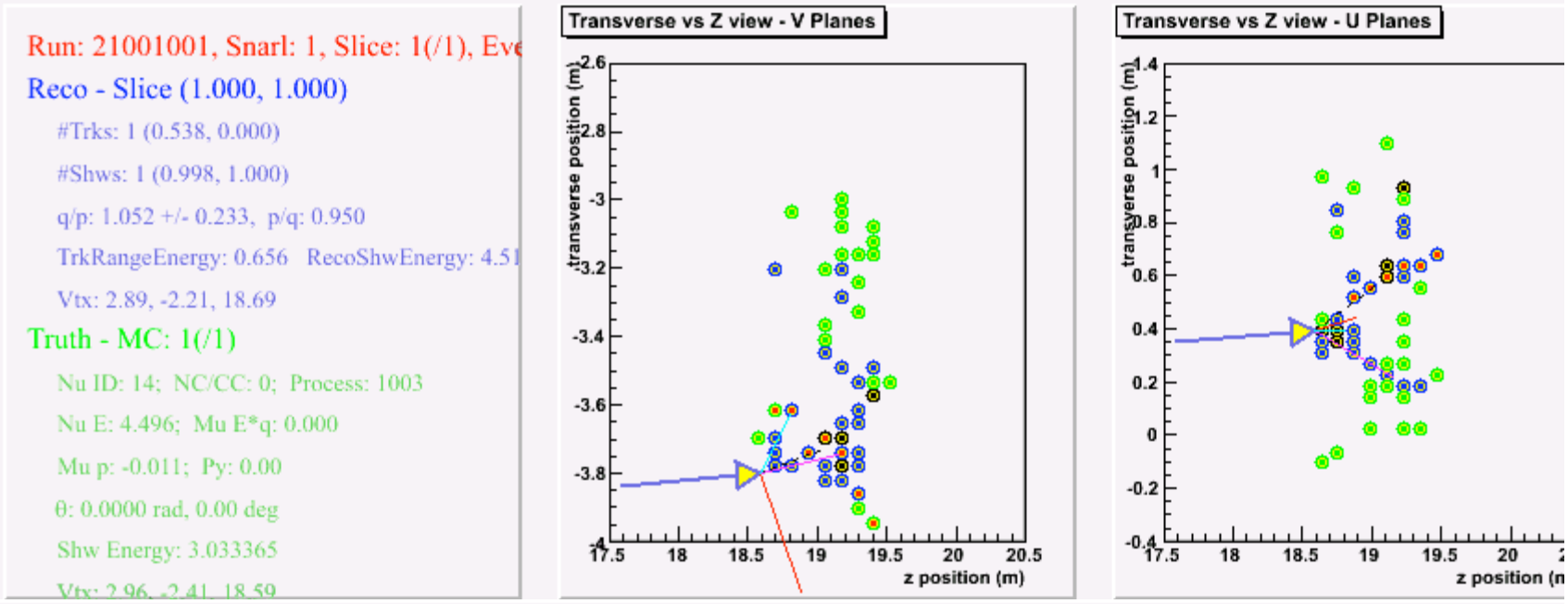


GMINOS

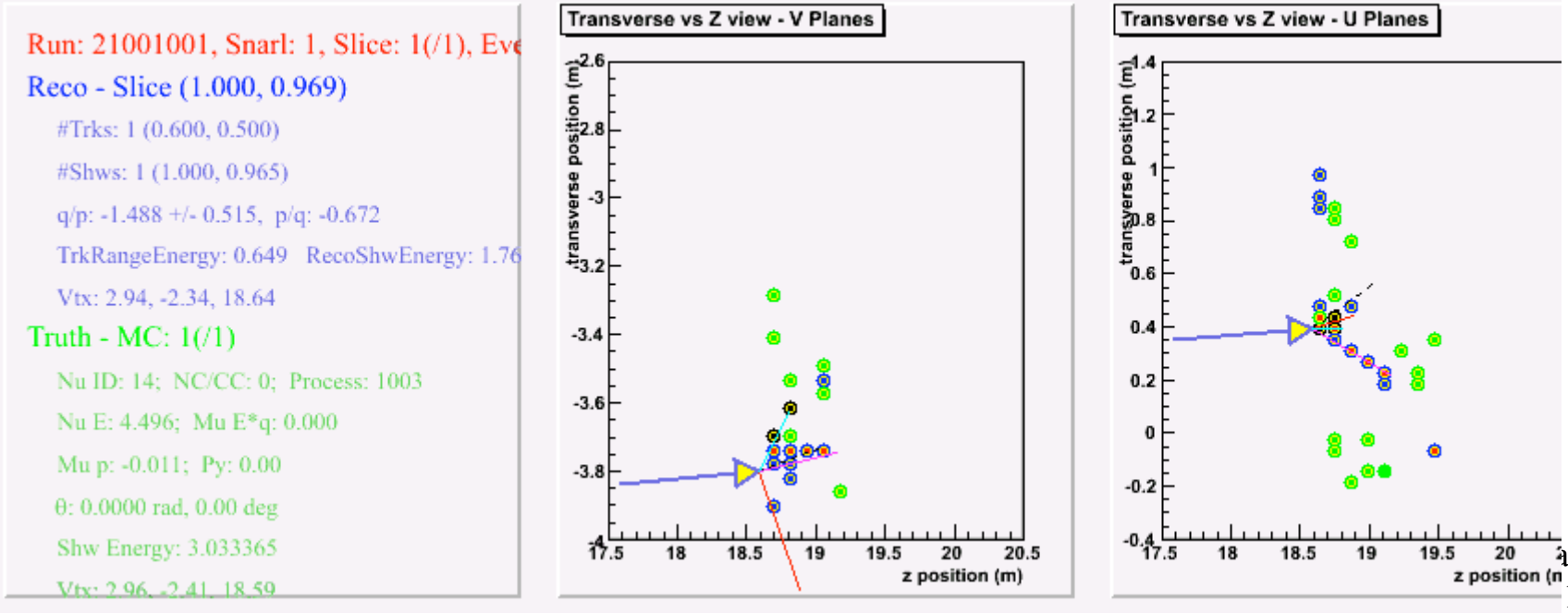


PTSim vs GMINOS - Beam Neutrinos/Far Detector

PTSim

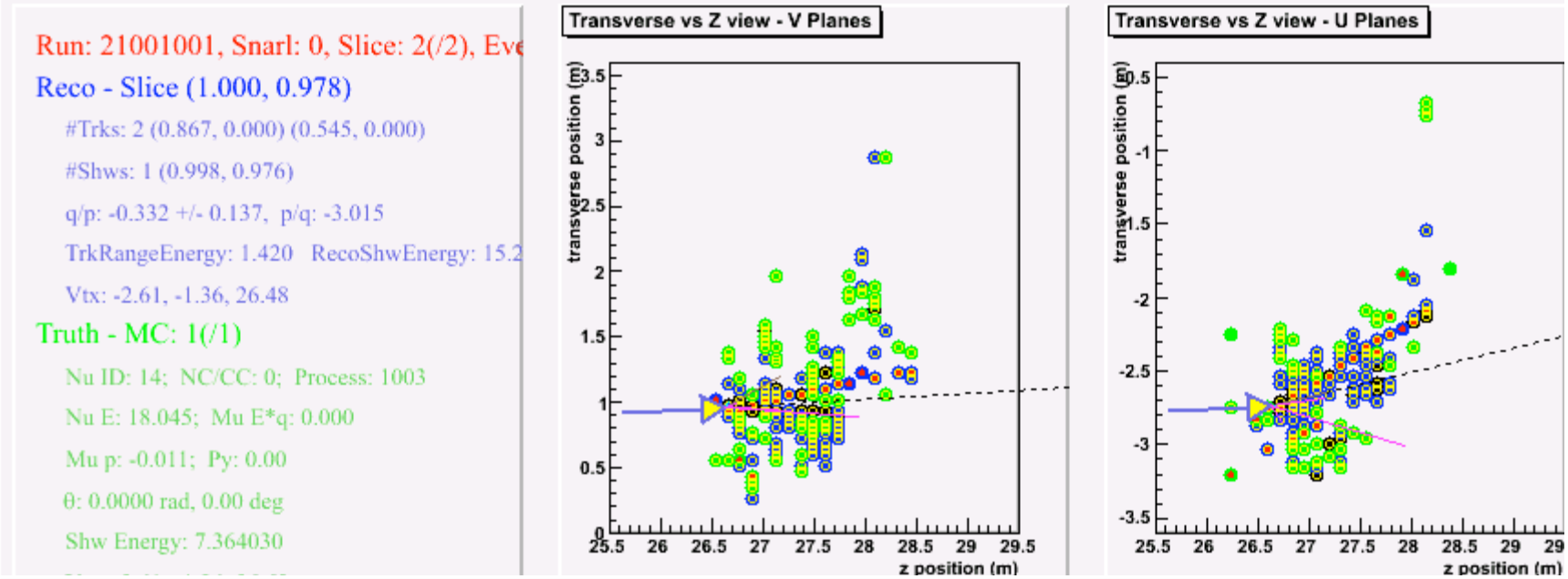


GMINOS

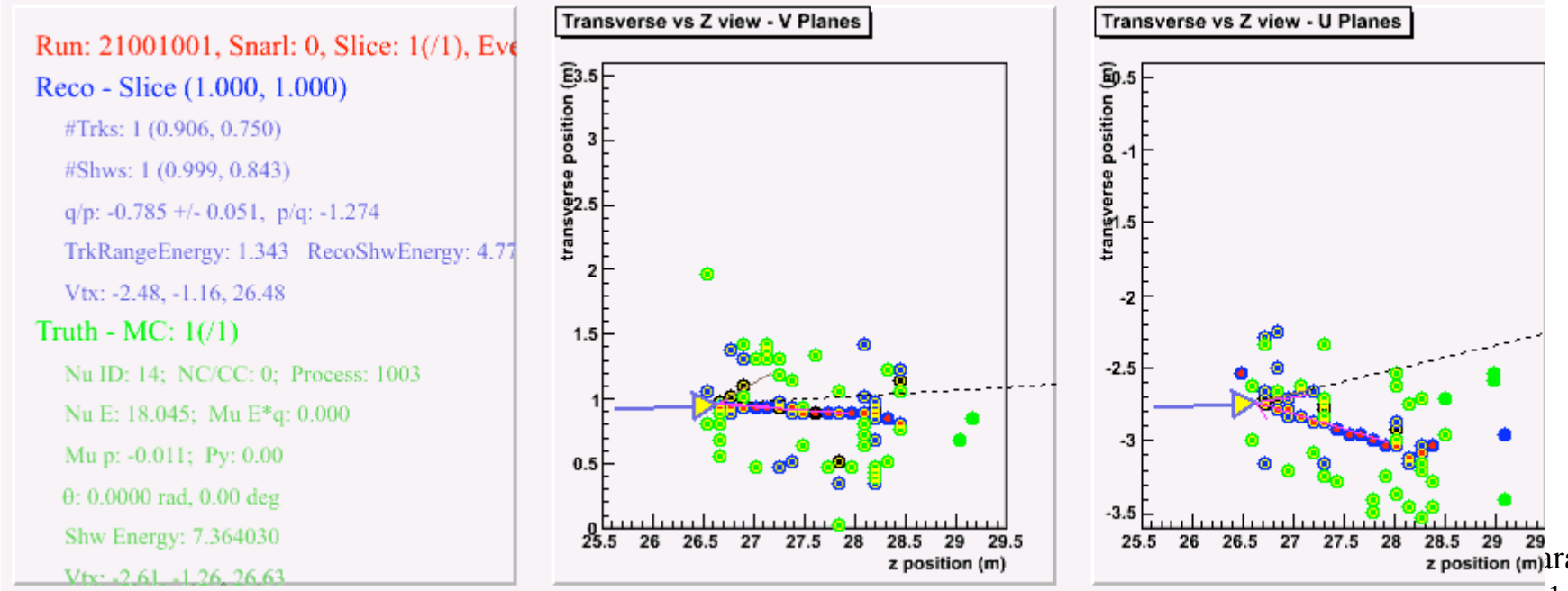


PTSim vs GMINOS - Beam Neutrinos/Far Detector

PTSim



GMINOS



PTSim/GeoGeometry Plans

- Large scale PTSim processing of neutrino events, to be performed at MN.
  - Using Kregg's batch processing setup, after R1.20 bug fixes are complete.
  - Such testing will force remaining PTSim bugs to surface & allow use of MCMonitor package for validation against GMINOS.
  - After this, PTSim could be made available for wider use?

- GeoGeometry coil geometry will be updated to match that of newer GMINOS.
  - Also need the BField package to be updated (Robert) to make use of detailed maps near coil.

- The latest update from the ROOT team (Andrei) is that a preliminary version of the GEANT4 interface to TGeo navigation will be available at the "beginning of March", 2006.
  - It will be his primary focus over the next couple months.