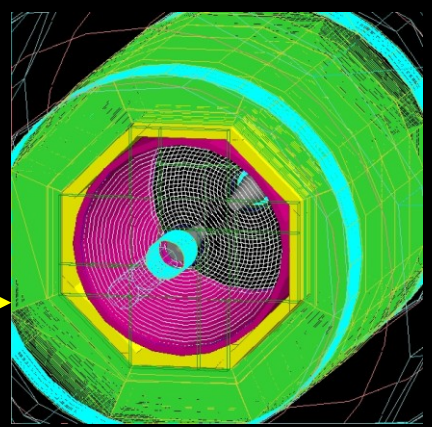
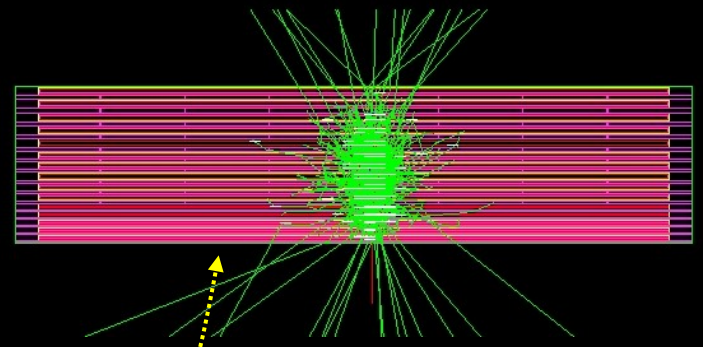
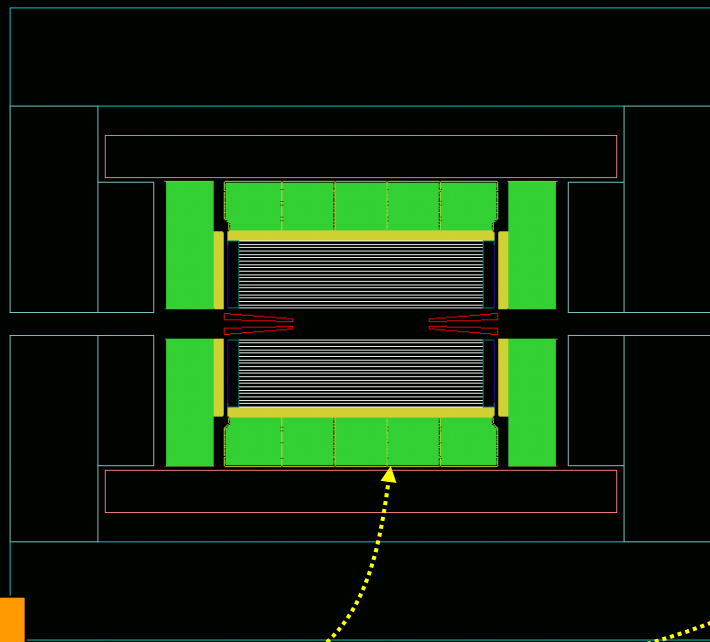


Mokka and integration of the geometry

AIDA kick-off meeting
WP2 session: Common software tools
17 February 2011 – CERN

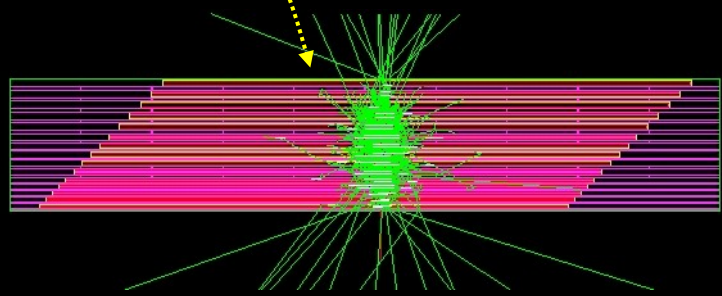
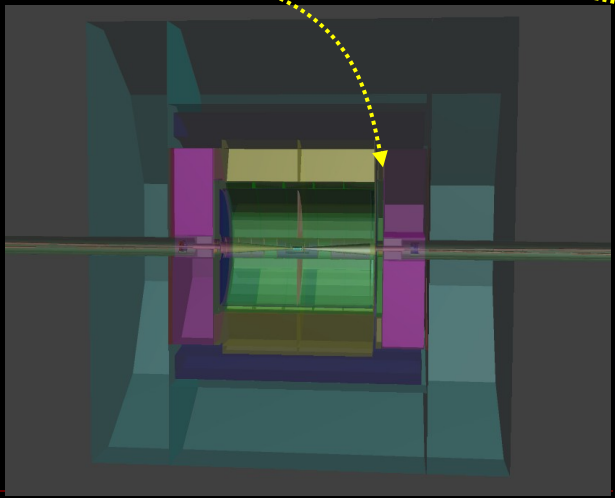
Paulo Mora de Freitas and Gabriel Musat
Ecole polytechnique

