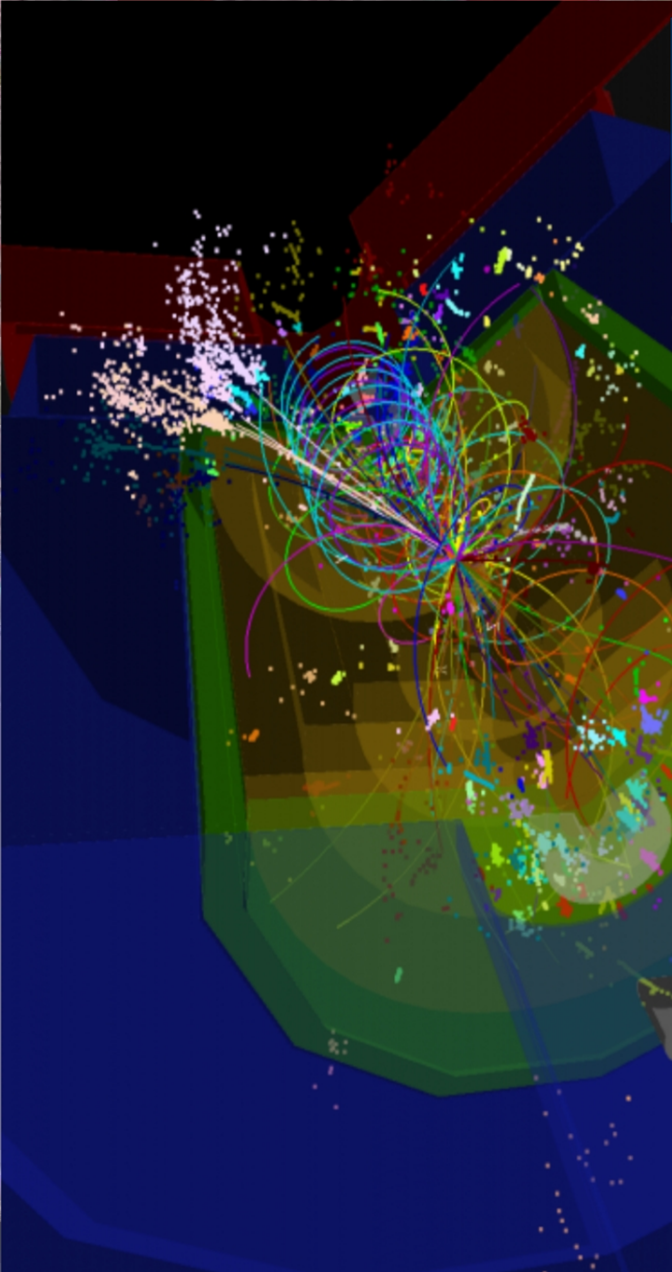


DD4hep

Recent Developments and Experience with DD4hep

- Introduction
- Simulation
- Conditions
- Alignment
- Miscellaneous topics

M.Frank, F.Gaede, M.Petric, A.Sailer

CERN

This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement no. 654168.

AIDA 2020

Motivation and Goal

DD4hep

- **Develop a detector description**
  - **For the full experiment life cycle**
    - **detector concept development, optimization**
    - **detector construction and operation**
    - **“Anticipate the unforeseen”**
  - **Consistent description, single source, supporting**
    - **simulation, reconstruction, analysis**
  - **Full description, including**
    - **Geometry, readout, alignment, calibration etc.**

This project has received funding from the European Union’s Horizon 2020 Research and Innovation programme under Grant Agreement no. 654168.

AIDA 2020

Philosophy of DD4hep & Co

DD4hep

- **Effort of very few people with a simple, humble and comprehensive vision**

Detector description for the lazy
Minimal effort, pragmatic, no technical restrictions,
No obstacles induced by religious wars

- **DD4hep is the "glue"**
  - **Bring together what belongs together:**
    Detector structure, geometry, simulation, conditions, etc
  - **Reuse existing modules: TGeo, Geant4, Assimp, etc**
  - **CAD support**

Main Entities

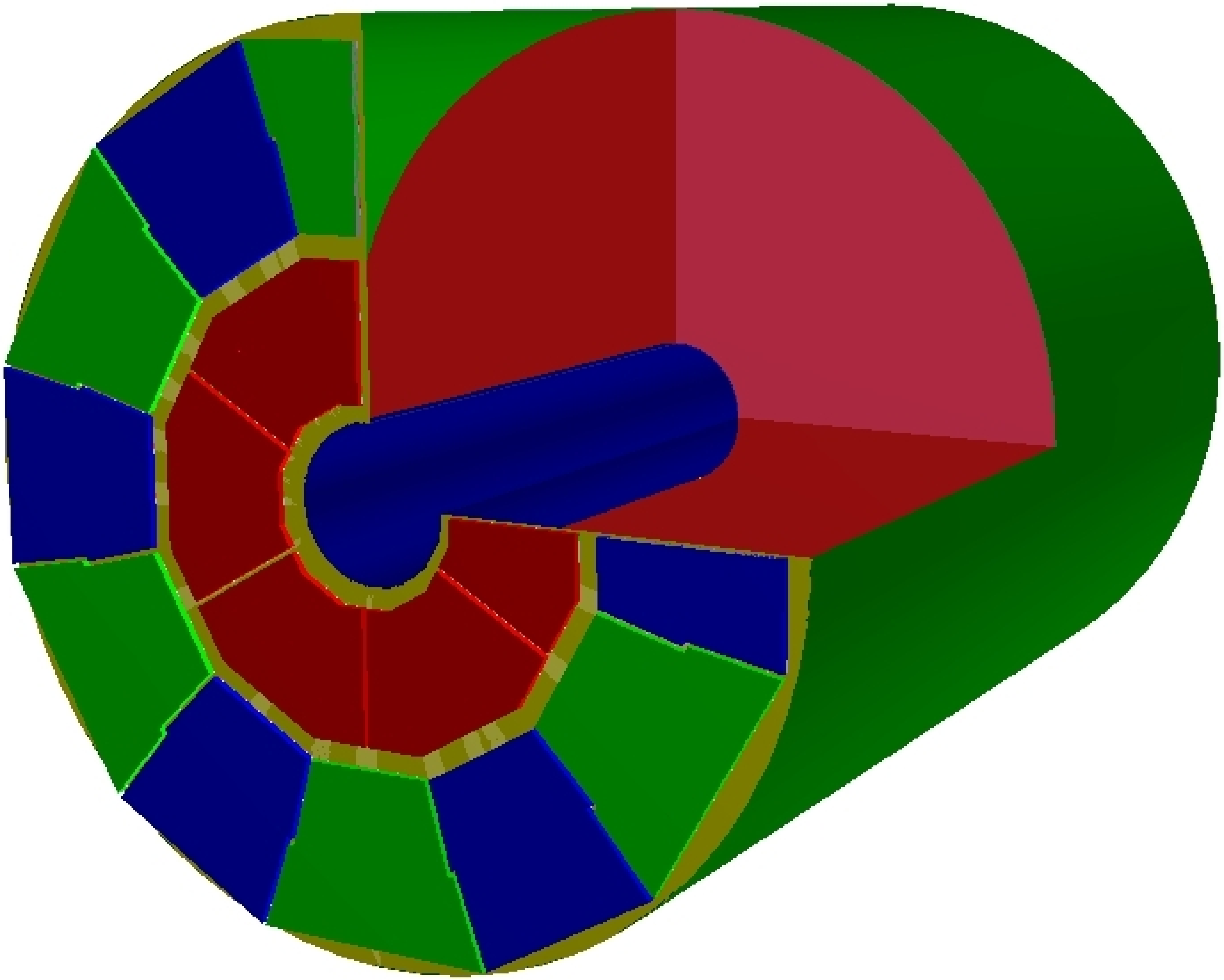
Detector description is not only geometry!

DD4hep

- **Geometrical hierarchy**
  - Volume: Shape + material
  - PlacedVolume Volume + placement matrix → mother
- **Structural hierarchy**
  - Detector Experiment
  - DetElement Parts of the experiment

- **What is the difference between geometrical and structural hierarchy**

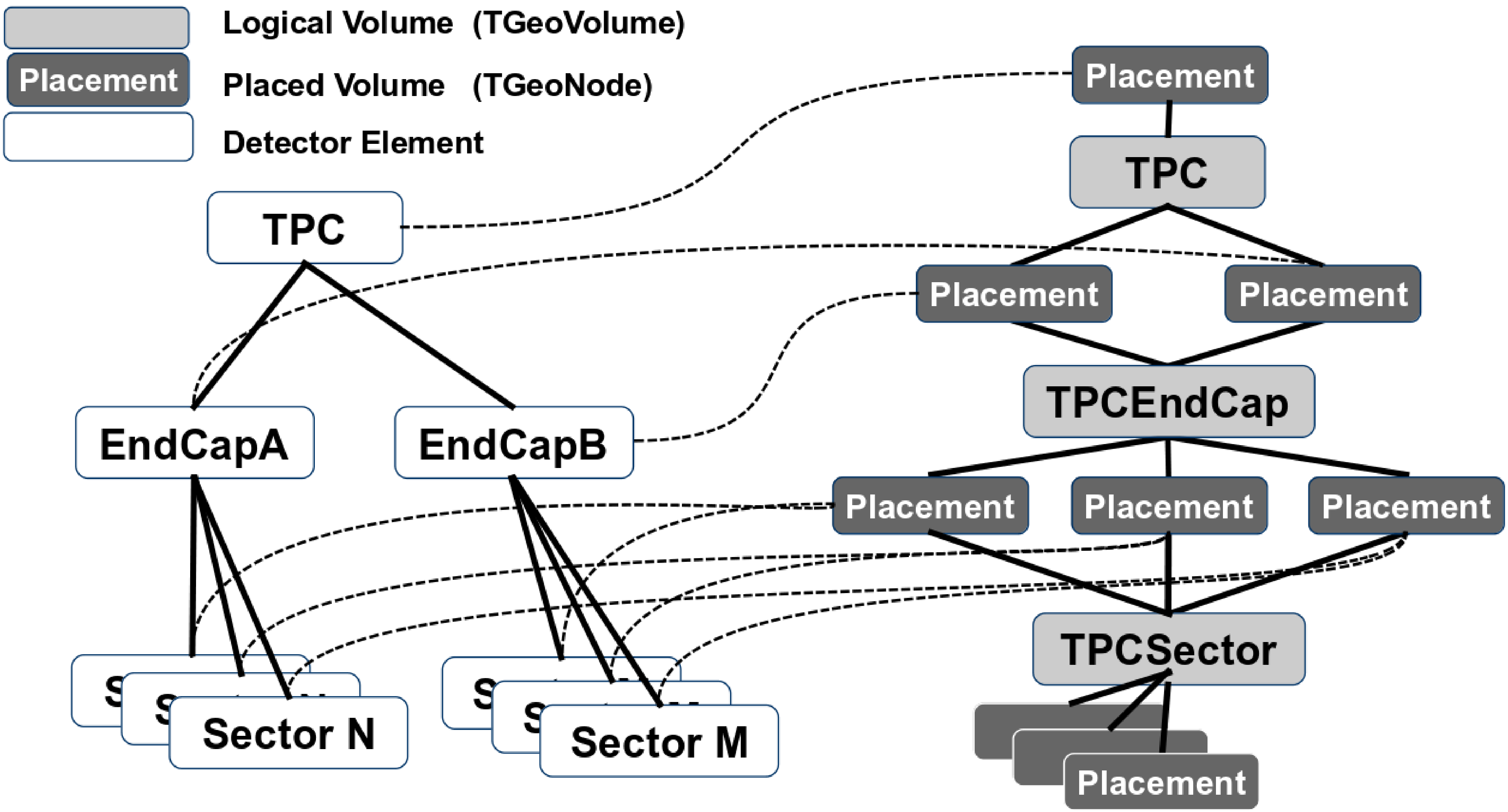
Example: ALEPH TPC



$> geoDisplay -input examples/AlignDet/compact/AlephTPC.xml

Structural and Geometrical Hierarchy

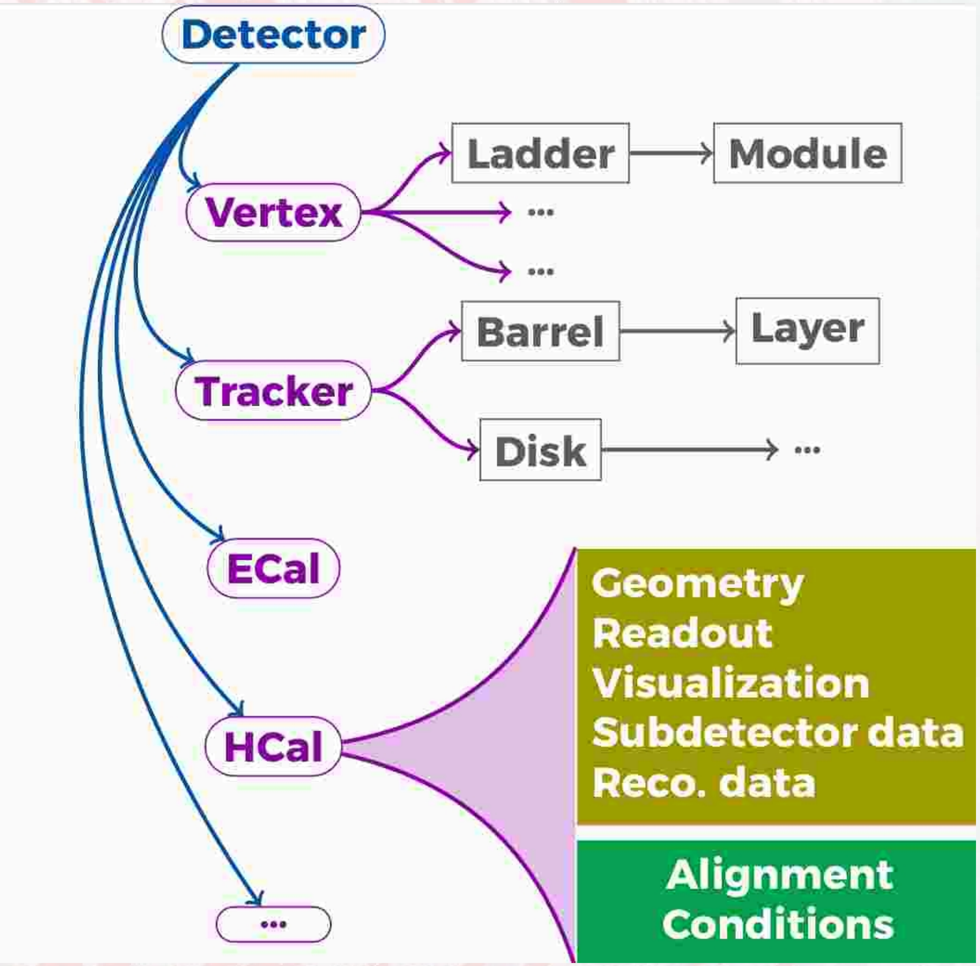
DD4hep



What is Detector Description ?

