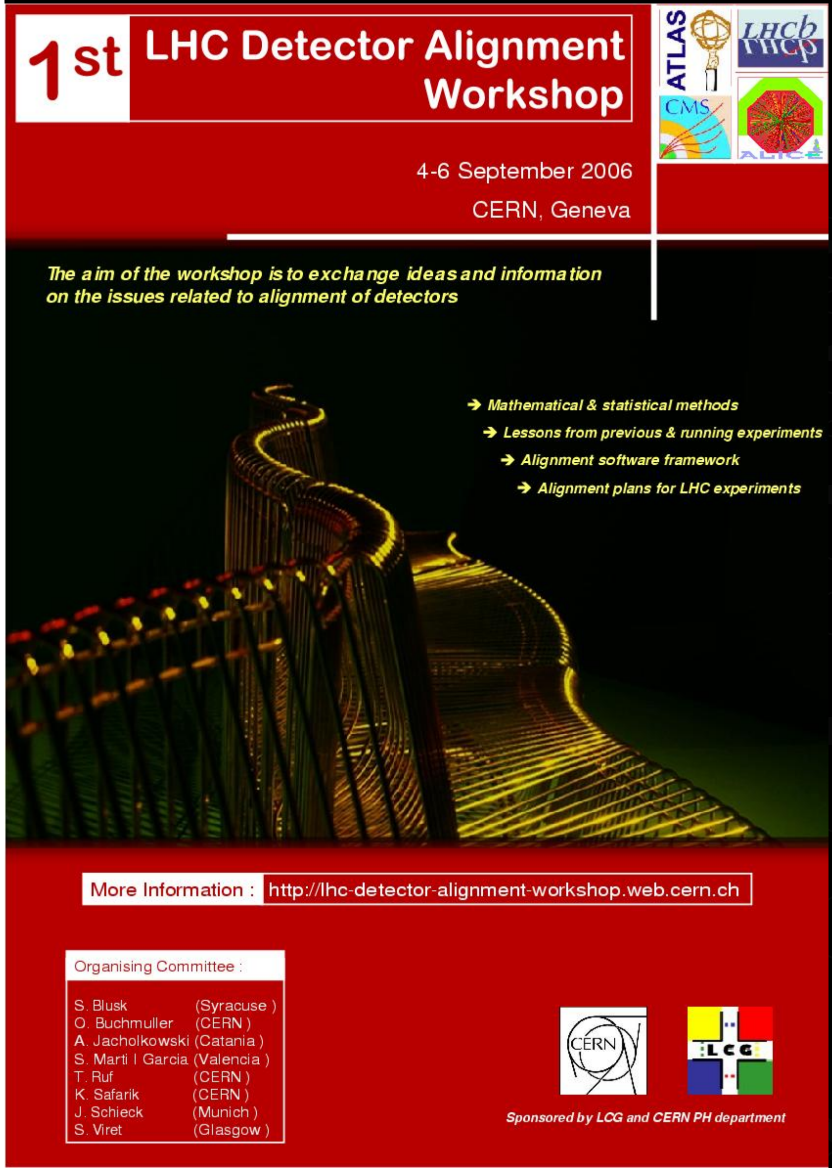


Detector Description (Overview)

C.Cheshkov

Outline

- Terminology
- Overview on:
  - Detector geometry implementation
  - Geometry support for misalignments or implementation of misalignments
  - Definition of alignments constants
  - Alignment related Condition DB issues
  - Application of misalignments for both simulation and reconstruction
- The above items for the 4 LHC experiments
- Summary Table (instead of conclusions)

Many Thanks!

- Juan Palacios, Andrei Gheata, Mihaela Gheata, Frédéric Ronga, Grant Gorfine, Raffaele Grosso, Marco Clemencic et al.

