

# Full Chain MC Hands-on

Tongguang Cheng

# Why do we need MC?

- Feasibility study
- Calculate efficiency and acceptance
- Get the property of signal (or background)
- Optimize selections
- .....

# Login IHEP farm and setup CMSSW

```
ssh your\_asf\_name@lxlogin.ihep.ac.cn  
source /cvmfs/cms.cern.ch/cmsset_default.sh (setup cmssw)  
mkdir PrivateMC  
cd PrivateMC  
cp /publicfs/cms/user/tocheng/Neutrino\_E-10\_gun\_Premix.root .  
cmssw-el8 (use singularity)  
export SCRAM_ARCH=el8_amd64_gcc10  
cmsrel CMSSW_12_4_14_patch3  
cd CMSSW_12_4_14_patch3/src/  
cmsenv  
cd -
```

# Optional: setup grid certificate

```
cd ~  
mkdir .globus  
cp myCertificate.p12 .globus  
cd .globus  
rm -f usercert.pem  
rm -f userkey.pem  
openssl pkcs12 -in myCertificate.p12 -clcerts -nokeys -out usercert.pem  
openssl pkcs12 -in myCertificate.p12 -nocerts -out userkey.pem  
chmod 400 userkey.pem  
chmod 400 usercert.pem  
  
voms-proxy-init
```

**GEN**  
**Hard scattering**  
**Hadronization**  
**Validation**

What physics are we studying?  
What generator should be used?  
Which parameters should be modified?

**SIM**

CMS Geometry, Magnetic field,...

**DIGI,L1,DIGI2RAW,HLT**

Detector electronics simulation, Pileup situation,  
Alignment-Calibration, Trigger menu,...

**RAW2DIGI,L1Reco,RECO,  
VALIDATION,DQM**

Reconstruction algorithms,  
High level object creation

**Your Analysis**

Your analysis code

https://cmsweb.cern.ch/das/



results format:  ,  results/page, dbs instance  ,

[Show DAS keys description](#)



### Help: DAS queries

DAS queries are formed by **key=value** pairs, for example

- dataset=/ZMM\*/\*\*
- release=CMSSW\_2\_0\_\*
- run=148126

The wild-card can be used to specify the pattern. The list of supported DAS **keys** can be found in [Services](#) section. For more details please read DAS [Frequently Asked Questions](#).

[hide](#)

https://cmsweb.cern.ch/das/request?instance=prod/global&input=config+dataset%3D%2FUpsilon2Mu\_UpsilonFilter\_2MuFilter\_TuneCP5\_13p6TeV\_pythia8%2FRun3Summer22EEMiniAODv3-124X\_mcRun3\_2022\_realistic\_postEE\_v1-v2%2FMINIAODSIM

results format:  ,  results/page, dbs instance  ,

config dataset=/Upsilon2Mu\_UpsilonFilter\_2MuFilter\_TuneCP5\_13p6TeV\_pythia8/Run3Summer22EEMiniAODv3-124X\_mcRun3\_2022\_realistic\_postEE\_v1-v2/MINIAODSIM

[Show DAS keys description](#)

 mongoDB

Showing 1—3 records out of 3.

[<first](#) | [prev](#) | [next](#) | [last](#)>

Request name: **cmsRun**

Created by: /DC=ch/DC=cern/OU=computers/CN=wmagent/vocms0255.cern.ch Creation time: 2022-11-29 04:11:09 Global Tag: 124X\_mcRun3\_2022\_realistic\_postEE\_v1 Pset hash: GIBBERISH Release: CMSSW\_12\_4\_11\_patch3 Request urls:

ReqMgr info Sources: [dbs3](#) [hide](#)

DAS service: [dbs3](#) DAS api: outputconfigs

```
release_version:"CMSSW_12_4_11_patch3"
module_label:"Merged"
created_by:"/DC=ch/DC=cern/OU=computers/CN=wmagent/vocms0255.cern.ch"
creation_time:1669695069
output_module_label:"Merged"
pset_hash:"GIBBERISH"
global_tag:"124X_mcRun3_2022_realistic_postEE_v1"
pset_name:
creation_date:1669695069
create_by:"/DC=ch/DC=cern/OU=computers/CN=wmagent/vocms0255.cern.ch"
name:"cmsRun"
```

