Table of Contents

How to implement a new custom geometry in FATRAS

In ATLAS, testing alternative layouts through the full simulation and reconstruction chain is a work-intensive program, which is probably not worth following to full detail for all the test layout configurations.

This guide presents how to implement fast definitions of realistic detector layouts in the ATLAS TrackingGeometry library and run the fast simulation program (FATRAS), followed by fast digitisation and truth tracking.

The XML configuration files used to build the TrackingGeometry in FATRAS can then be re-used to build FastGeoModels and produce full simulation samples.

Presentations about the package:

- INDICO - January 28th 2015 (J. Leveque)
- INDICO - January 21st 2015 (R. Lafaye)
- INDICO - February 3rd 2015 (N. Calace)

Setup

Use latest stable release (replace rel_X with the correct release) from:
http://atlas-nightlies-browser.cern.ch/~platinum/nightlies/info?tp=g&nightly=20.20.X-VAL

```
mkdir NewDIR
cd NewDIR
cp -rf  ~leveque/public/FATRAS/XMLGEO/* .

setupATLAS
asetup 20.20.2.1,gcc49,AtlasProduction,here

cmt co -r InDetReadoutGeometry-02-00-22  InnerDetector/InDetDetDescr/InDetReadoutGeometry

cmt co -r InDetTrackingGeometryXML-00-00-17   InnerDetector/InDetDetDescr/InDetTrackingGeometryXM
cmt co -r TrkGeometryAlpine-00-00-08  Tracking/TrkDetDescr/TrkGeometryAlpine
cmt co -r ISF_FatrasDetDescrAlpine-00-00-28   Simulation/ISF/ISF_Fatras/ISF_FatrasDetDescrAlpine

source make.sh
```

Important notes:

- Make sure you are checking out the latest tags of the packages. Check the Available Tags section for more information.

Edit the layout XML configuration files

Change your layout configuration by editing the XML files located in:

```
InnerDetector/InDetDetDescr/InDetTrackingGeometryXML/share/
```

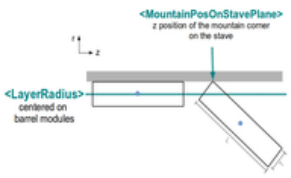
For each layout, 3 files can be edited:

- Barrel part: `Layout_PixelLayer.xml` and `Layout_PixelStave.xml`
- Endcap part: `Layout_PixelEndcap.xml`

- The same files exist for extended Very Forward layouts (pixel only)
- Modules and materials are described in separate XML files
- Check Documentation.xml for a description of the input variables

Important notes:

- Angle unit is rad
- Units for all dimensions is mm
- The numbers in barrel/stave XML files should given for a half-stave. Non-integer numbers are allowed
- The numbers in endcap XML files should be given for the z-positive side only
- The RingModuleZOffset parameter allows to avoid modules overlaps on discs
- The mountain modules z positions MountainPosOnStavePlane should be given for the z-positive side only (half-stave)
- To understand the precise definitions of the layer radius and mountain positions, click on the drawing below
- Material description: ongoing
- If you have questions or comments regarding the code or the instructions, - Request



Layer radius and mountain module position definitions

Build your layout and validate it

VP1 vizualisation

After compiling, simply do:

```
export LD_PRELOAD=/usr/lib64/libGL.so
cd WorkArea/run/
athena TrackingGeometryTest_jobOptions.py
```

The VP1 display should open automatically:

- In the "TrkGeo" tab, click on the "Retrieve" button.
- Click on the detector parts to display the corresponding tracking surfaces.

You can also run a small macro created during the geometry building to vizualize the layout in the R-z plane:

```
root InDetLayoutRZ.C
```

The total silicon surface is also printed on the screen.

Change the displayed layout by setting the appropriate GeometryAlpineLayoutFlags in your jobOptions:

```
GeometryAlpineLayoutFlags.PixelLayout.set_Value( "LoI" )  ## ATLAS, LoI, ExtBrl32Ref, ExtBrl4Ref,
GeometryAlpineLayoutFlags.PixelEndcapLayout.set_Value( "ECRing4Ref" )  ## ATLAS, LoI, ECRing32Ref
GeometryAlpineLayoutFlags.doSCT.set_Value( False )        ## True or False
GeometryAlpineLayoutFlags.SCTLayout.set_Value( "LoI" )    ## ATLAS, LoI, FourLayersNoStub
```

The full list of available layouts can be found by looking into the `share` directory of the `InDetTrackingGeometryXML` package:
InnerDetector/InDetDetDescr/InDetTrackingGeometryXML/trunk/share

