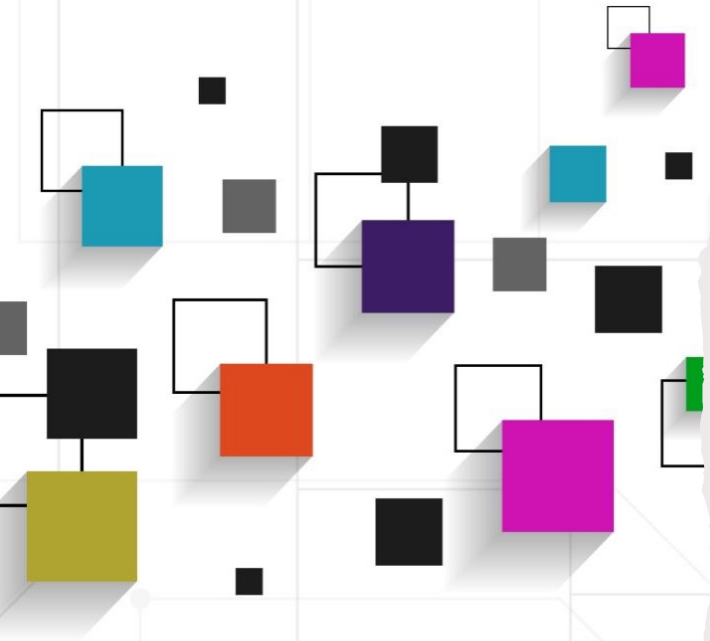


Detector Description Session in SW week. A status report

Joe Boudreau/Vakho Tsulaia

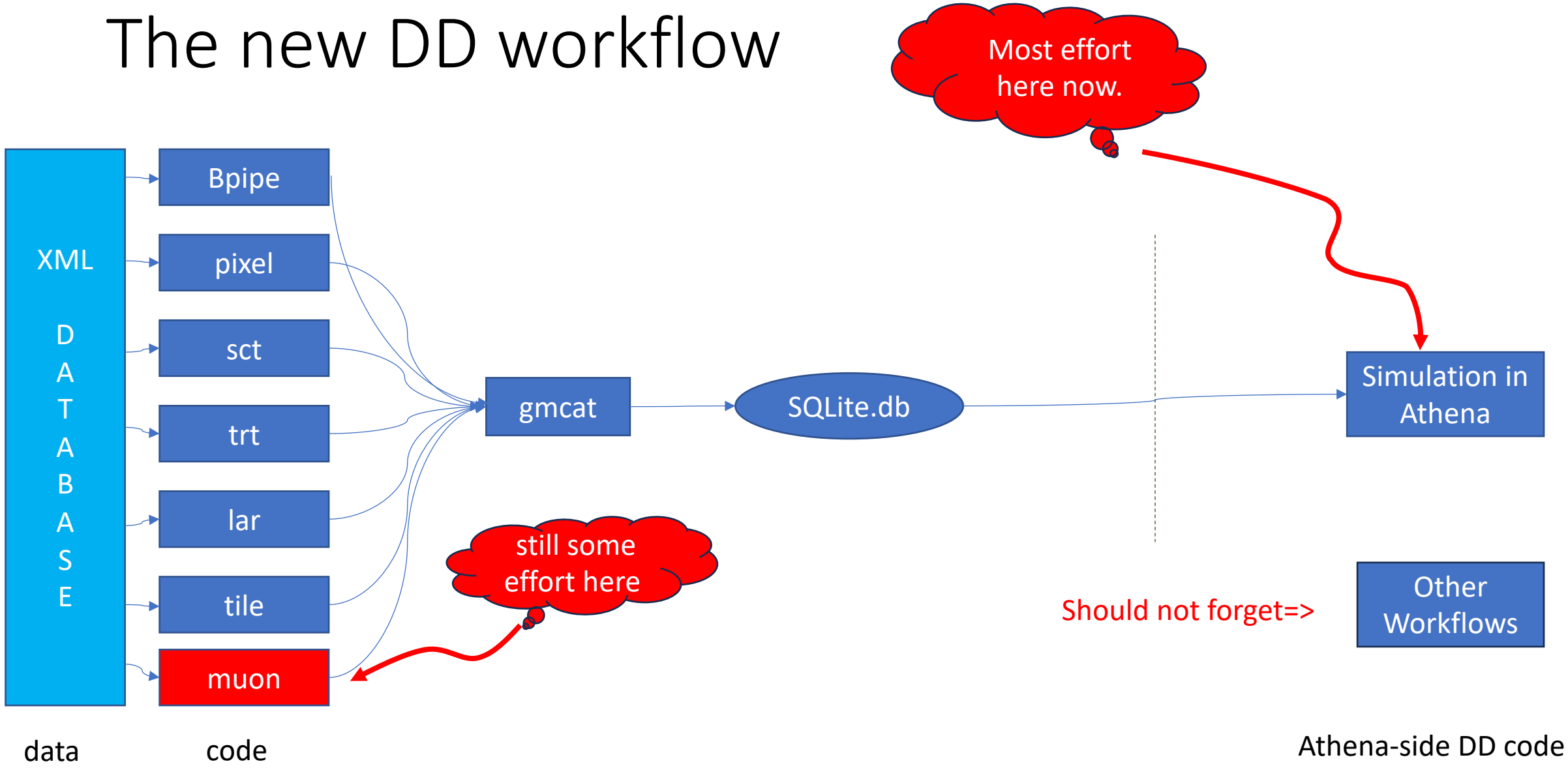
Detector Description meeting in Software week