Terminology

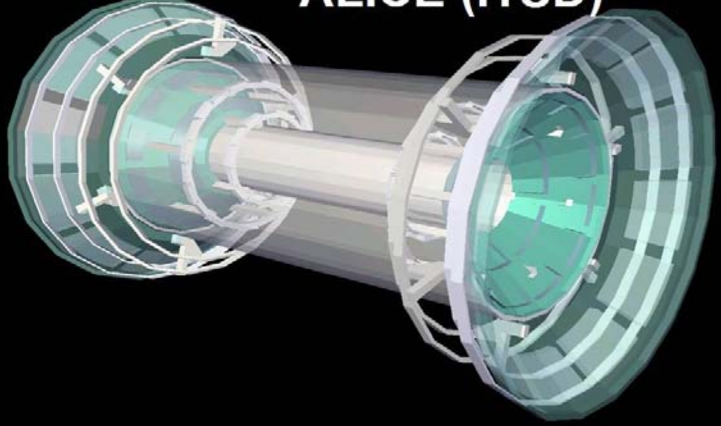
- Logical Volume or Node:
  - Describe volume a' la GEANT (one logical volume represents many real detector elements)
  - Used to define the geometry hierarchy
  - Can not be misaligned
- Physical Volume or Node:
  - Coupled to corresponding Logical volume
  - Represent single (unique) detector element
  - Used to handle the misalignment (or any other detector-element specific) information
- Alignment constants – actual objects stored in Condition DB and used to misalign the detector geometry
- Default transform – detector volume position and orientation in case of ideal (no misalignments) geometry
- Delta transform – correction to Default transform in case of misaligned geometry
- Geometry overlaps (caused by misalignment) – overlaps between Physical Volumes at simulation level

ALICE

Geometry Implementation

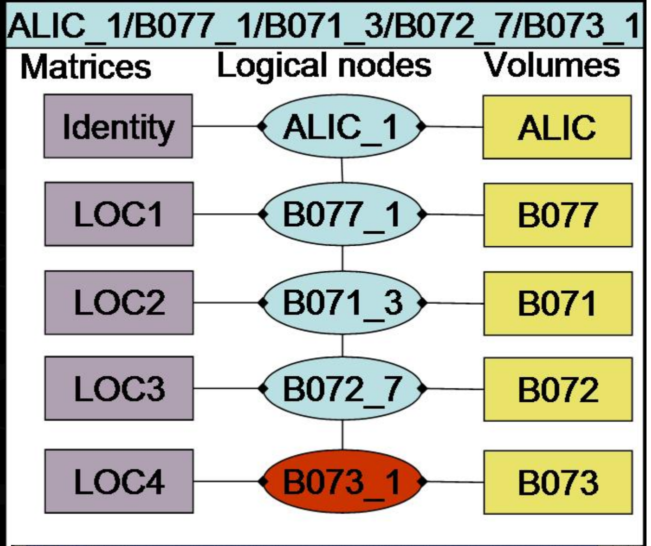
- ROOT Geometrical Modeler (TGeo):
  - framework for building, browsing, tracking and visualizing a detector geometry
  - Hierarchical model (G3, G4 alike) but independent from transport MC
  - same geometry for tracking, reconstruction or visualization
  - advantage of using ROOT features related to bookkeeping, I/O, histograming, browsing and GUI's

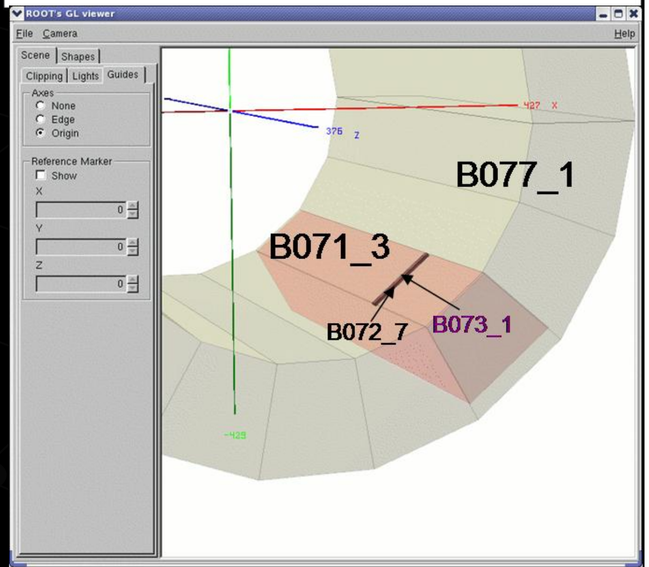
ALICE (ITSD)