Hits maps and hermeticity

To produce hits in your custom detector, run the following command:

```
cd WorkArea/run/
athena ExtrapolationEngineTest_jobOptions.py
```

The hits are stored in the ExtrapolationEngineTest.root output file. A tiny macro can also be used:

```
root macros/Coverage.C
```

Change the displayed layout by setting the appropriate GeometryAlpineLayoutFlags in your jobOptions:

```
GeometryAlpineLayoutFlags.PixelLayout.set_Value( "LoI" )  ## ATLAS, LoI, ExtBrl32Ref, ExtBrl4Ref,
GeometryAlpineLayoutFlags.PixelEndcapLayout.set_Value( "ECRing4Ref" )  ## ATLAS, LoI, ECRing32Ref
GeometryAlpineLayoutFlags.doSCT.set_Value( False )        ## True or False
GeometryAlpineLayoutFlags.SCTLayout.set_Value( "LoI" )    ## ATLAS, LoI, FourLayersNoStub
```

Run the FATRAS digitization and truth tracking

Instructions can be found on HowToFatrasDigitisationTruthTracking

Full digitization and tracking : FastGeoModel XML building

Once the tracking geometry is validated, you can build a more detailed GeoModel, starting from the same XML templates, and use them to produce fully simulations samples with pile-up.

Instructions can be found here: FastGeoModelGeometryImplementationGuide

Available Tags

InDetTrackingGeometryXML

- InDetTrackingGeometryXML-00-00-17
  - New pixel module naming convention
- InDetTrackingGeometryXML-00-00-12
  - fixed surfaces to layer association for alpine staves
  - fixed layout vizualization macro
  - additional tilt angles parameters added inside stave XML files, on top of the global tilt angle defined in layer XML.
- InDetTrackingGeometryXML-00-00-11
  - N. Calace : Adding surfaces to layer association
- InDetTrackingGeometryXML-00-00-10
  - added LayerPosition for ring layers in Endcap XML description files. LayerNumber starts at 0 and increments by 1, but LayerPosition doesn't have to and can be used to create empty layers (see ECRing32Ref and ECRing4Ref for examples)
  - added rows/columns info for chip readout

- InDetTrackingGeometryXML-00-00-09
  - new XML for module description to match GeoModel description
  - moved pitch info from barrel/endcaps to front-ends

- InDetTrackingGeometryXML-00-00-08
  - new SiDetElementProvider allowing to delete duplicated code in ISF_FATRAS.
  - the tracking geometry is now fully built from the InDetTrackingGeometryXML code
  - corrected SCT Endcap XML file

- InDetTrackingGeometryXML-00-00-07
  - new config files for reference layouts: `ECRing32Ref`, `ECRing4Ref`, `ExtBrl32Ref`, `ExtBrl4Ref`, `IExtBrl4Ref`

- InDetTrackingGeometryXML-00-00-06
  - test config files for SLIM

- InDetTrackingGeometryXML-00-00-05
  - new PlanarDetElement interface

- InDetTrackingGeometryXML-00-00-04
  - Preliminary 5 Layers layout

- InDetTrackingGeometryXML-00-00-03
  - Corrected ieta value in identifier for non-tilted layouts in barrel

- InDetTrackingGeometryXML-00-00-02
  - Additional parameter phi offset for endcap modules
  - Corrected configuration file for ATLAS SCT Endcap

- InDetTrackingGeometryXML-00-00-01
  - First version to be used both with FATRAS and Fast Geomodel

TrkGeometryAlpine

- TrkGeometryAlpine-00-00-08
  - from Andi S. : adding CL, changing to const BoundaryCheck?& version

- TrkGeometryAlpine-00-00-07
  - for MIG5. Correct compiler warnings

- TrkGeometryAlpine-00-00-06
  - FullSim compatible version

- TrkGeometryAlpine-00-00-05
  - Fixed phi overlap
  - All sensitive hits are found

ISF_FatrasDetDescrAlpine

- ISF_FatrasDetDescrAlpine-00-00-28
  - update CustomStagedBuilder to run the fully inclined layout

- ISF_FatrasDetDescrAlpine-00-00-24
  - new XML for module description to match GeoModel description
  - moved pitch info from barrel/endcaps to front-ends

- new python setup scripts from Noemi required for truth tracking in FATRAS

- ISF_FatrasDetDescrAlpine-00-00-21
  - new PlanarDetElementProvider allowing to delete duplicated code.
  - the Tracking geometry is now fully built from the InDetTrackingGeometryXML code

