Evolution of ROOT geometry package

Andrei Gheata

ROOT

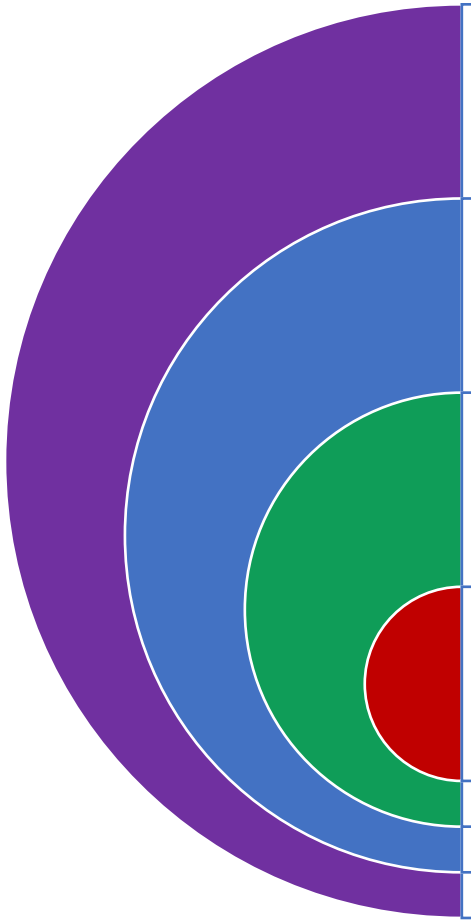
Data Analysis Framework

https://root.cern

Outlook

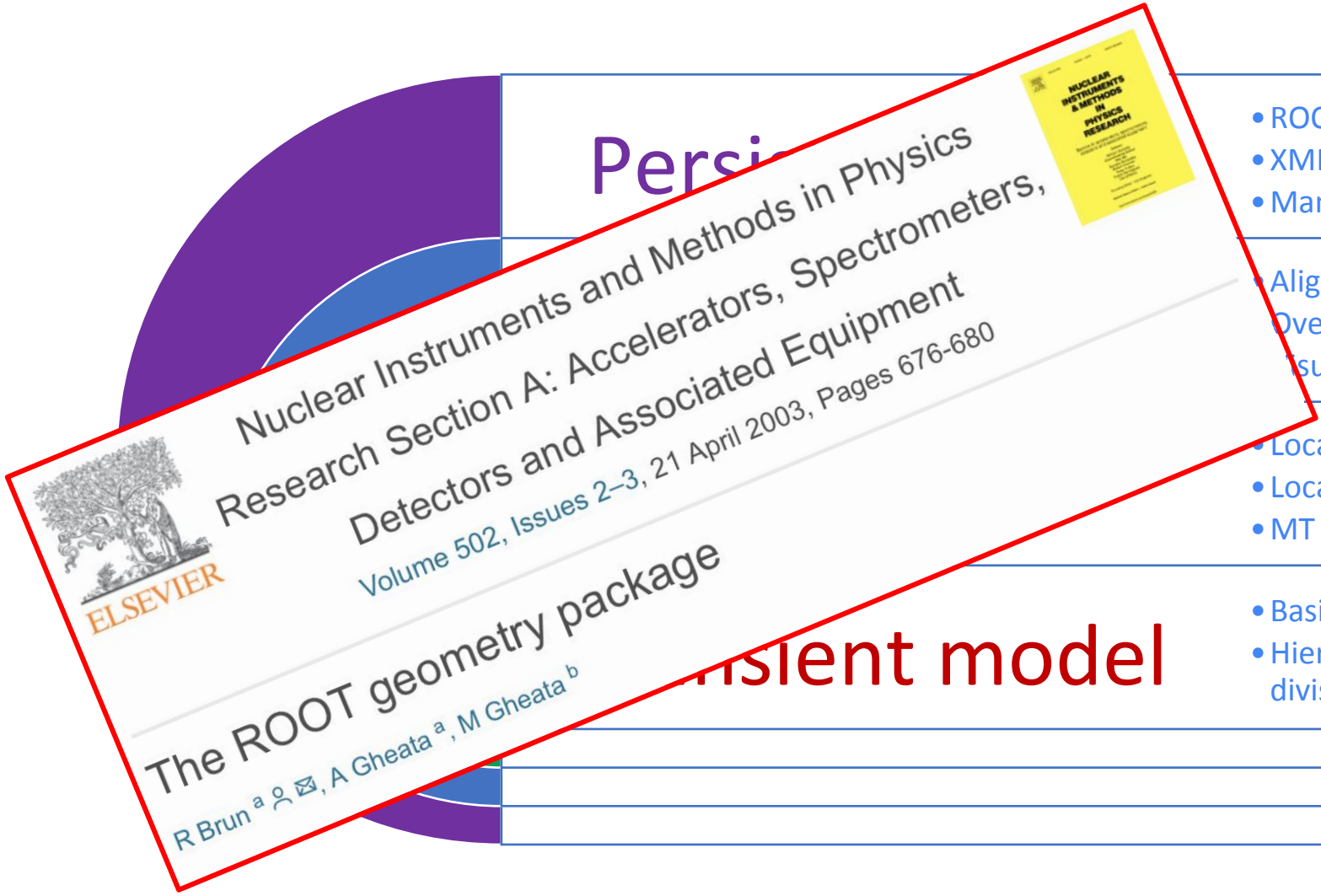
- Geometry features, status, usage
- VecGeom features
- TGeo evolution, mix and match with VecGeom
- Open questions

TGeo main components



Persistent layer	• ROOT binary format • XML, GDML, code generation • Many converters available
Features	• Alignment on top of ideal geometry • Overlap checker • Visualization, event display easy integration
Navigation	• Localizing points, computing distances, … • Local (solid) algorithms, optimizers (voxels), global • MT support on demand
Transient model	• Basic types: shapes, transformations, materials • Hierarchical elements: volumes, placements, divisions