June 12 2023

Outline

- Integration of the new SQLite-based workflow in athena
  - Run 3, 2, ..1, test beams
  - Run 4 (to be discussed, this session).
  - Identification and resolution of open issues (Geometry tags, Identifiers; see Vakho Tsulaia's talk)
  - Definition of procedures & roles for maintenance & operation of "newDD" components.
- Muon DD
  - ongoing developments (see presentation by Spyros Merianos)
  - Repackaged geomodel/amdb interim solution
- Ongoing issues
  - Optimize format of the SQLite database
  - Proper concatenation of "auxiliary tables" in gmcat
- New ideas to consider
  - Virtual geometry
  - Volume/surface/mass calculations

The new DD workflow

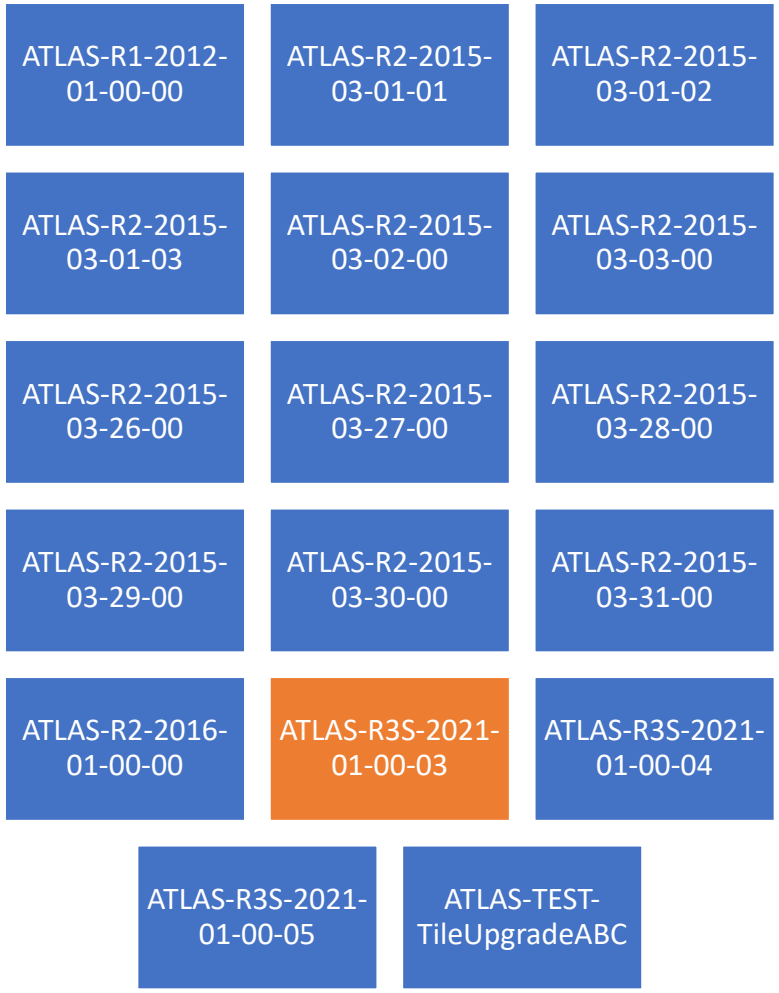


… and we need recruitment to maintain & operate all of this.

Principles to maintain in the production system.

- The SQLite is the only source of geometry input seen by Athena
  - The raw geometry is frozen and entirely captured in an SQLite file.
  - No further change in raw geometry can be applied on the Athena side.
    - Eg. today in Athena the geometry database contains some of the switches and tables that steer geometry construction, then the TRT gas is overriden in job options. We do not foresee this in the future. No code, either python or C++, can alter geometric or material properties anymore.
    - *However, one can construct such things as material-distorted SQLite files, for example.*
- The code to build ATLAS geometry is portable.
  - It does not depend upon `root`, in particular. Nor any other colossus.

Branches in the XML repository



- *On this page we list the geometry tags that we presume we will have to handle.*
- *We would further propose to limit this to geometries that can already be handled in the existing system, once that list is known.*
- *We highlight ATLAS-R3S-2021-01-00-03 which will be the first geometry tag to handle.*
- *All major detector systems work in the simulation workflow*
  - *Except the NSW which is under test right now.*

Status of athena integration today

- Configurations:
  - Only r3s-2021-01-00-03 corresponds to a branch in the sqlite DB
  - It is not tagged, frozen, validated, or even complete.
  - Run 3 testing start from this tag.

- Plugin code
  - Beampipe, Pixel, SCT, TRT, InDetServMat, LAr, Tile, and Toroids present.
  - Muons:
    - GeoModelXML-based solution under development by Andrea Dell'acqua & Spyros; athena integration work is being undertaken by Johannes Junggeburth. See talk by Spyros, this session.
    - Raw Geomodel interrim solution under development by Joe Boudreau can almost be used now going into validation.

