Updates on the misalignment framework

Raffaele Grosso

ALICE Offline Group

October 5, 2006

Outline

Problem to solve

The problem:

allow the alignment objects to keep their validity although the volume paths to which they refer to would change (e.g. addition of an intermediate parent volume).

Our solution: symbolic volume names

save in the alignment object a “symbolic name” for the volume, which is univocally, but dynamically, associated to the volume path to allow the alignment constants to keep their validity in the case of modified volume path.

Symbolic volume names

$$\begin{array}{ccc} \text{alignment constants} & & \text{TGeo} \\ \text{symbolic volume name} & \Leftrightarrow & \text{volume path} \end{array}$$

Alignment constants can then be valid

- till the end of the experiment,
- for every ALICE geometry, with no need to reference a specific geometry version.

Plus:

We have the chance to give meaningfull and “human readable” names to alignable volumes.

Underlying functionality in TGeo

TGeo offers the `TGeoPNEntry` class, a `TNamed` storing as name the symbolic volume name and as title the corresponding volume path. When the constructor is called the validity of the path is checked (in case of invalid path the `TGeoPNEntry` object is not created). TGeoManager will than store the list of the `TGeoPNEntry`s as a `THashList` which is saved together with the geometry.
The list can be queried by means of the following `TGeoManager` methods:

- `TGeoPNEntry *GetAlignableEntry(const char *name) const;`
  retrieve object by symbolic name
- `TGeoPNEntry *GetAlignableEntry(Int_t index) const;`
  retrieve object by index (useful in iterations)
- `Int_t GetNAlignable() const;`
  get number of objects in the list

Underlying functionality in TGeo

The following methods of **TGeoManager** create the **TGeoPhysicalNode** object, either from the PNEntry or from the symbolic name, and connect it to the **TGeoPNEntry**:

1. `TGeoPhysicalNode *MakeAlignablePN(const char *name);`
2. `TGeoPhysicalNode *MakeAlignablePN(TGeoPNEntry *entry);`

Implementation in the framework

The `AliAlignObj`ects don't refer any more to the node by knowing the path, but by knowing the symbolic name associated to that path.
The calls to the AliAlignObj constructors will look exactly the same as now, with the only difference that we pass the simbolic name instead of the real path.
Previous calls of `MakePhysicalNode` (`AliAlignObj::ApplyToGeometry()` and `SetLocalPars()`) have been substituted with calls of `gGeoManager->MakeAlignablePN`, after making sure that the object exists with `gGeoManager->GetAlignableEntry()`.

Implementation in the framework

If the `TGeoPNEntry` is not found the framework treats the symbolic name as a volume path, issuing a warning. This allows to smoothly go from volume paths to symbolic volume names and, in the future, to allow the user to apply displacements to volumes which are not a priori alignable.

Rules for valid symbolic volume names

Rules for symbolic volume names:

1. the symbolic volume name has to contain a leading substring `DET_` identifying the detector it belongs to
2. using `/` as separator, the symbolic volume name has to contain, hierarchically, the parent volumes which are also alignable (e.g. `DET/sector/stack/chamber`)

This is enough to guarantee one by one mapping between symbolic names and paths (if names are unique inside each DET) and will allow to use also symbolic volume names consistently when misaligning nested volumes.

To be done by detectors

It is then detectors' responsability:

1. to choose once for all the symbolic volume names
2. to provide for each of them a correspondent valid path, keeping it synchronized with possible changes in the geometry.

Issue number 1 is done overloading `AliModule::AddAlignableVolumes` in the detectors' class (`AliDETvX`). In this method each alignable volume has to be "declared" by means of calls like:

```
TGeoPNEntry *SetAlignableEntry(const char *symbolic_name,
            const char *path)
```

e.g.:

```
gGeoManager->SetAlignableEntry("TPC/EndcapC/Sector11/InnerChamber"
"ALIC_1/TPC_M_1/TPC_Drift_1/TPC_ENDCAP_1/TPC_SECT_11/TPC_IROC_1")
```

Ready detectors

According to this morning's repository, the following detectors have already implemented the `AddAlignableVolumes` method: MUON,PHOS,TPC,TRD.
The task is not heavy, it's mainly choosing good names.

Note

The implementation of symbolic volume names alone will not solve the problem of huge output of error messages during simulation due to presence of alignment objects for inexistent volumes (i.e. TRD sectors switched off).

To solve this problem, the methods which switch off some parts of the geometry (like `AliTRDgeometry::SetSMstatus` or `AliTOF::SetTOFSectors` in TRD) should somehow inform AliSimulation of which `AliAlignObj`ects to discard.