TGeo main components



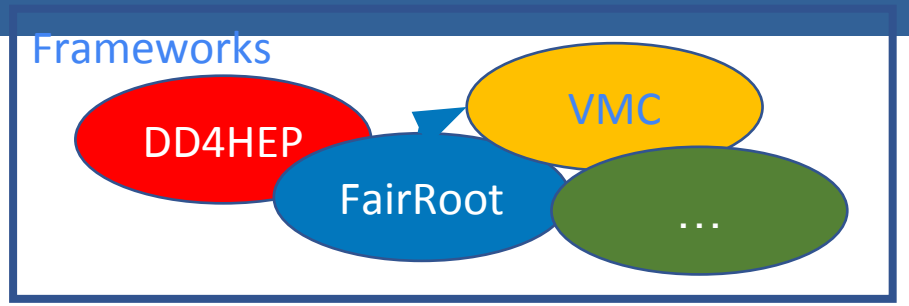
- ROOT binary format
- XML, GDML, code generation
- Many converters available

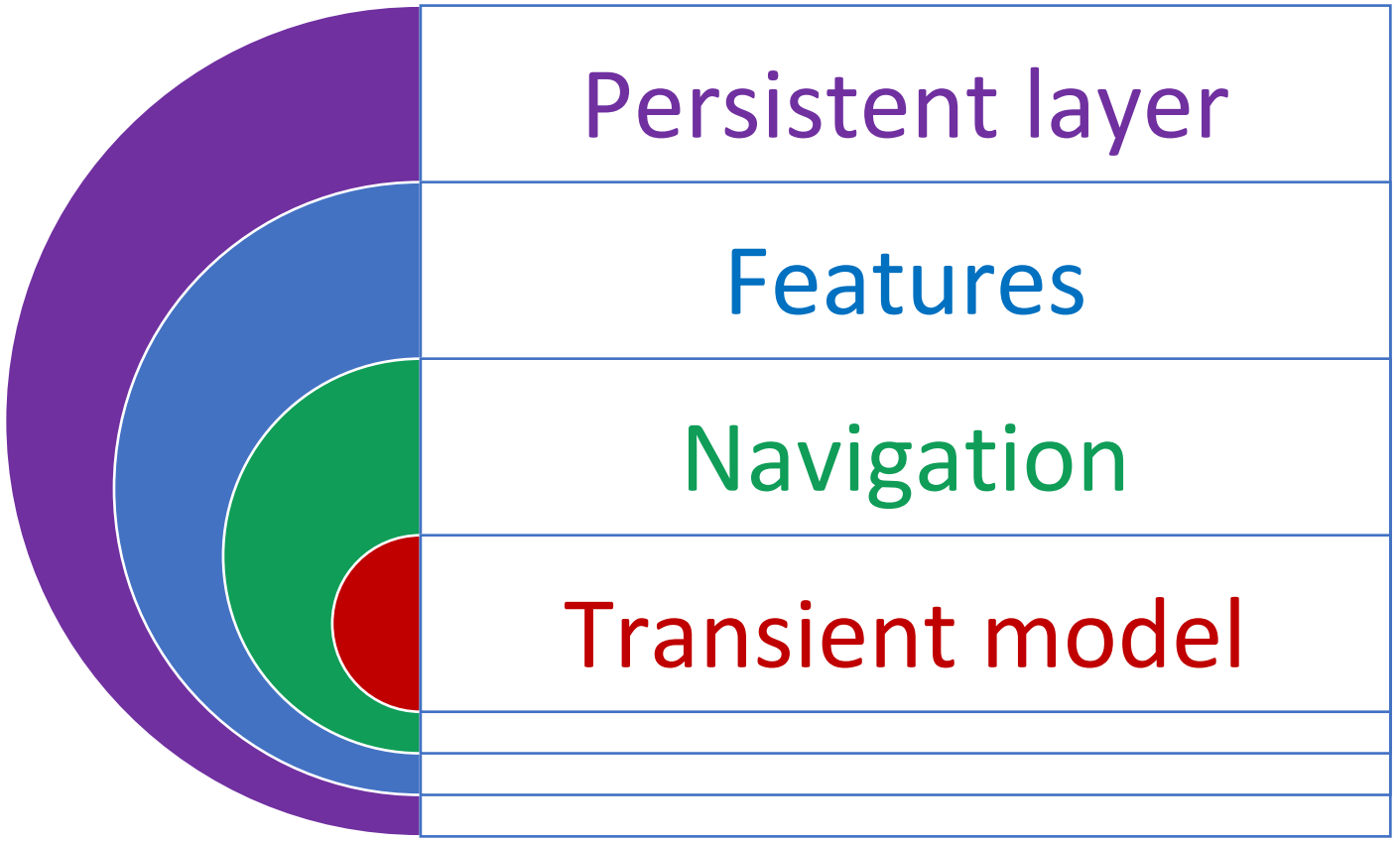
- Alignment on top of ideal geometry
- Overlap checker
- Visualization, event display easy integration

- Localizing points, computing distances, …
- Local (solid) algorithms, optimizers (voxels), global
- MT support on demand