Validation

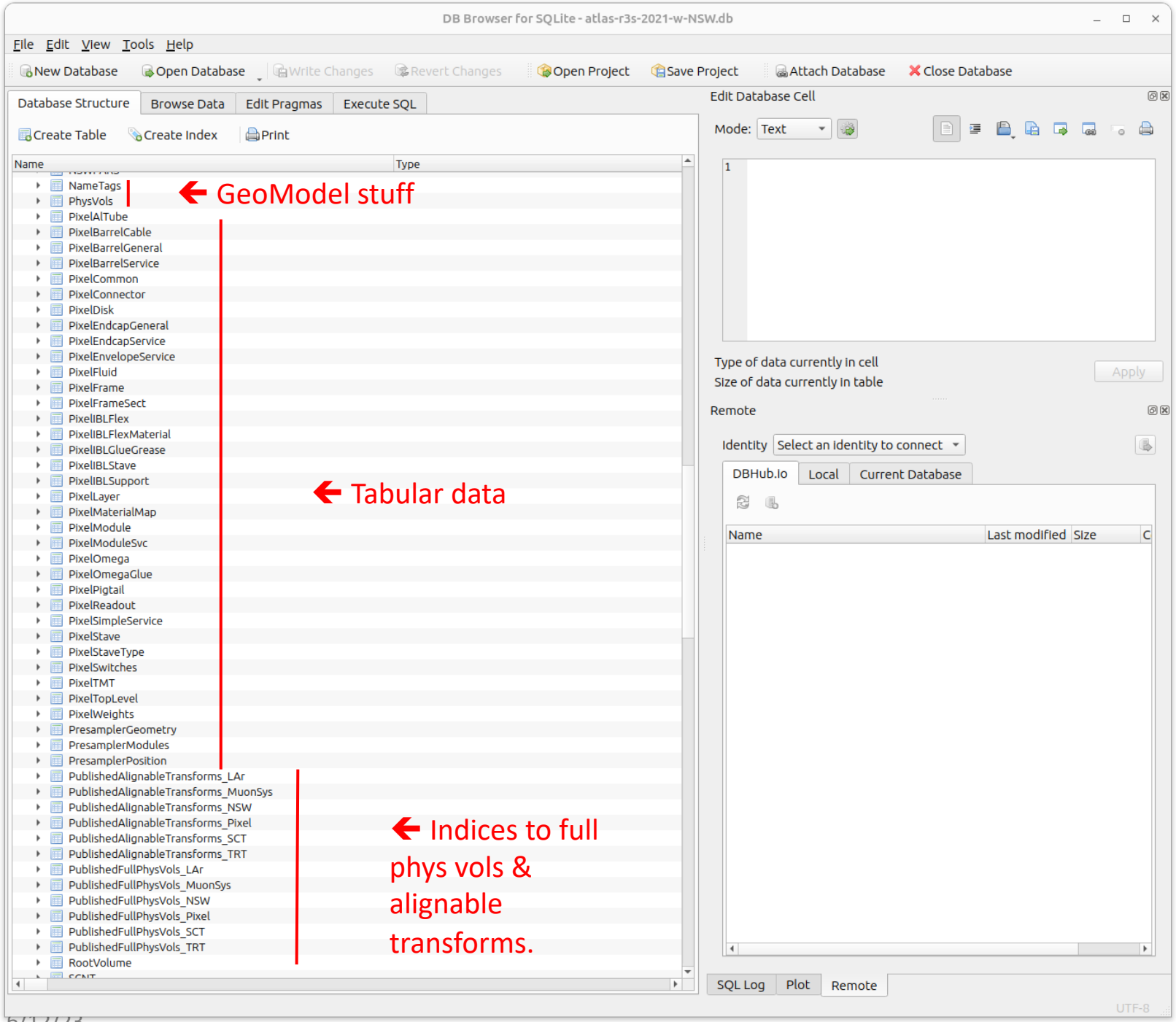
- This is our next major target!
- We propose to use the validation infrastructure (is the any choice?) to test sqlite inputs to the offline software.

- But before we get to this, we need
  - to address certain issues that Vakho will bring up in his slides.
  - do more in-house testing.

- We expect the tests to reveal multiple integration problems that all need to be addressed.

Integration is more than just raw geometry.

- The SQLite file is declared to GeoModelSvc via job options.
- GeoModelSvc restores the geometry from the SQLite file.
- The DetectorTools/Factories detect the pre-built geometry model.
- They read additional database tables from the SQLite file.
  - Most of these have been exported from Oracle.
- They locate the full physical volumes from an index within the SQLite file.
- They combine the full physical volumes with additional tabular data to construct and manage detector elements.
  - To do this would normally not be difficult, however retrofitting the existing system to this model may be the most challenging thing we have to do.
  - That's because the mechanisms to do this vary widely from subsystem to subsystem.


DB Browser for SQLite - atlas-r3s-2021-w-NSW.db
File Edit View Tools Help
New Database
Open Database
Write Changes
Revert Changes
Open Project
Save Project
Attach Database
Close Database
Database Structure
Browse Data
Edit Pragmas
Execute SQL
Create Table
Create Index
Print
Name
Type
NameTags
PhysVols
PixelAlTube
PixelBarrelCable
PixelBarrelGeneral
PixelBarrelService
PixelCommon
PixelConnector
PixelDisk
PixelEndcapGeneral
PixelEndcapService
PixelEnvelopeService
PixelFluid
PixelFrame
PixelFrameSect
PixelIBLFlex
PixelIBLFlexMaterial
PixelIBLGlueGrease
PixelIBLStave
PixelIBLSupport
PixelLayer
PixelMaterialMap
PixelModule
PixelModuleSvc
PixelOmega
PixelOmegaGlue
PixelPigtail
PixelReadout
PixelSimpleService
PixelStave
PixelStaveType
PixelSwitches
PixelTMT
PixelTopLevel
PixelWeights
PresamplerGeometry
PresamplerModules
PresamplerPosition
PublishedAlignableTransforms_LAr
PublishedAlignableTransforms_MuonSys
PublishedAlignableTransforms_NSW
PublishedAlignableTransforms_Pixel
PublishedAlignableTransforms_SCT
PublishedAlignableTransforms_TRT
PublishedFullPhysVols_LAr
PublishedFullPhysVols_MuonSys
PublishedFullPhysVols_NSW
PublishedFullPhysVols_Pixel
PublishedFullPhysVols_SCT
PublishedFullPhysVols_TRT
RootVolume
GeoModel stuff
Tabular data
Indices to full phys vols & alignable transforms.
Edit Database Cell
Mode: Text
Type of data currently in cell
Size of data currently in table
Apply
Remote
Identity
Select an identity to connect
DBHub.io
Local
Current Database
Name
Last modified
Size
SQL Log
Plot
Remote
UTF-8

Pre-validation integration test platforms:

- Vakho's ART test
  - Tested with r3s-2021-01-00-03
  - Simulation-only
- Sarka's test platform
  - Simulation+Digitization+Reconstruction
  - Fails on construction of tracking geometry + what else?
- Nick's detailed instructions
  - Runs ITK in simulation+ digitization
  - Runs in a pipeline
  - Could be the basis for a run 4 test setup w/ multiple detectors.
  - It's a topic for discussion here.