Request name: **cmsunified\_task\_BPH-Run3Summer22EEGS-00008\_v1\_T\_221223\_075332\_7923**

Request urls: BPH-Run3Summer22EENanoAODv11-00011\_0

ReqMgr info Sources: [reqmgr2](#) [show](#)

https://cmsweb.cern.ch/reqmgr2/fetch?rid=cmsunified\_task\_BPH-Run3Summer22EEGS-00008\_\_v1\_T\_221223\_075332\_7923

**Request cmsunified\_task\_BPH-Run3Summer22EEGS-00008\_\_v1\_T\_221223\_075332\_7923**

Table JSON Splitting Config **Comment**

Status: normal-archived to normal-archived ▾

**Submit**

Field	Value
-------	-------

**Config Cache List**

- [Step1: BPH-Run3Summer22EEGS-00008\\_0: ConfigCacheID: 54565de7a0feb08440ada71d671cf86a](#)
- [Step2: BPH-Run3Summer22EEDRPremix-00012\\_0: ConfigCacheID: b075b519c2308daf80c7c5762fd6f7ac](#)
- [Step3: BPH-Run3Summer22EEDRPremix-00012\\_1: ConfigCacheID: b075b519c2308daf80c7c5762fd70c5f](#)
- [Step4: BPH-Run3Summer22EEMiniAODv3-00012\\_0: ConfigCacheID: b075b519c2308daf80c7c5762fd8dea3](#)



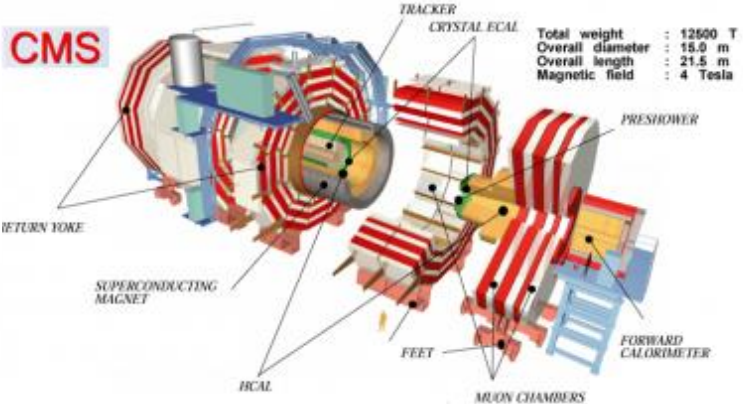
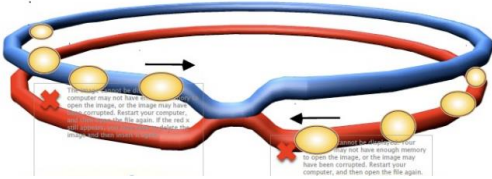
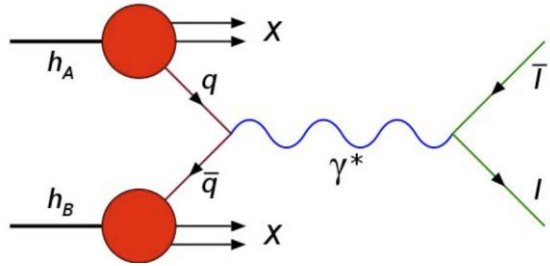
# GEN

- Madgraph, POWHEG, ...
- Pythia, Herwig, Sherpa, ...



# SIM

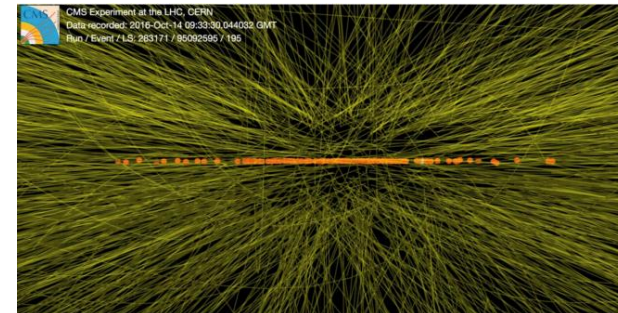
- Geant4, Delphes, ...



# GEN, SIM

Try:

```
cmsDriver.py Configuration/GenProduction/python/BPH-  
Run3Summer22EEGS-00008-fragment.py  
--python_filename step1_gen_sim.py --eventcontent RAWSIM  
--customise Configuration/DataProcessing/Utils.addMonitoring  
--datatier GEN-SIM --fileout file:BPH-Run3Summer22EEGS-00008.root  
--conditions 124X_mcRun3_2022_realistic_postEE_v1  
--beamspot Realistic25ns13p6TeVEarly2022Collision  
--step GEN,SIM --geometry DB:Extended --era Run3 --no_exec --mc  
-n 200
```



# GEN, SIM

Try:

```
cmsDriver.py Configuration/GenProduction/python/BPH-Run3Summer22EEGS-00008-fragment.py
--python_filename step1_gen_sim.py --eventcontent RAWSIM
--customise Configuration/DataProcessing/Utils.addMonitoring
--datatier GEN-SIM --fileout file:BPH-Run3Summer22EEGS-00008.root
--conditions 124X_mcRun3_2022_realistic_postEE_v1
--beamspot Realistic25ns13p6TeVEarly2022Collision
--step GEN,SIM --geometry DB:Extended --era Run3 --no_exec --mc
-n 200
```

What's wrong?

Instead copy the following config and cmsRun by :

```
cp /publicfs/cms/user/tocheng/step1_gen_sim.py .
```

```
cmsRun step1_gen_sim.py
```

# Configuration for GEN

<https://twiki.cern.ch/twiki/bin/view/CMS/GitRepositoryForGenProduction>

## A new repository for generator fragments

After the transition to *git*, Configuration/GenProduction is no more available in the CMSSW release. The repository for this package is now in <https://github.com/cms-sw/genproductions>.

Please refer to <http://cms-sw.github.io/cmssw/> for general information about *git* and register to it.

Below a short guide on the usage of the *genproductions* repository is given.

## How to use fragments from the repository

One can browse the repository on the web and copy or download the fragment with *curl*, e.g.

```
$ curl https://raw.githubusercontent.com/cms-sw/genproductions/master/genfragments/ThirteenTeV/WToENU/WToENU_M_1000_TuneCP5_13TeV_pythia8_cfi.py --create-dirs -o Configuration/GenProduction/python/Thir
```

Otherwise you can create a new empty directory and clone the full repository

```
$ mkdir -p Configuration/GenProduction/  
$ git clone --single-branch --depth=1 git@github.com:cms-sw/genproductions.git Configuration/GenProduction/
```

**Don't forget to do *scram b* before trying to use the fragments.**

GEN

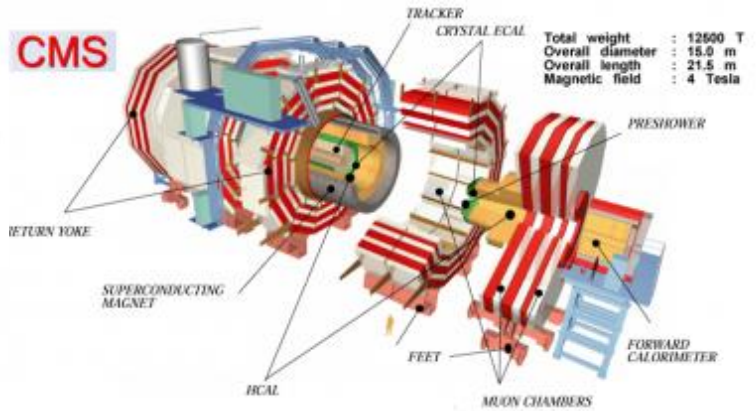
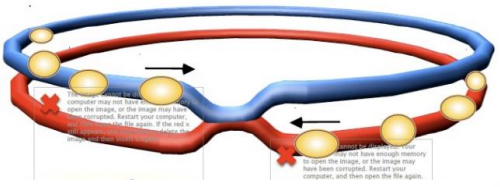
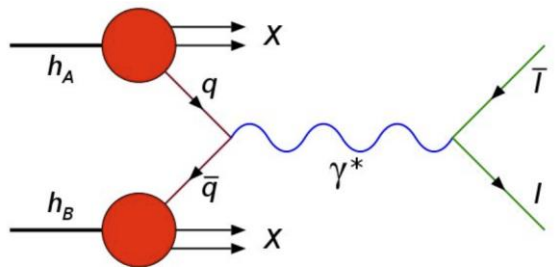


SIM



- Madgraph, POWHEG, ...
- Pythia, Herwig, Sherpa, ...

- Geant4, Delphes, ...



# Convert LHE file to ROOT format

(If you have created your own lhe file in the previous exercise, please copy it your working directory)

```
cp DIRECTORY_TO_YOUR_LHEFILE ./unweighted_events.lhe
```

```
cmsDriver.py step0 --filein file:unweighted_events.lhe --fileout file:LHE-13TeV.root --mc --eventcontent LHE --datatier GEN  
--conditions 124X_mcRun3_2022_realistic_postEE_v1  
--step NONE --python_filename LHE_13TeV_cfg.py --no_exec  
--customise Configuration/DataProcessing/Utils.addMonitoring -n -1
```

```
cmsRun LHE_13TeV_cfg.py
```

# GENSIM from LHE

cmsDriver.py

```
Configuration/Generator/python/Hadronizer_TuneCUETP8M1_13TeV_  
generic_LHE_pythia8_cff.py --filein file:LHE-13TeV.root  
--fileout file:GENSIM-13TeV.root --mc --eventcontent RAWSIM  
--datatier GEN-SIM --124X_mcRun3_2022_realistic_postEE_v1  
--beamspot Realistic25ns13p6TeVEarly2022Collision --step GEN,SIM  
--nThreads 4 --geometry DB:Extended --era Run3  
--python_filename GENSIM_13TeV_cfg.py --no_exec  
--customise Configuration/DataProcessing/Utils.addMonitoring -n -1
```

cmsRun GENSIM\_13TeV\_cfg.py

# GEN

- Madgraph, POWHEG, ...
- Pythia, Herwig, Sherpa, ...



# SIM

- Geant4, Delphes, ...



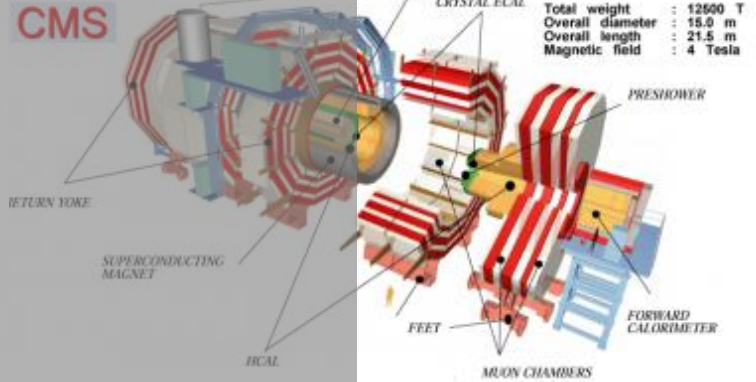
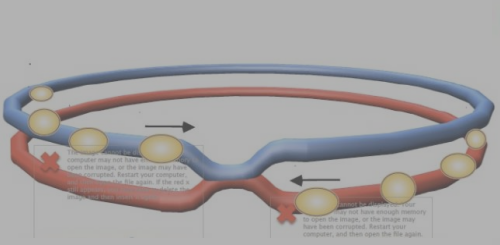
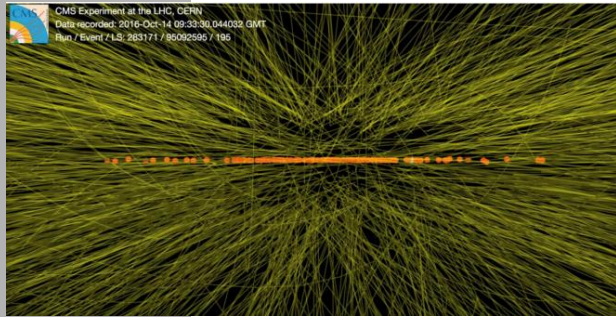
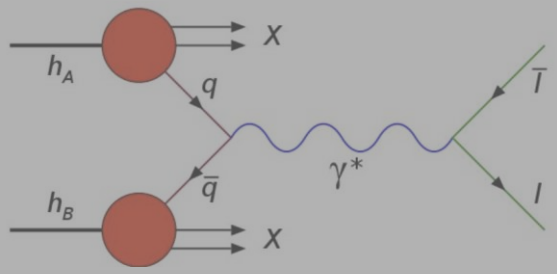
# DIGI, MIX

- Need CA to access DAS

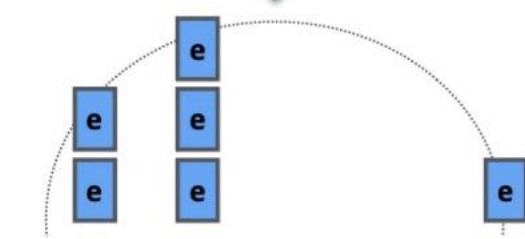
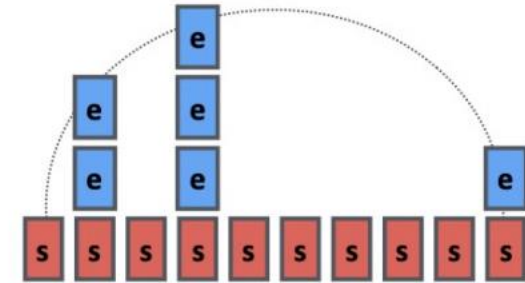
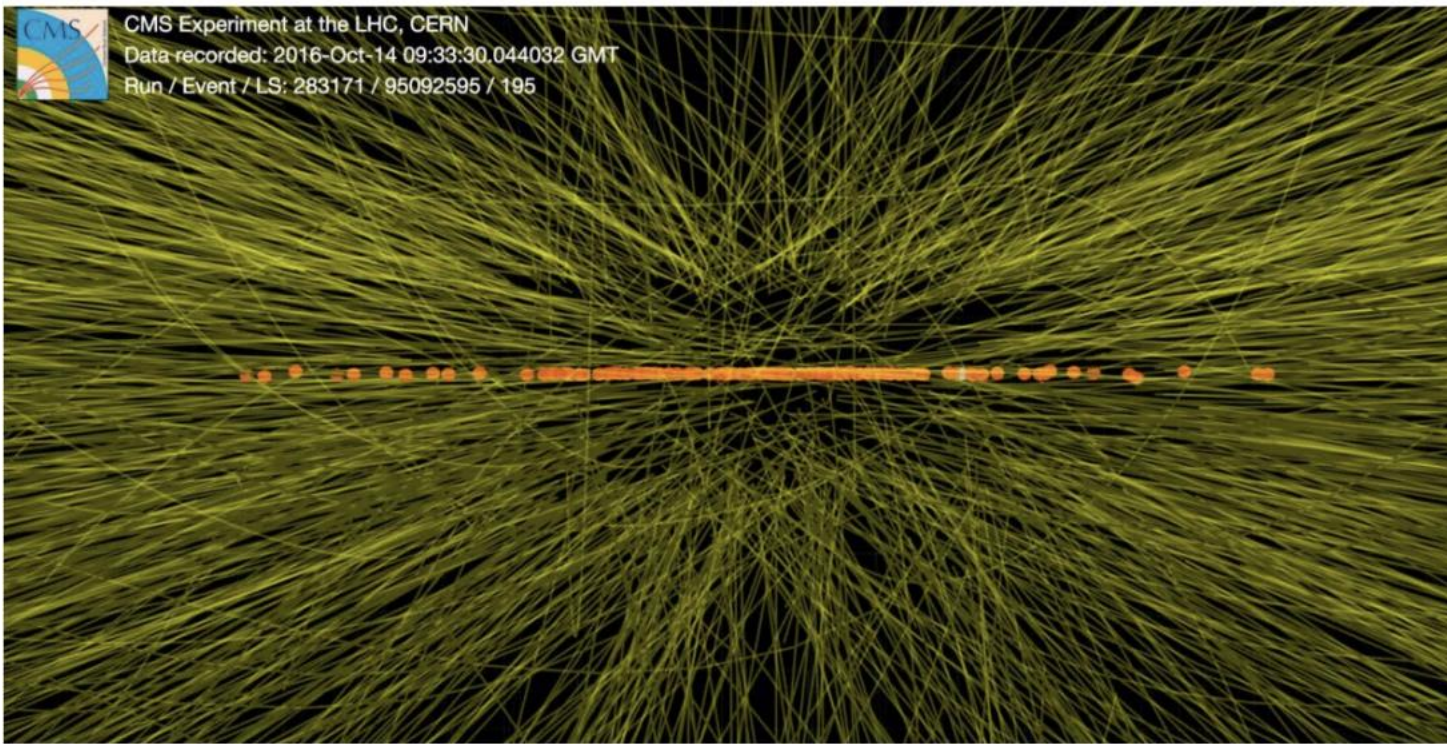


