Table of Contents

Migration from CMSSW 1_7_7 to 3_1

Contents

Migration Trace (Jun 01 2009 - Aug 21 2009)

Roadmap:

Step	Task	Duration	Progress
1	Transportation	Jun 01 - June 15	Done
2	Translation	16 Jun - 28 July	Done
3	Testing	29 July - 21 Aug	30%

Comments:

As the migration work is progressing, people are still updating the code based on the Totem software @ CMSSW_1_7_7. Version synchronization work (merge the updated code to the new Totem software @ CMSSW_3_1_X) is done at some certain check points. The following list shows synchronization history. For detail information about each synchronization, you can check the **Synchronization Log** section.

The work is carried out by:

1. Check out the latest code @ CMSSW_1_7_7 from the CVS.
2. Compare the code with migrating version.
3. Merge the code.

Syn ID	Date	CVS Check Out Date
1	10 July 2009	10 July 2009
2	31 July 2009	31 July 2009

Step 1: Transportation

Transport all the modules in the old Totem package (@ CMSSW_1_7_7) to the new framework CMSSW_3_1_0. Make sure that all modules get compiled successfully.

The module transporting trace is decided by the increasing dependencies among all these modules.

ID	Module
[1-1]	DataFormats

[1-2]	IOMC
[1-3]	TotemCondFormats
[1-4]	Configuration
[1-5]	SimG4CMS
[1-6]	SimG4Core
[1-7]	Geometry
[1-8]	EventFilter
[1-9]	TotemDataReadout
[1-10]	RecoTotemRP
[1-11]	TotemAlignment
[1-12]	SimTotem
[1-13]	L1TriggerTotem
[1-14]	RecoTotemT1T2
[1-15]	TotemAnalysis
[1-16]	TotemBackground
[1-17]	TotemRPValidation
[1-18]	TotemT1T2Validation
[1-19]	TotemDQM

NOTE:

1. IORawData module in the original Totem package (@ CMSSW_1_7_7) is obsolete, thus it is not migrated into the new version of Totem package (@ CMSSW_3_1_0).
2. Geometry/Records (@ CMSSW_1_7_7) will not be included into the new version of Totem package (@ CMSSW_3_1_0). Instead A new submodule Geometry/TotemRecords, which contains only RealGeometryRecord.[h,cc] and MisalignedGeometryRecord.[h,cc] about Totem, will take its place.

References:

- Offline Simulation Software - getting started
- Migration to newer version of CMSSW framework
- CMSSW Software Cross-Reference

Step 2: Translation

Transfer the configuration files in CMS-specific configuration language to the equivalent ones in Python Configuration Syntax.

The translation trace follows the transporting trace in step 1.

ID	Module
[2-1]	IOMC
[2-2]	Configuration
[2-3]	Geometry
[2-4]	TotemDataReadout
[2-5]	RecoTotemRP
[2-6]	TotemAlignment
[2-7]	SimTotem
[2-8]	L1TriggerTotem
[2-9]	RecoTotemT1T2

[2-10] TotemBackground
[2-11] TotemRPValidation
[2-12] TotemT1T2Validation
[2-13] TotemDQM

NOTE:

There are some old format configuration files which have not been translated because they are obsolete. The list for these files are:

- `RecoTotemT1T2/T2MakeCluster/test/*.cfg > ==idealOnlyTotemGeometryXML.cfi` is missing
- `RecoTotemT1T2/T2RecHit/test/*.cfg > ==idealOnlyTotemGeometryXML.cfi` is missing
- `RecoTotemT1T2/T2RoadProducer/*.cfg > ==idealOnlyTotemGeometryXML.cfi` is missing
- `TotemBackground/BeanGas/data/RP_sim_*.cfg` ==> refer to many obsolete cfi files

References:

- Introduction to CMS Configuration Files
- Description of the cmsRun Configuration File Language
- Description of the cmsRun Python Configuration Synta

Step 3: Testing

Run the test suit (`offline/cmssw/examples/`) to make sure that the preceding transportation process does not change any functionality of the Totem software. This process is done by running the `cmsRun` with a series of test configuration files. The resulting `root` files will be sent to the experts from T1/T2/RP, who are responsible for certain modules for further validation if necessary. This process may require to go back to the ***Transportation*** and ***Translation*** phases, because errors would be included in these two phases or some modules need to be updated.

