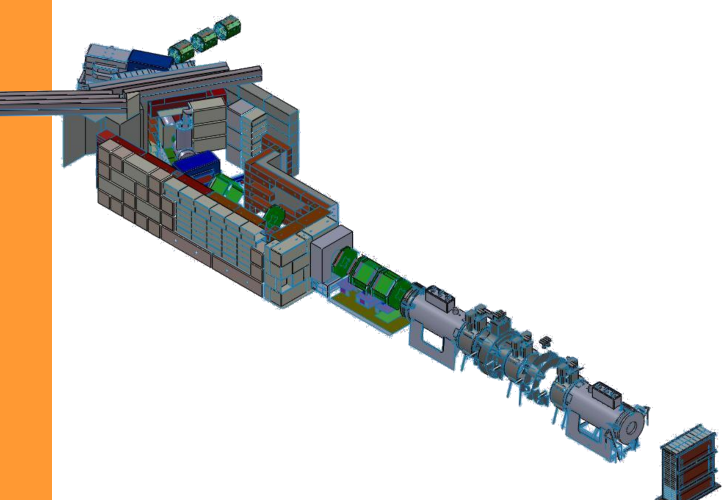
Active CAD Geometry Handling System

The system went live last collaboration meeting and is now in MAUS

$MAUS_ROOT_DIR/bin/utilities

- upload_geometry.py
- download_geometry.py
- get_geometry_ids.py



Matthew.Littlefield@brunel.ac.uk

Brunel UNIVERSITY

What the executables do?

1. upload_geometry.py

Upload Geometry

1. Set up the Configreader class and read the values provided by ConfigurationDefaults.py or by custom config files.
2. Instantiate an Uploader class object using the upload directory and geometry note taken from the configuration file.
3. Create a list of geometry files which are to be compressed into one zip file in preparation for uploading.
4. This zip file is then used as the argument for the upload_to_CDB method which takes the contents of the zip and then uploads this, as a single string to the CDB.

Optional If cleanup is specified in the configuration file then the file list and the original GDML files are the deleted leaving only the zip file.

Matthew.Littlefield@brunel.ac.uk

What the executables do?

2. download_geometry.py

Download Geometry

1. Set up the Configreader class and read the values provided by ConfigurationDefaults.py or by custom config files.
2. Instantiate a Downloader class object which downloads either the current, time specified or run number zipped geometry to a temporary cache location.
3. The zip file is then unzipped in this location.
4. The Formatter class is then called and this class formats the GDMLs. The formatting alters the schema location of these files and points them to the correct local locations of the Materials GDML file and schema. Formatting leaves the original GDMLs in the tmeporary cache and places the new formatted GDML files in the download directory specified in the configuration file.
5. GDMLtoMAUS is then called with the location of the new formatted files as its argument. This class converts the GDMLs to the MICE Module text files using the XSLT stylesheets previously described.

Optional If specified in the configuration file the temporary cache location is removed along with the zip file and unzipped files.

Matthew.Littlefield@brunel.ac.uk

What the executables do?

3. get_geometry_ids.py

Get Geometry IDs

1. Set up the Configreader class and read the values provided by ConfigurationDefaults.py or by custom config files. This file takes start and stop time arguments to specify a period to search the CDB.
2. A CDB class object is then instantiated with the server specified in the configuration file.
3. The get ids method from the CDB class is called and the python dictionary which is downloaded is formatted and either printed to screen or to file as specified in the configuration file.

All arguments are in the ConfigurationDefaults.py file.

User Documentation can be found in MAUS.

MICE Note on the entire system will soon follow.

Matthew.Littlefield@brunel.ac.uk

Current Geometries

The ConfigDB now has three geometries saved.

```
Geometry Number = 1
Geometry Note   = Version 0point1 of the officical geometry contains D2 Q456789 tof and ckov placeholders The legacy detectors will be placed within the placeholders
upon download
ValidFrom       = 2012-04-25 14:17:45.447614
Date Created    = 2012-04-25 14:17:45.527000

Geometry Number = 2
Geometry Note   = Version 0point1 of the officical geometry contains D2 Q456789 tof and ckov placeholders The legacy detectors will be placed within the placeholders
upon download
ValidFrom       = 2012-03-21 09:00:00
Date Created    = 2012-04-27 15:09:53.801000

Geometry Number = 3
Geometry Note   = Version 0point2 of the officical geometry contains beamline and legacy TOF CKOV and tracker single station
ValidFrom       = 2012-05-08 09:00:00
Date Created    = 2012-05-22 15:47:34.856000
```

More will follow.

Geometry 3 is an intermediate step for the single station test. Surveys are currently being carried out and the resulting geometry will replace Geometry Number 3.

Matthew.Littlefield@brunel.ac.uk

Teething Problems

As with all new systems there are and will be a few problems. So far there has been nothing major except a few bugs and problems placing legacy detectors. The main problems are;

1. The Angle of D2. The beam cannot (at the moment) have its direction rotated therefore the beam cannot be guide into D2s aperture. Code is being developed.

2. Fields. The field of the magnets have been placed but their directions are not correct. This is because we cannot see the direction of the field. This is easy to fix and will be completed soon.

Matthew.Littlefield@brunel.ac.uk

Future Features?

Modules will be named in a more coherent manner as current convention is a little vague.

Anything else users and developers want/need?

Matthew.Littlefield@brunel.ac.uk

Future Work

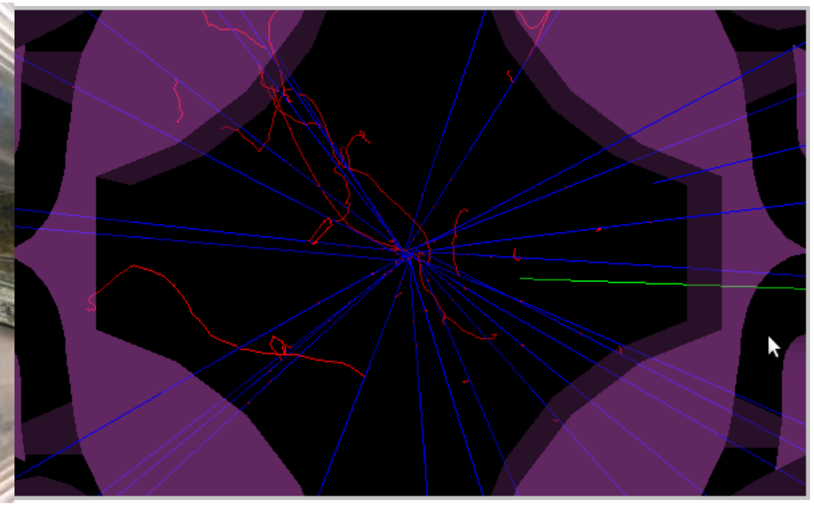
- Survey of single station geometry shall be uploaded and some minor management details ironed out.
- Management process shall be finalised. Fastrad is moving to RAL where Jason Tarrant and the engineers shall be responsible for this part of the system in conjunction with CAD.
- Collaborate with detector groups to improve detector geometries and update legacy models. If need be.
- Anything else people would like to see?

Please start using the system to help me improve it.

I can and am happy to help to translate CAD models people wish to simulate and any other geometrical issues people have.

Matthew.Littlefield@brunel.ac.uk

Any Questions?



Thank You for Listening

Matthew.Littlefield@brunel.ac.uk