Table of Contents

B Physics Analysis Software

MC generation and detector simulation

The CMS management recommendation is to centrally produce the Monte Carlo sample, managed by the Generator group, for that you should refer to the current generator contact Ye Chen. Here you can find the page where you can insert the production requests. The private production is allowed, even if discouraged, but an expert revision of the simulation steps and configuration files is required. A good practise would be to publish the privately generated samples in the DBS system and let them reachable in the Grid.

For a general introduction on how to use the CMSSW framework to generate events, we refer the user to the WorkBook, in particular chapter 6. CMSSW allows to generate events using the "full" and the "fast" simulation routines. The former requires the following simulation steps: generation, (Geant) simulation, digitization, optionally the emission of a trigger signal, and the subsequent reconstruction. The latter ("fastsim") replaces the full detector simulation part by a parameterized detector response. The recent productions are anyway done in the fullsim mode.

The MC generator relevant for particle generation in the B-physics group is PYTHIA which is used for the p-p collision simulation, parton showering and hadronisation. Typical decay generators are EvtGen (EvtGen's usage in CMS and in LHCb) and TAUOLA. Pythia is particularly inefficient in the Bc production and a dedicated generator is usually used: Bc generator.

Current centrally produced samples can be checked in the MCM webpage.

Analysis software

The software used in the analyses is developed in the CMSSW framework (EDAnalyzer). The collaboration provides reconstructed data and simulated samples in the RECO tier, however it is recommended to add the PAT tier on top of the RECO tier and use it as the main source of the physical objects needed in the analysis (electrons, muons, jets, etc). The following webpage gives a generic description on how PAT is used in the B-physics group, containing links to examples in the HeavyFlavorAnalysis package PAT related B-Physics packages.

The common B physics analysis software should reside in the HeavyFlavorAnalysis package in the CMSSW git directory. Currently the site in not populated. In principle the analysis softwares developed by the analysts should be put into the HeavyFlavorAnalysis directory, but that implies a steady maintenance to make the software available and reliable in each of the future CMSSW releases. This option is preferable in case your code is a) common to the B physics group or b) should serve as an example / test for our group then you have to submit the code to the CMSSW area. The other way is to put the code outside the cmssw framework. It's normal practise to make the analysis software available to all the collaboration for checks and for the benefit of the future analyses. The current "official" private repository is the github host, here you can find a guide written for the cms users.

The note "Version control policy for CMS physics analysis codes", CMS IN-2009/017 gives recommendations on how physics analysis code (developed within the PAG groups) should be organised and also where it should be placed. It suggests to keep the directory structure based upon the organisation of PAGs.

Running on data

Valuable information on the datasets usable in the data analysis can be found in the Physics Data And Monte-Carlo Validation twikipage.

Analysis tools

- Global Tags for Conditions Data: twikipage where you can find info about the global tags used in data and usable in the analysis reconstruction.
- LXR: webtool where you can find the information about the CMSSW packages.
- MCM: is the Monte Carlo production management tool, in there you can find the information about the produced samples.
- DAS: is a webtool that merges many information, you can use it to retrieve information about the Monte Carlo and data datasets.
- WBM: is a webtool where you can get all the information about the experiment at running time.
- ConfDB: is a webtool where you can find detailed information about the trigger tables used in the Monte Carlo samples and in the data harvesting.

Tutorials

1.) In the following we show, using the J/psi production example, how the steps from MC generation to reconstruction, using fullsim, look like.

▸ Tutorial: J/psi production and full detector sim... ▾ Hide Contents
We describe a recipe that works on the lxplus cluster, using CMSSW_3_1_0_pre11, following the generic instructions given in CMS.ProdCfgFileGuide21X.

Make a directory for the Starter Kit and create a new project area. (You might want to consider creating this directory in your scratch0 directory, since your assigned space on lxplus is limited):

```
mkdir MyQuarkonia
cd MyQuarkonia
scramv1 list CMSSW   #provides you with a list of currently installed CMSSW versions
scramv1 project CMSSW CMSSW_3_1_0_pre11
cd CMSSW_3_1_0_pre11/src
cmsenv
```

Check out the generation config files from the CVS repository:

```
cvs co Configuration/CMS.GenProduction
```

In case you have already a config file (or have modified a file in this directory) compile it:

```
cp /PATH/TO/YOUR/CFG/YOUR-CONFIG-FILENAME_cff.py  ./Configuration/CMS.GenProduction/python
scramv1 b
```

The cmsDriver.py utility (which you can use once you've setup your CMSSW session with the command "cmsenv") can be used to generate a full configuration (python) script that uses your generation fragment, and adds code to do the individual steps from the generation up to the digitization and HLT emission (clearly, depending on how far you want to go in your simulation, you can only use a subset of the tags following the -s option). Go into the directory in which you want to do your generation and the subsequent analysis. E.g. you could checkout the code for the analyzer you want to use lateron (e.g. cvs co HeavyFlavorAnalysis/Onia2MuMu) and work in the subdirectory /test or /scripts. Now you are ready to create the generation and simulation configuration script:

```
cmsDriver.py Configuration/CMS.GenProduction/python/PYTHIA6_JPsi_10TeV_cff.py \
                -s GEN:ProductionFilterSequence,SIM,DIGI,L1,DIGI2RAW,HLT \
                --eventcontent RAWSIM \
                --datatier GEN-SIM-RAW \
                --conditions FrontierConditions_CMS.GlobalTag,STARTUP31X_V1::All \
                -n 5000 \
                --no_exec
```

Note that for versions 34X onwards the snippet "GEN:ProductionFilterSequence" should be replaced by "GEN" only. (See also Updates on cmsDriver.py and Updates on Generation for releases of 34X onwards). This will produce a file called "PYTHIA6_JPsi_10TeV_cff_py_GEN_SIM_DIGI_L1_DIGI2RAW_HLT_STARTUP31X.py" that will generate 5000 events. For an explanation of all the option of the "cmsDriver.py" command, consult ComposeFullSimConfig; for explanations concerning the language used in the simulation config file, consult WalkThroughCfg.