The testing trace is shown as follows. The test suit can be found from `offline/cmssw/examples/`. There are two folders under this folder: `production/` and `validation/`. The configuration files under `production/` aim to simulate events ,do the track reconstruction, and produce the results. The configuration files under `validation/` aim to validate the results made by the files under `production/` and produce the validation summary results.

Target Detector	Production	Validation	Process
T1T2	gunT1T2mu_cfg.py	valgunT1T2mu_cfg.py	OK
	gunT1T2pi_cfg.py	valgunT1T2pi_cfg.py	OK
RP	RPelastic90_cfg.py	valRPelastic90_cfg.py	Fail
	RPelastic1535_cfg.py	valRPelastic1535_cfg.py	Fail
	RPinelastic2_cfg.py	valRPinelastic2_cfg.py	Fail
	RPinelastic90_cfg.py	valRPinelastic90_cfg.py	Fail
	RPinelastic1535_cfg.py	valRPinelastic1535_cfg.py	Fail
RP + T1T2	RPT1T2pythiaSDbeta2_cfg.py	valRPT1T2pythiaSDbeta2_cfg.py	Fail
	RPT1T2pythiaSDbeta90_cfg.py	valRPT1T2pythiaSDbeta90_cfg.py	Fail

-- ZhengkuiZhang - 07 August 2009

Change Log (Jun 01 2009 - Aug 21 2009)

Roadmap:

Step	Task	Duration	Progress
1	Transportation	Jun 01 - June 15	Done
2	Translation	16 Jun - 28 July	Done
3	Testing	29 July - 21 Aug	30%

NOTE: Because the change log is so long (>1000 lines), it is more advisable to only give a summary on what kinds of changes are made. The detail change log can be downloaded from https://twiki.cern.ch/twiki/pub/TOTEM/CompOfflineCMSSWMigration177to31/change_log.

Step 1: Transportation

Summary of changes

ID	1-1
Reason	Some head files have been moved to a new directory.
Change	Update the references in both BuildFiles and *.[h,cc] files accordingly.

Example-1	`<use name=SimDataFormats/HepMCProduct>`
	`-->  <use name=SimDataFormats/GeneratorProducts>`
	`#include "SimDataFormats/HepMCProduct/interface/HepMCProduct.h"`
	`-->  #include "SimDataFormats/GeneratorProducts/interface/HepMCProduct.h"`

Example-2	`#include "PluginManager/ModuleDef.h"` `-->  #include "FWCore/PluginManager/interface/ModuleDef.h"`

ID	1-2
Reason	Some root or C++ standard library file is not implicitly included any more.
Change	Include those library files explicitly in *.[h,cc] files.
Example-1	`#include "TClass.h"`
Example-2	`#include "TMath.h"`
Example-3	`#include <map>`
Example-4	`<use name=rootgraphics>` (change_log/[1-20])
Example-5	`#include <math.h>`
Example-6	`<use name=hepmc>` (change_log/[1-22])
Example-7	`<use name=SimDataFormats/GeneratorProducts>` (chang_log/[1-23])
Example-8	`<use name=clhep>` (change_log/[1-33])
Example-9	`#include "CLHEP/Vector/LorentzVector.h"` (change_log/[1-33])

ID	1-3
Reason	A circular dependency of the parent class and the child class happens. **`rootcint`** refuse to compile.

Change	Use trick fool the **`rootcint`**.
Example-1	change_log/[1-7]

ID	1-4
Reason	CMSSW_1_7_7 used **`CLHEP::Hep3Vector`** as the return type of **`CaloG4Hit::getEntry()`**, while from CMSSW1_8_4 **`math::XYZPoint`** was used instead.
Change	Replace **`Hep3Vector`** to **`math::XYZPoint`**
Example_1	change_log/[1-8]

ID	1-5
Reason	Some tools which were used in CMSSW_1_7_7 may not be needed in CMSSW_3_1_0 or be replaced by some other tools already.
Change	Remove them from **`BuildFiles`**.
Example_1	`<use name=Foundation/PluginManager>`
Example_2	`<use name=Framework/SealKernel>`
Example_3	`<use name=DataFormats/TotemData>`
Example_4	`<use name=RecoTotemRP/RPServiceRecords>`
Example_5	`<use name=SimGeneral/HepPDT>`