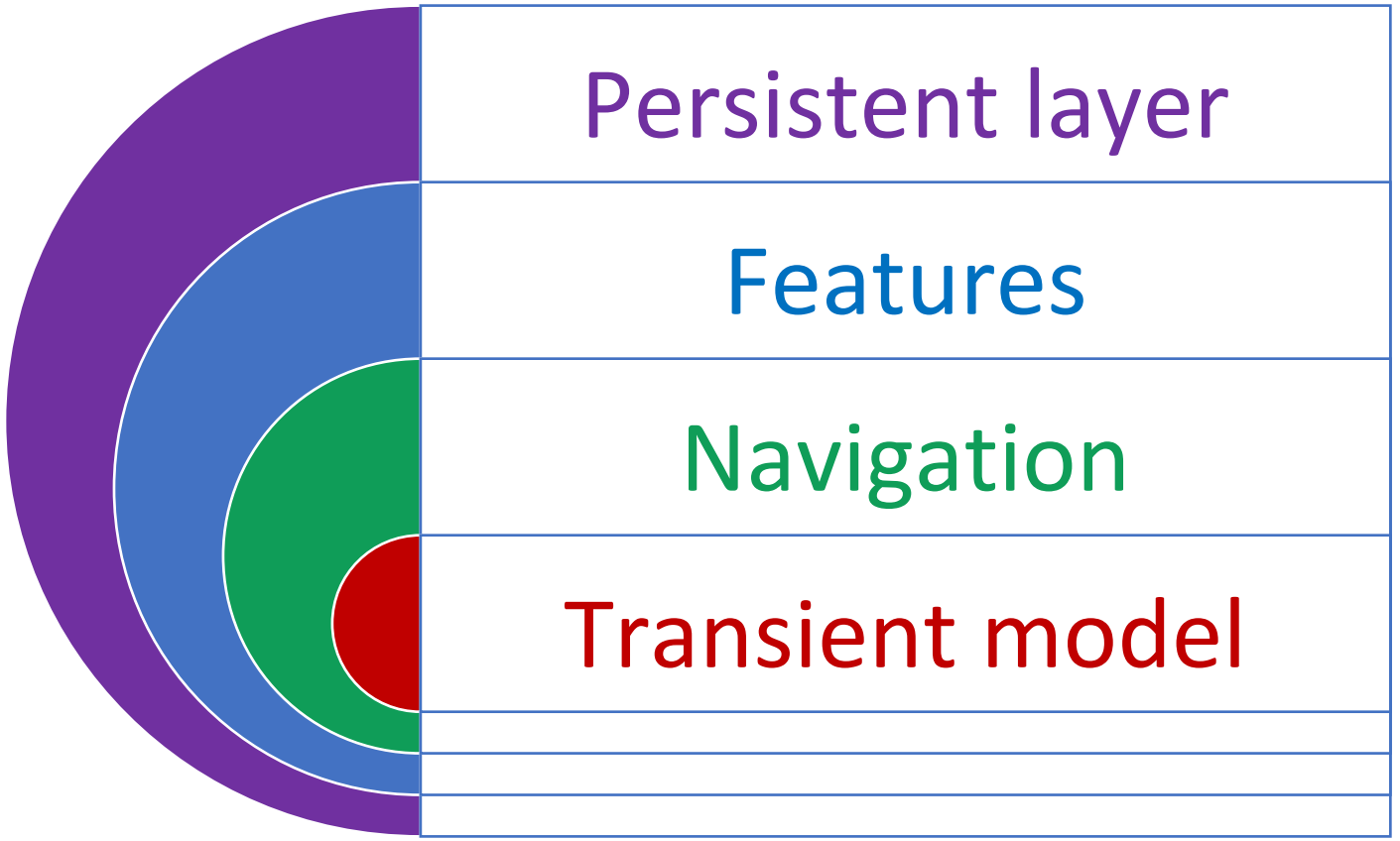
- Basic types: shapes, transformations, materials
- Hierarchical elements: volumes, placements, divisions

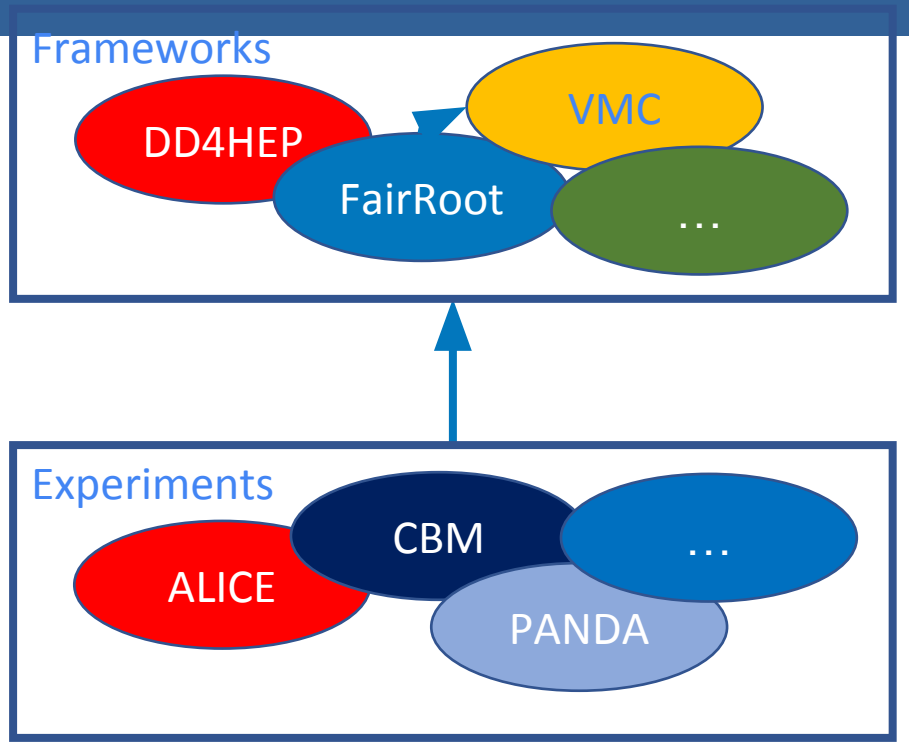
TGeo usage


Frameworks
DD4HEP
VMC
FairRoot
...

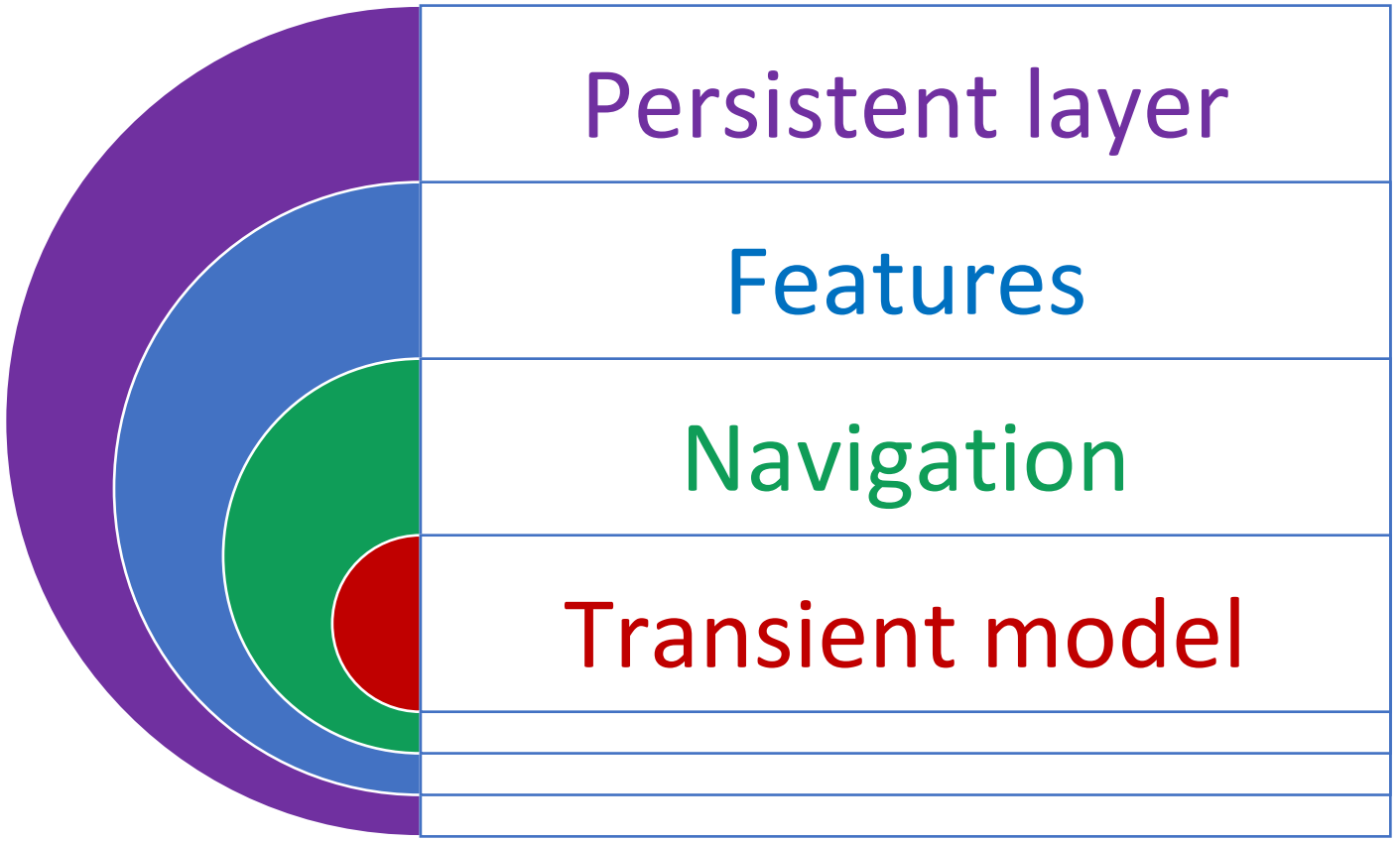

Persistent layer
Features
Navigation
Transient model

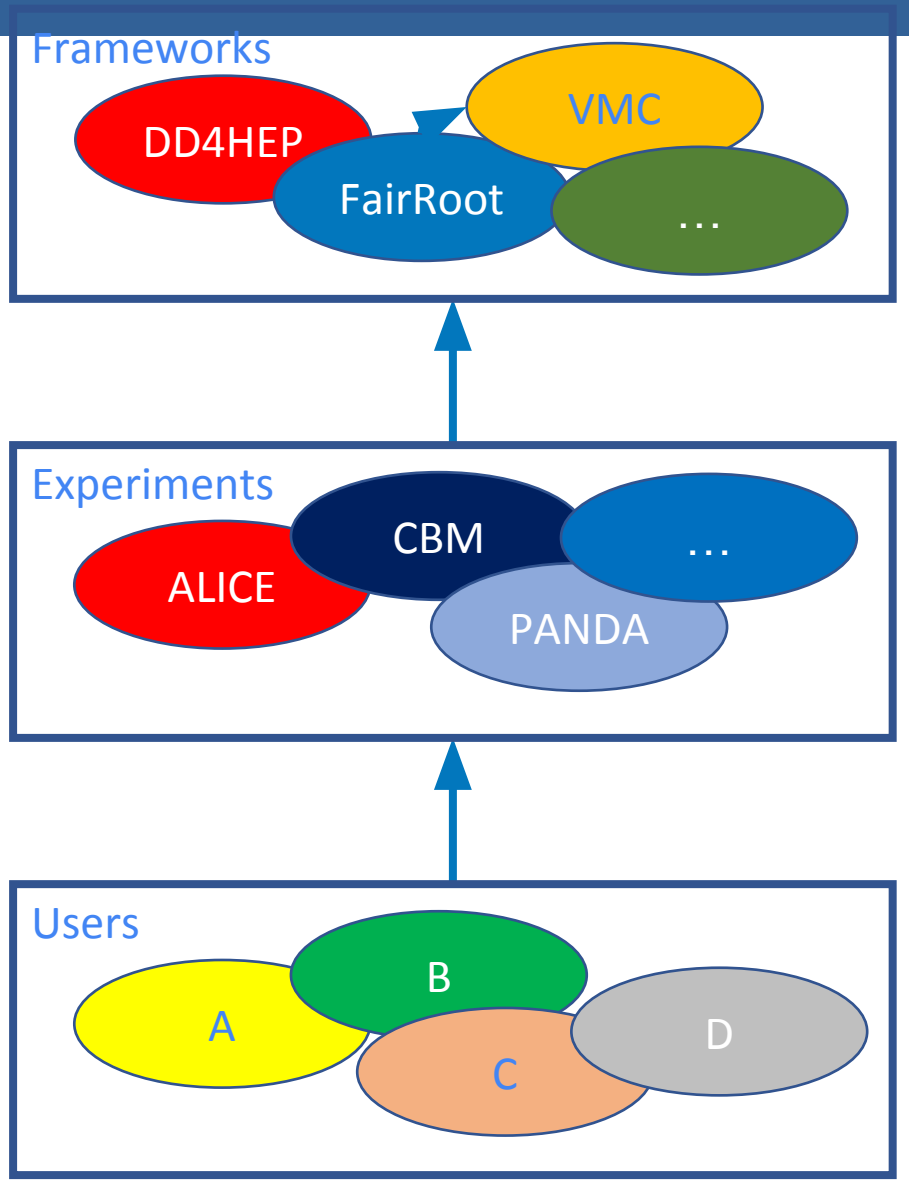
TGeo usage


Persistent layer
Features
Navigation
Transient model

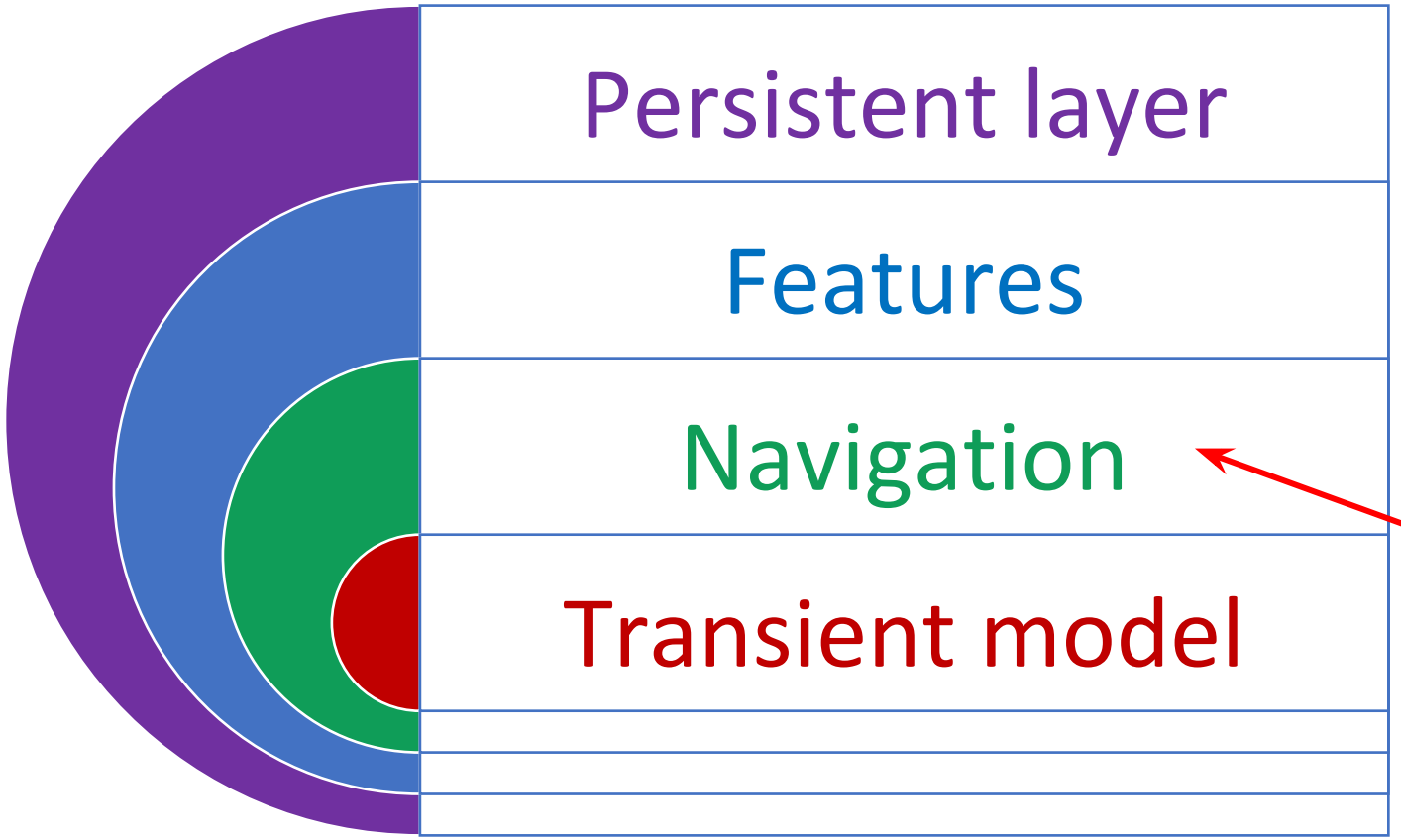

