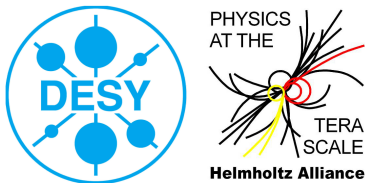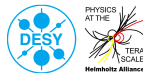# Terascale Monte Carlo School 2011

Event Generation in ATLAS

Liza Mijović for the ATLAS DESY group

Hamburg, 17th Mar. 2011

Event Generation in ATLAS

Event Generation is done within (C++) Athena framework with (part of the) generator configuration done with (Python) JobOptions file; this implies:

- differences in generator configuration syntax (wrt. standalone),
- using standard generator installations/versions rather than installing the generators locally,
- using pre-defined scripts to run the generators.

While this means learning some additional syntax, it has some **important benefits**:

- many aspects of the generation need to be set to the same values for the bulk of the samples (masses, couplings, tunes. . .); within Athena these are conveniently collected in common files that are #included in the files steering the generation.
- Event generation used for the central production is reproducible.
- The existence of rules how the events need to be generated also imply you know events were generated in case it wasn't you who was running the generator.

The aim of the ATLAS-side tutorial is to provide an overview of the event generation and give you hands-on experience with both running the generators and retrieving information about the central production generator runs. This talk is predominantly aimed at the former, the Tutorial exercises at the latter.

Event Generation Schema
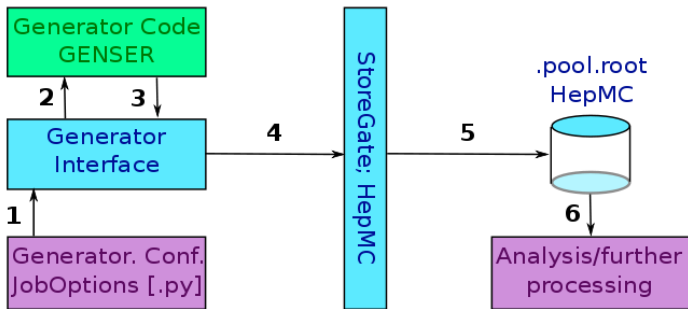
- Q: Which generators can be used acc. to this schema?
- A: https://twiki.cern.ch/twiki/bin/view/AtlasProtected/MonteCarloWorkingGroup

Using LHEF Generators

LHEF = Les Houches Event File Format:

- specialized generators (Alpgen, MC@NLO, POWHEG-box . . .) produce only parton level output,
- parton shower and hadronization should be provided by a general purpose generator (Herwig($++$),Pythia(8)).
- in order to pass events from the specialized gen. $\rightarrow$ general purpose gen. LHEF is used.

**Generating LHEF Generator + general purpose generator events within Athena:**

- generate the parton level events in the LHEF format outside Athena (use GENSER installations),
- make the input events available for access within Athena,
- run the general purpose generator set so that the ME-level events are read from the LHEF files.
- Example from the MC10 central production JobOptions (MC@NLO + Herwig):

```
from MC10JobOptions.McAtNloEvgenConfig import evgenConfig

# dummy needed
evgenConfig.inputfilebase = 'mcatnlo'
try:
  if runArgs.ecmEnergy == 7000.0:
    evgenConfig.inputfilebase = 'group09.phys-gener.mcatnlo341.105200.ttbar_7TeV.TXT.v1'
```

Event Generation Schema
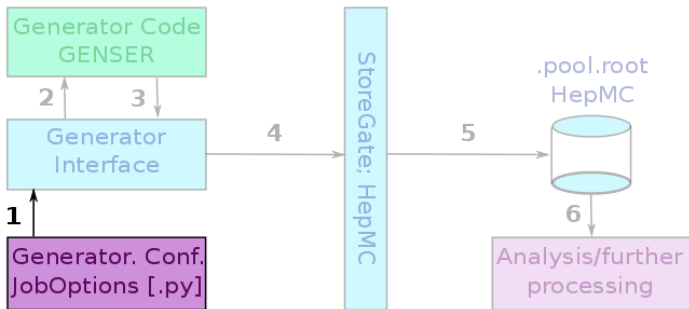
- ■ Q: Which generators can be used acc. to this schema?
- ■ A: https://twiki.cern.ch/twiki/bin/view/AtlasProtected/MonteCarloWorkingGroup