Next steps (in order..)

- Understand and deal with issues of geometry configuration and/or identifiers
  - See Vakho's talk.
- Extend the integration activities to all workflows
  - Digitization, Reconstruction, VP1, etc.
- Develop test programs, pipelines, etc
  - Use the physics validation infrastructure for this (?)
- Extend the integration & testing to run 4 geometries
  - Discussion on this issue to follow in this session
- Formalize (in a document) the procedures for maintaining and operating the new system.
  - As discussed in S&C week, January 2023.
- Implement the full set of geometry tags to be supported.

Key roles in a “newDD/SQLite”production system

- **Subsystem DD responsible:**
  - Responsibility for integrity of subsystem code and XML inputs
- **SQLite database curator**:
  - global responsibility for minting, maintenance, and integrity of a full set of SQLite DB’s.
- **Software coordination contact:**
  - responsible for insuring that all Athena-side DD code is maintained, pipelines run, and results are stable.

- A COM note outlining the procedures for maintaining and operating the New DD system is in preparation.

The software description of the ATLAS Detector: Procedures and Roles

Marilena Bandieramonte[a], Riccardo Maria Bianchi[a], Joseph Boudreau[a], Andrea Dell'Acqua[b], Vakhtang Tsulaia[c]

[a]*University of Pittsburgh*
[b]*CERN*
[c]*Lawrence Berkeley National Laboratory*

The generation of a complete software description of the ATLAS detector involves database management, software development, and validation activities that are crucial for stable and accurate simulation, reconstruction, and physics analysis. Much of the detailed knowledge of detector subsystems resides with subsystems experts, while generic detector description (DD) application software expertise resides with the detector description, simulation, and reconstruction groups. This document outlines the procedures for the development of tagged, self-contained SQLite databases for use in production workflows, including but not limited to simulation and reconstruction. In addition, it proposes a number of roles related to this task to be filled by the subsystems responsibles.

https://www.overleaf.com/project/6463755d053055eff91d846e

Muons

- The GeoModelXML based solution is to be the long term solution.
  - The corresponding plugin is called MuonPlugin.
  - It will only be integrated with Athena when significant code on the Athena side is written from scratch (thanks to Johannes)
  - Both run 3 and run 4 geometries will be integrated with Athena.
- The raw Geomodel based solution is a fallback solution.
  - The corresponding plugin is called the MTechPlugin
  - Can almost validate today.
  - Still missing NSW:
    - NSWPlugin exists.
    - It uses AGDD for geometry construction
    - Integration Athena in progress.
  - Run 3 geometries working with simulation (Vakho's ART test script)
  - It is not foreseen to carry this into run 4 geometries.

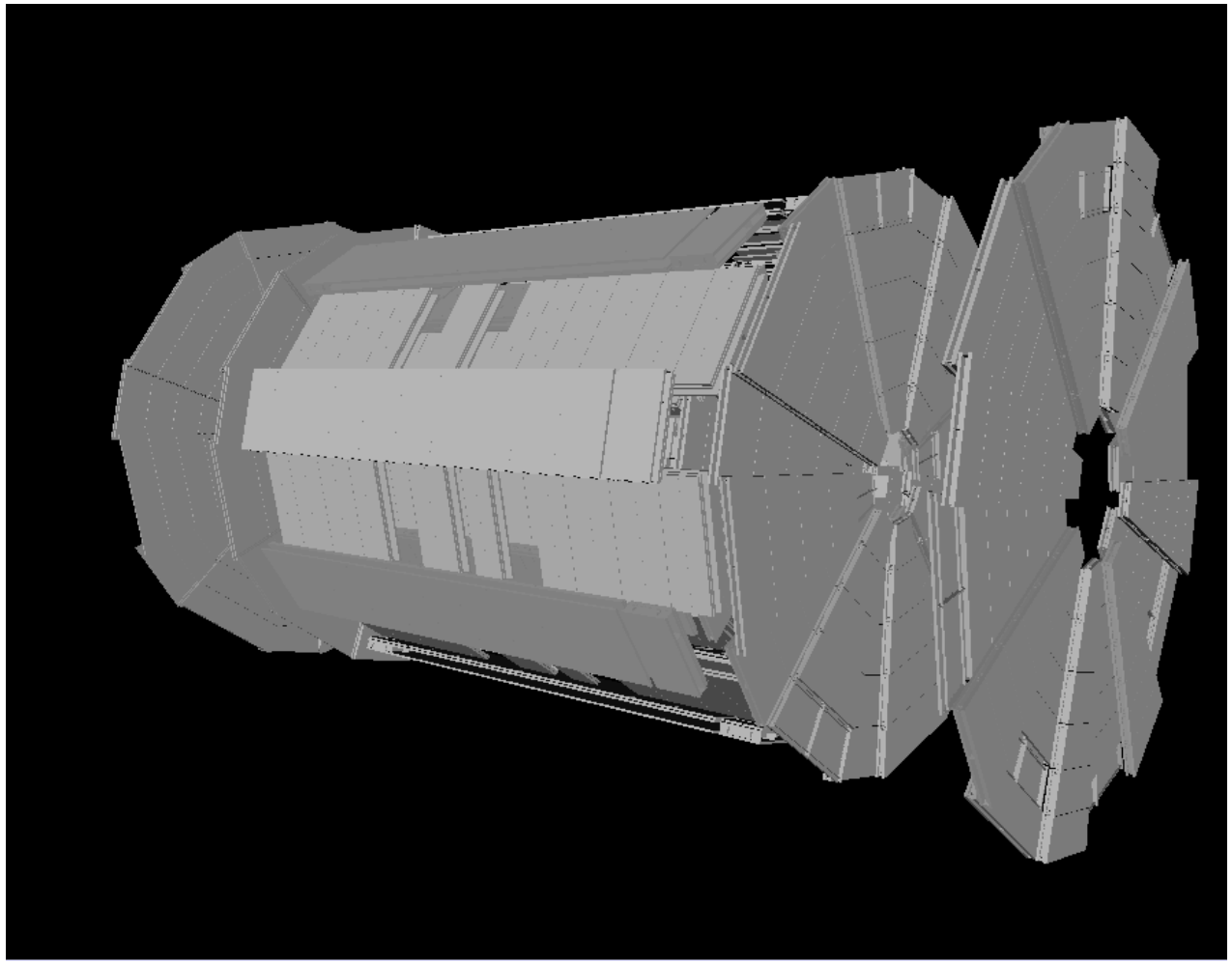
The GeomodelXML based solution nearing completion:

This is the status as of today.

