ATLAS Event Generators on 64-bit Architectures

H. Vogt
DESY Zeuthen

DESY Hamburg/Zeuthen ATLAS Meeting
Zeuthen, Tue July 3, 2007

- ATLAS Monte Carlo Interfaces:

```
ParticleGenerator, Isajet_i, Pythia_i, Herwig_i(Jimmy, DpEMC), Tauola_i,
Photos_i, Hijing_i, AlpGen_i, AcerMC_i, CompHep_i, PythiaB, MadCUP_i,
McAtNlo_i, (Lhapdf_i -> Pythia, Herwig), Phojet_i, Winhac_i, TopRex_i
```

- Some of them do a full event generation (Isajet, Pythia, Herwig) some use an external generator file or generate events on the parton level and use Herwig or Pythia for hadronization

- The event generator libs are mostly taken from the GENSER installations in

/afs/cern.ch/sw/lcg/external/MCGenerators

or from

/afs/cern.ch/sw/lcg/app/releases/GENSER

- The generator components

  `Phojet_i, Winhac_i, AcerMC_i, AlpGen_i, McAtNlo_i, Tauola_i, TopRex_i, Winhac_i`

  have ATLAS specific modifications and are taken from

  `/afs/cern.ch/atlas/offline/external`

- **64 bit** architectures peculiarities:

  The libs taken from **`/afs/cern.ch/sw/lcg/external/MCGenerators`** had to be changed from static to shared (only those are compiled with the `-fPIC` compiler option). Doing this one has to solve the "`BLOCK DATA`" problem (the Fortran BLOCK DATA code has to be included in the first shared lib used by "`athena.py`").

  -> This is solved now with the help of the GENSER group.

- 64 **bit** architectures peculiarities (continue):

To handle addresses of Fortran `COMMON BLOCKS` in C++ code a Fortran preprocessor macro `-DFVOIDP=INTEGER*8` must be used because the shared libs loaded by `athena.py` and the `COMMON BLOCKS` included are linked in the address space above the 32 bit address range.

Also one had to make sure that all such addresses used in the C++ code must be declared as type `void*`.

All this concerns a lot of generator interfaces:

`Pythia_i, Herwig_i, Cascade_i, Photos_i,Tauola_i, AcerMC_i, Hydjet_i, Hijing_i`

- 64 **bit** architectures peculiarities (continue):

  A special correction was required in the Herwig_i C++ code for the `HepMC::HEPEVT_Wrapper` class:
  `size(long int)` has to be replaced by `size(int)`
  (because `long int` is 8 Bytes on 64 bit architectures) .

- Present status:
  - Most of the Generator Interfaces are now running on 64 bit architectures.
  - Some Modifications are still required for the **jobOptions** file using configurables
  - The validation of the results running the Interfaces on 64 bit computers has began by the ATLAS MC group