- ISF_FatrasDetDescrAlpine-00-00-20
  - python modification to split Layer Providers in a more flexible way

- ISF_FatrasDetDescrAlpine-00-00-19
  - added error messages for bad detector configurations

- ISF_FatrasDetDescrAlpine-00-00-18
  - corrected phi neighbors bug

- ISF_FatrasDetDescrAlpine-00-00-17
  - new PlanarDetElement interface

- ISF_FatrasDetDescrAlpine-00-00-16
  - Python update for 5 Layers layout

- ISF_FatrasDetDescrAlpine-00-00-15
  - Corrected ieta value in identifier for non-tilted layouts in barrel

- ISF_FatrasDetDescrAlpine-00-00-14
  - Added a phi offset parameter for disc modules

- ISF_FatrasDetDescrAlpine-00-00-13
  - Fill IdHashDetElementCollection

- ISF_FatrasDetDescrAlpine-00-00-12
  - XML part moved to XML in InDetTrackingGeometryXML

- ISF_FatrasDetDescrAlpine-00-00-11
  - Automated production of IdDictionnary fully operationnal. Tag included in mig5.

- ISF_FatrasDetDescrAlpine-00-00-10
  - Automated production of IdDictionnary
  - Warning: you run to run the code first to generate the correct dictionary, then to build the geometry. To be fixed soon.

- ISF_FatrasDetDescrAlpine-00-00-09
  - Fixed stereo module finding and disc thickness computation
  - Still missing automated production of IdDictionnary

- ISF_FatrasDetDescrAlpine-00-00-08
  - SLIM XML files added (Pixel &SCT)
  - Still missing automated production of IdDictionnary

- ISF_FatrasDetDescrAlpine-00-00-07
  - SLIM XML files processing added
  - Still missing automated production of IdDictionnary

- ISF_FatrasDetDescrAlpine-00-00-06

- Switched to StagedBuilder
- Rings and extended barrel layers allowed
- LoI XML files and ATLAS files filled with official geometry DB inputs
- Still missing automated production of IdDictionnary

- ISF_FatrasDetDescrAlpine-00-00-04
  - SCT included
  - No pixel ring yet (work ongoing)
  - Layouts with no endcap are forbidden: dummy discs required to avoid a crash
  - LoI XML files contain arbitrary numbers.To be fixed. Volunteers welcome.

- ISF_FatrasDetDescrAlpine-00-00-03
  - Renaming of XML parameters
  - No SCT yet (work ongoing)
  - No pixel ring yet (work ongoing)
  - Layouts with no endcap are forbidden: dummy discs required to avoid a crash
  - LoI XML files contain arbitrary numbers.To be fixed. Volunteers welcome.

- ISF_FatrasDetDescrAlpine-00-00-02
  - First released version
  - No SCT yet (work ongoing)
  - No pixel ring yet (work ongoing)
  - Layouts with no endcap are forbidden: dummy discs required to avoid a crash
  - Conical and LoI XML files contain arbitrary numbers.To be fixed. Volunteers welcome.

To-Do List

ISF_FatrasDetDescrAlpine & TrkGeometryAlpine developments

- Add Ring Layout done
- Add automated generation of IdDict done
- Fix "dummy disc" issue done
- Update TrkDetDescrSvc done
- Take non-integer number of chips as input to compute module size (needed for petal modules) done
- Add NChipXMin and NChipXmax to create petal modules done
- Add BarrelModuleTiltAngle done
- Add z0 and d0 smearing in ExtrapolationEngineTest done
- Module naming convention to be changed to match Full GeoModel done
- Produce LoI XML files for FATRAS validation against fullsim samples done
- Add `StereoAngleInner` and `StereoAngleOuter` for SCT stave modules done
- Implement compound materials ongoing
- Implement navigation for neutrals in TrkGeometryAlpine not started

Trk::ExtrapolationEngine validation

The validation has to check the correct beheaviour of the samples that perform the interaction with matter in the iFatras::McMaterialEffectsEngine:

- iFatras::EnergyLossSamplerBetheBloch using muons (from 1 GeV to 100 GeV) done
- iFatras::MultipleScatteringSampler(Highland/GeneralMixture/GaussianMixture) using muons (from 1GeV to 100 GeV) ongoing (assigned to Artem)
- iFatras::EnergyLossSamplerBetheHeitler using electrons ongoing
- iFatras::PhotonConversionSampler ongoing
- iFatras::iFatras::G4HadIntProcessor not assigned

First of all, visualise it!