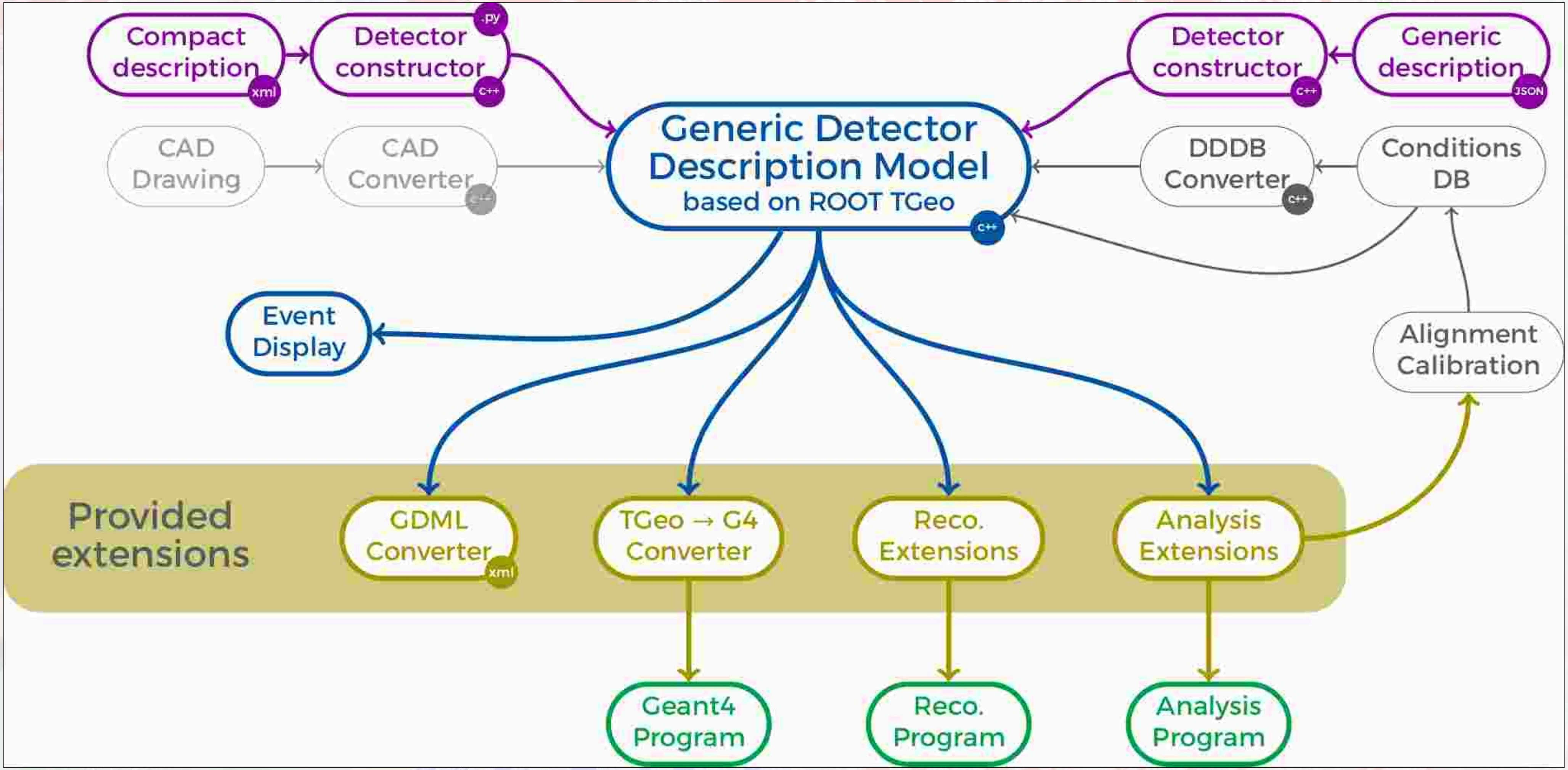
DD4hep

- Tree-like hierarchy of "detector elements"
  - Macroscopic (ie. not a strip)
  - Subdetectors or parts of subdetectors
- Detector Element
  - Geometry
  - Key to access
    - Environmental data
    - Alignment data
    - Derivatives of these
  - Optionally experiment, sub-detector or activity specific data



DD4Hep - The Big Picture

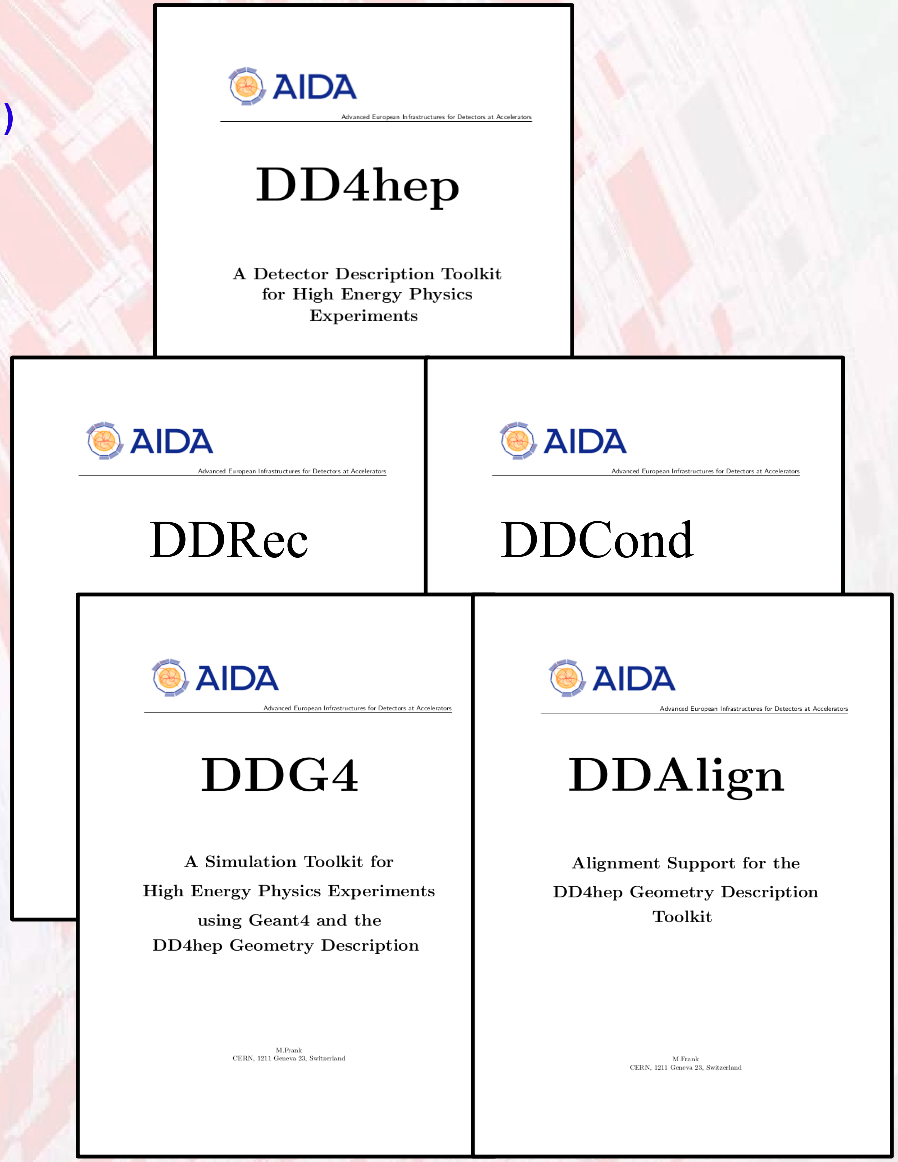
DD4hep



Saga in 5 Episodes

DD4hep

- DD4hep – basics/core (1)
- DDG4 – Simulation using Geant4 (1)
  - Fast simulation (4)
- DDRec – Reconstruction supp.(2)
- DDCond – Detector conditions (3)
- DDAlign – Alignment support (3)
- DDigi – Generic Digitization (4)


AIDA
Advanced European Infrastructures for Detectors at Accelerators

DD4hep

A Detector Description Toolkit for High Energy Physics Experiments

AIDA
Advanced European Infrastructures for Detectors at Accelerators

DDRec

AIDA
Advanced European Infrastructures for Detectors at Accelerators

DDCond

AIDA
Advanced European Infrastructures for Detectors at Accelerators

DDG4

A Simulation Toolkit for High Energy Physics Experiments using Geant4 and the DD4hep Geometry Description

M.Frank
CERN, 1211 Geneva 23, Switzerland

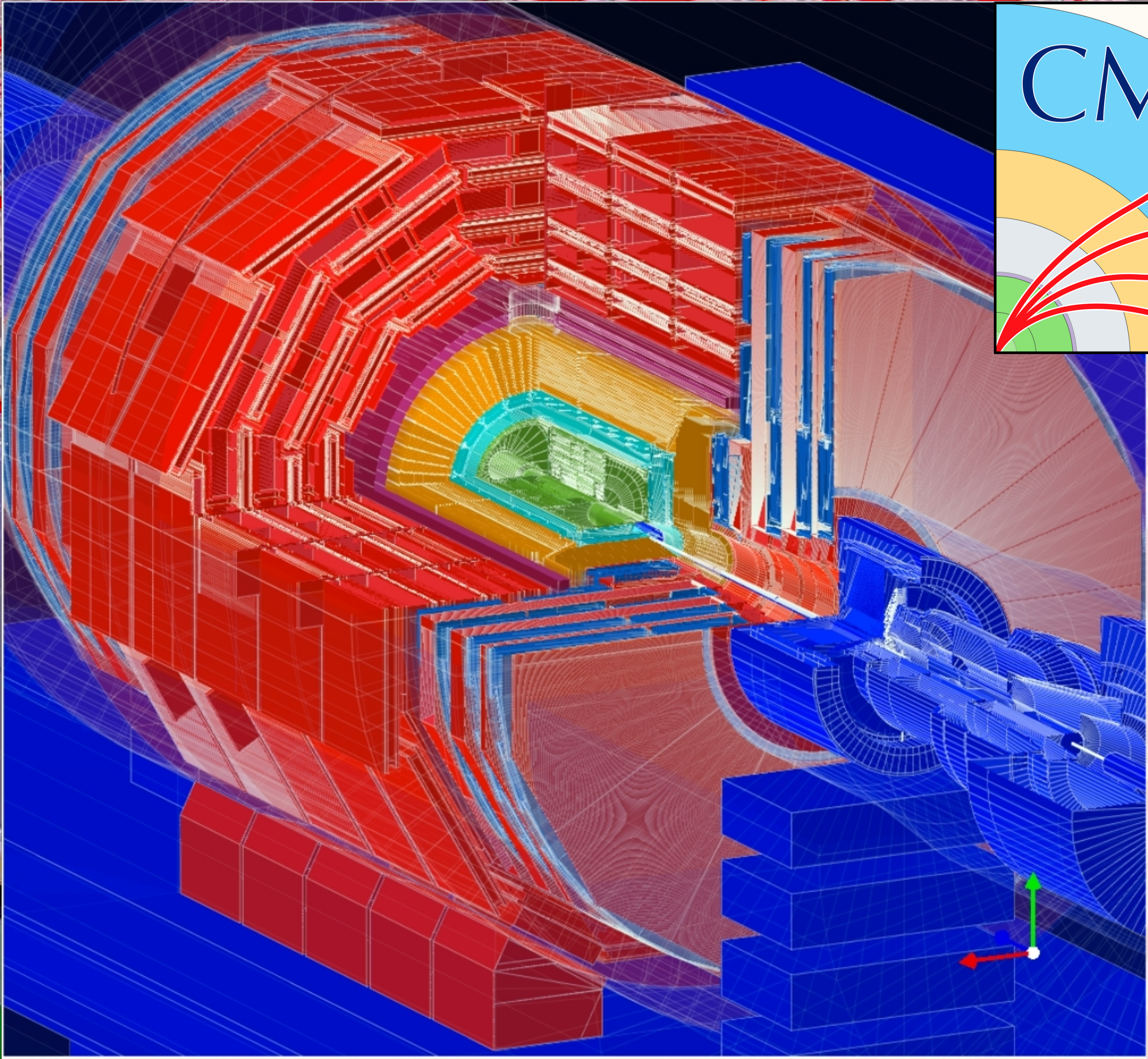
AIDA
Advanced European Infrastructures for Detectors at Accelerators