Geometry Support for Misalignment

- TGeo Physical Node (PN):
  - Describes unique geometry object (detector element)
  - Fully identified by path with names of logical nodes (positioned in their container with a LOCAL matrix)
  - Points to last logical node in the path
  - Global matrix is product of all local matrices in branch:
    GLOBAL = LOC1 * LOC2 * … * LOCn
  - Can be created for any logical node at any level in geometry hierarchy
  - ~ 2.5x10^6 in ALICE
    => they are created ON DEMAND
- PN misaligned by changing LOCAL matrix for last logical node in the path
- The default matrix is backup
- Possibility to address all PNs by unique symbolic names (created during geometry initialization)
- Misalignment is automatically active for transport MC's or any geometry access





Alignment Constants

- Simple ROOT objects which contain:
  - Delta transform (rotation + translation) in GLOBAL frame
  - Symbolic links to TGeo Physical Volume path (eg "TPC_Sector5_InnerChamber")

    $\Rightarrow$ independent of geometry version
  - For sensitive volumes - unique (ALICE-wide) identifier:

    $\Rightarrow$ used for fast navigation in alignment procedures
- Applied on several levels in geometry hierarchy via TGeo interface

Conditions Database

- Offline condition files are ROOT files
- These are available to applications running on the GRID via
  - Registration into the AliEn file catalogue (logical)
  - Storage in a GRID storage element (physical)
- Conditions data is uniquely identified by 3 parameters
  - Logical path: “TPC/Align/Data“, “TRD/Align/Data“...
  - Run range validity: [0,100], [1,1] ...
  - Version (local and Grid)
- Accessed only once per simulation/reconstruction job

- Alignment constants are stored:
  - As one array per detector
  - For different detectors as separate files in corresponding folders

- Ideal geometry stored as a single object in same DB

Misalignment at Simulation and Reconstruction level

- Misalignments are read from CDB and applied to TGeo in the same way for both simulation and reconstruction:
  - At initialization stage
  - All consecutive access is done via TGeo interface
  - Ordered by geometry hierarchy (high -> low level)
- In simulation:
  - Transport MC uses directly TGeo navigation (already with misalignments)
  - During the digitization of energy depositions, position and orientation of volumes are accessed via calls to TGeo only

Geometry Overlaps

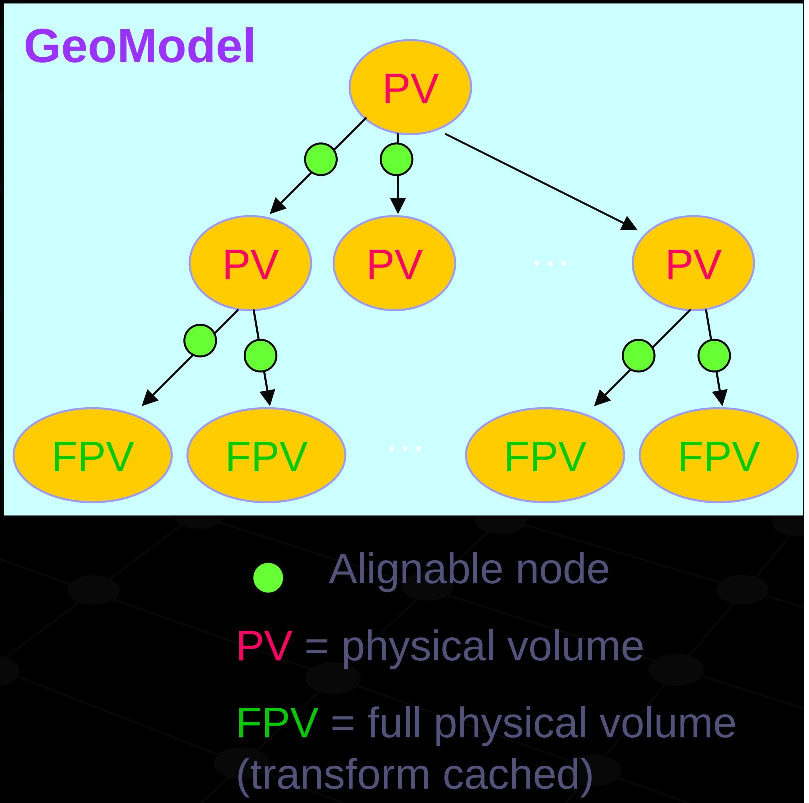
- TGeo overlap checker
- Ways to deal with detector overlaps:
  - Usage of TGeo assemblies

  - Misalignment also on high-level structures (detector, layers):
    - Big movements on high-level structures
    - Small (residual) movements at level of sensitive volumes