Outline

Rearrangements in the framework

- Minor rearrangements in the code have been done to make it more “symmetric” and avoid some duplicated implementation
- `GetVolPath()` is now `SymName()` and static `GetVolPath(volid)` is now `GetSymName(volid)`
- `AliAlignObj` constructors have been simplified and those taking both layerId and moduleId have been removed since they where error-prone (several detectors misused them).

We also put (are putting) in CVS `DET/MakeDETResMisAlignment.C` `DET/MakeDETFullMisAlignment.C` which are steered by `macros/MakeAllDETsResMisAlignment.C` and `macros/MakeAllDETsFullMisAlignment.C` to produce residual and full misalignment objects for all detectors).
Please check and maintain these two macros per detector.
We are updating to these changes the alignment page (http://aliceinfo.cern.ch/Offline/Activities/Alignment.html) and the internal note we are about to submit.

Outline

Survey issues

It seems that the time is getting really short for detectors to be aware of their survey requirements (C.Lasseur's talk at LHC alignment workshop).
Each detector has to make sure to be aware of and to agree with:

- which volumes of your detector will be surveyed;
- where are the fiducial marks;
- the names given to them;
- how often survey data will be produced (at mounting?);
- survey fulfills needs, if any, of your monitoring system (ITS, MUON).

Otherwise get in contact with the surveyors: Christian Lasseur, Henning Kvaerno, Antje Behrens.

Outline

Predicted Space Frame deformations

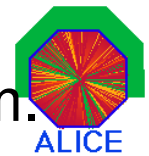
Expected space frame deformations evaluated by:

- a load test (water filled tanks representing TRD and TOF supermodules)
- finite element simulation (ANSYS)

The two are compatible although only a rough comparison is possible
Order of expected deformations:

1. $\sim$ 4 mm (tolerances)
2. $\leq$ 6 mm (deformations)
3. $\leq$ 20 mm (deflections of the rails)

3 can be almost completely shimmed $\Rightarrow$ overall expected deformations of order 7 mm
The space frame deformations will be monitored by the Spaceframe Monitoring System which will allow a determination of the absolute position of the 18 space frame corners with an accuracy of $\sim$0.5mm.



(Space frame issues were presented a bit more in-depth at the weekly offline meeting July the 6^{th})

Related changes in the framework

- Redefinition of the volumes ``B077'' and ``B076'' as TGeoAssemblies (previously defined as polygone volumes);
- Production of a set of alignment objects simulating the predicted deformations;
- Inclusion of non-sensitive detectors in the list of dets looped by `AliSimulation` when loading alignment objects and of corresponding entries in the CDB.

TRD and TOF volumes need no additional displacement being put in the conteainer volumes "`BSEGMO`".

In real life alignment objects will have to be produced by:

- survey and spaceframe monitoring system
- avareging the displacements of the TRD and TOF modules if this improves the precision (information from tracking).

Simulation of Space Frame deformations for PDC

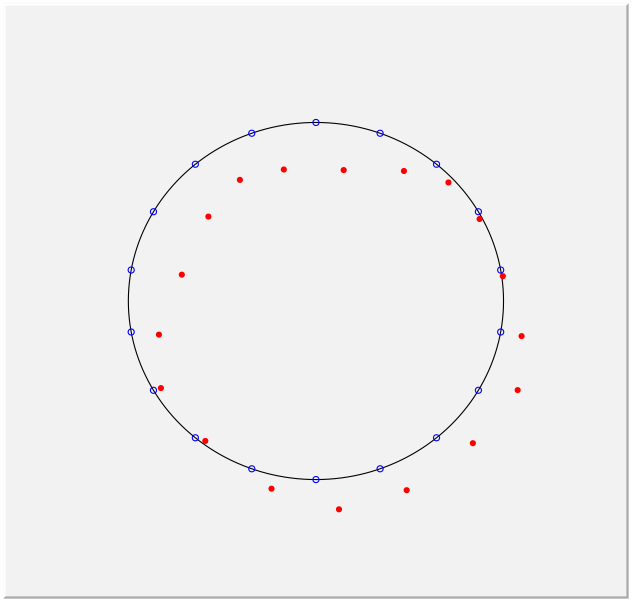


Figure: Shifts of the 18 sector centers (magnified by factor 100)

Objects for PDC obtained moving the center of the sectors; z-shifts neglected. The application of the corresponding alignment objects introduces few small overlaps (order 2mm) not concerning sensitive volumes (to do: final check to verify it's not dangerous).

Outline

Conclusions

If you did not already, we ask you to:

- provide symbolic volume names
- check the two macros per DET on CVS to produce alignment objects
- check your awereness and status on survey issues