Check now if the cuts on the generator level are to your satisfaction. In order to use this script with version 31X change the lines

```
process.source = cms.Source("PythiaSource",
comEnergy = cms.untracked.double(10000.0),
process.ProductionFilterSequence = cms.Sequence(process.oniafilter*process.mumugenfilter)
```

to

```
process.source = cms.Source("EmptySource")
process.generator = cms.EDFilter("Pythia6GeneratorFilter",
comEnergy = cms.double(10000.0),
process.ProductionFilterSequence = cms.Sequence(process.generator*process.oniafilter*process.mumu
```

and then start the simulation with the command

```
cmsRun PYTHIA6_JPsi_10TeV_cff_py_GEN_SIM_DIGI_L1_DIGI2RAW_HLT_STARTUP31X.py
```

The second step is to do the reconstruction. The corresponding configuration file can again be created using the cmsDriver.py utility. Note that HLT and RECO cannot be run at the same time (software conflicts) and, hence, two steps are necessary.

```
cmsDriver.py Configuration/CMS.GenProduction/python/PYTHIA6_JPsi_10TeV_cff.py \
                -s RAW2DIGI,RECO \
                --filein file:PYTHIA6_JPsi_10TeV_cff_py_GEN_SIM_DIGI_L1_DIGI2RAW_HLT.root \
                --eventcontent RECOSIM \
                --datatier GEN-SIM-RECO \
                --conditions FrontierConditions_CMS.GlobalTag,STARTUP31X_V1::All \
                -n -1 \
                --no_exec
```

This will create a second configuration file, called "PYTHIA6_JPsi_10TeV_cff_py_RAW2DIGI_RECO_STARTUP31X.py". The reconstruction can be achieved by executing the script via the command

```
cmsRun PYTHIA6_JPsi_10TeV_cff_py_RAW2DIGI_RECO_STARTUP31X.py
```

Its output is a root file with name "PYTHIA6_JPsi_10TeV_cff_py_RAW2DIGI_RECO.root". Due to the fact that only a small fraction of the initially generated 5000 events fell into the detector's acceptance (muons must be within |eta| < 2.4 and have pT > 2.5 GeV/c), the number of reconstructed events in this sample is (only) 34. Its output can be inspected in a root session using a TBrowser. The TTree "Events" contains a lot of RECO objects of which the branch "recoMuons_muons__RECO" is of interest to us. It contains the muon objects "recoMuons_muons__RECO.obj" that contain the info about the individual muons. Clicking on the muons' methods, like charge, eta, pT, IsCaloMuon, ... you can plot the distribution of the fully reconstructed muons. "size" is a variable that shows how many muons were reconstructed in each event that passed the generator filter cuts. The fact that there are events with up to 4 reconstructed muons shows that not only the muons of the J/psi give rise to reconstructed muons. Together with charmonium we simulated the underlying event that on one hand contains further muons (e.g. from pion, kaon, open charm decays) or contributes to reconstructing fake muons due to the larger hit multiplicites.

After these two steps (generation/detector simulation and reconstruction) you are ready to further process the events with a given analyzer. The usage of the individual analyzer of the B-physics group can be found in the section Analysis software.

2.) The following link gives a tutorial on how to generate the channel B --> J/psi and how to "reconstruct" events using fast sim:
Tutorial: Inclusive B--> J/psi and fast sim

3.) The next link shows a tutorial on how to do an analysis with PAT:
Tutorial: Sample B reconstruction with PAT

4.) The recent EJTERM V school contain two exercises relevant to quarkonium analyes:

- Measurement of first dimuons in CMS
- Upsilon Analysis

5.) CMSDAS2011:

- Upsilon exercise @ CMSDAS2011 (instructions, discussion forum)

In case you are interested in quarkonium related issues you can check out the dedicated workspace. The Quarkonia webpage contains software related discussions about quarkonium production, including explanations on the "filterEfficiency" and "crossSection" parameters of the simulation.

Contacts

- **Group home page**: B Physics working group main page
- **Conveners**: (Jack) Kai-Feng Chen and Hermine Woehri
- **Hypernews fora**:
    - General: https://hypernews.cern.ch/HyperNews/CMS/get/bphysics.html
    - Exclusive decays: https://hypernews.cern.ch/HyperNews/CMS/get/bphysics-exclusive.html
    - Inclusive measurements: https://hypernews.cern.ch/HyperNews/CMS/get/bphysics-inclusive.html
    - Leptonic B decays: https://hypernews.cern.ch/HyperNews/CMS/get/bph-bmm.html
    - Trigger: https://hypernews.cern.ch/HyperNews/CMS/get/bph-trigger.html

Review status

Reviewer/Editor and Date	Comments
KatiLassilaPerini - 18 April 2008	created template page

Responsible: HermineWoehri
Last reviewed by: