

Code sprint 26.2 - 1.3.18

Status update on VecGeom items

Sandro Wenzel

28.2.2018

Reminder: List of topics

- **general cleanup campaign**
 - remove/deprecate original USOLIDS code
 - remove other deprecated / unused code
- **factory for unplaced solids**
- **VecGeom navigator in G4 simulation**

USOLIDS (v1) cleanup

- we (in particular G4) are no longer using the original USOLIDS code, nor its interface; VecGeom supersedes USOLIDS v1.
- so geometry working group decided to remove all couplings of VecGeom to original USOLIDS
 - will simplify code
 - we reduce the number of build options (we no longer require distinct builds for special USOLID compatible mode)
- Based on previous effort from G.Lima, this task was completed in merge request [546](#).

Unplaced volume factory

- “As a user, I would like to be able to **create the best possible specialized instance** of a volume given **some generic parameters/description.**”
 - get FullTube instance when asking for a Tube without inner radius
- In VecGeom, such a mechanism existed for **PlacedVolumes** but not yet for **UnplacedVolumes** after the recent class layout changes

Unplaced volume factory

- Worked out a **uniform template interface** in **GeoManager** which is achieving this

```
// a factory for unplaced shapes  
template <typename UnplacedShape_t, typename... ArgTypes>  
static UnplacedShape_t *MakeInstance(ArgTypes... Args);
```

- In principle now **the only interface** we should use

Unplaced volume factory

- A default implementation and specialized treatment for tube done:

```
auto ubox = GeoManager::MakeInstance<UnplacedBox>(1., 1., 2.);  
assert(dynamic_cast<UnplacedBox *>(ubox));
```

```
auto utube = GeoManager::MakeInstance<UnplacedTube>(0., 1., ...);  
assert(dynamic_cast<SUnplacedTube<TubeTypes::NonHollowTube> *>(utube));
```

- Should now be easy to transfer to other volumes which might be appropriate for specialization
- To see if G4 can profit from this

Unplaced volume factory

- Immediate positive implication in the ROOT to VecGeom conversion, instead of using GenericUnplacedTube we can now just use the factory (see example blow)

```
// THE TUBE CONVERSION OLD -- HAD TO SPECIFY A CONCRETE VECGEOM TYPE
if (shape->IsA() == TGeoTube::Class()) {
    TGeoTube const *const tube = static_cast<TGeoTube const *>(shape);
    unplaced_volume = new GenericUnplacedTube(tube->GetRmin(), tube->GetRmax(), tube->GetDz(), 0., kTwoPi);
}

// // THE TUBE CONVERSION NEW -- FACTORY WILL GIVE BEST POSSIBLE TYPE
if (shape->IsA() == TGeoTube::Class()) {
    TGeoTube const *const tube = static_cast<TGeoTube const *>(shape);
    unplaced_volume = GeoManager::MakeInstance<UnplacedTube>(tube->GetRmin(), tube->GetRmax(), tube->GetDz(), 0., kTwoPi);
}
```

VecGeom navigation in G4

- started implementation based on the existing package G4Root (A. Gheata) which provides a Geant4 navigator using ROOT geometry
- Setup of repository with files / cmake / example app: Have a first G4 application actually instantiating a real VecGeom geometry hierarchy ... but not yet coupled to the navigator
- Will have to fill a couple of holes here and there to have a first complete example, but it seems within relatively easy reach