DDAlign

Alignment Support for the DD4hep Geometry Description Toolkit

M.Frank
CERN, 1211 Geneva 23, Switzerland

(1) Mature state: bug-fixes and maintenance
(2) F. Gaede (WP3, Task 3.6)
(3) Work since start of AIDA2020
(4) Planned extensions



CMS described with DD4hep

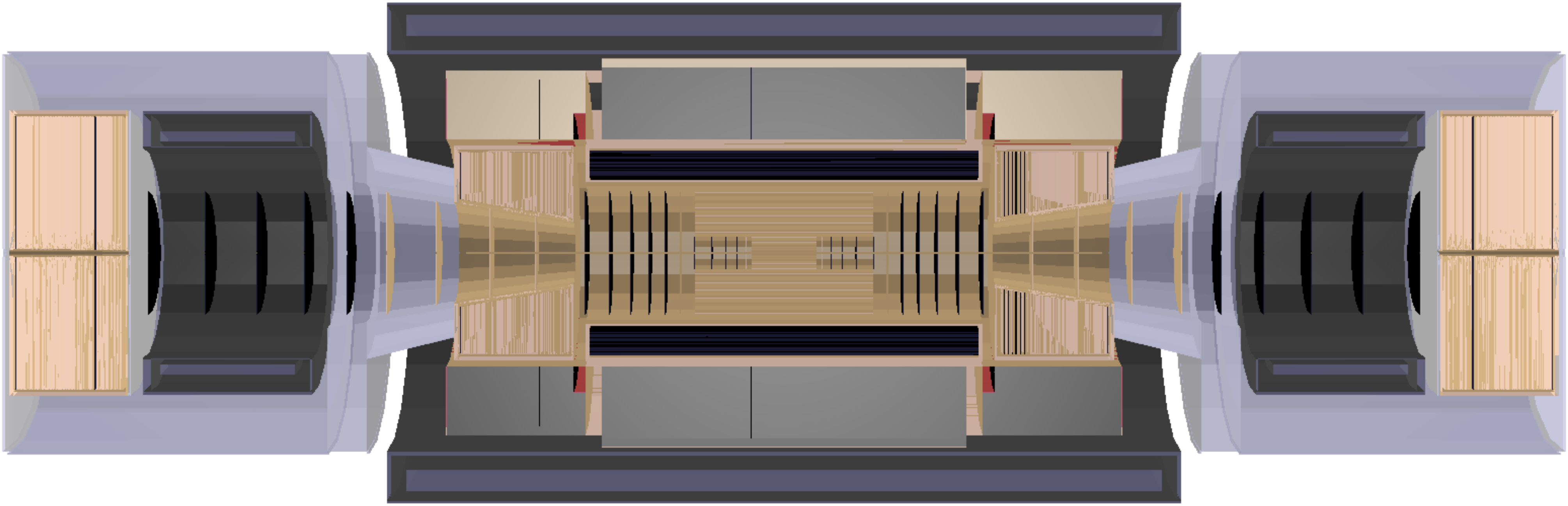
DD4hep baseline for Run3

CHEP 2019, Adelaide, AU

(C.Vuosalo / CMS)

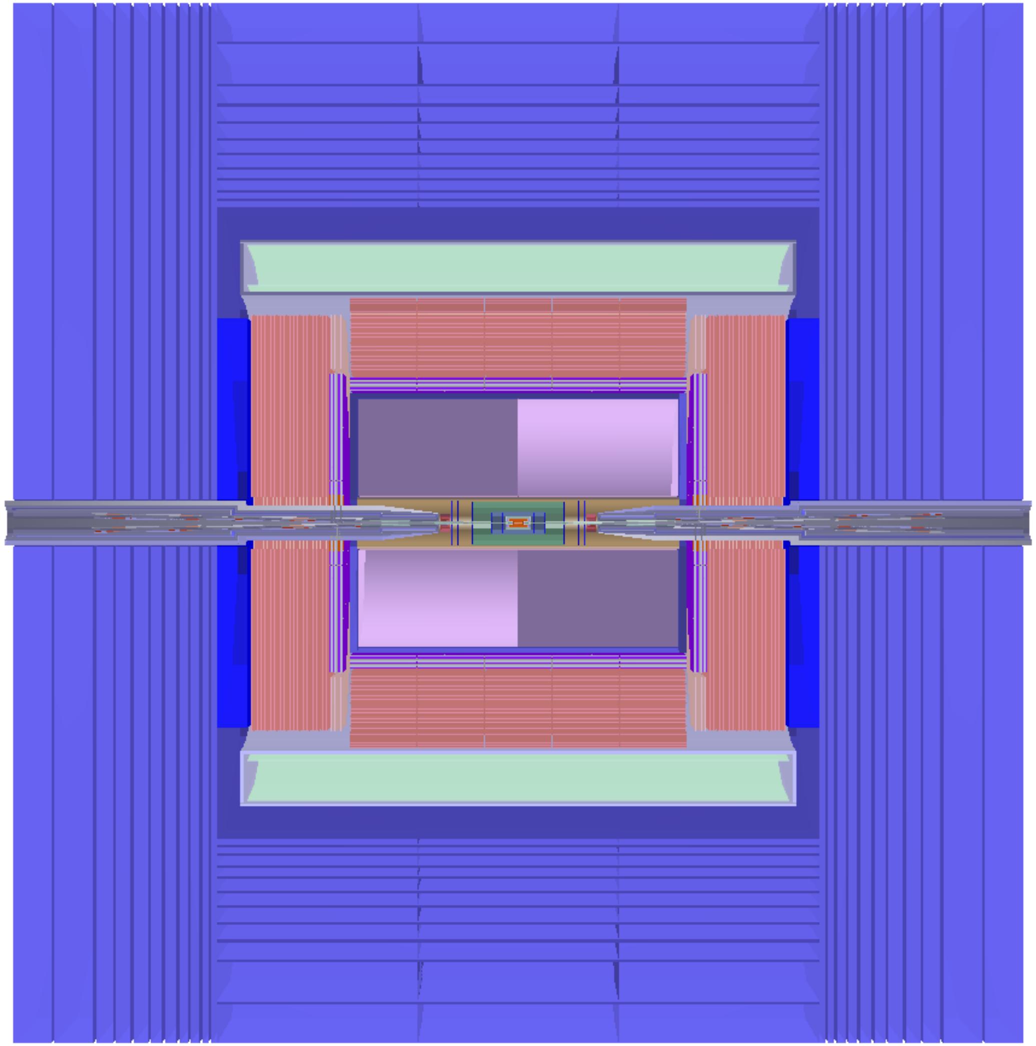
FCC Design Study

DD4hep



Circular Electron Positron Collider

DD4hep



DD4hep Core

DD4hep

- Handles the detector element functionality
- Basically stable
  - Bug fixes, enhancements
- Objects are fully reflective
  - C++ dictionary defined
  - Intrinsic support for cross-language development
- Reflection supports interactivity
  - Cint (Cling) and python (cppyy)
- CHEP 2013

DD4hep: A Detector Description Toolkit for High Energy Physics Experiments

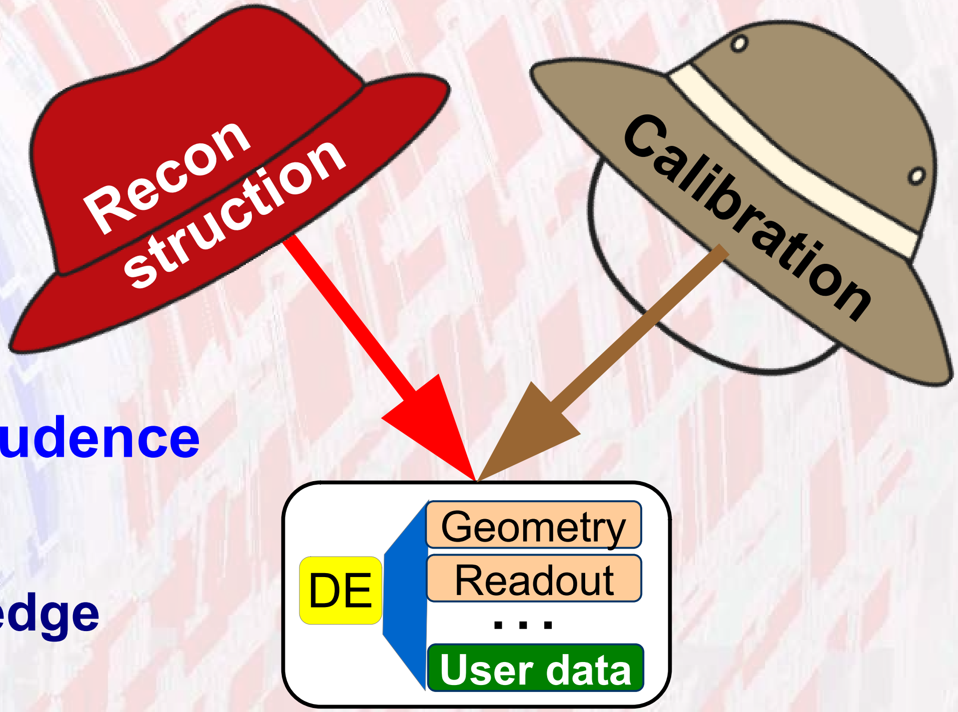
Views & Extensions:

Users Customize Functionality

DD4hep

DD4hep is based on handles (smart pointers)

- Rarely deal with data directly
- Possibility of many views based on the same DE data
  - Same 'data' associated to different 'behaviors'
  - All views are consistent and creation is efficient: pointer-copy
- User data to be used with prudence
  - Blessing and a curse
  - User data: common knowledge

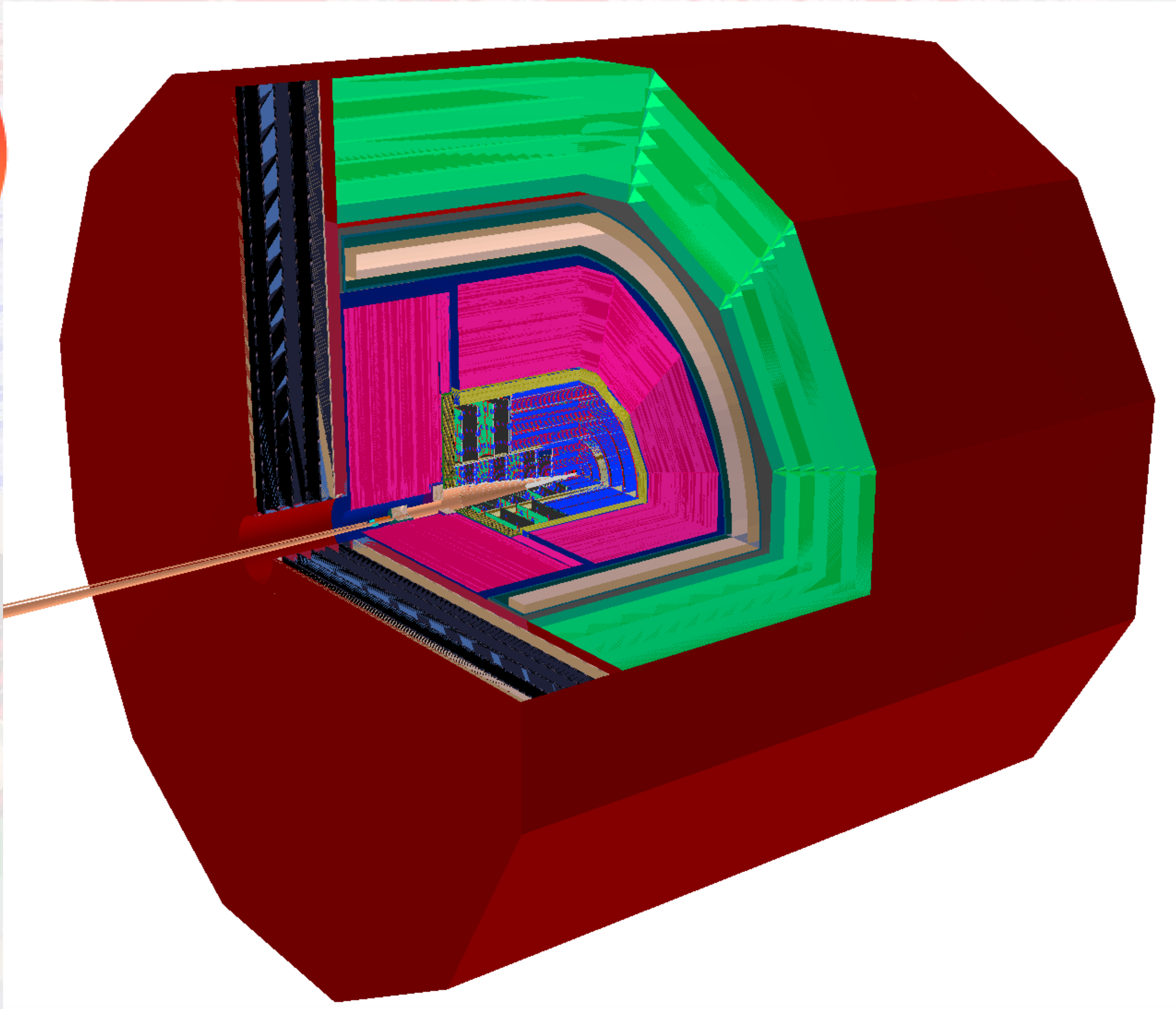


Standard Detector Palette DDDetectors

DD4hep

- Used for design studies (LC, FCC-eh)
- Origin from the SiD detector model
  - Layer based detectors
  - Tracker barrel & endcap
  - Several calorimeter constructs
- Partially with measurement surfaces (F. Gaede)
  - Uses plugin mechanism to enhance detector elements
    - Mechanism to attach user defined optional data
      => Proof that 'anticipate the unforeseen' works
    - NOT intrusive to detector constructors