Mokka detectors

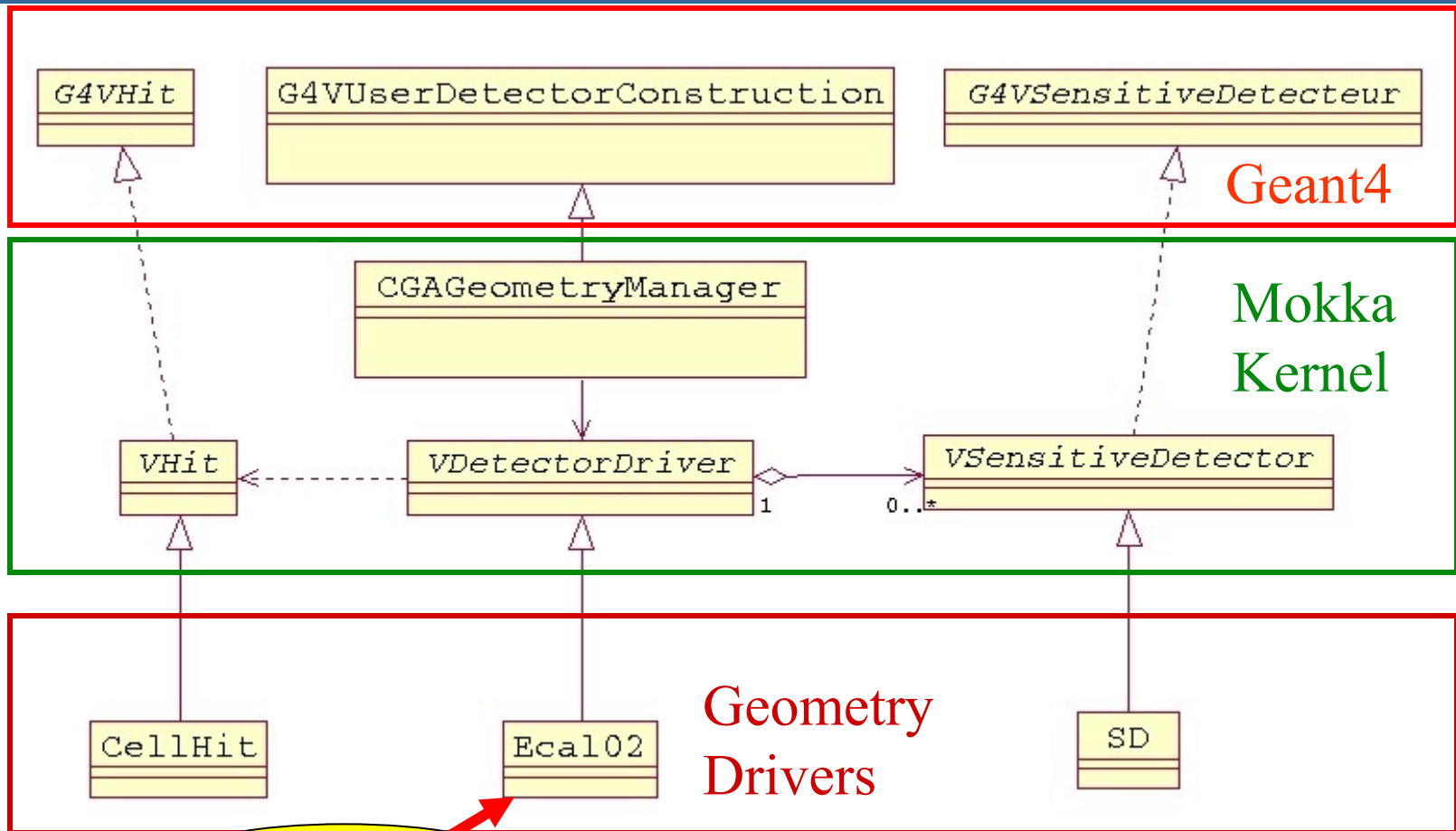


Mokka
Geometry drivers

**Geometry
Database**

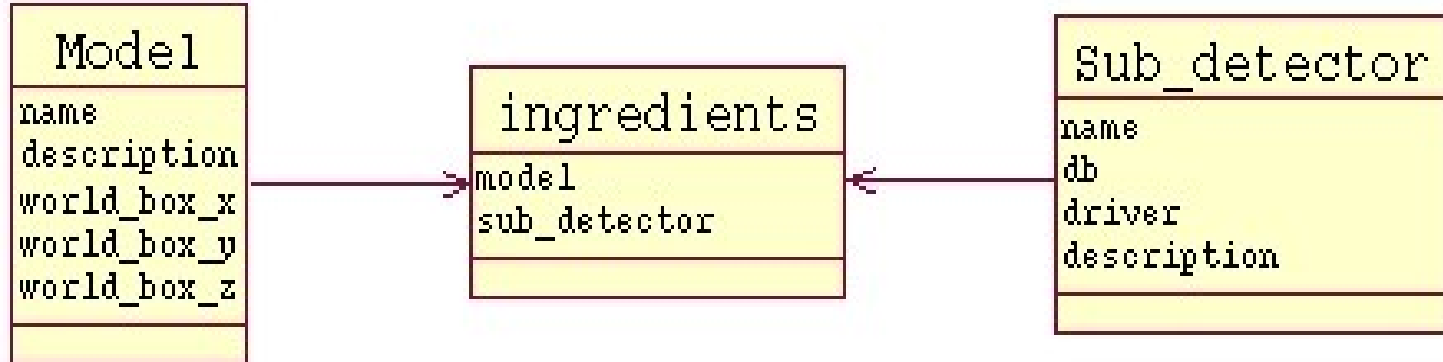


Mokka's kernel framework



Geometry Database

The detector models in DB