ID	1-6
Reason	The interfaces and implementations of some classes were changed from CMSSW_1_7_7 to CMSSW_3_1_0.
Change	Compare and merge the code.
Example_1	change_log/[1-12]
Example_2	change_log/[1-32]

ID	1-7
Reason	Some modules are obsolete.
Change	Do not migrate these modules.
Example_1	**`IORawData`** (change_log/[1-10])
Example_2	**`Geometry/Records`** (change_log/[1-13])

Step 2: Translation

Translation is done by mainly done by running the translation tool from the CMS workbook: Translation Utilities

1. Dump to the screen a translation of your configuration file: **`python cfg2py.py`**
2. Put the output file in the appropriate python/subdirectory of your package: **`python translate.py <subsystem/package/dir/your cfg, cff, or cfi>`**
3. Find the comments in your old file, and tries to translate them over to the python file: **`python comments.py`**
4. Test if your new python file will compile: **`python <subsystem/package/python/your py file>`**

If translation fail, manual modification is carried out. After translation, manual checking is carried out, because the translated configuration file will seldom keep the statements' sequence as in the old configuration file.

Summary of changes

ID	2-1
Reason	Some cfg files are obsolete.
Change	Delete these files.
Example-1	`IOMC/Elegent/test`
Example-2	`IOMC/Elegent/validation`
Example-3	`IOMC/EnergyVertexGenerator`
Example-4	`Geometry/TotemRPGeometryBuilder/data/PrintGeometry.cfg`

ID	2-2
Reason	"." which means reference should not be used to inside a variable name.
Change	Remove the dot inside the variables' name.

	Rename
	
	`"error.txt", "warning.txt", "info.txt", "debugmessage.txt"`
	`-->  "errors", "warnings", "infos", "debugs"`

Example-1	`Configuration/TotemCommon/python/LoggerMax_cfi.py`

ID	2-3
Reason	**`mix_none.cfi`** is obsolete and does not exist anymore since CMSSW_1_8_4. **`mixNoPU.cfi`** is the closest alternative.
Change	`include "SimGeneral/MixingModule/data/mix_none.cfi"` `-->  process.load("SimGeneral.MixingModule.mixNoPU_cfi")`
Example-1	`L1TriggerTotem/T1Trigger/test/triggerGunT1_cfg.py`

ID	2-4
Reason	Don't use number in the beginning of a file name. scram build complains about it.
Change	Rename the files' name.
Example-1	chang_log/[2-6]

ID	2-5
Reason	The volumeBasedMagneticField_cfi.py does not exist @ CMSSW_3_1_0, instead there are a few versions.
Change	We choose the 3.8 T default magnetic field map (**`Configuration/StandardSequences/python/MagneticField_38T_cff`**).
	`process.load("Configuration.StandardSequences.MagneticField_38T_cff")`
Example-1	`RecoTotemT1T2/T2TrackProducer/test/T2RecoProdExample_cfg.py`
Example-2	`RPInelasticReconstruction/test/runGun_transport_primvert_2stations_cfg.py`
Comment	The newest way on how to use the magnetic field which is compatible with CMSSW_3_1_1 can be found at Magnetic Field Interface

ID	2-6

Reason	The variable
	
	`g4SimHits.Generator.ApplyPtCuts`
	
	is renamed to
	
	`g4SimHits.Generator.ApplyPCuts`

Change	Rename the variable accordingly.
Example-1	change_log/[2-12]

ID	2-7
Reason	In cfg file, it is possible to use ***using PSet block*** to include a PSet block from other cfi files. However, in python, there is no equivalent mechanism.
Change	Make use of python's assign-by-reference property to bypass. **`RPTransportVolumesDimensions`** is obsolete, because it has the same contents as **`BeamProtTransportSetup`** which is the outer scope block of **`RPTransportVolumesDimensions`**. So **`BeamProtTransportSetup`** will take the place of **`RPTransportVolumesDimensions`**. The source code for reading parameters from **`RPTransportVolumesDimensions`** should also be changed to read from **`BeamProtTransportSetup`**.
	#1 **`BeamProtTransportSetup`** is declared in:
	`Configuration/TotemOpticsConfiguration/python/*_cfi.py`
	#2 we add a placeholder for **`BeamProtTransportSetup`** in:
	`RecoTotemRP/RPInelasticReconstruction/python/*_cfi.py` `Configuration/TotemCommon/python/g4SimHits_cfi.py`
	#3 when you include those files (#2), the reference to **`BeamProtTransportSetup`** needs to be redirected to the original **`BeamProtTransportSetup`** declarations (#1).

Example-1	`Configuration/TotemOpticsConfiguration/python/OpticsConfigBeta_(1535/05/2/90)_cfi.py`
	`Configuration/TotemCommon/python/g4SimHits_cfi.py`
	`TotemRPValidation/Examples/Data/*_cfg.py`
	`L1TriggerTotem/CoincidenceChip/test/simdigi_cfg.py`

Comment	More information about this problem and how it is solved can be found from: Assign-By-Reference in Python, change_log/[2-15] and change_log/[2-19]

ID	2-8
Reason	**`GeometryFilesBlock`** as an outer scope block from **`geomXMLFiles`** is not necessary. There is also no equivalent mechanism in python as the "using" in old configuration file.
Change	#1 **`GeometryFilesBlock.geomXMLFiles`** is delcared in:
	`Configuration/TotemCommon/data/geometry*.cfi`
	#2 **`GeometryFilesBlock.geomXMLFiles`** is appended with some xml file in:

	`L1TriggerTotem/CoincidenceChip/test/simdigi_cfg.py`
	`TotemRPValidation/HitDistributions/test/hitsBeta90_cfg.py`
	`TotemRPValidation/Examples/Data/inelastic_*_cfg.py`

	In (#1) where **`GeometryFilesBlock.geomXMLFiles`** is delcared : First put the **`GeometryFilesBlock.geomXMLFiles`** inside **`XMLIdealGeometryESSource`**. Then delete **`GeometryFilesBlock`**.
	In (#2) where **`GeometryFilesBlock.geomXMLFiles`** is appended with some **`xml`** file : use the following statement to append the **`xml`** file: `process.XMLIdealGeometryESSource.geomXMLFiles.append('an xml file name')`
Example-1	`Configuration/TotemCommon/data/geometryRP.cfi` `TotemRPValidation/Examples/Data/inelastic_2_smeared_sim_cfg.py`

ID	2-9
Reason	Make the pythia configuration files of Totem compatible with the ones in CMSSW_3_1_1.
Change	#1 **`PythiaSource`** is not registered as a plugin anymore, but **`Pythia6GeneratorFilter`** should be used.
	#2 The pythia module should be created as a **`generator`** of **`EDFilter`** not a **`source`** of **`EDSource`** any more. In order to use the pythia generator an empty source needs to be created. The random number generator should also include the seed for pythia generator.
	#3 **`'MSTP(52)=1'`** is added to the FORTAN code block named **`pythiaMinBias`**.
	#4 **`comEnergy = cms.double(X)`** is added.
Example-1	`SimG4CMS/Forward/python/test/PythiaSD_cfi.py`
Example-2	`SimG4CMS/Forward/python/test/PythiaDD_cfi.py`

Example-3	`SimG4CMS/Forward/python/test/PythiaMB_cfi.py`
comment	More info about the pythia fortun parameter switches: PYTHIA 6.4 Physics and Manual --> 9.3 The General Switches and Parameters (P 201 ~ P 225)

ID	2-10

Reason	No reflex dictionary info for the following classes:
	`pair<T1DetId,vector<T1Cluster> >`
	`pair<T1DetId,vector<T1DigiVfat> >`
	`pair<T1DetId,vector<T1DigiWire> >`
	`pair<T2DetId,vector<T2PadDigi> >`
	`pair<T2DetId,vector<T2StripDigi> >`
	`pair<T2DetId,vector<T2Cluster> >`

Change	Add reflex info for the target class into corresponding files:

Sources	`DataFormats/T1Cluster/src/classes_def.xml`
	`DataFormats/T1Cluster/src/classes.h`
	`DataFormats/T1DigiVfat/src/classes_def.xm`
	`DataFormats/T1DigiVfat/src/classes.h`
	`DataFormats/T1DigiWire/src/classes_def.xml`
	`DataFormats/T1DigiWire/src/classes.h`
	`DataFormats/T2Digi/src/classes_def.xml`
	`DataFormats/T2Digi/src/classes.h`
	`DataFormats/T2Cluster/src/classes_def.xml`
	`DataFormats/T2Cluster/src/classes.h`
	`DataFormats/T1DetId/src/classes_def.xml`
	`DataFormats/T1DetId/src/classes.h`

ID	2-11
Reason	Current version of Mixing module provides different naming schema for **`CrossingFrame`** products. One have to use ***g4SimHitsDetectorLabel*** instead of ***DetectorLabel***.

Change	`TotemHitsT2Gem --> g4SimHitsTotemHitsT2Gem`
	`TotemHitsT1 --> g4SimHitsTotemHitsT1`
	`TotemHitsT2Gem --> g4SimHitsTotemHitsT2Gem`
	`TotemHitsRP --> g4SimHitsTotemHitsRP`

Sources	`SimTotem/T2Digitizer/src/T2DigiProducer.cc`
	`SimTotem/T1Digitizer/plugins/T1DigiProducer.cc`
	`TotemT1T2Validation/T2RecoValidation/src/T2RecoAnalyzer.cc`
	`SimTotem/RPDigiProducer/src/RPDigiProducer.cc`

ID	2-12
Reason	Currently in dictionary of **`CrossingFrame`**, there is active flag (**`persistent="false"`**), which results in not saving **`CrossingFrames`** to ROOT file (described here: Creating New Products). There is new class, called **`PCrossingFrame`** which is persistent. As a result, validation run, defined in separate configuration file cannot access data saved by mixing module.

Change	Copy **`SimDataFormats/CrossingFrame/`** from CMSSW_3_1_1's reposity, save them to Totem software. Add reflex info into classes.h and src/classes_def.xml for:
	`std::vector<edm::HepMCProduct*>`
	`edm::Wrapper<CrossingFramePlaybackInfo >`
	`edm::Wrapper<CrossingFrame<PSimHit> >`
	`edm::Wrapper<CrossingFrame<PCaloHit> >`
	`edm::Wrapper<CrossingFrame<SimTrack> >`
	`edm::Wrapper<CrossingFrame<SimVertex> >`
	`edm::Wrapper<CrossingFrame<edm::HepMCProduct> >`
	`boost::shared_ptr<const edm::Wrapper<PCaloHit> >`
	`boost::shared_ptr<const edm::Wrapper<PSimHit> >`
	`boost::shared_ptr<const edm::Wrapper<SimTrack> >`
	`boost::shared_ptr<const edm::Wrapper<SimVertex> >`
	`boost::shared_ptr<const edm::Wrapper<edm::HepMCProduct> >`
	`boost::shared_ptr<const edm::Wrapper<std::vector<PCaloHit> > >`
	`boost::shared_ptr<const edm::Wrapper<std::vector<PSimHit> > >`
	`boost::shared_ptr<const edm::Wrapper<std::vector<SimTrack> > >`

	`boost::shared_ptr<const edm::Wrapper<std::vector<SimVertex> > >`
	`boost::shared_ptr<const edm::Wrapper<std::vector<edm::HepMCProduct> > >`
	`boost::detail::shared_count`
	`boost::detail::sp_counted_base*`
	`edm::Wrapper<PCaloHit>`
	`edm::Wrapper<PSimHit>`
	`edm::Wrapper<SimTrack>`
	`edm::Wrapper<SimVertex>`
	`edm::Wrapper<std::vector<edm::HepMCProduct> >`

Comment	Useful Link: Creating New Products

ID	2-13
Reason	Dictionary **`SimDataFormats/GeneratorProducts/src/classes_def.xml`** seems to be corrupted (**`"<"`** instead of **`"<"`**), this could be a reason of missing dictionary for **`HepMC::GenVertex`**. As a result, validation run, defined in separate configuration file cannot access data saved by event source.
Change	Copy **`SimDataFormats/GeneratorProducts/`** from CMSSW_3_1_1's reposity, save them to Totem software. In src/classes_def.xml replace "<" to "<" and ">" to ">".
Comment	Useful Link: CMSSW/SimDataFormats/CrossingFrame/src/classes_def.xml

ID	2-14
Reason	The **`EvtHandle`** label of **`TotemT1T2Validation.T2RecoValidation.T2RecoValidation`** needs to be exte it to "source" is not a good way. Because the label may not always be "source".

Change	`+ Add HepMCProductLabel = cms.string('source')`
	`into TotemT1T2Validation/T2RecoValidation/src/T2RecoValidation_cfi.py, Line 14`
	`+ Add HepMCProductLabel = cms.string('source')`
	`into TotemT1T2Validation/T2RecoValidation/src/T2RecoValidation2_cfi.py, Line 14`
	`+ Add HepMCProductLabel as a string variable`
	`into TotemT1T2Validation/T2RecoValidation/interface/T2RecoValidation.h, Line 104`
	`+ Add HepMCProductLabel = iConfig.getParameter<std::string>("HepMCProductLabel")`
	`into TotemT1T2Validation/T2RecoValidation/src/T2RecoAnalyzer.cc, Line 44`
	`+ Replace iEvent.getByLabel("source", EvtHandle) --> iEvent.getByLabel(HepMCProductLa`
	`in TotemT1T2Validation/T2RecoValidation/src/T2RecoAnalyzer.cc, Line 215`

Example-1	`offline/cmssw/examples/validation/valgunT1T2mu_pro_and_val_cfg.py`
Example-2	`offline/cmssw/examples/validation/ T1T2pythiaMBbeta2_pro_val_fullchain_cfg.py`

Remarks (Mirko Berretti)

The label of can no longer be 'source' for CMS-maintained generators.

In the old version of the framework all event generators were implemented as EDSources and as such, they all forced label `source'. In 3.x version, some generators (those maintained by CMS) became EDProducers and those cannot give the label `source'. A single label has been chosen for CMSSW 3_1_x (CMS-maintained) generator to be used by all (CMS-maintained) generator modules in the future, and it is 'generator'. This implies that further steps in the event processing chain that use HepMCProduct have been adapted, i.e. their 'InputTag' has changed to 'generator'. Please account for this in case your private analysis code uses HepMCProduct taken from (CMS-maintained) generator as Pythia Generator or Particle Gun.

For example, in order to retrieve HepMC particles in your analyzer, the following lines can be utilized (both for Particle Gun or Pythia Generator):

```
Handle<HepMCProduct> EvtHandle;
iEvent.getByLabel(HepMCProductLabel, EvtHandle);
```

where the std::string HepMCProductLabel is initialized with the value "generator".

| **Example-1** | TotemT1T2Validation/T2RecoValidation/T2RecoValidation_cfi.py

| **Example-2** | Further information:
https://twiki.cern.ch/twiki/bin/view/CMS/SWGuideEventGenerationReleaseNotes31X

Step 3: Testing

The following tables show the testing procedure for the example test configuration files under offline/cmssw/examples/production/ (and validation/) @ CMSSW_3_1_1.

ID	3-1
Source	`production/gunT1T2mu_cfg.py` `validation/valgunT1T2mu_cfg.py`

Steps	`[1] cmsRun gunT1T2mu_cfg.py ==> gunT1T2mu.root`
	`[2] copy production/gunT1T2mu.root to validation/`
	`[3] mkdir gunT1T2mu/ under validation/`
	`[4] cmsRun valgunT1T2mu_cfg.py ==> valT1PlotsMu.root, valT2PlotsG4Mu.root, valT2PlotsDigiMu.root, valT2PlotsRecoMu.root, valT2PlotsRecoMu2.root under validation/gunT1T2mu/`

ID	3-2
Source	`production/gunT1T2pi_cfg.py` `validation/valgunT1T2pi_cfg.py`

Steps	`[1] cmsRun gunT1T2pi_cfg.py ==> gunT1T2pi.root`
	`[2] copy production/gunT1T2pi.root to validation/`
	`[3] mkdir gunT1T2pi/ under validation/`
	`[4] cmsRun valgunT1T2pi_cfg.py ==> valT1PlotsPi.root, valT2PlotsG4Pi.root, valT2PlotsDigiPi.root, valT2PlotsRecoPi.root, valT2PlotsRecoPi2.root under validation/gunT1T2pi/`

-- ZhengkuiZhang - 15 July 2009

Synchronization Log (Jun 01 2009 - Aug 21 2009)

Synchronization 1 (10 July 2009)

CVS Check Out Date: 10 July 2009

Full synchronization trace can be downloaded from:
https://twiki.cern.ch/twiki/pub/TOTEM/CompOfflineCMSSWMigration177to31/syn_1st.