Frameworks
DD4HEP
VMC
FairRoot
…
Experiments
ALICE
CBM
…
PANDA

TGeo usage

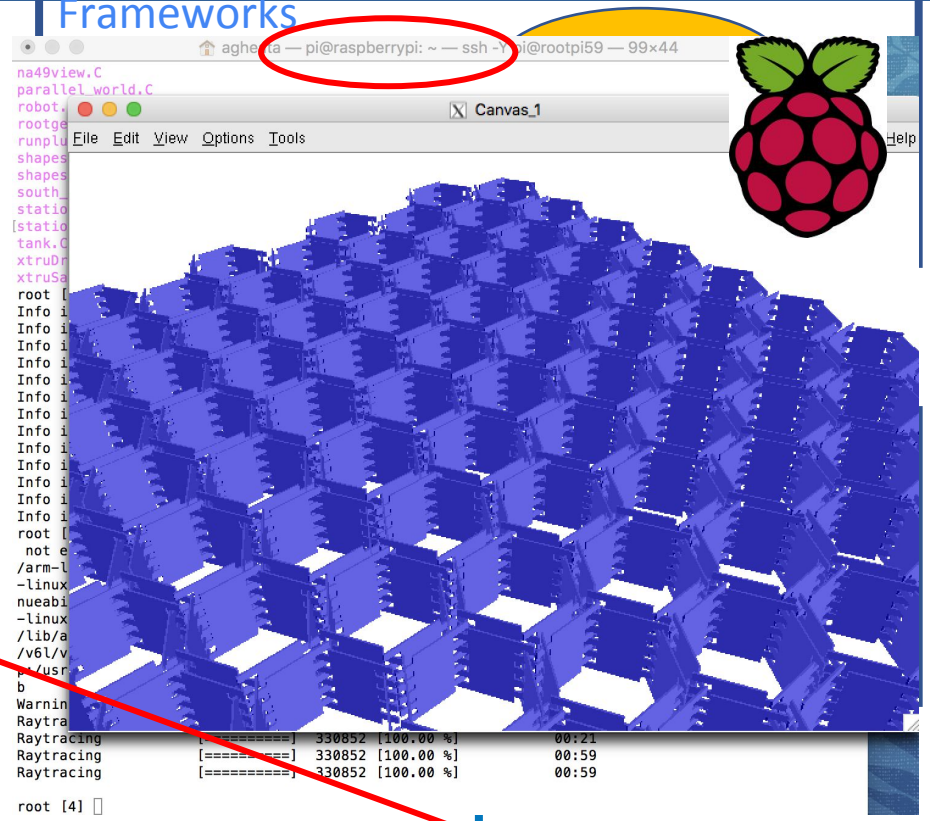

Persistent layer
Features
Navigation
Transient model

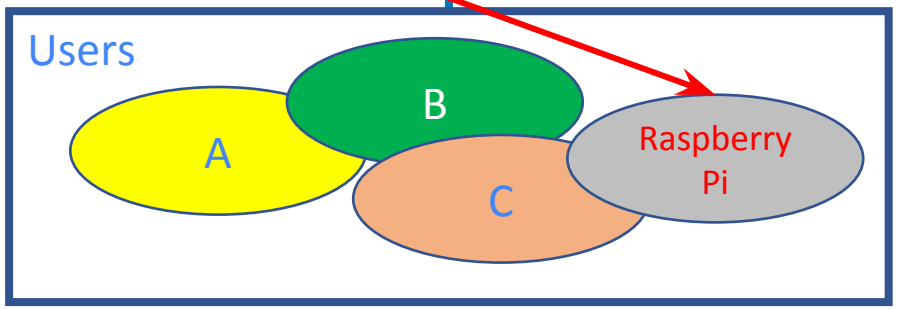

Frameworks
DD4HEP
VMC
FairRoot
...