1) Passing Generator Configuration to the Interface

Technicalities:

- generator configuration is steered via Python (JobOption) files,
- the JobOption commands are parsed by the generator interface (Athena; C++).
- Examples of how to write the JobOptions follow in the Exercises section.

Getting DataSet Info from the JobOptions:

- The fact that JobOptions specify the generator configuration (for non-LHEF generators) makes them useful for extracting the info about the centrally produced samples.
- Example: the (in)famous QCD dijet samples 105009-1059017 conventionally called *Pythia J0 Sample*,...*Pythia J8 Sample*.
- The best way of really knowing what the contents of these samples are (I can think of) is to look into the JobOption files.
- For the MC10 production the JobOption files reside in Generators/MC10JobOptions package, svn repo location is:

  https://svnweb.cern.ch/trac/atlasoff/browser/Generators/MC10JobOptions/trunk/share/MC10.105009.J0_pythia_jetjet.py
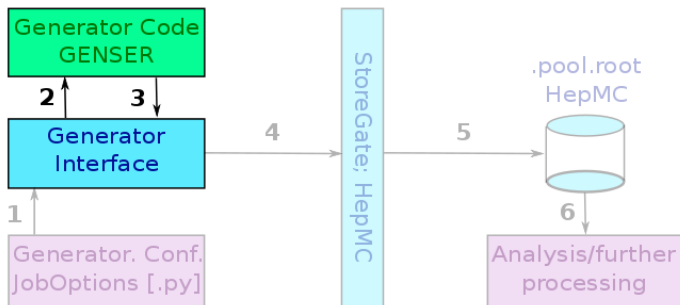
  https://svnweb.cern.ch/trac/atlasoff/browser/Generators/MC10JobOptions/trunk/share/MC10.105010.J1_pythia_jetjet.py

  ...

- You should be able to do/follow the interpretation after the Pythia exercise session.

Event Generation Schema

- **Q:** Which generators can be used acc. to this schema?
- **A:** https://twiki.cern.ch/twiki/bin/view/AtlasProtected/MonteCarloWorkingGroup

2) - 3) Generator Interface - MCGenerator Code Interaction

Technicalities:

- Generator interfaces operate within Athena and are written in C++ for all generators (incl. the F77 ones).
- Packages with the interface code are located in the `Generators` package; `https://svnweb.cern.ch/trac/atlasoff/browser/Generators` .
- There are many examples of JobOption files in the `share` dir.-s of `Generators/MC0XJobOptions` packages; these can be used as templates on which you base your JobOptions file; solid understanding of the interface code in not necessary for generating events within Athena (but can be useful, especially for more advanced use-cases).

GENSER repository:

- generator interfaces provide links to the GENSER builds of the generator codes;
- the code is located here:

```
[atlas14] /afs/cern.ch/sw/lcg/external/MCGenerators $ ls
Sherpa   blackmax      epos          herwig++  hydjetpp  mcfm      photos++    professor  rivet
acermc   cascade       evtgenlhc     herwigpp  isajet    mctester  photospp    pyquen     sherpa
agile    charybdis     hej           hijing    jimmy     nlojet++  pomwig      pythia6    stagen
alpgen   charybdis2    hepmcanalysis hydjet    lhapdf    phojet    powheg      pythia8    starlight
baurmc   distribution  herwig        hydjet++  mcatnlo   photos    powheg-box  qcdloop    tarFiles
```
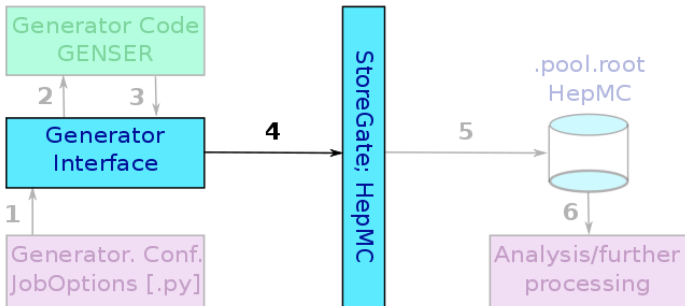
- DESY mirror: `/afs/desy.de/group/alliance/mcg/public/`.
- From the user-side useful for e.g. running the LHEF generators (no Athena interface providing the linking).