ATLAS

Geometry Implementation

- GeoModel (non ATLAS-specific)
  - GeoModel Kernel
    - toolkit of geometry primitives: shapes, materials, etc
    - Physical Volumes (**PV**) have attached nodes – connects other physical volumes via transform
  - Special "Alignable" nodes
    - Default transform + Delta transform
    - Can be at several levels
  - Full Physical Volume (**FPV**)
    - Calculates and caches local to global transform.
    - When alignable node higher up in hierarchy is modified:
      - cache is invalidated
      - transform recalculated on next access



Readout Geometry

- Detector Manager
  - Access to detector elements
  - Manages transfer of alignment constants from Conditions Database to GeoModel alignable nodes.
  - Alignments updated via callback (or explicit call in geometry initialization)
- Detector Elements
  - Points to GeoModel Full Physical Volume
  - Derived quantities cached for fast access
- Reconstruction only accesses geometry information via detector elements in readout geometry
  - Always gets aligned positions
  - Most clients do not need to worry about alignment infrastructure

Alignment Constants

- Defined as "Delta Transform" of "Alignable" nodes:
  - for modules – in a local frame
  - for higher level structures (sub-system,layers) – in global frame
- Rigid body transforms (rotation + translation)
  - Use CLHEP HepTransform3D
  - Applied directly in Detector Description at several levels in hierarchy (eg sub-system, layers, modules)
  - Alignment data is container of Identifier and HepTransform3D pairs
    - Common ATLAS Identifier uniquely identifies detector element or higher level structure (eg a silicon layer)
- Fine corrections. Eg module distortion, wire sag
  - Vector of floats
  - Interpretation is detector specific
  - Not dealt with at Detector Description level. Applied at Reconstruction level at time track is known

Conditions Database

- Alignment constants written to LCG POOL ROOT - same persistency technology as event data.
- LCG COOL database records IOV (Interval Of Validity) and reference to POOL file.
- Clients register callbacks on conditions data object.
- IOV services takes care of loading new data if IOV changes and triggers callbacks.

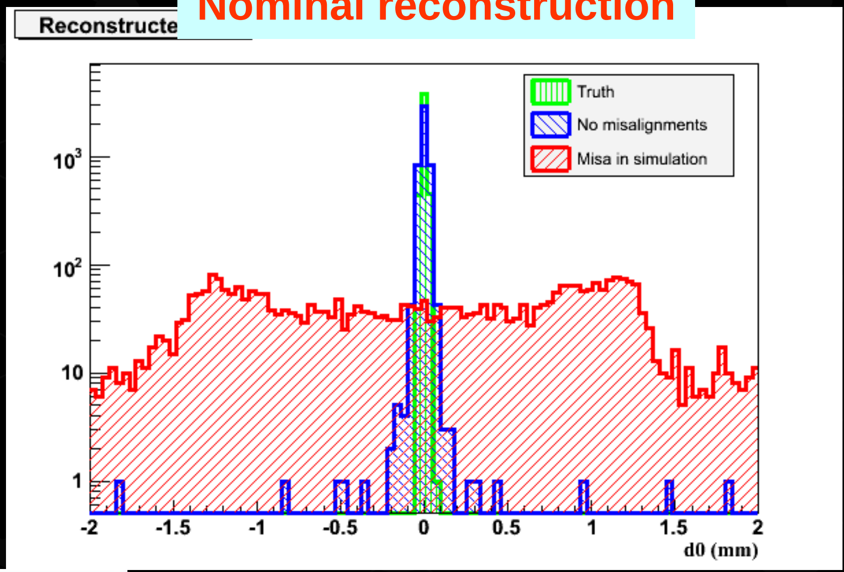
- Ideal geometry description stored in a separate DB (Primary numbers in DB ->GeoModel->Default Transforms)
- Conditions data will contain tag pointing to corresponding geometry description