- ❑ A model = a set of sub detectors (TPC, Ecal, Hcal, etc.)
- ❑ A sub detector = a driver \leftrightarrow DB association

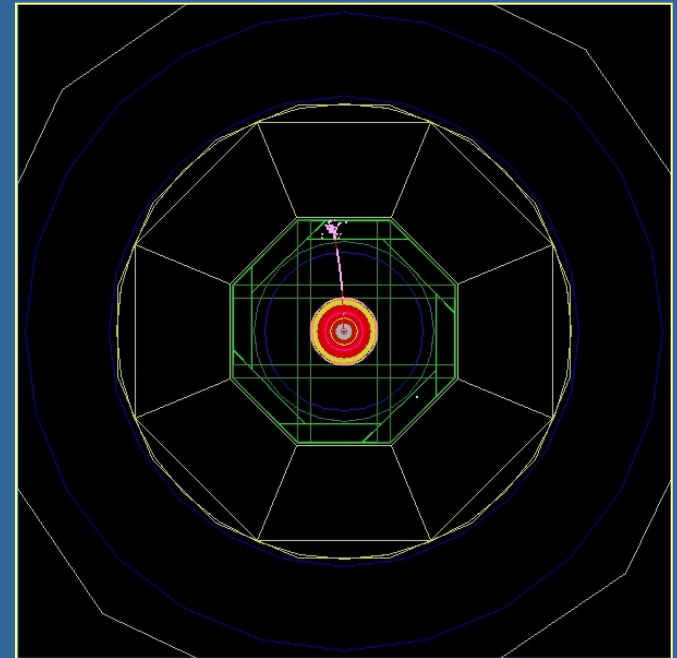
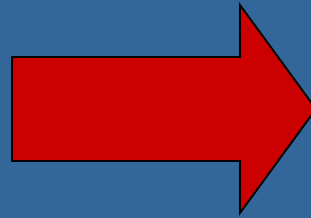
Mokka, other geometry features

