Atlas Event Generators

Ian Hinchliffe LBNL

June 20, 2003

Overall Strategy

We have one full time person and a few part time volunteers working in this area.

- Code from MC authors build unchanged in external
  Not inside Atlas CVS
  Not rebuilt for each atlas release
  Used for version control

- Link and include paths defined by External area inside of CVS and is part of the release build

- Generators packages to take the events and repackage them into common format for use downstream (currently HepMC)

- ROOT I/O persistency for the HepMC

- Downstream packages see same structure independent of generator used

- Particle output of Geant-3 is mapped back into complete HepMC structure – Single structure used for both generated and Geant secondaries.

Documentation here
http://www-theory.lbl.gov/ ianh/monte/Generators/

Current supported Packages

Complete event Generators

- Pythia
  Supporting two versions concurrently 6.203 (default) and 6.217 (Test), only possible if their output and interfaces (common blocks) are same.

- Herwig Supporting 6.5

- Phojet Supporting 112135

- Hijing Supporting 1.831

- Isajet Supporting 7.64

Special Purpose

- AcerMC, AlpGen, Comphep that use the Les Houches interface
  events are read, hadronized with pythia or herwig and passed downstream
  adding others is simple provided they follow the standard

- Tauola/Photos
  Pythia integration uses atlas code
  Herwig integration uses Herwig options

- Single Particle (used for testing mainly)

- LHAPDF; own copy in “external”, interfaced to Herwig. Pythia interface pending.

- HepMC (atlas version) and I/O packages

Current clients

- Altfast simulation, uses parametrized detector response – widely used for physics studies

- Geant-3 Legacy code reads events (HepMC) for full simulation
  output from G-3 is also repacked into a HepMC structure to handle truth for existing simulation/reco

- Geant-4. will use a truth association object to map between G4 and HepMC (waiting for LCG Simulation project)

Productions

- Just completed DC1, $\sim 10^7$ events
  Pile up handed at the hit level from a repository of simulated events
- Used Pythia (mainly) and Herwig
- Tuned Pythia to get underlying event to agree with Tevatron
  Tunings are loaded at runtime
- DC2 starts next year, Generation should be validated before Christmas

Current and Future Actions

- Use HepPDT
  We have been using Particle Properties from HepMC, this is being replaced by HepPDT – transition will be complete before end of July
  Athena Particle Properties service

- Integrate EvtGen – need dedicated B-decay package for use in DC2
  This is already late

- Drop pdflib; replaced by LHAPDF

- add Jimmy; presently Herwig cannot be tuned properly

- Herwig++; would like to try this in DC2 even if functionality is severly limited.

- Interface to MC event store DB

- Migrate to CLHEP version of HepMC from atlas version?????

Current Problems and Issues

- Usual conflict between the need for stability and need for latest feature, trying to handle this with more than one version in the releases (will not work with a static build)

- Use of external means that export of atlas code to remote sites is more difficult (not just a problem for event generators)

- Legacy issues; G3 truth in particular.

- Lack of enforced standards; example “no standard HEPEVT”, code is tied to pythia, then breaks with Herwig.

- G4 Truth
  No factorization in MC truth in G3; need to test G4 Truth strategy early so we do not get trapped into a bad design.

- HepMC
  Must have routines to translate event from generators into HepMC
  We are using our own “preliminary” translators
  No translators in CLHEP version
  Trival to get 4-vectors and stable status codes correct
  Full history is more tricky

- We are stuck with fortran well into data era

Wish list

- Replace "external" with LCG supported release
  This can be done incrementally starting with Pythia, Herwig...
  Provided the LGC release contains the same versions that we are using, we can switch during a software release cycle (3 weeks)

- Consistent Particle properties
  Ideally everyone should use same thing so there is only one source
  Hope that Herwig++ will use HepPDT
  Can someone fix pythia so that it can use HepPDT as an option?
  What about G4?

- Properly maintained HepMC
  CLHEP version has no translators; this is where all the hard work is.
  We will not move from Atlas-HepMC to another product unless that product has validated translators. Need team to do this for all experiments and all time.

- Coordinate tuning effort between experiments