Misalignments in Simulation and Reconstruction

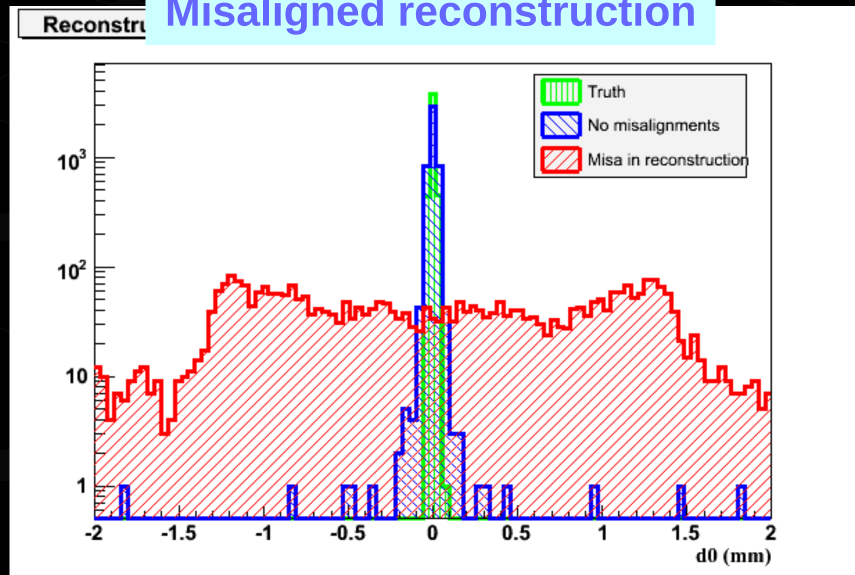
- GeoModel is common source for both Geant4 simulation and reconstruction:

  ⇒ Same infrastructure for misalignments in both.

Misaligned simulation, Nominal reconstruction



Nominal simulation, Misaligned reconstruction



Geometry overlaps

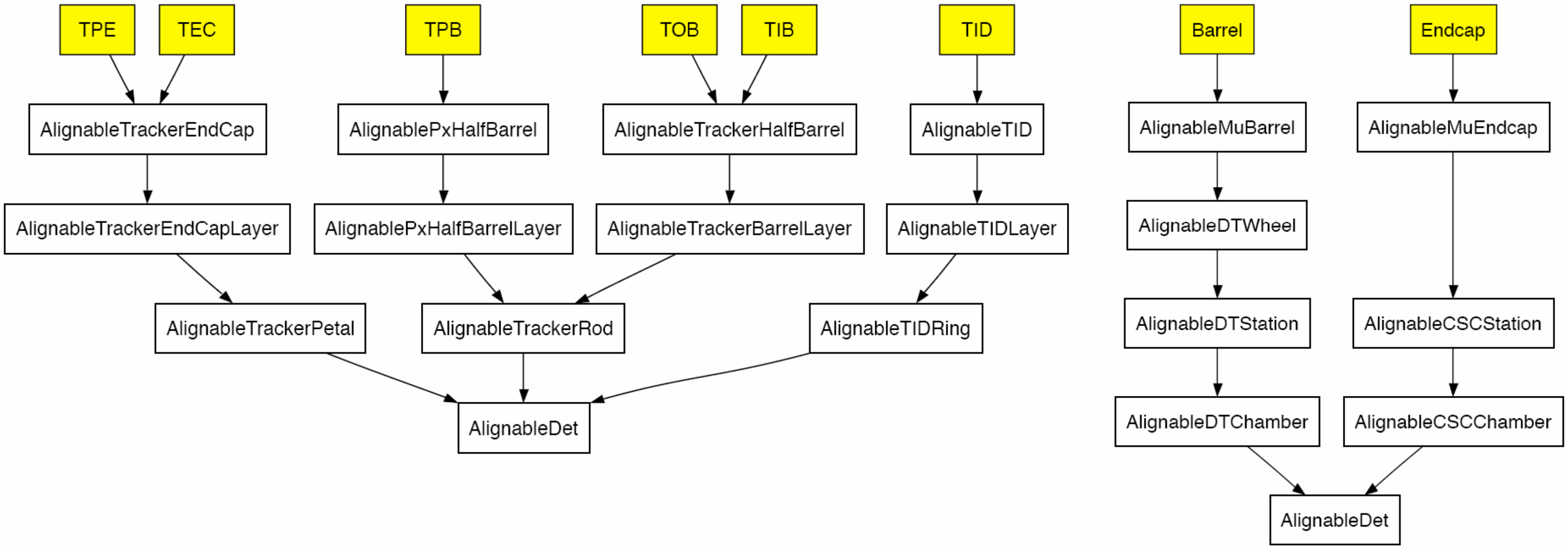
- Some extra challenges for misalignments in simulation geometry in order to avoid overlaps:
  - Enough clearance:
    - Mostly a matter of resizing/reshaping some envelopes
    - Some services had to be artificially thinned or moved
  - Facilitated by alignable nodes at several levels:
    - Allows larger movements of big structures (eg subsystem) and smaller movements of lower structure (eg modules)