4) Event Generation Schema

- Q: Which generators can be used acc. to this schema?
- A: https://twiki.cern.ch/twiki/bin/view/AtlasProtected/MonteCarloWorkingGroup

4) Event Generation Output

- Internal Event Record differs from generator to generator;
- one of the gen. interface's functions is the output of the events to StoreGate in the generic **HepMC** format;
- HepMC provides 4-vectors and utils, rules for status-codes and utils to traverse the event (GenVertex,GenParticle); you will be working with this format when analyzing the output obtained by running MC generator within Athena (without further processing).
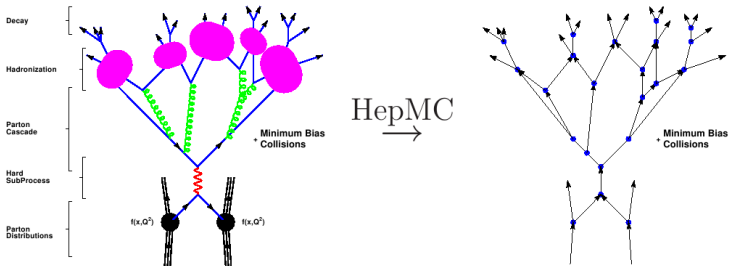

Figure from the HepMC docu source of choice:
http://lcgapp.cern.ch/project/simu/HepMC/206/HepMC2_user_manual.pdf

4) Event Generation Output Cont'd

Apart from the events in HepMC format further relevant output stems from the event generation stage;

- output from the generator (typically including header, info on included processes, warnings, errors ...),
- output from the interface (typically including tune/parameter setup info, warnings, errors ...).
- The output of the standard event generation run within Athena is dumped to a file conventionally named Evgen.log.
- Evgen.log files also contain the info on the generator-level Meta Data (e.g. process x-section, more info in the next slide).
- Evgen.log contains the transform exit status; N.B. if a transform doesn't exit with 0 the output events are not stored for the grid jobs.

```
Py:Athena          INFO leaving with code 0: "successful run"
/afs/cern.ch/atlas/software/releases/16.6.2/AtlasCore/16.6.2/InstallArea/share/bin/athena.py -b
runargs.Evgen.py EvgenJobTransforms/skeleton.Evgen.py Evgen_messageSvc_jobOptions.py - exit code 0.
```

- The Evgen.log files are stored for all available centrally produced DataSets; we'll have a closer look at them in the Tutorial Sessions.
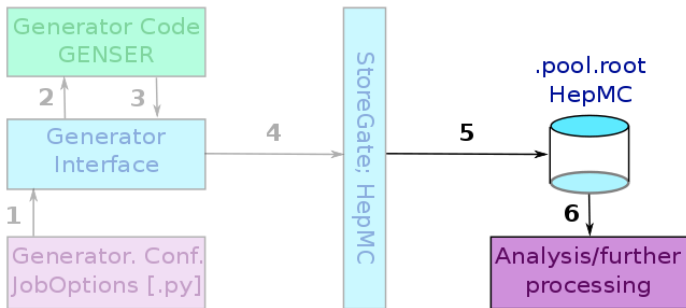
AMI Info [Hands-on]