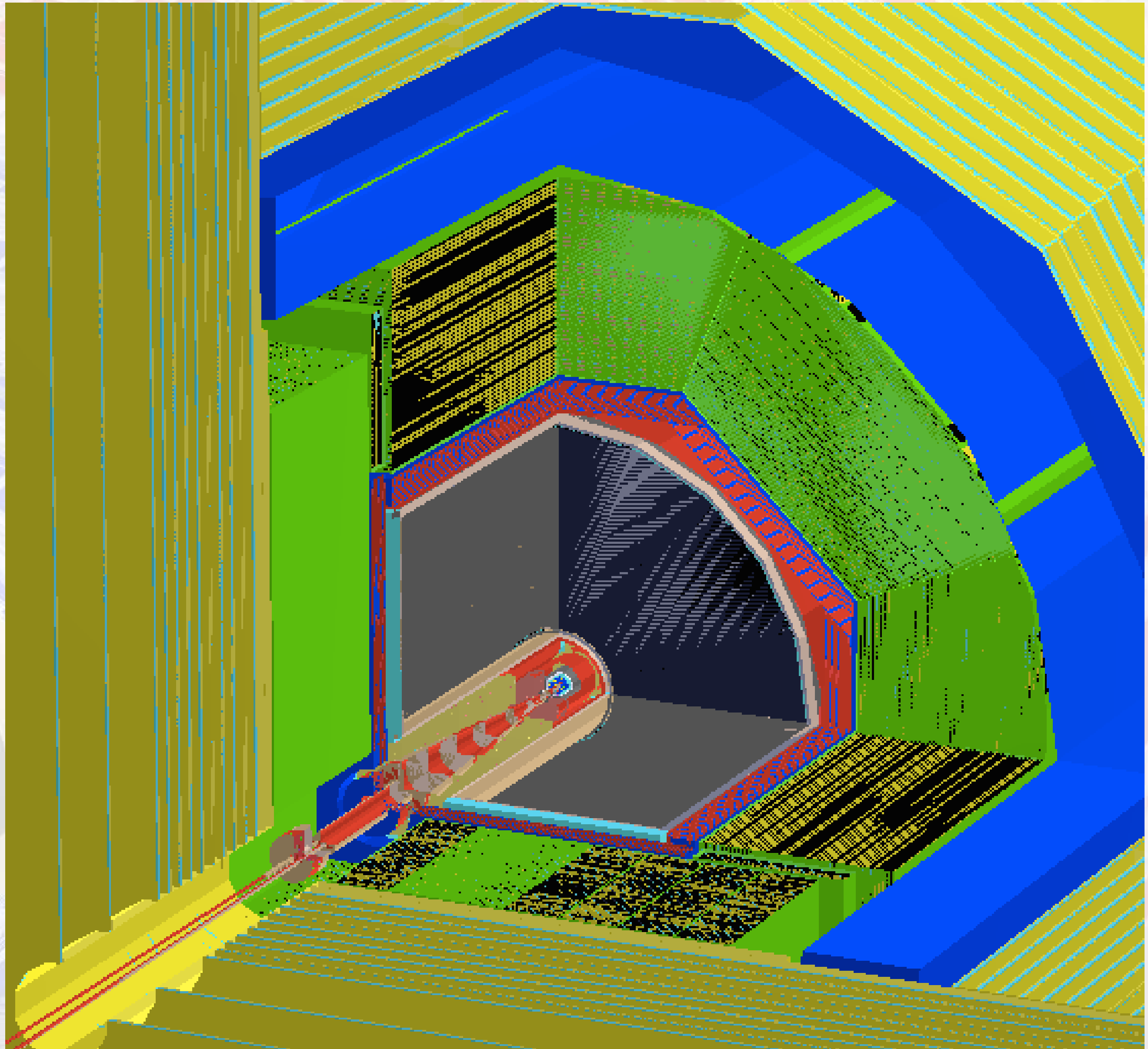
CLICdp CLIC Detector Project



International Linear Collider

(F.Gaede, L.Shaojun)

DD4hep



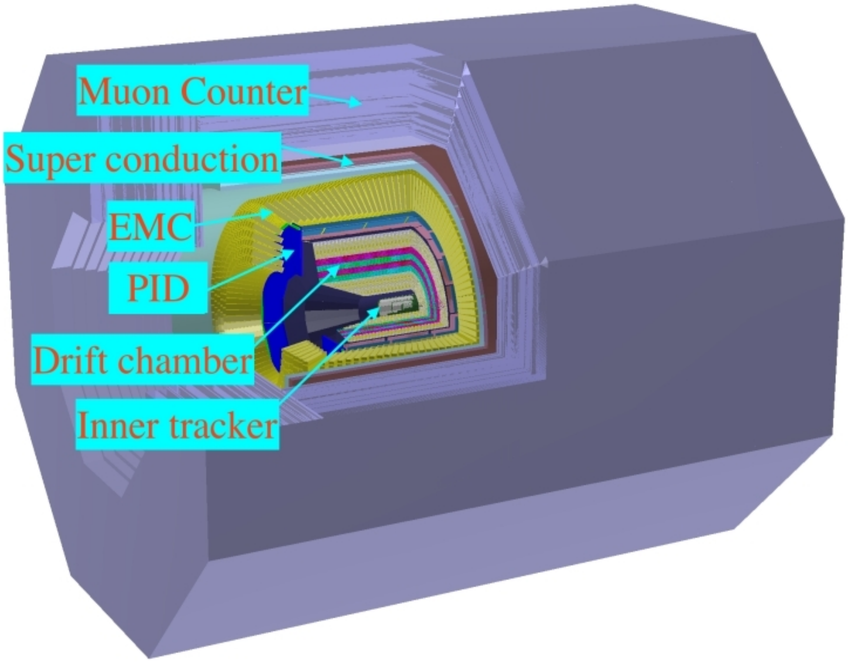
Xiaorong Zhou
State Key Laboratory of Particle Detection and Electronics
University of Science and Technology of China

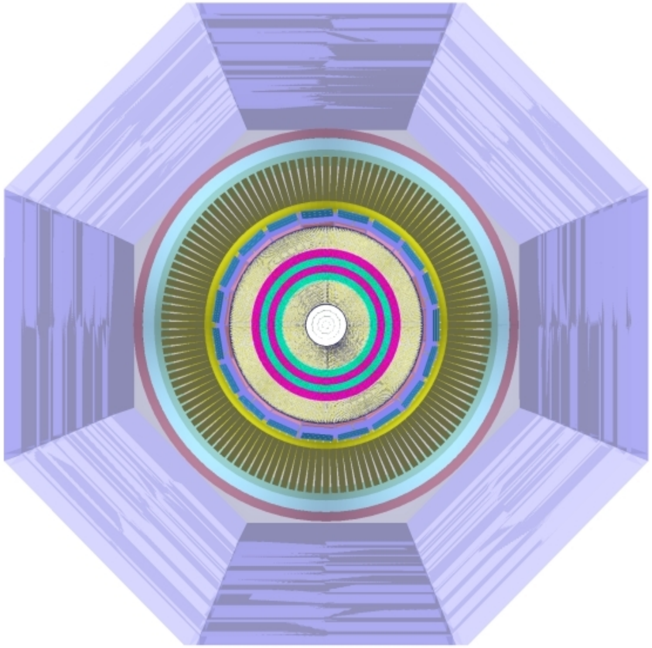
Joint Workshop on Future Tau-Charm Factory
2018.12.4-2018.12-7, Paris

DD4hep

Progress on detector simulation

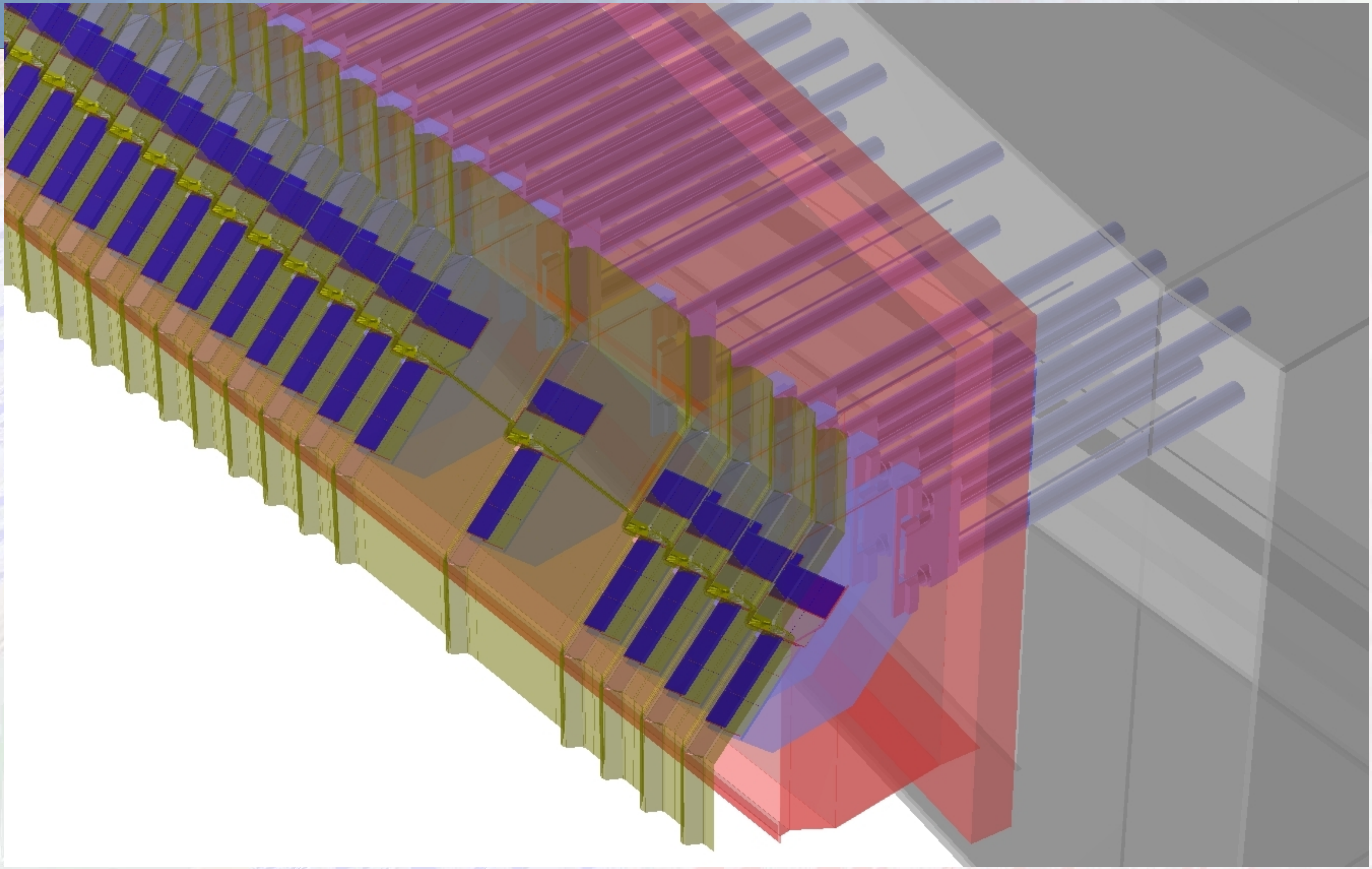
- STCF software team has been formed.
- OSCAR: Offline Software of Super Tau-Charm Facility.
- Detector geometry with DD4hep.