Use latest stable release (replace rel_X with the correct release) from:
http://atlas-nightlies-browser.cern.ch/~platinum/nightlies/info?tp=g&nightly=20.20.X-VAL

Let's start setting our working space.

```
asetup 20.20.X-VAL,rel_X,slc6,gcc49,opt,here,runtime
```

Be sure you are using the following or more recent tags for the packages shown below.

```
ISF_FatrasDetDescrModel-00-00-10
ISF_FatrasDetDescrTools-00-00-09
AtlasGeoModel-00-03-34
TrkDetDescrSvc-01-02-08
InDetServMatGeoModel-00-04-27
PixelGeoModel-00-09-43
SCT_SLHC_GeoModel-00-00-25
```

The detector we want to create is fully described in the `ISF_FatrasDetDescrTools package`. Check this package out and have a look at the `python/CustomInDetTrackingGeometryBuilder.py`.

Here you have the definition of the builders used to create the custom ID with all the information regarding the geometry:

- BARREL:
  - BarrelLayers : (size_t) Number of barrel cylinders
  - LayerSCTlike : (bool) Enable the back surface creation
  - LayersZsectors : (std::vector<size_t>) Segmentation in z for all cylinders
  - LayerPhiSectors : (std::vector<size_t>) Segmentation in phi for all cylinders
  - LayerTilt : (std::vector<double>) Tilt angle of the detector elements with respect to the tangent to the support cylinder surface in the plane perpendicular to the cylinder axis and the detector elements.
  - LayerMinPhi and LayerMaxPhi : (std::vector<double> and std::vector<double>) Cylinder's phi range
  - LayerMinZ and LayerMaxZ : (std::vector<double> and std::vector<double>) Cylinder's z range
  - LayerRadius : (std::vector<double>) Cylinders' radii
  - LayerSeparation : (std::vector<double>) Vertical separation between two consecutive rings in the same cylinder
  - LayerThickness : (std::vector<double>) Detector element thickness
  - LayerLengthY : (std::vector<double>) Detector element Y dimension
  - LayerLengthXmin : (std::vector<double>) Detector element Xmin dimension
  - LayerLengthXmax : (std::vector<double>) Detector element Xmax dimension (to be used for trapezoidal elements)
  - LayerPitchX : (std::vector<double>) X dimension of the sensors on the same detector element
  - LayerPitchY : (std::vector<double>) Y dimension of the sensors on the same detector element
  - LayerRotation : (std::vector<double>) Detector element rotation with respect to the radius direction
  - LayerStereo : (std::vector<double>) Rotation angle in the detector element plane. This is used if you have a SCT-like structure
  - LayerStereoSeparation : (std::vector<double>) Separation in R between a detector element and its back surface. This is used if you have a SCT-like structure
  - AdditionalLayerRadius : (std::vector<double>) Radii of additional passive cylinders
- ENDCAPS:
  - EndcapDiscs : (size_t) Number of disks in the endcap

- DiscSCTlike : (bool) Enable the back surface creation
- DiscsZpos : (std::vector<double>) Position of the disks in Z
- DiscPhiSectors : (std::vector<std::vector<double>>) Segmentation in phi for all rings
- DiscMinPhi and DiscMaxPhi : (std::vector<std::vector<double>> and std::vector<std::vector<double>>) Phi range for all rings
- DiscRingMinR and DiscRingMaxR : (std::vector<std::vector<double>> and std::vector<std::vector<double>>) Minimum and maximum radii for all rings
- DiscSeparation : (std::vector<std::vector<double>>) Z separation between two consecutive detector elements on the same ring
- RingDisplacement : (std::vector<std::vector<double>>) Z separation between two consecutive rings on the same disk
- DiscThickness : (std::vector<double>) Detector element thickness
- DiscLengthY : (std::vector<std::vector<double>>) Detector element Y dimension
- DiscLengthXmin : (std::vector<std::vector<double>>) Detector element Xmin dimension
- DiscLengthXmax : (std::vector<std::vector<double>>) Detector element Xmax dimension (to be used for trapezoidal elements)
- DiscPitchX : (std::vector<std::vector<double>>) X dimension of the sensors on the same detector element
- DiscPitchY : (std::vector<std::vector<double>>) Y dimension of the sensors on the same detector element
- DiscStereo : (std::vector<std::vector<double>>) Rotation angle in the detector element plane. This is used if you have a SCT-like structure
- DiscStereoSeparation : (std::vector<std::vector<double>>) Separation in Z between a detector element and its back surface. This is used if you have a SCT-like structure
- AdditionalDiscZpos : (std::vector<double>) Z positions of additional passive disks