# L1

- 100 kHz
- ASIC
- FPGA
- PLD
- RAM







## Classic mixing

- GENSIM Signal (MC Hard-scatter event) is overlaid with GENSIM MinBias with chosen pileup configuration.

## Pre-mixing

- MinBias events in RAWSIM format are overlaid on empty single neutrino events using a chosen pileup configuration. Digis made in this step are converted to RAW.
- 1-1 combination of PreMixed event - signal event. RawToDigi is done on-the-fly to premixed events before overlay.

# DIGI, MIX, L1, RAW, HLT

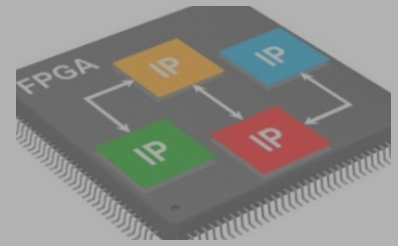
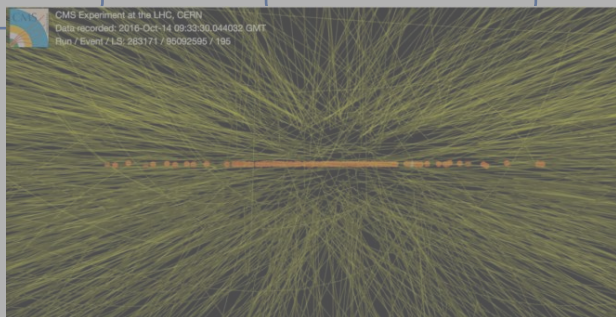
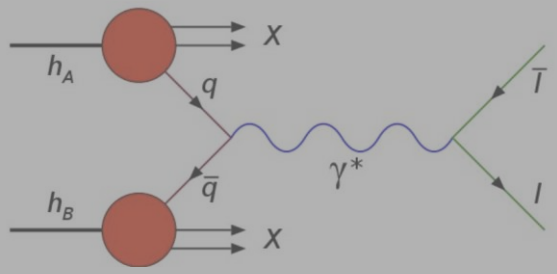
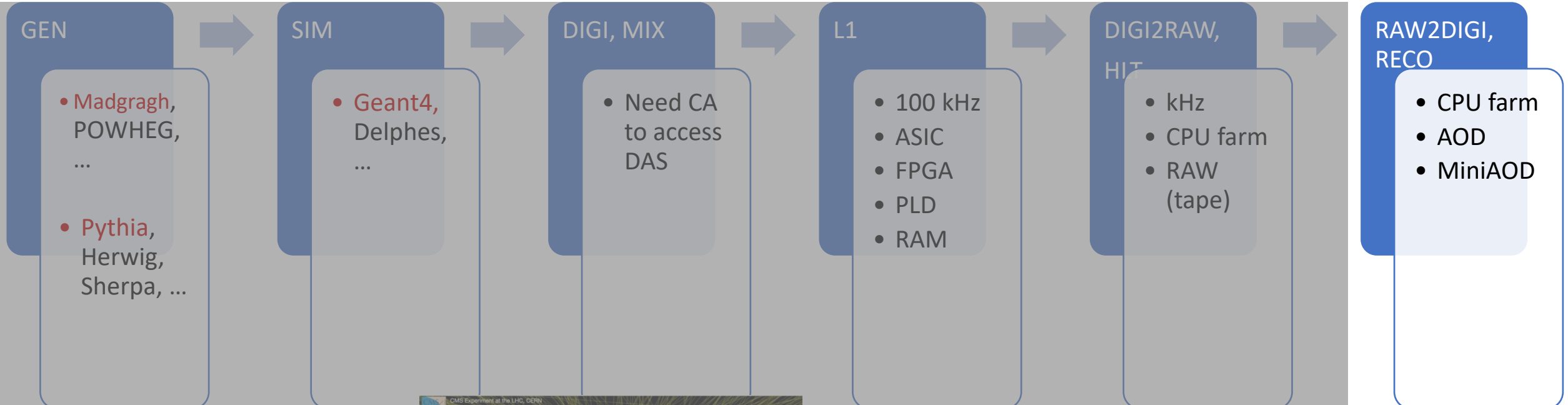
```
cmsDriver.py --python_filename step2_digi_mix_L1_HLT.py
--eventcontent PREMIXRAW
--customise Configuration/DataProcessing/Utils.addMonitoring
--datatier GEN-SIM-RAW
--fileout file:BPH-Run3Summer22EEDRPremix-00008.root
--pileup_input file:Neutrino_E-10_gun_Premix.root
--conditions 124X_mcRun3_2022_realistic_postEE_v1
--step DIGI,DATAMIX,L1,DIGI2RAW,HLT:2022v14
--procModifiers premix_stage2, siPixelQualityRawToDigi --nThreads 4
--geometry DB:Extended --filein file:BPH-Run3Summer22EEGS-00008.root
--datamix PreMix --era Run3 --no_exec --mc -n 100
```

[cmsRun step2\\_digi\\_mix\\_L1\\_HLT.py](#)