Experiments
ALICE
CBM
...
PANDA
Users
A
B
C
D

TGeo usage

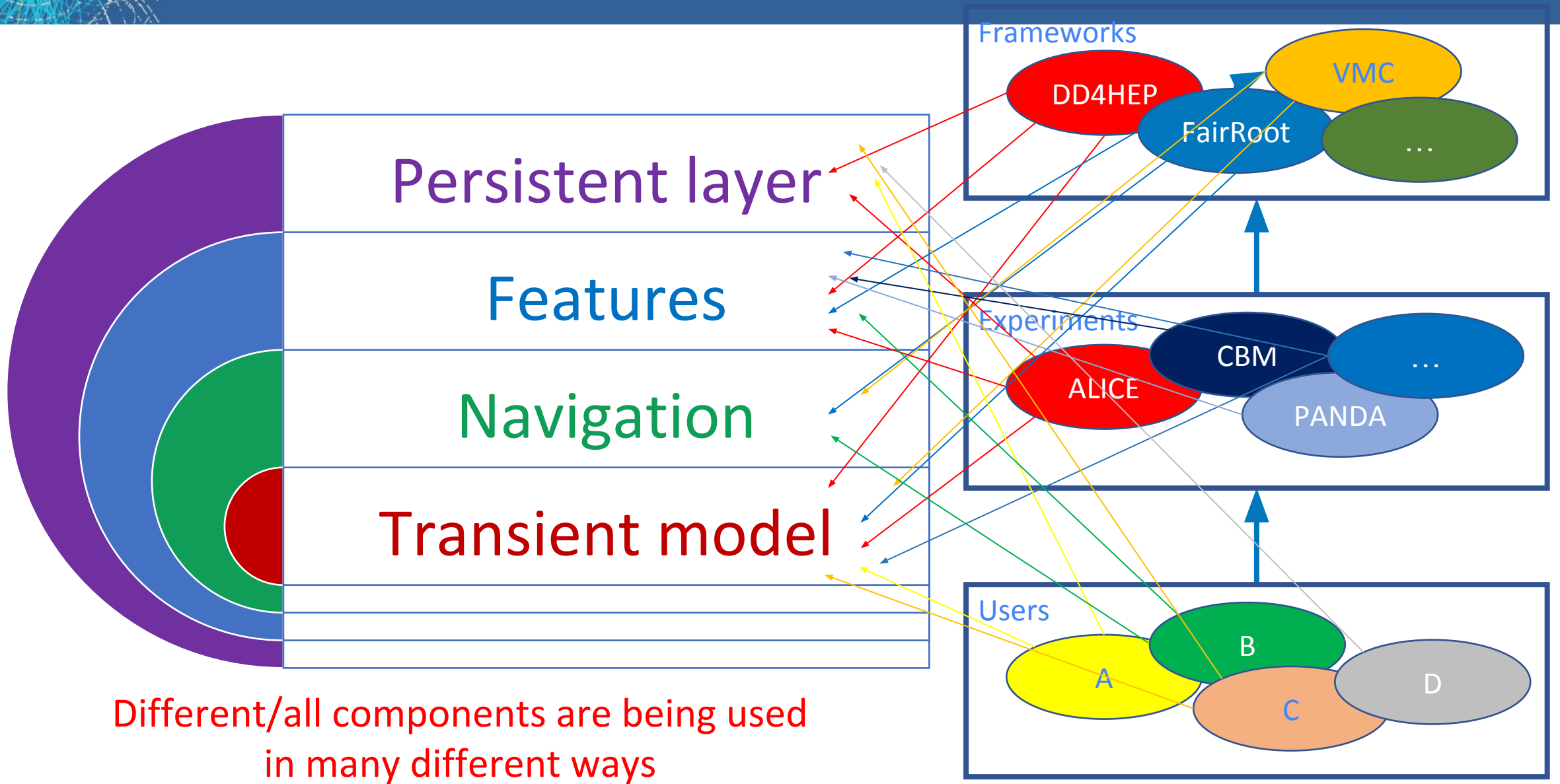


Frameworks





TGeo usage



Different/all components are being used in many different ways

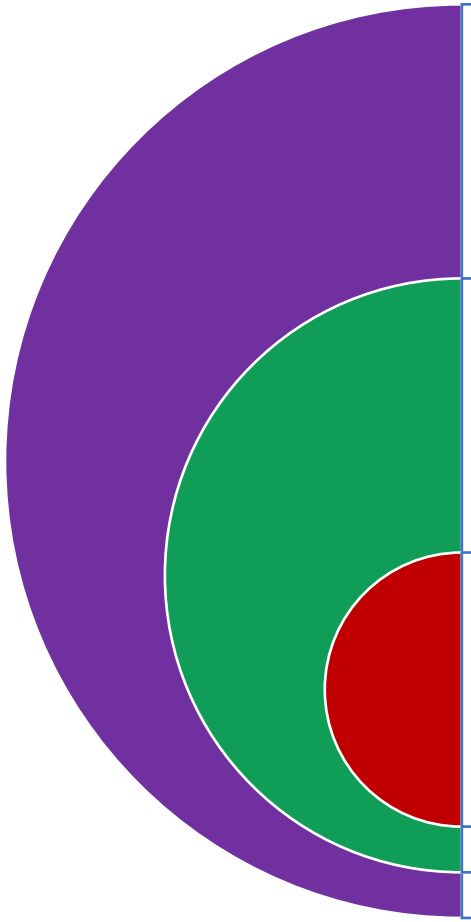
Evolution?