CMS

Geometry Support for Misalignment

- Dedicated set of objects (AlignableTracker, AlignableMuon):
  - Map tracker and muon geometry including
    - High-level structures
    - Sensitive volumes used by the reconstruction
  - Propagate movements through detector hierarchy
  - Provide access to global position and orientation and detector ID for all sensitive volumes ("Alignment"s)
  - Misalignment consists in movements applied at various geometry levels and propagated down to sensitive volume (if needed)
  - Custom misalignments flexible set of pre-defined parameters in configuration files
  - Several "misalignment scenarios" are prepared (eg from engineering desing, survey,…)

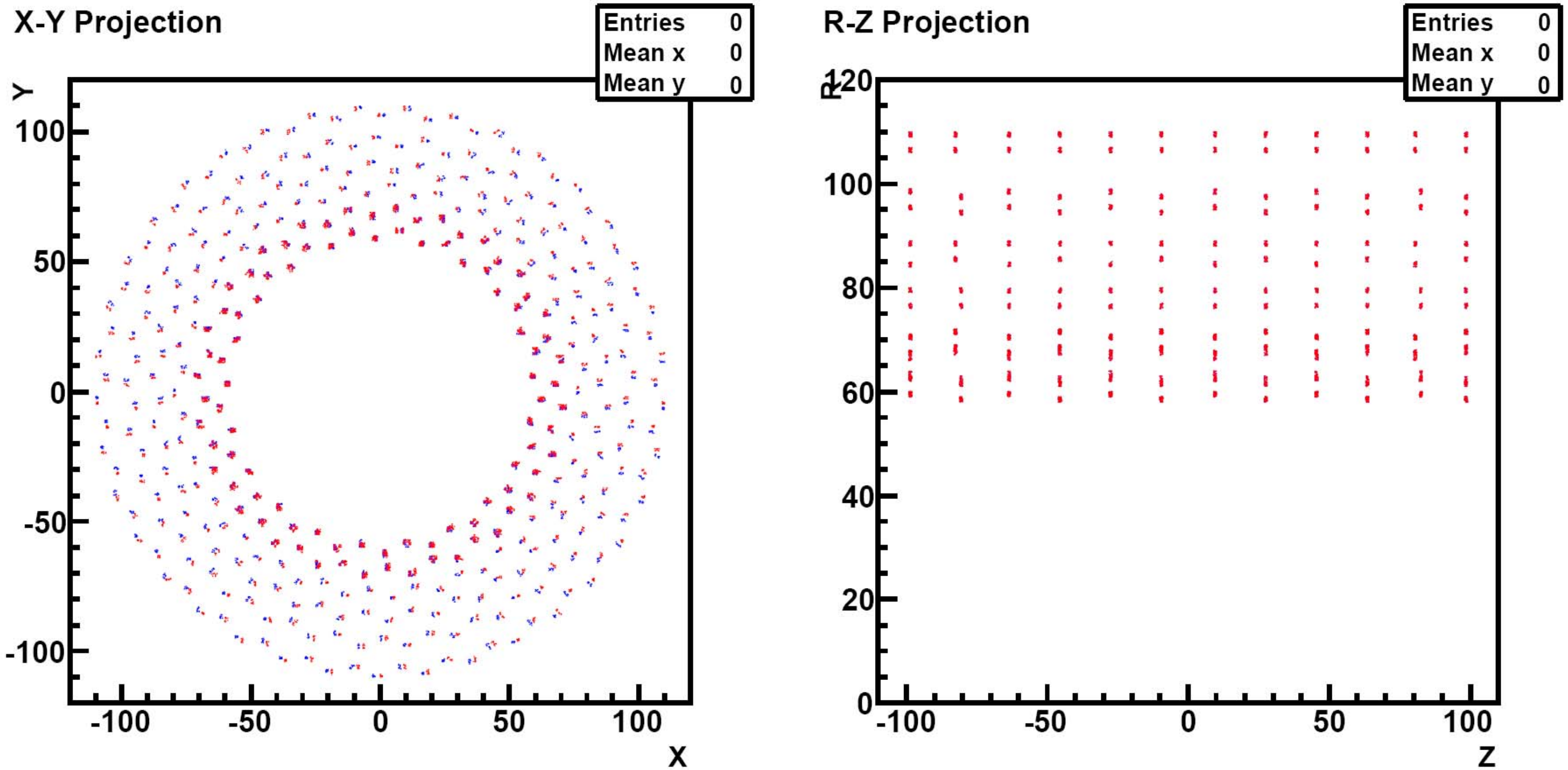
Hierarchy of structures in `AlignableTracker` and `AlignableMuon`.