Lots of new stations added by Spyros.

Certain tables required for readout geometry area also exported.

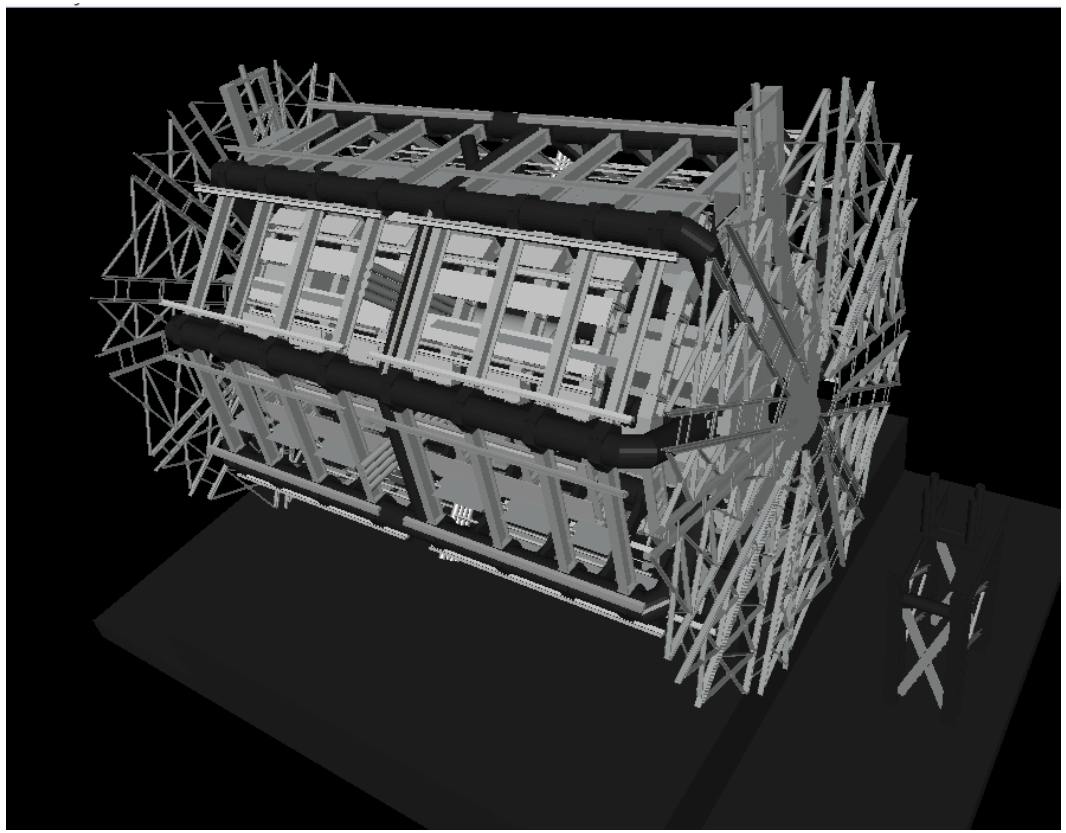
We express our gratitude for this.