Merged file list:

```
IOMC/Elegent/src/Elegent.cc

TotemDataReadout/Testbeam/interface/RawDataSource.h
TotemDataReadout/Testbeam/interface/RPDataDigiProducer.h
TotemDataReadout/Testbeam/plugins/BuildFile
TotemDataReadout/Testbeam/plugins/RawDataSource.cc
TotemDataReadout/Testbeam/plugins/RPDataDigiProducer.cc

TotemAlignment/RPRates/data/MCScoringPlaneProfiles.cfi
TotemAlignment/RPRates/data/ScoringPlaneProfiles.cfi
TotemAlignment/RPRates/plugins/BuildFile
TotemAlignment/RPRates/plugins/MCScoringPlaneProfiles.cc
```

```
TotemAlignment/RPRates/plugins/ScoringPlaneProfiles.cc
TotemAlignment/RPRates/interface/ScoringPlaneDistributions.h (a new file)
TotemAlignment/RPRates/src/ScoringPlaneDistributions.cc (a new file)

L1TriggerTotem/CoincidenceChip/interface/RPCoincidenceAnalyzer.h
L1TriggerTotem/CoincidenceChip/interface/RPCoincidenceProducer.h
L1TriggerTotem/CoincidenceChip/plugins/RPCoincidenceAnalyzer.cc
L1TriggerTotem/CoincidenceChip/plugins/RPCoincidenceProducer.cc

RecoTotemT1T2/PrimaryVertexProducer (a new module)

TotemBackground/PP/data/PPBckgAnalysis.cfi (a new file)
TotemBackground/PP/data/PPBckgSource.cfi
TotemBackground/PP/plugins/BuildFile
TotemBackground/PP/plugins/PPBckgAnalysis.cc (a new file)
```

Jan Kaspar made further changes and checked in to TOTEM repository's CMSSW_3_1_x branch. The changes Jan made is as follows:

Modifications by Jan Kaspar (14 July 2009)

TotemDataReadout package renamed and restructured

The package has been renamed to TotemRawData (it make more sense and is closer to CMS terminology). Moreover, the contents have been reshuffled into 3 submodules: DataFormats, Readers (alogorithms to read raw data files) and RawToDigi (algorithms to covert raw data to digi).

TotemBackground/BeamGas

has been freed from THnSparse class definition (it is now a part of the official ROOT distribution)

Shared packages under Geometry directory

There 3 packages originated by CMS and modified by TOTEM: CMSCommonData, ForwardCommonData and ForwardSimData. At the version 177, I compared files in these packages between CMS framework and TOTEM SW. Modified/new files have been then merged/copied to the 311 framework version, resulting in TOTEM 311 files.

In CMSCommonData, those were:

- recoConfiguration.xml (copied)
- rotations.xml (copied as rotations2.xml)
- totem_rotations (copied)
- normal/cmsextent (merged)

In ForwardCommonData (all copied unless stated differently)

- castor1.xml
- castor2.xml
- castor_fibres.xml
- cuts-beampipe.xml
- ionpump.xml (copied as ionpump2.xml)
- totem2cuts.xml

In ForwardSimData (all copied)

- TotemProdCuts.xml
- a.xml

- totemProdCuts.xml
- totemsensGem.xml

g4SimHits_cfi.py

3_1_1 framework version merged with 1_7_7 TOTEM version.

-- JanKaspar - 14 July 2009

Synchronization 2 (31 July 2009)

`Visualization` module authored by Maciej Besta is added to Totem software @ CMSSW_3_1_1. This module is created for using Frog.

-- ZhengkuiZhang - 07 August 2009

This topic: TOTEM > CompOfflineCMSSWMigration177to31
Topic revision: r28 - 2009-10-20 - MirkoBerretti