# DIGI, MIX, L1, RAW, HLT

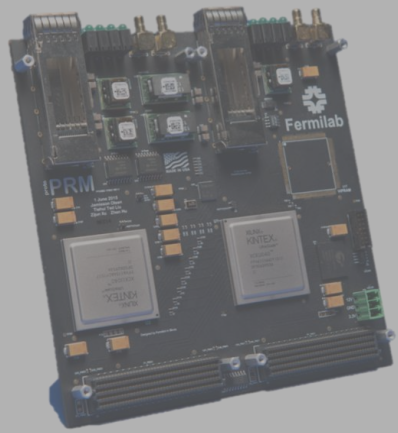
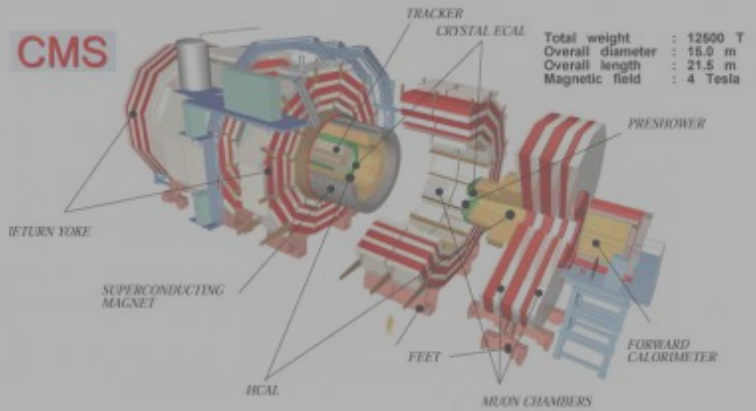
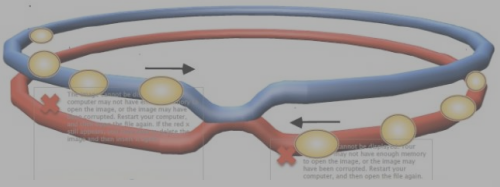
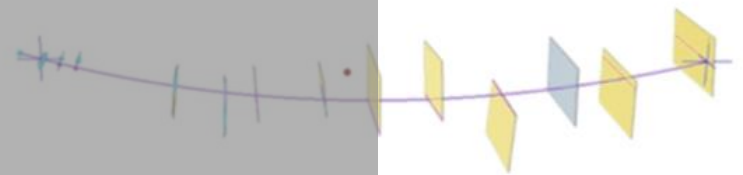
```
cmsDriver.py --python_filename step2_digi_mix_L1_HLT.py --eventcontent  
PREMIXRAW --customise Configuration/DataProcessing/Utils.addMonitoring  
--datatier GEN-SIM-RAW --fileout file:BPH-Run3Summer22EEDRPremix-  
00008.root  
--pileup_input dbs:/Neutrino_E-10_gun/Run3Summer21PrePremix-  
Summer22_124X_mcRun3_2022_realistic_v11-v2/PREMIX  
--conditions 124X_mcRun3_2022_realistic_postEE_v1  
--step DIGI,DATAMIX,L1,DIGI2RAW,HLT:2022v14  
--procModifiers premix_stage2,siPixelQualityRawToDigi --nThreads 4  
--geometry DB:Extended  
--filein file:BPH-Run3Summer22EEGS-00008.root  
--datamix PreMix --era Run3 --no_exec --mc -n 100
```

[step2\\_digi\\_mix\\_L1\\_HLT.py](#)



Partial Reco

Full Reco



# RECO

```
cmsDriver.py --python_filename step3_reco.py
--eventcontent AODSIM --customize
Configuration/DataProcessing/Utils.addMonitoring --datatier AODSIM
--fileout file:BPH-Run3Summer22EEDRPremix-00008_stage2.root
--conditions 124X_mcRun3_2022_realistic_postEE_v1
--step RAW2DIGI,L1Reco,RECO,RECOSIM