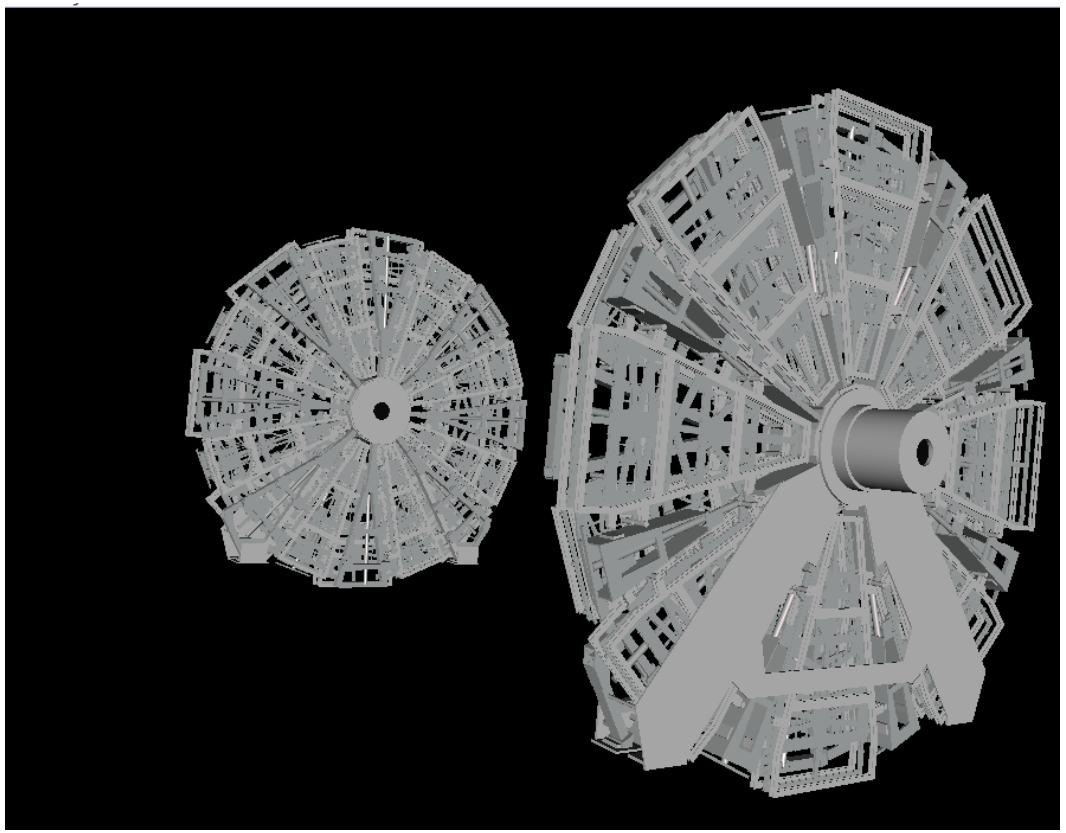


About the NSWPlugin Interim solution

- It produces a raw geometry & database tables for readout geometry construction in Athena.
- Today in Athena, the readout geometry information is purveyed through and AGDDService which is disabled when initializing geometry from SQLite.
  - The required data is now (local release!!)
    - Coming from the WMM table and WSTGC tables created by Spyros
    - put into GeoModelData xml repository.
    - transferred to the SQLite database.
    - read into Athena through RDBAccessSvc
    - And if the same data were put into Oracle, we could today access it in the same way for SQLite/NonSQLite jobs.

Reminder: AGDD (exported outside athena) still generates toroids and NSW support





Ongoing issues

- Optimization of the SQLite database schema
  - Much of the GeoModel tree is stored as ASCII within the SQLite DB.
  - We expect better performance (speed and file size) if stored as floats or ints.
  - This is not a big worry, but it would be good to sort it out before we go into production.

- gmcat concatenates data from:
  - SQLite files
  - Plugins
  - Any combination
  - …but it does not (yet) concatenate the tabular data or the indices originating from input files. To be fixed.

- Identifiers present multiple issues; can we find coherent ways of
  - Keeping them in synch with geometry?
  - Indexing the geometry outside of athena?

- It is scary the large fraction of geometrical shapes which are implemented as Boolean volumes.

- And in addition the visualization of these Boolean volume is imperfect, maybe misleading.

New ideas

New Idea #1: endow GeoModel with more powerful geometrical computation (Evgueni Tcherniaev's idea)

- GeoModel is essentially a raw data layer.
- Apart from caching position information, the geometrical primitives do not do much.

- Geant4 has a more extensive set of geometry calculations:
  - Inside/outside/on surface.
  - Generate point on surface.
  - (Correctly) compute volume..

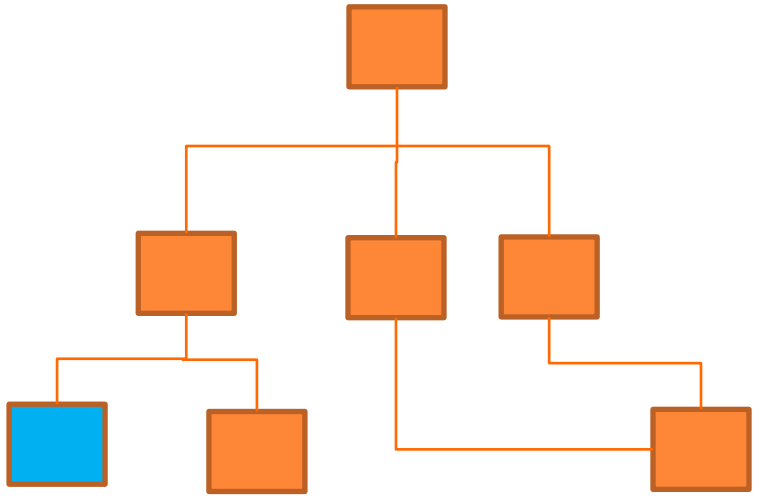
- These could be useful for visualization and material mapping.
- They could be introduced to GeoModel (e.g. via multiple inheritance)
- This idea is not yet widely discussed but appears worthwhile.

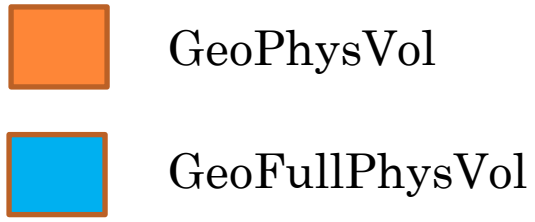
New Idea #2: Virtual geometry

- There are certain geometrical constructs in ATLAS software that do not represent physical structures and are not declared to Geant.
- An example which has come up recently is Tracking Surfaces.
- This type of object (are there more like that) is what I propose to call Virtual Geometry.
- I believe there are some advantages to embedding this within the GeoModel tree.
- The possibility of so doing has been foreseen from the earliest stages of the development of GeoModel.

GeoPhysVol and GeoFullPhysVol

- `GeoPhysVols` are assembled into a tree structure.
- There are two kinds: `GeoPhysVol` and `GeoFullPhysVol`
- They have (almost) the same interface but `GeoFullPhysVols` are optimized for the kinds of operations required of active detector elements.