- Many use patterns
  - Using geometry as transient store for detector geometry, transformations, materials
  - Using special features: visualization, overlap checking, alignment - convert original geometry to TGeo, then use directly
  - Using navigation interfaces on top of TGeoNavigator (e.g. VMC), cross-check for solid navigation algorithms, ray-tracers (single or multi-threaded)
- ROOT geometry will have to stay for a while…
  - In the context of geometry components being improved outside ROOT
- The package needs to evolve
  - Stable API - future changes should preserve it as much as possible

VecGeom

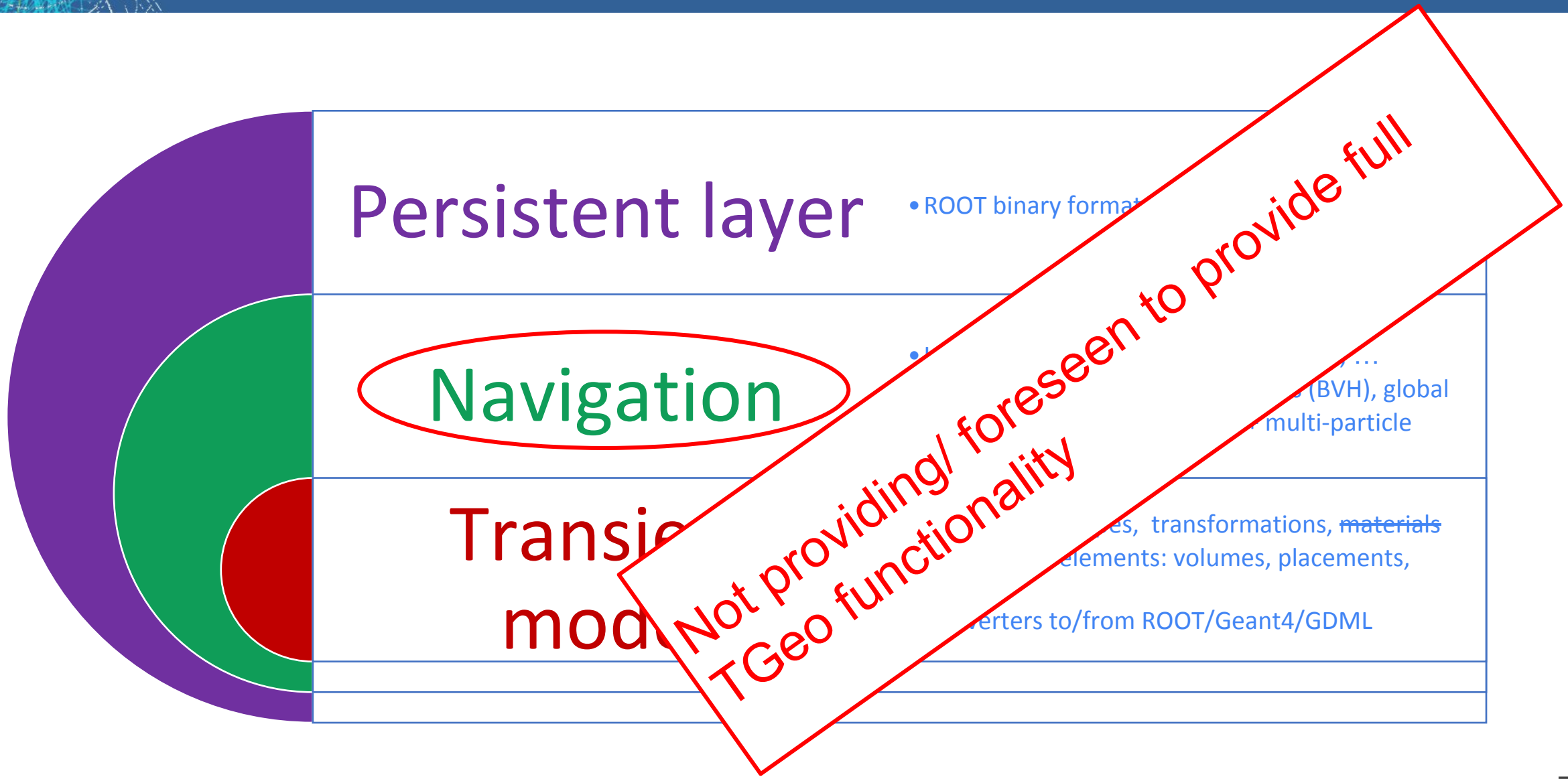
- Project for modernization of geometry algorithms (extension of USolids)
  - Stateless navigation, support for both scalar/vector API, multi-architecture, multi-platform support
  - Now a project with v1.0.0 released this June: integrated with Geant4 (solids level), used in production by CMS
- Better performance than ROOT/Geant4 in both scalar and multi-particle mode
  - Intended to replace the navigation functionality of both Geant4 and ROOT
  - Interfacing already partially done in Geant4
    - Wrapper classes as typedef to native Geant4 solids, calling VecGeom ones
    - Work on enabling VecGeom global navigation behind G4VNavigator ongoing (S. Wenzel)