18

LHCb: Velo Pixel Single Side



Simulation: DDG4

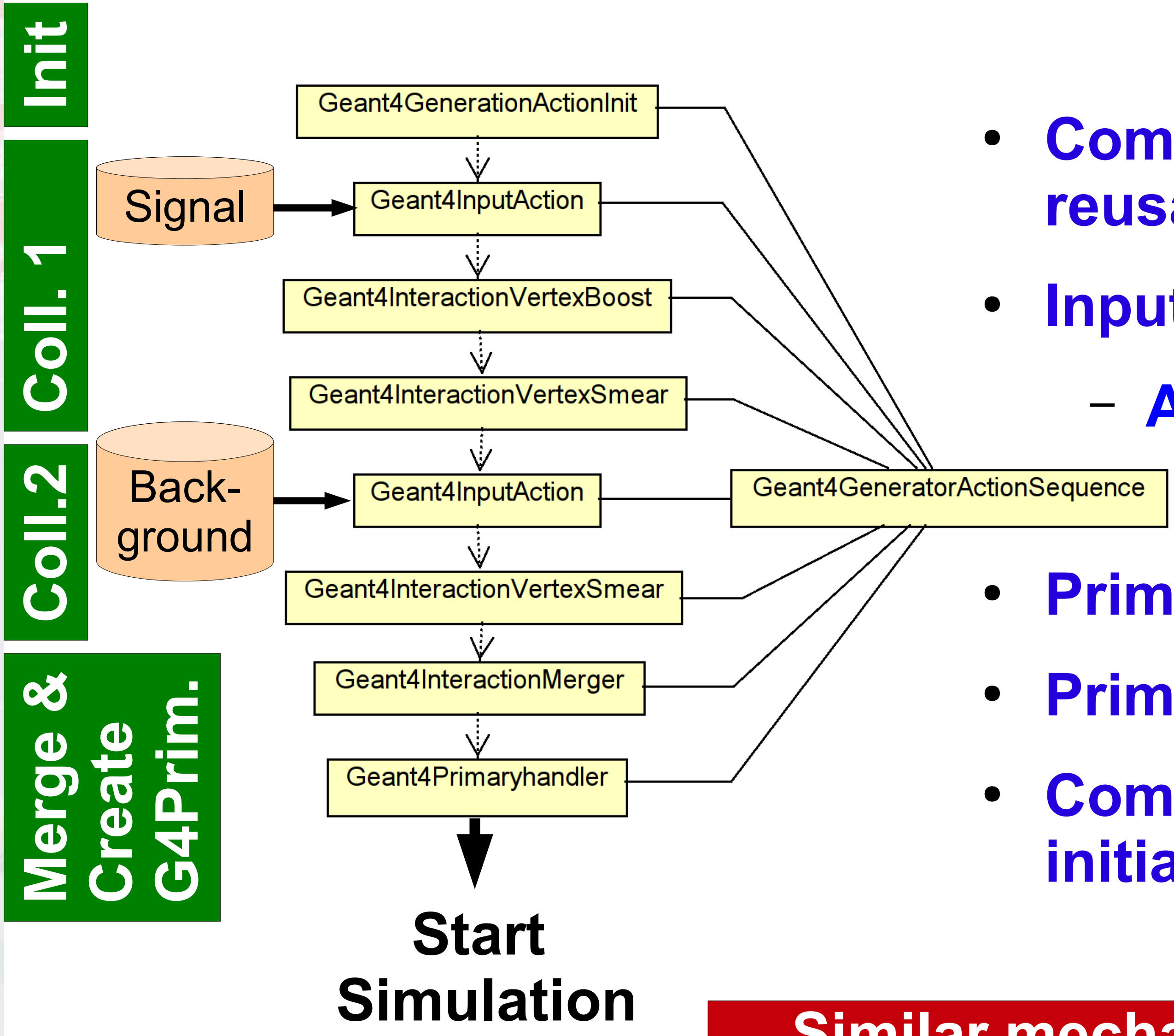
DD4hep

- **Simulation = Geometry +**
  Detector response +
  Physics
- **Mature status**
  - **Eventual bug fixes, smaller improvements**
  - **Phase of constant re-validation**
- **Automatic geometry conversion**
- **Extensive use of plugin mechanism → configuration**
- **Palette of standard sensitive detectors**
- **Support for MC truth handling**
- **CHEP2015**

DDG4 A Simulation Framework based on the DD4hep Detector Description Toolkit

Example of an Action Sequence

Generator Action: Event Overlay with Features

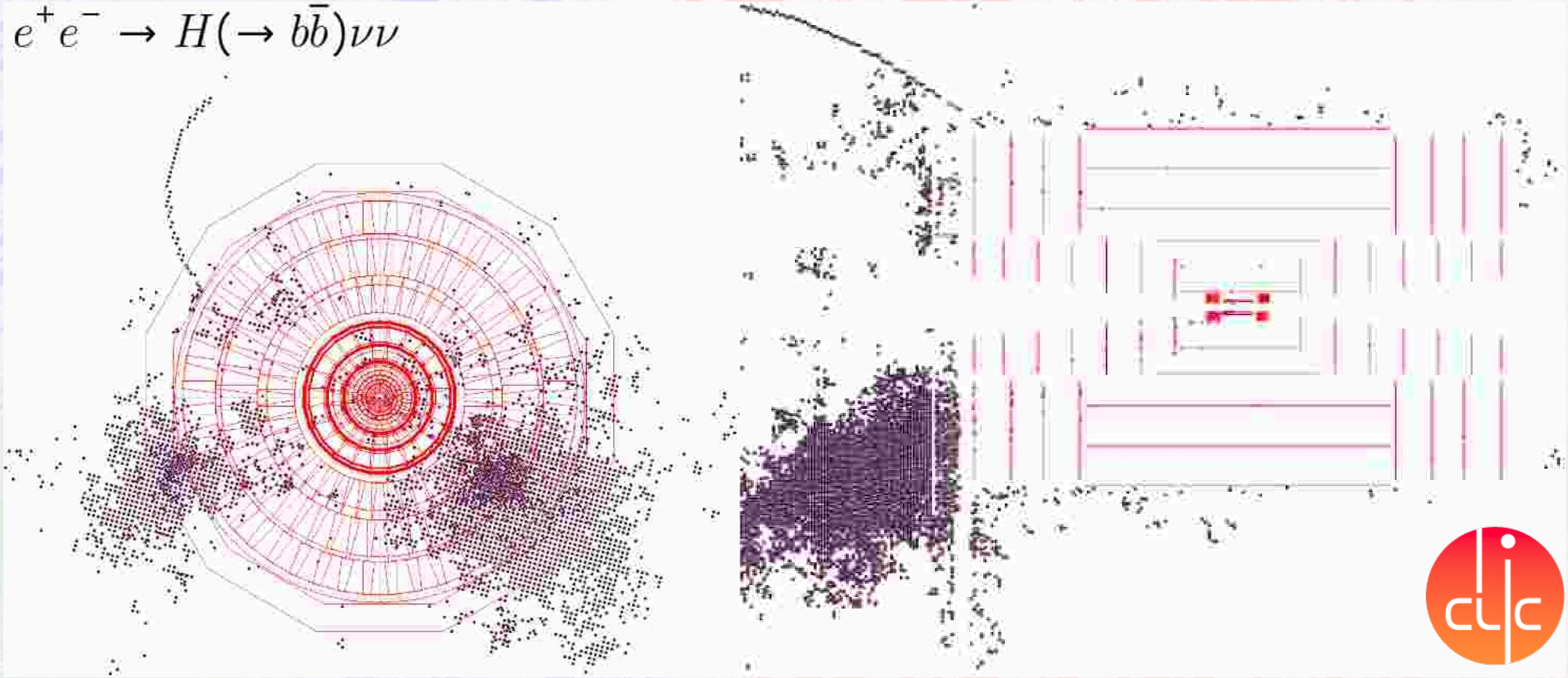


- **Combine simple and reusable modules**
- **Input module**
  - **Any data format**
- **Primary vertex smearing**
- **Primary vertex boost**
- **Common: initialization, final merge**

Similar mechanism for sensitve, tracking, event and run actions

DDG4 in Production

- **Deployed for CLICdp in DIRAC**
  - **For every detector study (now ~14) central generation**
- **ILC started mass production**



SCTF - Novosibirsk

L. Shekhtman, A. Sokolov, Vijayanand KV, T. Maltsev
Budker Institute of Nuclear Physics (BINP)

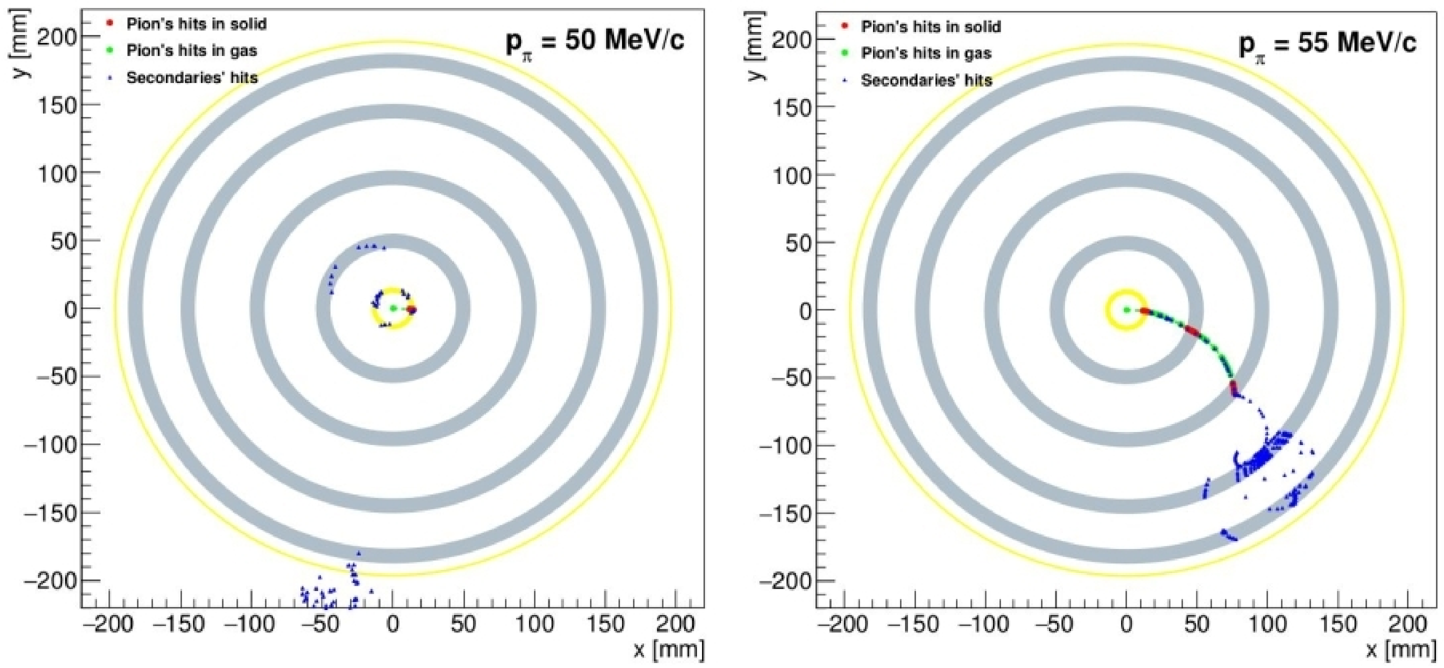
Joint Workshop on Future Tau-Charm Factory
2018.12.4-2018.12-7, Paris

DD4hep

Inner Tracker

DD4hep simulation

CGEM



- Pions with momenta less than 50 MeV/c do not pass through the beampipe
- Starting from p_π = 55 MeV/c two layers can be reached by pions

12

DDSim: DDG4 CLI

DD4hep

- **Python based command line interface to DDG4**
- **DDSim offers to the all usually used plugins of DDG4 and supports most detector models of the linear collider community**
  - **Detector description, simulation input, G4 steering, vertex offsets, mag.field setup, physics list, etc.**
  - **~ 100 command line arguments for nearly all wishes**
- **DDSim accepts python code snippets for fine grained user specialization**
- **Allows the creation of steering files**
  - **Re-produce results**
  - **Mass production**

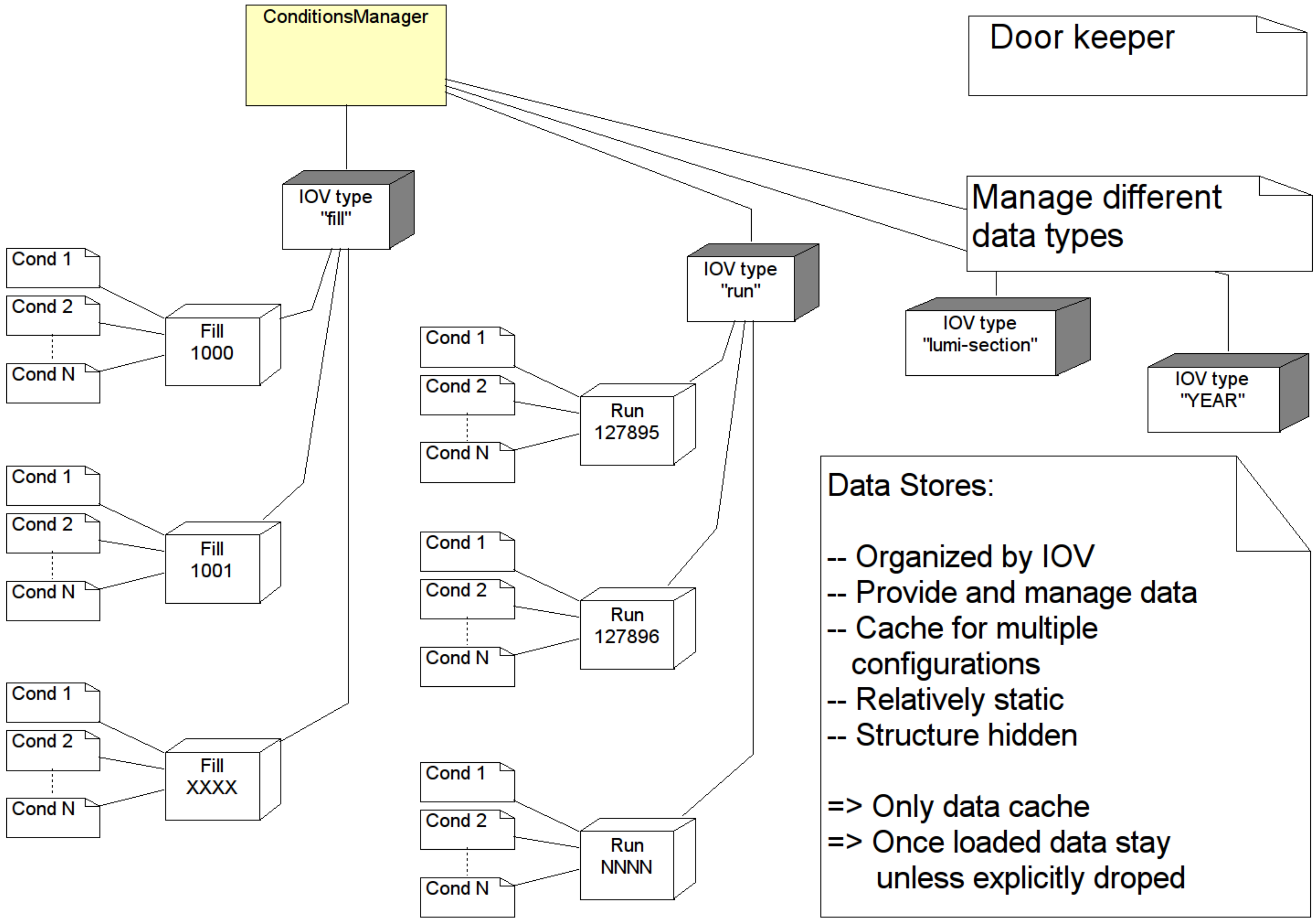
DDCond: Conditions Data

DD4hep

- Time dependent data necessary to process the detector response [of particle collisions]
  - slowly changing: every run O(1h), lumi section O(10min) ...
  - multiple conditions change in batches: require discipline
  - conditions may be the result of computation(s)
- DDCond deals with the management of these data
  - Efficient and fast, if used according to design ideas
  - Manages resources
  - Supports multi threading by design
    Well defined locking points
  - Cache where necessary but no more
- CHEP2018
  Conditions and Alignment extensions to the DD4hep Detector Description Toolkit

DDCond: Data Cache

Access key: Hash of DetElement path and condition name



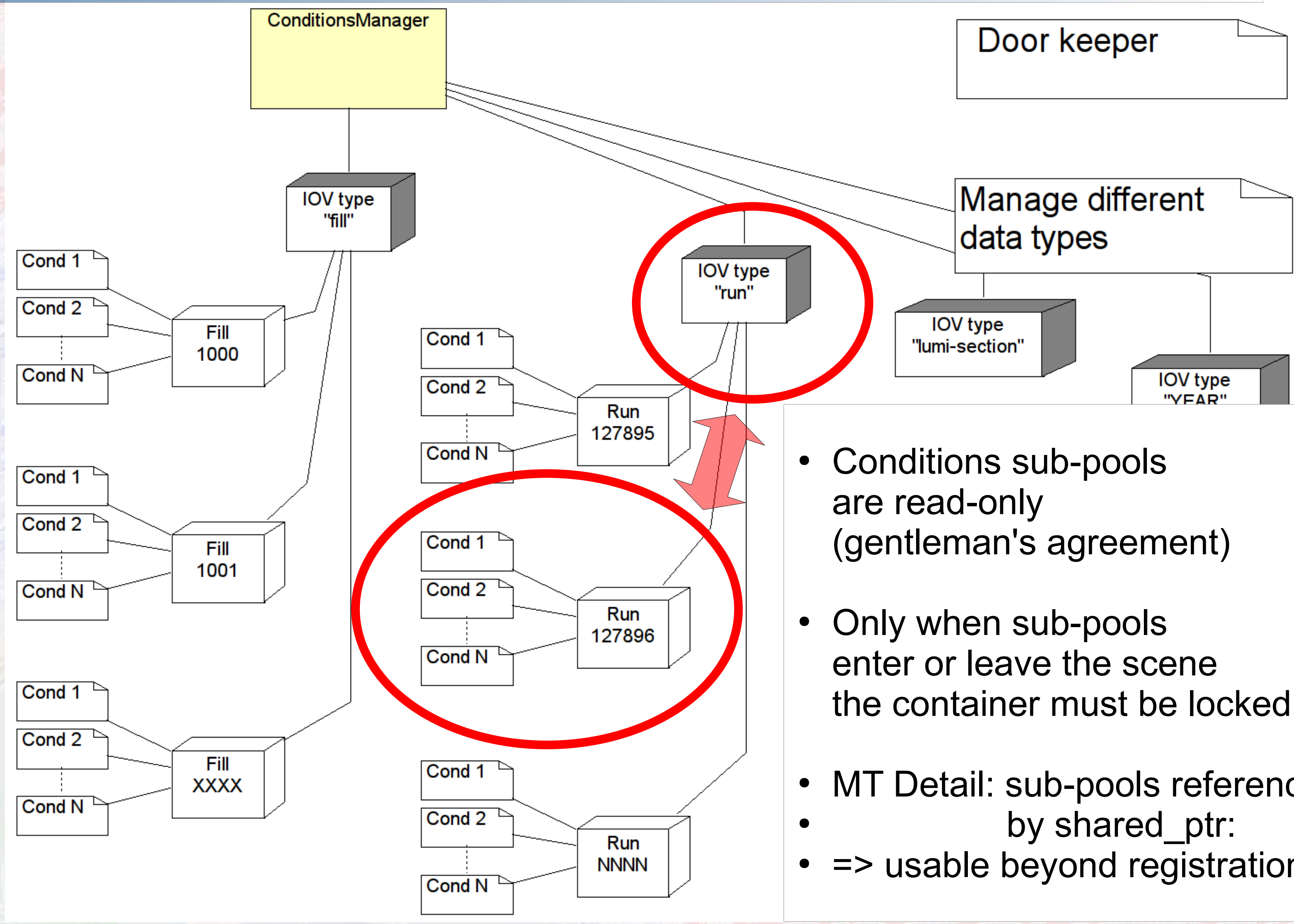
Data Stores:

-- Organized by IOV
-- Provide and manage data
-- Cache for multiple configurations
-- Relatively static
-- Structure hidden

=> Only data cache
=> Once loaded data stay unless explicitly droped

DDCond Implementation
The Data Cache

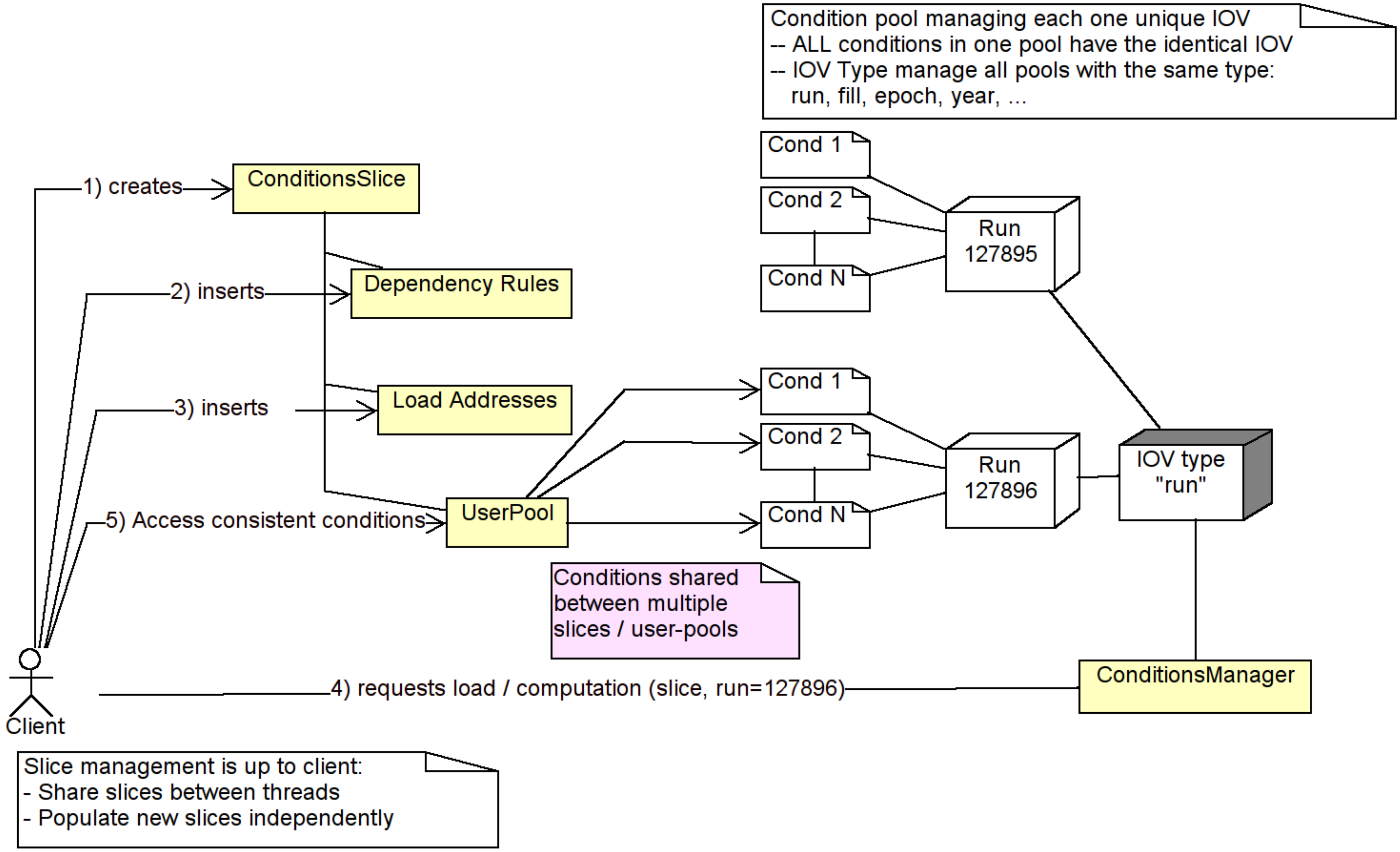
DD4hep



- Conditions sub-pools are read-only (gentleman's agreement)
- Only when sub-pools enter or leave the scene the container must be locked
- MT Detail: sub-pools referenced
- by shared_ptr:
- => usable beyond registration

DDCond Implementation
IOV Slice Projection

DD4hep



Derived Conditions

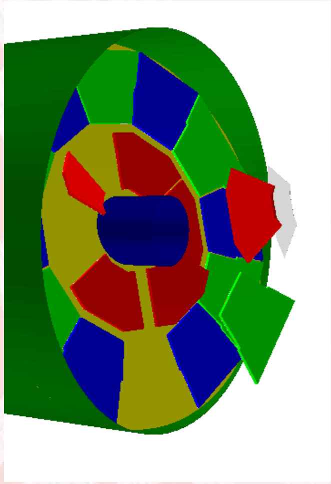
DD4hep

- Derived conditions are built using callback mechanism
  - Data computed from “measured” conditions
  - Callbacks are equivalent to persistent data addresses
- Computation is part of the slice creation
  - First load static conditions
  - Then compute dependent data
    ...and also data dependent on dependent data etc.
- Since conditions in existing pools still can be shared while preparing new IOV depending conditions
  - No locking strategy necessary
- Prime example: Alignments
  - Alignments must for efficiency be computed ‘en block’

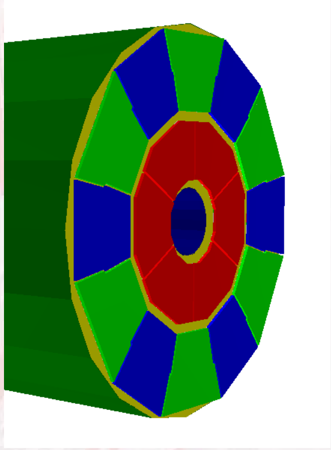
Global and Local Alignments

DD4hep

- **Global alignment corrections**
  - Physically alters geometry
    Intrinsically supported by ROOT
  - By construction not multi-threaded
  - Possibility to simulate misaligned geometries



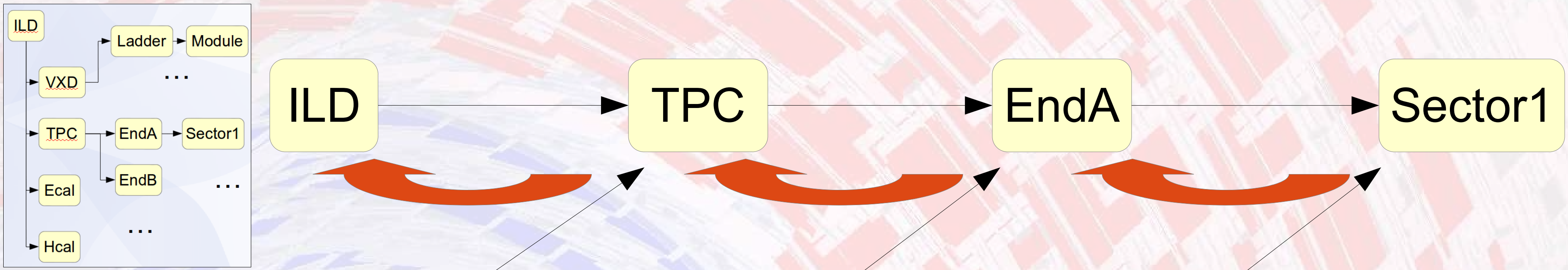
- **Local alignment corrections**
  - Geometry stays intact (either ideal or globally aligned)
  - Multi-threading supported, multiple versions
  - Local alignment corrections are conditions
  - Provide matrices from ideal geometry to world
    e.g. to adjust hit positions



- **Both supported (global with caveat)**

Local Alignment Δ - Parameters

DD4hep



$$Tr^{World}_{Sec\,1} = Tr^{World}_{EndA} \times \left(Tr^{Parent(EndA)}_{Sec\,1} + \Delta_{Sec\,1} \right)$$

$$Tr^{World}_{EndA} = Tr^{World}_{TPC} \times \left(Tr^{Parent(TPC)}_{EndA} + \Delta_{EndA} \right)$$

$$Tr^{World}_{TPC} = Tr^{World}_{ILD} \times \left(Tr^{Parent(ILD)}_{TPC} + \Delta_{TPC} \right)$$

- Trickle-up the hierarchy and compute the matrices the most effective way with re-use of intermediate results
- Math verified by AIDA^{2020} alignment task force (C.Burr)

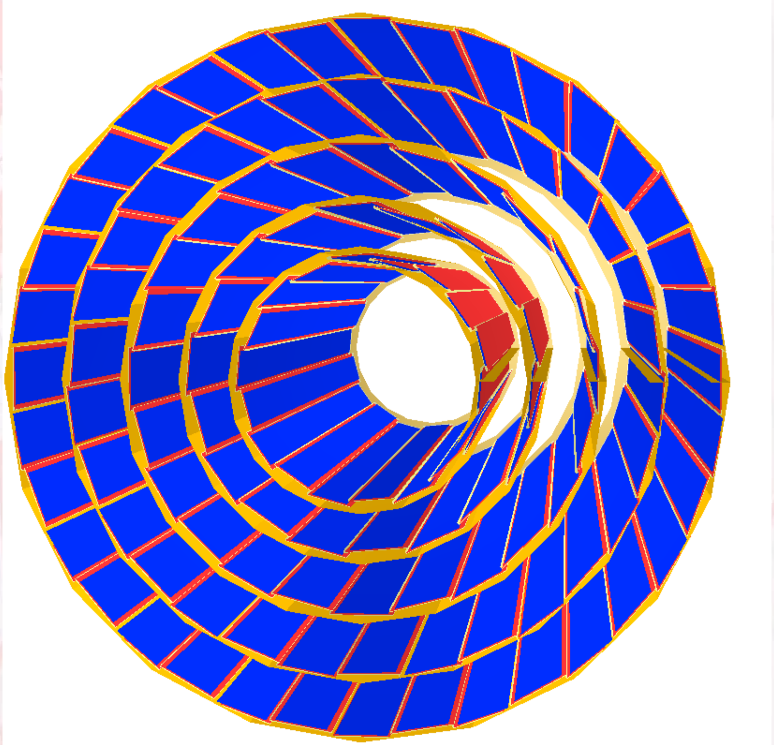
CAD Import, Export and Round-Trips

DD4hep

- If supported by Assimp, DD4hep supports
  - Import of shapes/volumes defined in CAD files into DD4hep
  - Export of partial geometries to CAD format

Import from CAD (STL)

Round-trip:
DD4hep / TGeo => CAD => TGeo



CAD: Limitations and Remarks

DD4hep

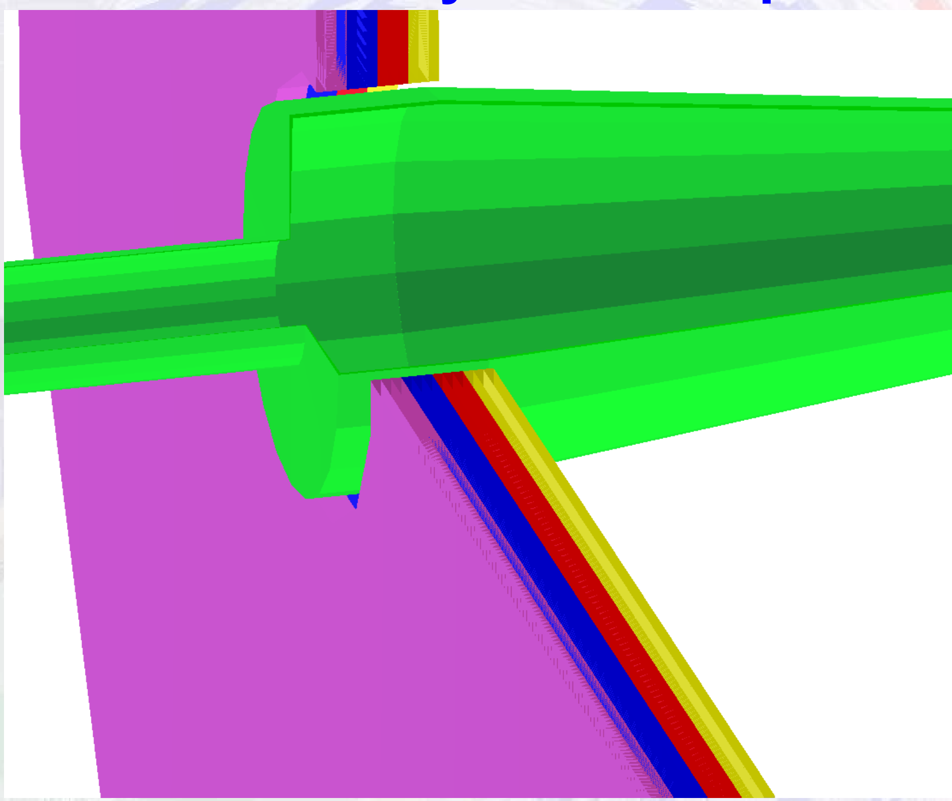
- **CAD Meshes are complex**
  - **Limitation of the total number of manageable vertices/facets**
  - **Analytical shapes are simpler than tessellated shapes and likely far better performing for tracking**
- **CAD comes in many dialects**
  - **Assimp supports many formats: STL, MD2/3/4, Collada, X3D, …**
    For details see: https://github.com/assimp/assimp/tree/master/code/AssetLib
  - **Single mesh CAD, multiple mesh CAD**
  - **Not all support materials, visualization attributes etc. Need to be injected by import mechanism**
  - **Need to choose optimal format**

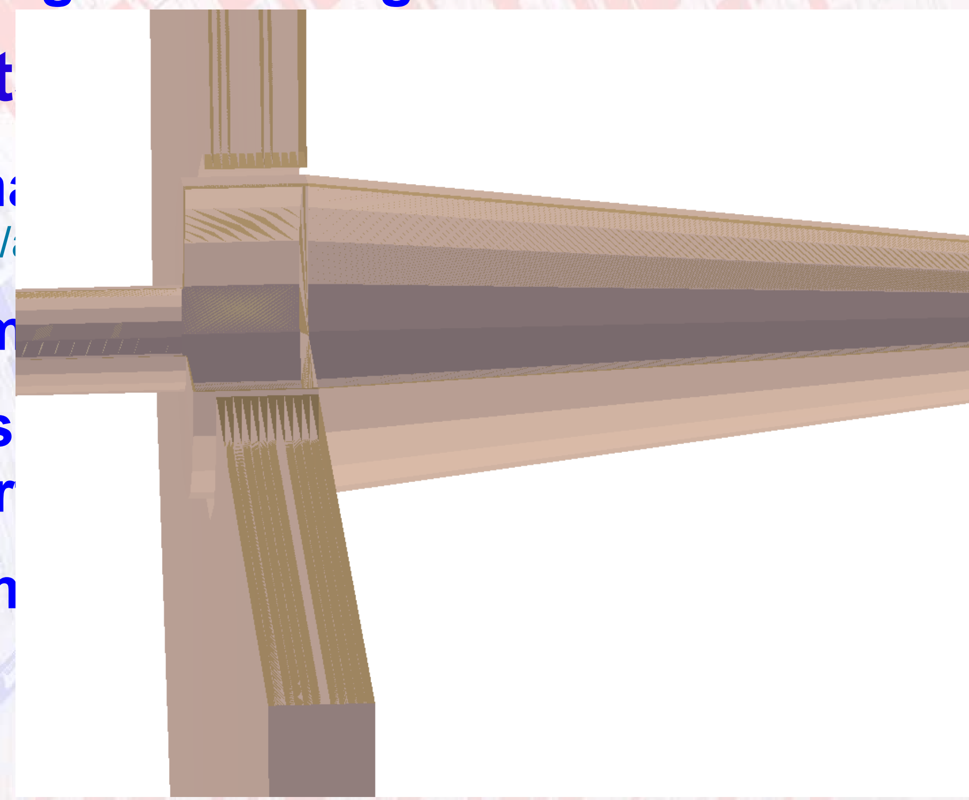
CHEP2021
Conditions and Alignment extensions to the DD4hep Detector Description Toolkit

CAD: Limitations and Remarks

DD4hep

- **CAD Meshes are complex**
  - **Limitation of the total number of manageable vertices/facets**
  - **Analytical shapes are simpler than tessellated shapes and likely far better performing for tracking**





CAD: Limitations and Remarks

DD4hep

- Can use round-trips for views in CAD tools
  - Round-trips are not unambiguously reversible
  - Example: Tube => Tessellated cannot be converted back
- Recently discovered
  - G4 complains when placing multiple CAD/tessellated shapes in assemblies
  - Works fine if CAD/tessellated shapes are placed in "real volumes"
  - Not yet understood...last word not yet phrased out
- Shape/Volume creation uses DD4hep plugin mechanism

Import / Export of the Detector Description

DD4hep

- **We do not develop such functionality**
  - **DD4hep is opportunistic on top of ROOT, Geant4, etc.**
- **Full snapshots can only be created using ROOT**
  - **ROOT is complete I/O machine and hence can save:**
    - **Geometry**
    - **Structural hierarchy**
    - **Conditions and alignments**
- **Imports / Exports of (partial) geometries possible using GDML**
  - **Plugin to save a sub-tree of the geometry**
  - **Plugin to load a sub-tree and attach it structurally to a detector element**

Support for EDM4hep in DDG4

DD4hep

- DDG4 is the DD4hep toolkit to support Geant4
  - Automatic geometry conversion
  - Plugin based, flexible programming of all user callbacks
- DDG4 supports intrinsically output to ROOT files, LCIO and now EDM4hep
  - New event model developed by Key4hep team, part of HSF
    Independent talks at this conference elaborate the issue
    See CHEP2021 talks:
    Key4hep: Status and Plans, https://indi.to/HNBpp
    EDM4hep and podio - The event data model…., https://indi.to/MbMcJ
- Ensures the interoperability of the full DDG4 functionality in the key4hep framework
- Support for EDM4hep based experiment specific digitization and reconstruction programs

How to Define an Experiment ?

DD4hep

- All starting is difficult
  - Lower entrance hurdles
- Beginners guide
  - https://dd4hep.web.cern.ch/dd4hep/page/beginners-guide
  - Fastest track from checkout to simulation
- Other documentation
  - http://dd4hep.cern.ch
  - CHEP presentations
  - User Manuals
    - Not always perfect, but give the overview
  - Up-to-date code reference (doxygen)

Toolkit Users

DD4hep

Increasing interest in the HEP community

- ILC F. Gaede et al.
- CLICdp A. Sailer et al.
- SiD D. Protopopescu et al.
- FCC-eh P. Kostka et al.
- FCC-hh A. Salzburger et al.
- FCC-ee O. Viazlo (CLD design), N. Alipour, G. Voutsinas
- SCTF Super-Charm-Tau Factory designs (Novosibirsk, Bejing)
- EIC Evaluation considered/started (W. Armstrong et al.)
- CEPC W. Li et al, IHEP

- CMS Base line for Run3 upgrade (C.Vuosalo et al.)
- LHCb LHCb Upgrade for Run III (B.Couturier et al.)
- CALICE Calorimeter R&D, started

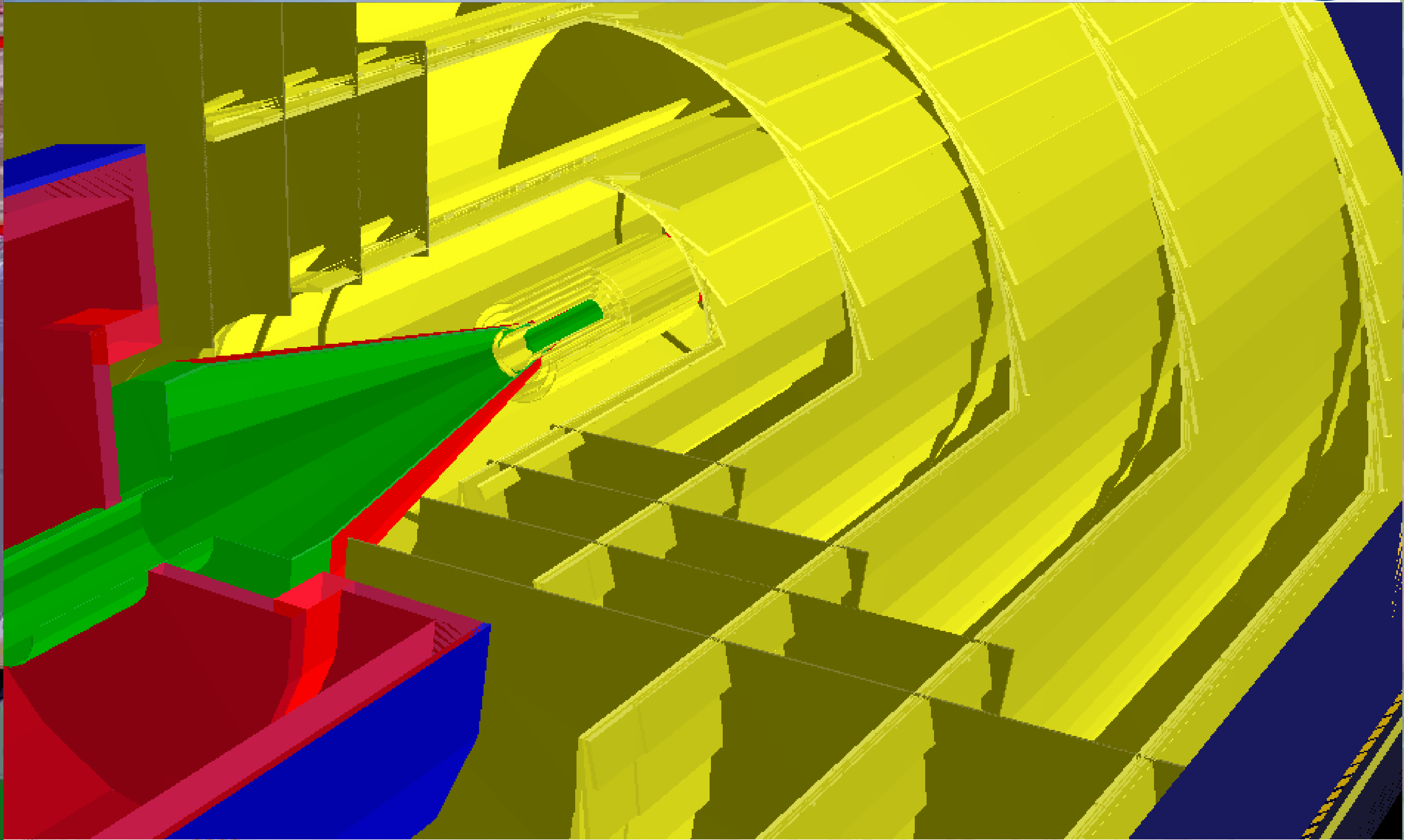
Summary

DD4hep

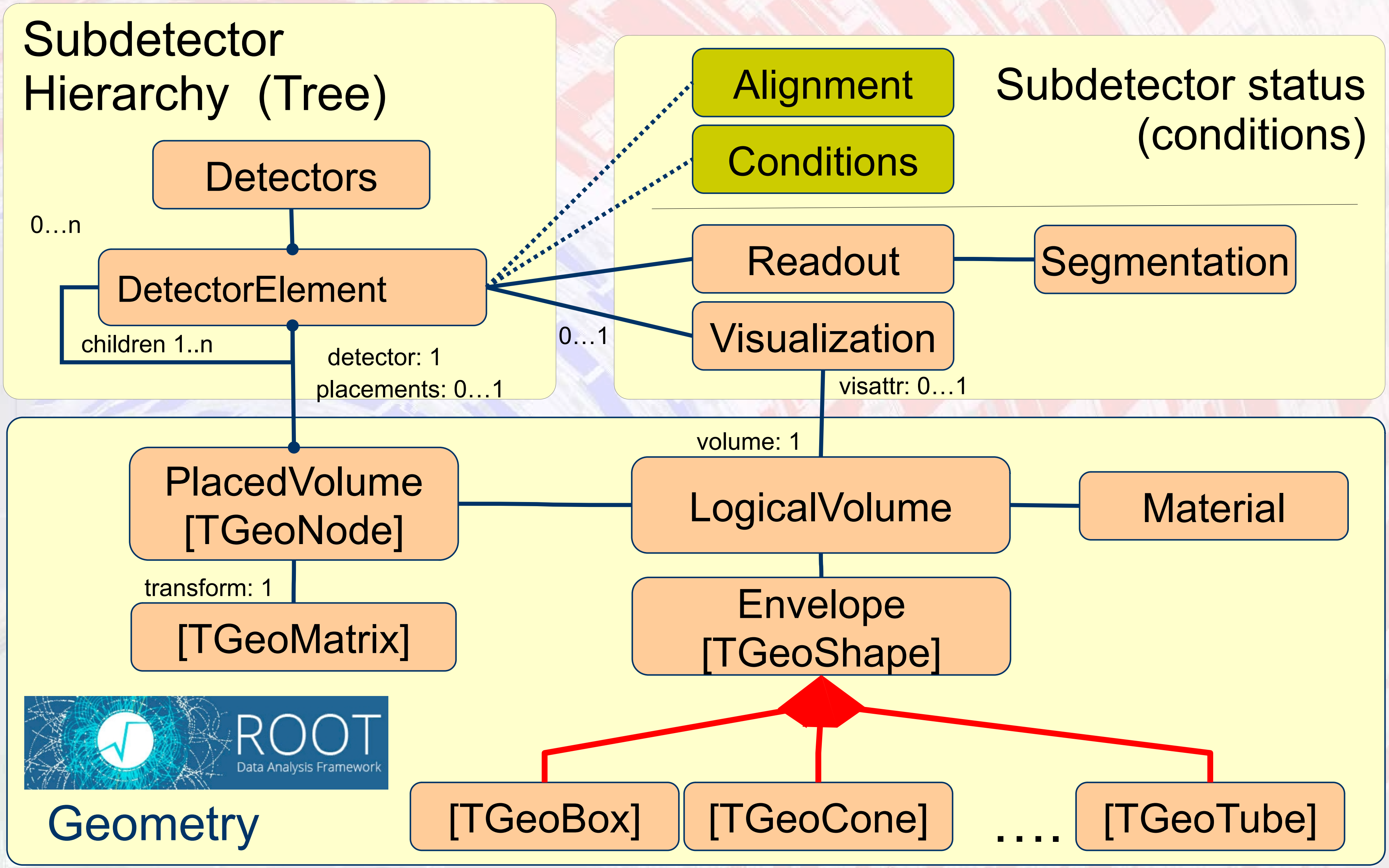
- DD4hep is getting mature
- Starts being capable of handling all aspects of detector description for the lifetime of an experiment
- Increasing interest in the community and increasing number of users
- Visit us on:
  - `http://dd4hep.cern.ch`
  - Beginners guide
  - Up to date doxygen information
  - User Manuals

Questions and Answers

DD4hep

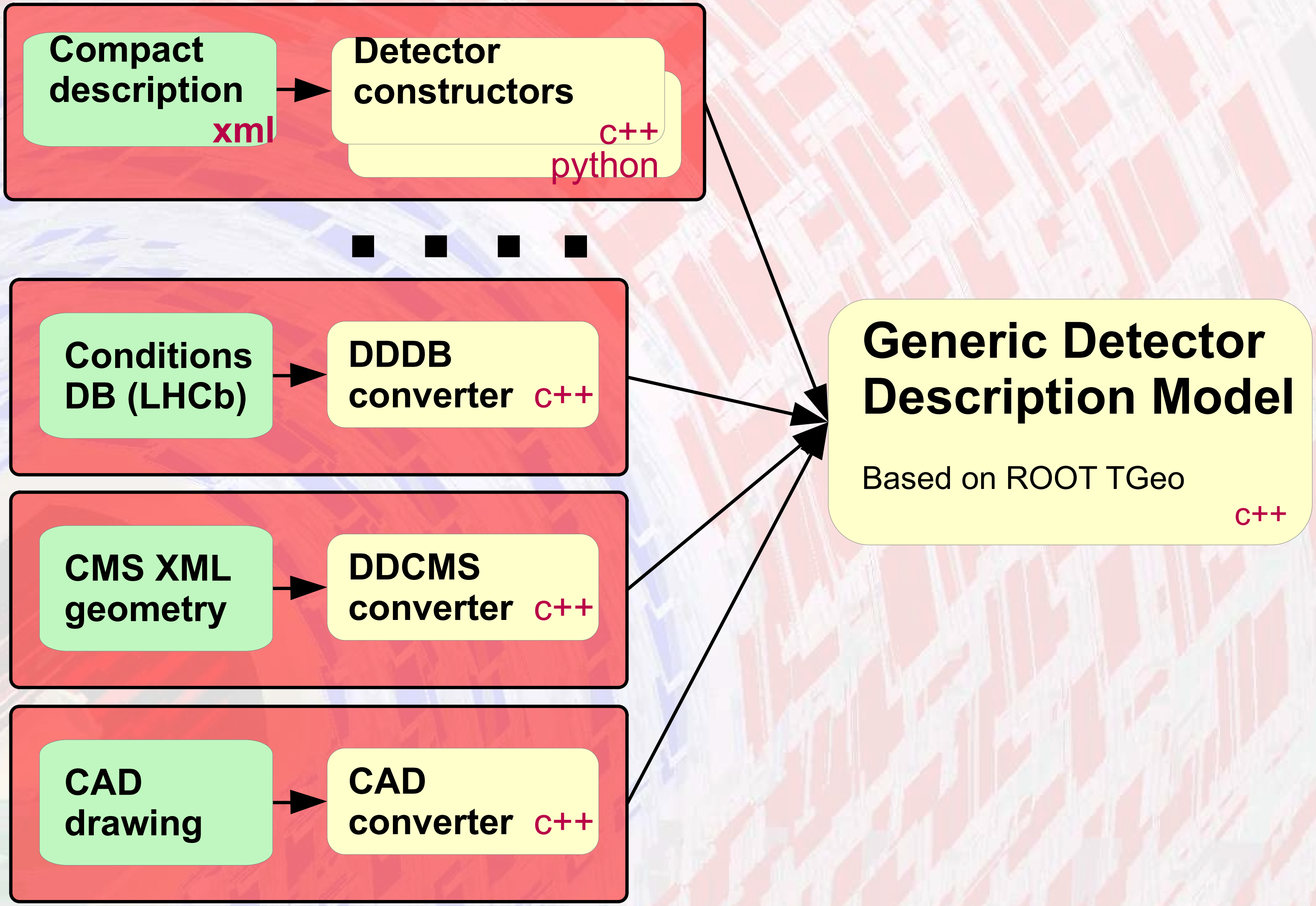


Class Diagram: Detector Element Sort of Standard...



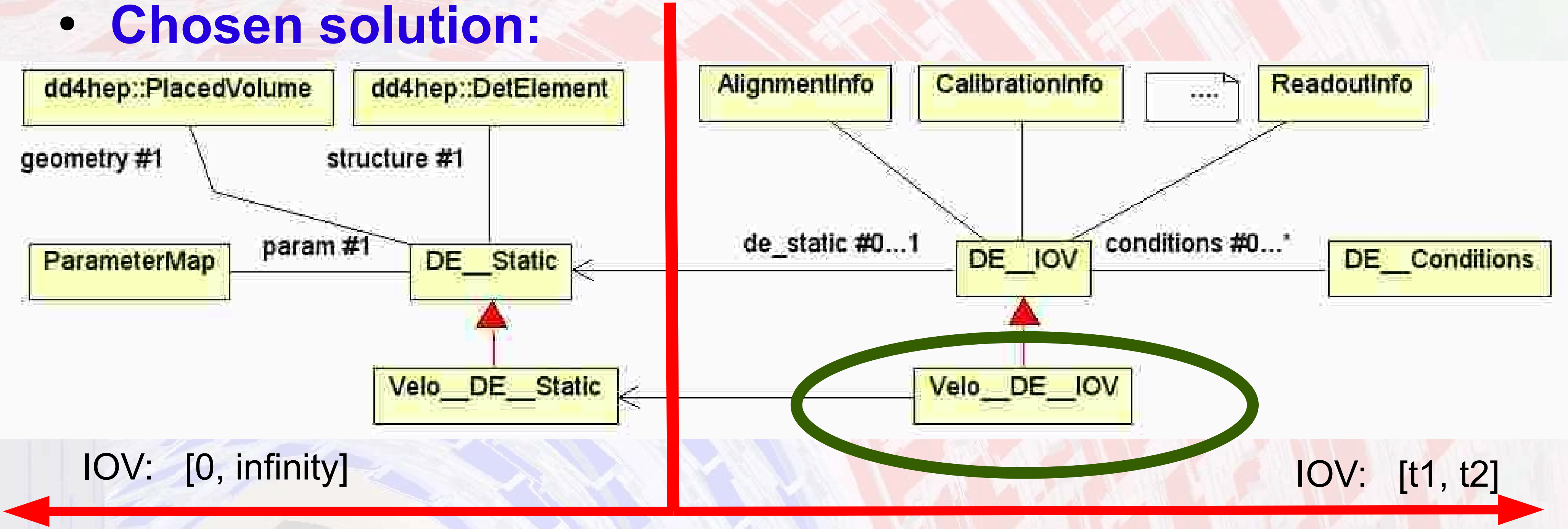
Multiple Input Sources

DD4hep



Real World Use Case
LHCb Velo Detector

- **Chosen solution:**



- **Use IOV dependent projection for event processing**
  - **This is our new “detector element”**
  - **Keeps reference to the not changing properties**
- **Dress with facade to provide required functionality(ies)**