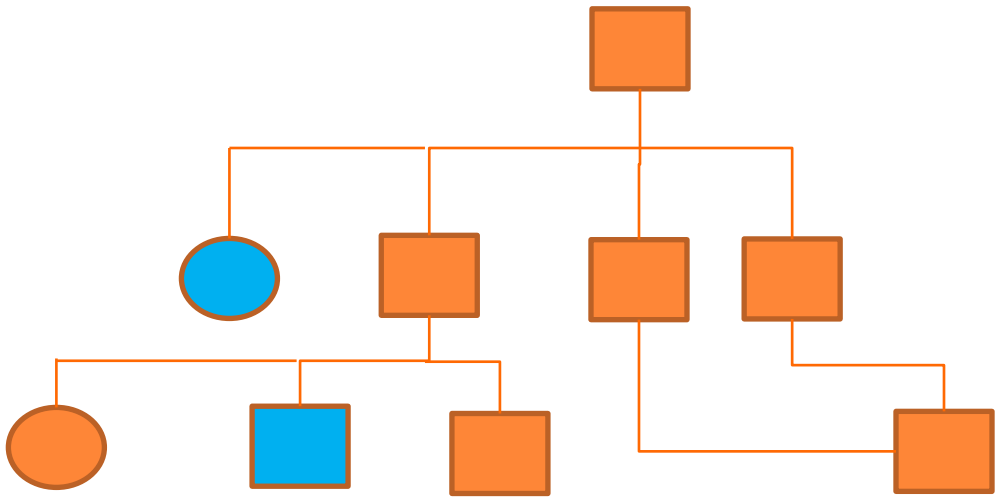


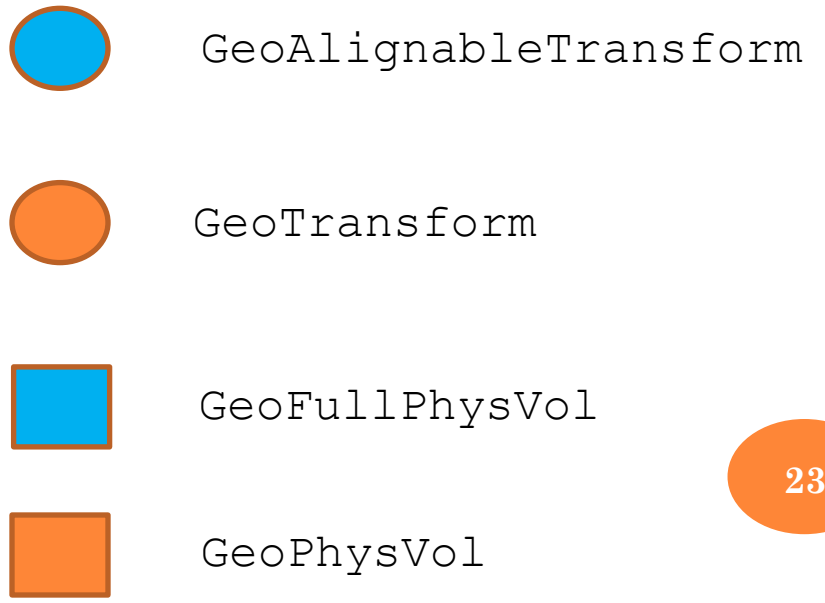


Transformations

- `GeoPhysVols` do not *have* a position. Instead, their position in space is determined by transformations inserted into the geometry tree upstream of the volume.

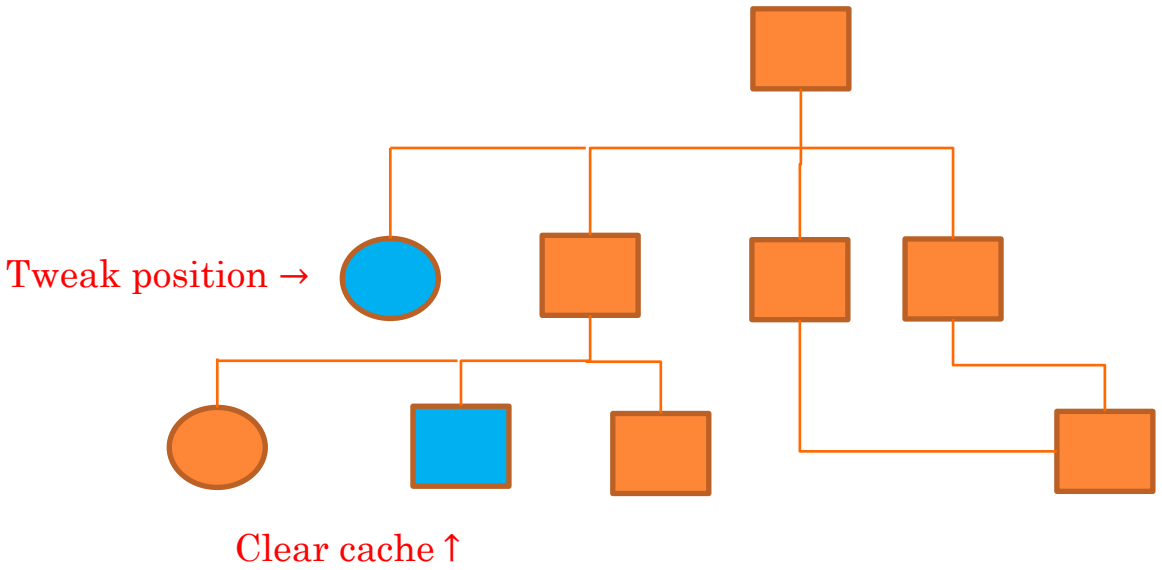
- It's a memory optimization.
- The global position of any volume is the composition of all transformations encountered during tree traversal
- But in `GeoFullPhysVols`, the result is cached.
- This is the difference between full and ordinary physical volumes.