--procModifiers siPixelQualityRawToDigi --nThreads 4 --geometry DB:Extended
--filein file:BPH-Run3Summer22EEDRPremix00008.root --era Run3 --no_exec
--mc -n -1
```

```
cmsRun step3_reco.py
```

# MiniAOD(Analysis Object Data)

```
cmsDriver.py --python_filename step4_miniAOD.py
--eventcontent MINIAODSIM
--customise Configuration/DataProcessing/Utils.addMonitoring
--datatier MINIAODSIM
--fileout file:BPH-Run3Summer22EEDRPremix-00008_miniAOD.root
--conditions 124X_mcRun3_2022_realistic_postEE_v1 --step PAT --nThreads 2
--geometry DB:Extended
--filein file:BPH-Run3Summer22EEDRPremix-00008_stage2.root
--era Run3 --no_exec --mc -n -1
```

```
cmsRun step4_miniAOD.py
```

# NanoAOD([v11](#))

```
cmsrel CMSSW_12_6_0_patch1
```

```
cd CMSSW_12_6_0_patch1/src
```

```
cmsenv
```

```
cd -
```

```
cmsDriver.py step5 --mc --eventcontent NANOAODSIM --datatier NANOAODSIM  
--conditions 126X_mcRun3_2022_realistic_postEE_v1 --step NANO  
--nThreads 4 --scenario pp --era Run3,run3_nanoAOD_124  
--filein file:BPH-Run3Summer22EEDRPremix-00008_miniAOD.root  
--fileout file:BPH-Run3Summer22EEDRPremix-00008_nanoAOD.root  
--no_exec -n -1
```

```
cmsRun step5_NANO.py
```