Example

And the result of a rotation around the z axis of the outer barrel tracker:

```
PSet TOBs = { double phiZ = 0.02 }
```



Alignment Constants

- "Alignment" objects:
  - Default + Delta Transform (global position and orientation of sensitive volumes)
  - Volume ID
  - Same object contains all (mis)alignment information:
    - Misalignment scenarios
    - Alignment from survey data
    - Hardware alignment
    - Track-based alignment

Conditions Database

- Condition DB based on POOL-ORA
- Transparent to the user
- "Alignment" objects provided by AlignableTracker and AlignableMuon are stored in DB
- Different instances in DB have different tags (string defining content) and different intervals of validity
- Ideal geometry description is stored in XML format

Misalignment at Simulation and Reconstruction level

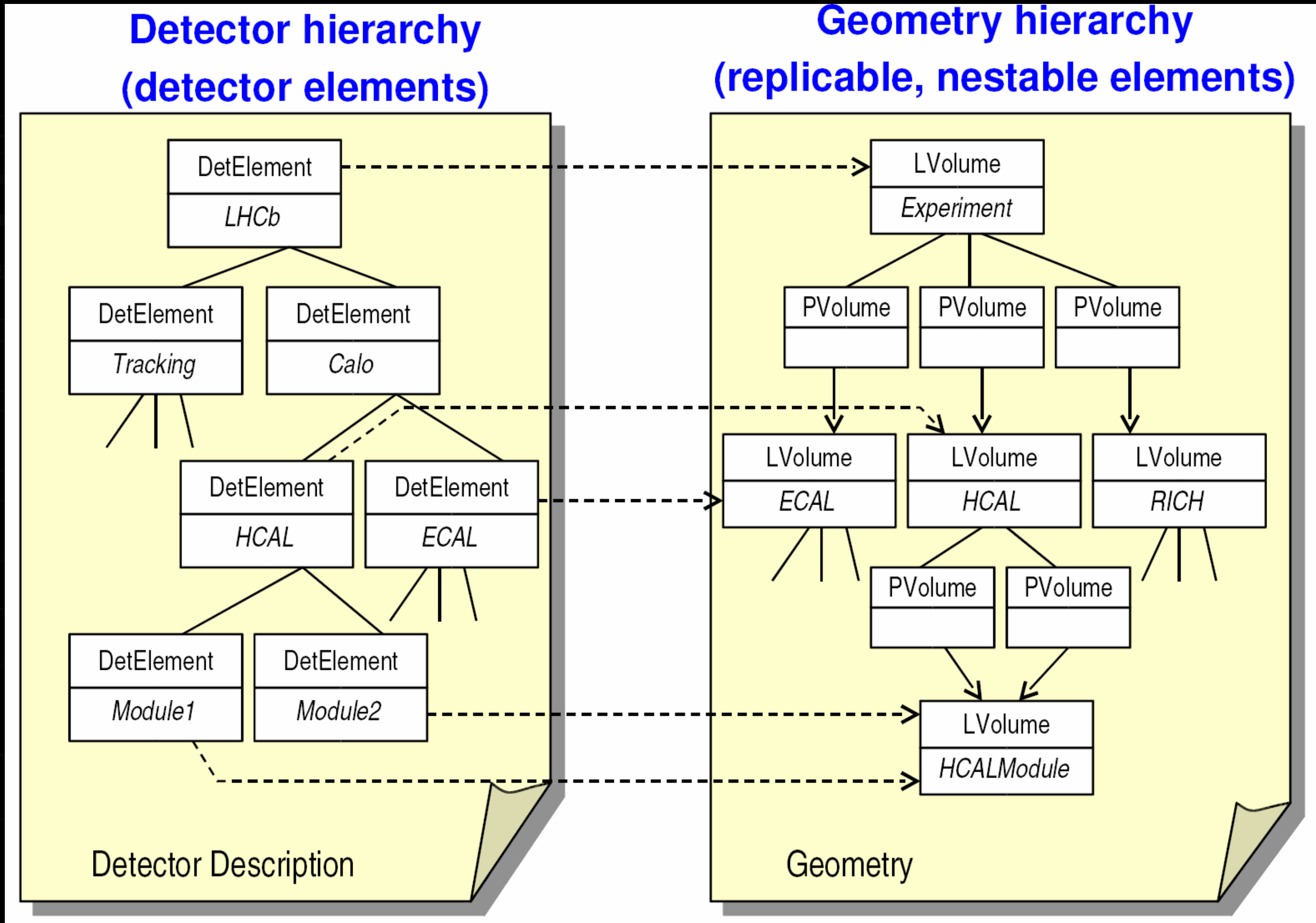
- Misalignment is performed at reconstruction level, just before global reconstruction:
  - "Alignment"s are applied to the reconstruction geometry
  - The reconstructed space-point positions are misaligned while converted from local to global frame
- Satisfactory for (mis)alignment at global level:
  - Main difference in the overlapping regions (at volume edges)
- The approach allows misalignment of any simulated sample
- Misalignment at simulation level is done using the same tools (via interface to GEANT4)?