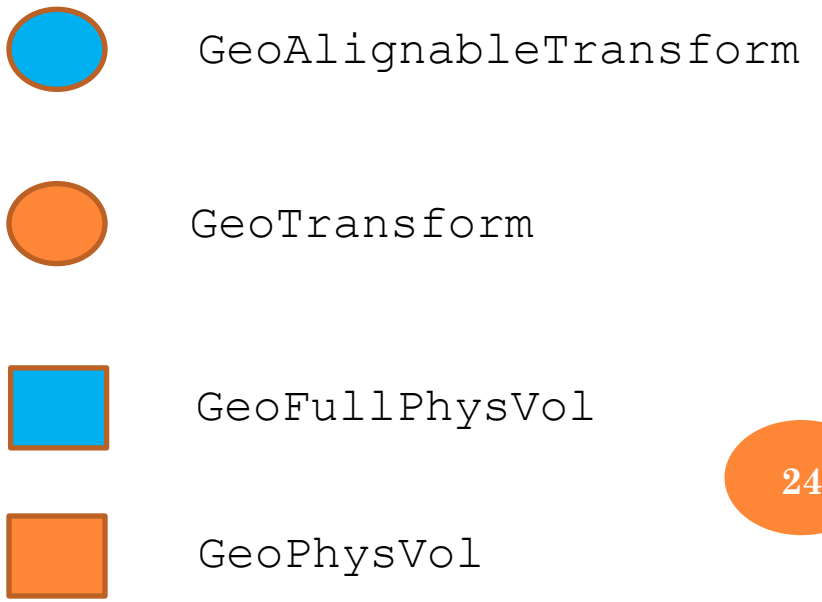




Alignment

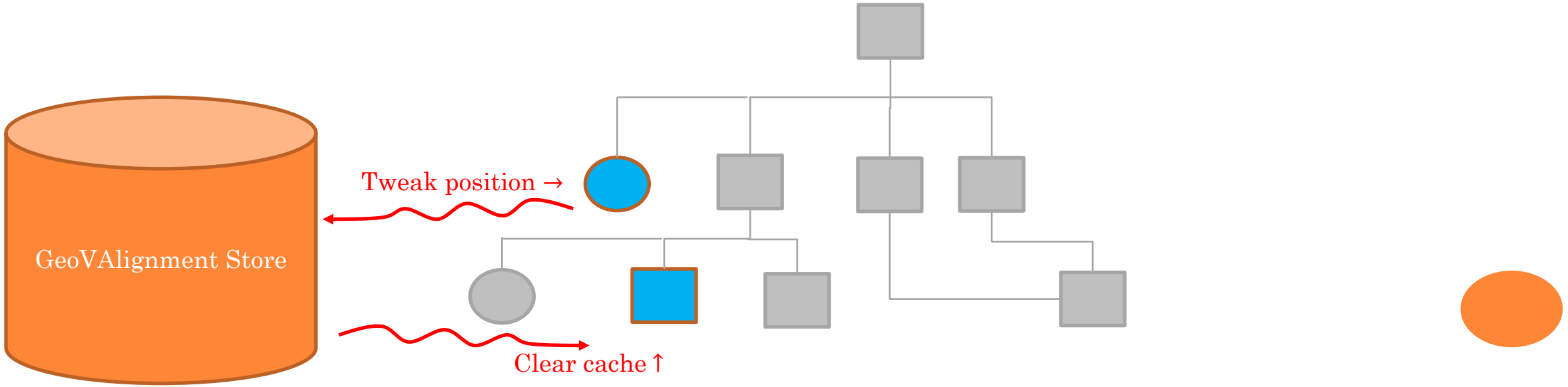
- `GeoAlignableTransforms` may be modified by adding a small "delta" transformation to the transformation which defines the default position.
- When this happens the caches of all affected physical volumes are cleared.





Thread issues

- With multithreaded Athena, the geometry can ”be in two places at the same time”.
- This occurs when two events from different run periods are simultaneously processed.
- In that case the alignment information is stored in a dedicated class which does essentially the same job caching and updating, but per-thread.



Tracking surfaces

- Tracking surfaces are conceptual surfaces located within the geometry and used within reconstruction.
- Example: there “is” a plane in the muon MDTs that goes through the center of a layer of tubes.
- This plane must move when the alignment is changed.
- Reconstruction groups have had to handle this alignment in an ad-hoc way, it appears.
- I believe we could incorporate those surfaces and their alignment in GeoModel at very low cost in effort.
- Do we (ATLAS SW) see an advantage in doing so?

How to introduce virtual geometry

- Extend `GeoGraphNode` with more classes that may be added as children to `GeoPhysVol` (or `GeoFullPhysVol`).
- Call the first of these: `GeoPlanarSurface`.
- Along with that: `GeoFullPlanarSurface`.
- These would respond to alignment updates in the same way as `GeoPhysVol/GeoFullPhysVol`
- Of course a serious design effort would have to examine very carefully the details

Wider impact

- Input/output would have to change to support the new classes.
- gmex would clearly be updated to display the Virtual Geometry.
- The Alignment store would need to handle the VirtualGeometry in the same way as the real geometry .
- Looking forward to informal meetings with interested parties during SW week!

Detector Description Session During S&C Workshop

Monday 12 Jun 2023, 10:30 → 12:15 Europe/Zurich

40/S2-D01 - Salle Dirac (CERN)

Joseph Boudreau (University of Pittsburgh (US)), Vakho Tsulaia (Lawrence Berkeley National Lab. (US))

Videoconference ATLAS Detector Description Weekly — Join — 40/S2-D01

10:30 → 10:55 **New DD inftrastructure.** 25m

Speakers: Joseph Boudreau (University of Pittsburgh (US)), Joseph Boudreau (University of Pittsburgh)

10:55 → 11:15 **Discussion** 20m

1. Handling of ID dictionaries
2. Passing geometry version names/tags to Athena/JobTransforms

Speaker: Vakho Tsulaia (Lawrence Berkeley National Lab. (US))

11:15 → 11:35 **Update on Muons** 20m

Speaker: Spyridon Merianos (Aristotle University of Thessaloniki (GR))

11:35 → 11:55 **New EMEC description with constructed solids** 20m

Speakers: Evgueni Tcherniaev (University of Pittsburgh (US)), Evgueni Tcherniaev (Tomsk State University (RU))

11:55 → 12:15 **Simplification of the ATLAS CAD geometry for Geant4 simulation** 20m

Speakers: Alexander Sharmazanashvili (Georgian Technical University (GE)), Salome Vashakidze (Georgian Technical University (GE))