The material associated to each detector element is defined by the iFatras::InputLayerMaterialProvider class. It is also possible to assign custom material also to the layers using the properties:

- CustomMaterial : boolean to enable custom material assignment to the layers
- CustomMaterialThickness
- CustomMaterialX0
- CustomMaterialL0
- CustomMaterialA
- CustomMaterialZ
- CustomMaterialRho

N.B.: In the new framework, it is possible to access to the material information of the single detector element. For this reason, it is no needed to assign custom material to the layers since it is already given by the iFatras::InputLayerMaterialProvider to every iFatras::PlanarDetElement object.

The CustomTrackingGeometry is initialised preincluding the `share/CustomTrackingGeometryInit.py`. You can define which subdetectors you want to create easily changing the customPixel, customSCT, customTRT flags from False into True.

The description shown as default in the `python/CustomInDetTrackingGeometryBuilder.py` and the CustomTrackingGeometry initialisation (in `share/CustomTrackingGeometryInit.py`) allow to build pixel and SCT detectors as in ATLAS standard layout (at the time being there is no TRT). Let's visualise it!

A new GeoModel tag has been crated in order to create the empty ID geometry, that will be filled with the custom one. We need to define the new DetDescrVersion in the jobs we want to run.

The job used to visualise the geometry is `TrackingGeometryTest_jobOptions.py` in the `TrkDetDescrUnitTests` package.

```
pkgco.py TrkDetDescrUnitTests
```

Now we need to edit the job option:

- Switch the calorimeter and the muon system off using

```
DetFlags.Calo_setOff()
DetFlags.Muon_setOff()
```

- Switch the TRT off (at the time being there is no custom TRT) using

```
DetFlags.TRT_setOff()
```

- Change the DetDescrVersion into

```
DetDescrVersion ='ATLAS-PX-ITK-00-00-00'
```

Setup the work area and compile.

```
setupWorkArea.py
cd WorkArea/cmt/
cmt broadcast "cmt config ; cmt make binclean ; cmt make"
cd -
```

Run the command

```
athena Simulation/ISF/ISF_Fatras/ISF_FatrasDetDescrTools/share/CustomTrackingGeometryInit.py \
Tracking/TrkDetDescr/TrkDetDescrUnitTests/share/TrackingGeometryTest_jobOptions.py 2>&1 | tee out
```

Now you should have several *.C files in your area. We are interested in the `TrackingGeometrySurfaceDisplay.C`. Open it with ROOT to visualise the surfaces defined in the new geometry (back surfaces are not shown).

```
root -l TrackingGeometrySurfaceDisplay.C
```

Check navigation through the custom geometry

To check if the navigation through the geometry is performed in the correct way, we can run the `share/ExtrapolationEngineTest_jobOptions.py`. This test will collect all the information about the hit surfaces (sensitive, passive and boundary surfaces). Check out the `TrkExUnitTests` package and compile it.

```
pkgco.py TrkExUnitTests
cd Tracking/TrkExtrapolation/TrkExUnitTests/cmt
make
cd -
```

As previously said, we need to change the DetDescrVersion and turn the TRT off (at the time being there is no custom TRT) adding the following line to the jobOption:

```
DetFlags.TRT_setOff()
DetDescrVersion ='ATLAS-PX-ITK-00-00-00'
```

Moreover, since the tracking layer material map doesn't exist for this GeoModel tag, we need also to add

```
TrkDetFlags.MaterialSource          = 'None'
```

Run the command

```
athena Simulation/ISF/ISF_Fatras/ISF_FatrasDetDescrTools/share/CustomTrackingGeometryInit.py \
Tracking/TrkExtrapolation/TrkExUnitTests/share/ExtrapolationEngineTest_jobOptions.py 2>&1 | tee o
```

The output of this job is a root file containing the hit information. To prove the correct behaviour of the navigation you should be able to visualise the correct position of disks and cylinders, their segmentation and back surfaces (if you have SCT-like structures).

- ITK Layout Task Force
- Ongoing Developments in ISF_Fatras
- Inner Detector Geometry Description
- IBL Services
- ITK
- ATLAS Upgrade

-- Main.NoemiCalace - 2015-01-08

This topic: Sandbox > CustomGeometryImplementationGuide
Topic revision: r91 - 2016-08-11 - JessicaLeveque