LHCb

Geometry Implementation

- Resides in Detector Description DB
- Accessed via Gaudi transient detector data store
- Two hierarchical structures:
  - Geometry (Geant-like)
    - Re-usable blocks of geometry description
      - Volumes with shape, material (Logical Volumes)
      - Hierarchy from positioning of volumes within volumes
  - Detector structure
    - Coupled to physical structure of LHCb
    - Hierarchy of “interesting” Detector Elements (Physical Volumes)
    - One-to-one correspondence of real detector components
    - Knowledge of positions in global and in parent frames
    - Knowledge of place in hierarchy
    - Knowledge of daughter volumes (not necessarily detector elements)
    - Handle very sophisticated information

Geometry Implementation



Geometry support of Misalignment

- Misalignments are applied through Detector structure
- Handled by Detector Elements (Physical Volumes):
  - Combine local misalignment with local transform to obtain new local position and orientation
  - Use links to parents to global position after misalignment
  - Use links to daughters to propagate misalignments to daughter global position and orientation
- Users “see” misaligned detector description automatically

Alignment Constants

- Defined as Delta Transforms of Detector elements in their parents frame
- Rotation about pivot point + translation
- Detector Element ID: sub-detector specific
- It is possible to update the constants run-time
  - Changes are propagated to all the necessary parts with the update manager service, even if the database itself is not updated.
  - Possible usage in an iterative alignment job
- Alignment Constants can be easily generalized:
  - Euler angles <-> matrix representation

Condition Database

- LCG COOL database records
- Alignment constants stored as XML strings
- Validity intervals, versions
- Condition DB Manager handles:
  - Conditions data validity
  - Loading of (sets of) alignment constants in a job
  - Gaudi transient store on demand

Misalignment at Simulation and Reconstruction level

- Both simulation and reconstruction use the detector description service and access the misalignment framework
- Misalignment in simulation:
  - done in the same way with some constraints (eg geometry overlaps)
  - Detector elements -> Geant4
  - No run-time dependent conditions-like misalignments for simulation

  $\Rightarrow$ Simulation is done with one set of constants in a job

Summary Table

	ALICE	ATLAS	CMS	LHCb
Geometry Implementation	ROOT TGeo	GeoModel	Dedicated set of classes	Detector Description framework
Geometry Misalignment	Physical Volumes on top of hierarchy of Logical Volumes	Hierarchy of Physical Volumes	Hierarchy of Physical Volumes	two hierarchies: one of Physical Volumes and one of Logical Volumes
Alignment Constants	Delta Transforms; Global frame	Delta Transforms; Local + Global frame	Default + Delta Transforms	Delta Transforms; Local frame
Condition DB	Arrays of objects; ROOT files on GRID	XML strings LCG COOL		
Misalignment in Sim and Rec	Same framework; Sim: Trans. MC uses TGeo navigation	Same framework; Sim: GeoModel is source for GEANT4	Only In Rec; Sim: Interface to GEANT4	Same framework; Sim: Framework interfaced to GEANT4
Geometry Overlaps	TGeo assemblies; Misalignment on various geom levels	Clearance between volumes; Misalignment on various geom levels		In progress