- What is the x-section of the ttbar sample with ID 105200?
- Information is available via the **A**TLAS **M**etadata **I**nterface .
- AMI search utils can be used to find the event generation output files for the sample;
- to save some time let's use the link below to access the information directly;
- click on details link, located as shown in the snap below.

```
https://ami.in2p3.fr/AMI/servlet/net.hep.atlas.Database.Bookkeeping.AMI.Servlet.Command?Converter=
/AMIXmlToAMIProdHtml.xsl&Command=FormBrowseDatasetPerParameter
+-datasetNumber=105200+-dataType=EVNT+-version=e598
```

Event Generation Schema

- Q: Which generators can be used acc. to this schema?
- A: https://twiki.cern.ch/twiki/bin/view/AtlasProtected/MonteCarloWorkingGroup

5)-6) Analysis and Further Processing

Analysis of the evgen-level outputs:

- occasionally the generator-level events are also used for analysis (generator validation, gen. level studies),
- For the analysis it might be useful to add truth jets and MET info:

```
[lmijovic@lxplus247]/tmp/lmijovic% csc_addTruthJetMet_trf.py --help
Setting runOpts.help to True
JobTransform csc_addTruthJetMet version EvgenJobTransforms-00-06-81
Add TruthJet/TruthMet on evgen Based on csc_atlfast_trf.py
usage: csc_addTruthJetMet_trf.py [options] <inputevgenfile> <outputaodfile> [maxevents] <skipevents>
```

- Friday: hands-on generator-level event analysis sessions.
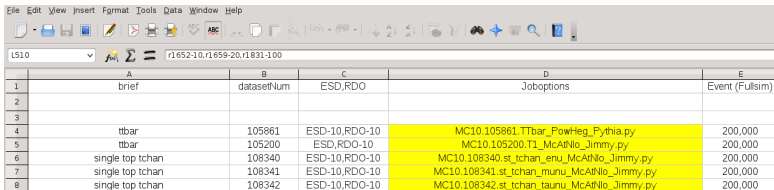
Further Processing:

- for the central production the output events from running the generator-level JobTransforms are stored to disk from where further processing steps (adding detector simulation) using them as inputs read them.
- What you will typically encounter when looking into truth-level info in e.g. AODs are not the generator level HepMC events, but its derivatives obtained after further processing;
- both the format and the contents of the events are affected by the processing; https://twiki.cern.ch/twiki/bin/view/AtlasProtected/McEventCollection

Generator-level Info About the Central Prod. Samples [Hands-on]

Example use-case:

- you are performing a SUSY search and you suspect ttbar production is a relevant source of background.

- Which DataSet(s) should you use for the backgrounds study?

- Centrally produced DataSets docu at the AtlasProduction wiki pages:

- `https://twiki.cern.ch/twiki/bin/view/AtlasProtected/AtlasProductionGroup`

- `wget http://www-f9.ijs.si/∼kersevan/Physics_mc10_7TeV.xls`

	A	B	C	D	E
1	brief	datasetNum	ESD,RDO	Joboptions	Event (Fullsim)
2					
3					
4	ttbar	105861	ESD-10,RDO-10	MC10.105861.TTbar_PowHeg_Pythia.py	200,000
5	ttbar	105200	ESD,RDO-10	MC10.105200.T1_McAtNlo_Jimmy.py	200,000
6	single top tchan	108340	ESD-10,RDO-10	MC10.108340.st_tchan_enu_McAtNlo_Jimmy.py	200,000
7	single top tchan	108341	ESD-10,RDO-10	MC10.108341.st_tchan_munu_McAtNlo_Jimmy.py	200,000
8	single top tchan	108342	ESD-10,RDO-10	MC10.108342.st_tchan_taunu_McAtNlo_Jimmy.py	200,000

- It is also a good idea to consult your colleagues from the SUSY group and visit the MC pages of the group that has requested the samples;

- ⇒ **Sample of choice: 105200**.

Tutorials After the Coffee Break and on Friday

- Thu: work on writing the jO files / generating events and interpreting the jOpts of the central production jO.
- Fri: Analysis Tutorial sessions.

Thank You