VecGeom components



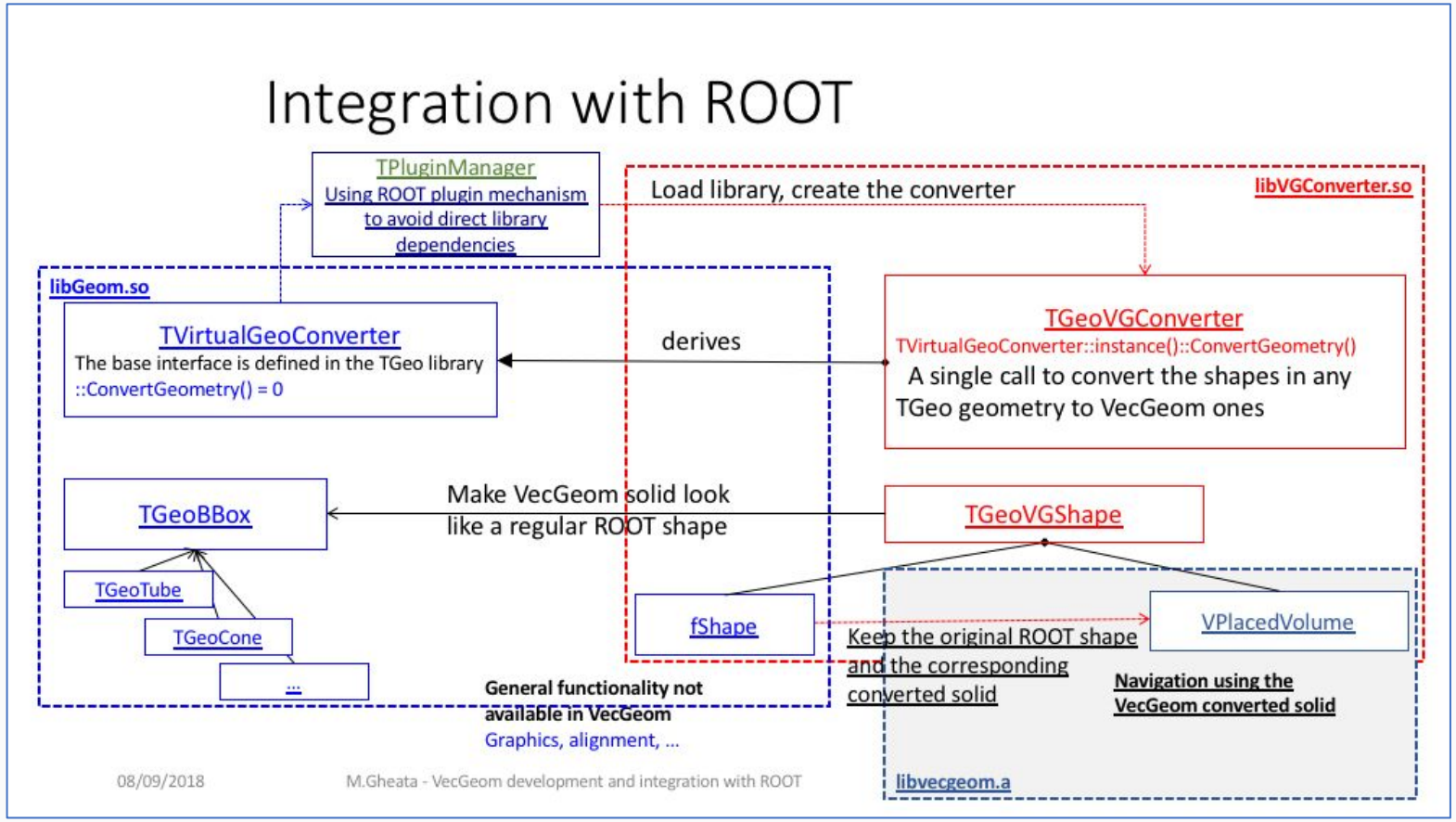
Persistent layer	• ROOT binary format, GDML (ongoing work)
Navigation	• Localizing points, computing distances, … • Local (solid) algorithms, optimizers (BVH), global • Intrinsic MT (no state), scalar + multi-particle
Transient model	• Basic types: shapes, transformations, ~~materials~~ • Hierarchical elements: volumes, placements, ~~divisions~~ • Converters to/from ROOT/Geant4/GDML

VecGeom components



Integrating VecGeom in TGeo - solids

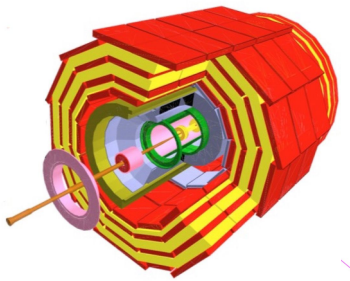
- How to enable VecGeom behind TGeo classes?
  - A first integration of VecGeom solids behind TGeoShape interfaces already done (TGeoVGShape)
  - Still one additional virtual layer & global TGeo navigator
- Final goal: achieving full VecGeom global navigation performance but preserve TGeo API

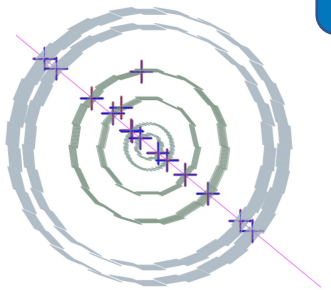


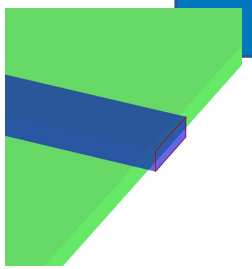
Mix and match

- Using full navigation performance of VecGeom, but having access to the rest of ROOT geometry features
- Support for VecGeom solids in TGeo
  - Via conversion
  - Solids not existing in TGeo (tessellated, multi-union solids) can be wrapped
    - Work for tessellated solid started as GSoC project this summer (M.Petric et al)
- Native VecGeom navigation behind **TGeoNavigator** interface
  - Similar to approach adopted for Geant4: perform navigation using VecGeom, then synchronize TGeo state.
  - Possible to expose also the native VecGeom navigation interfaces

VecGeom navigation behind the scene







Input
(XML/GDML/root/converters

TGeoManager

vecgeom::
GeoManager

Visualization

Alignment

Checker

Navigation

state

Navigation

TGeoNavigator<VecGeomImpl>

Price to pay: 2 geometries in memory,
but w/o ROOT geometry voxelization

Open questions

- Other use cases in relation with VecGeom features?
  - E.g. new solids support w/o navigation
- Other features besides VecGeom integration?
- Flaws/issues when used by external frameworks
  - E.g. concurrency
- Time scale, contributions?
- …