- “Scaling”, the user is able to modify the model's main parameters at launch time
 - To easily be able to study different detector options, like TPC size, number of layers in calorimeters, etc.
- “Cooking”, the user is able to modify the model ingredients at launch time
 - To easily be able to study different detector technologies, like analogical versus digital Hcal, etc.

Scaling

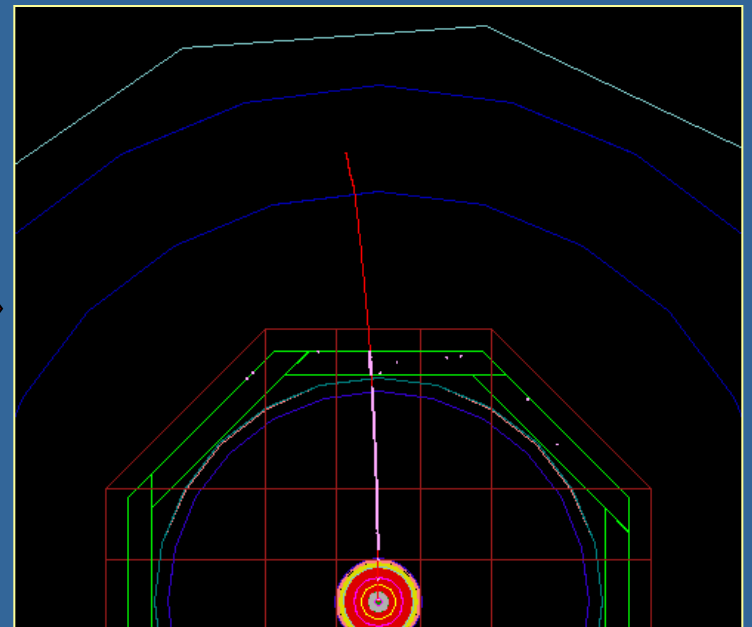
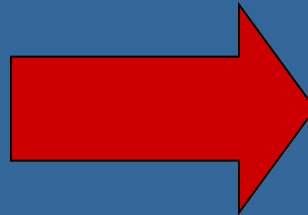
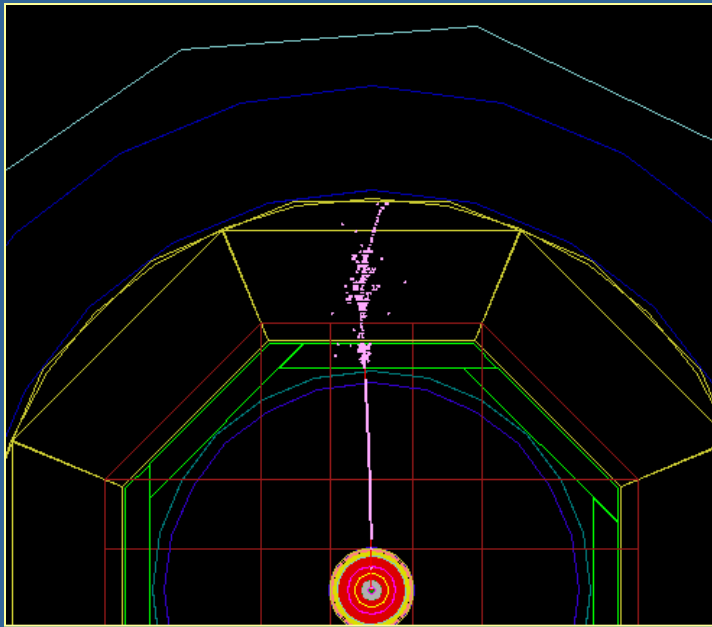
- Example :

/Mokka/init/globalModelParameter TPC_outer_radius 800

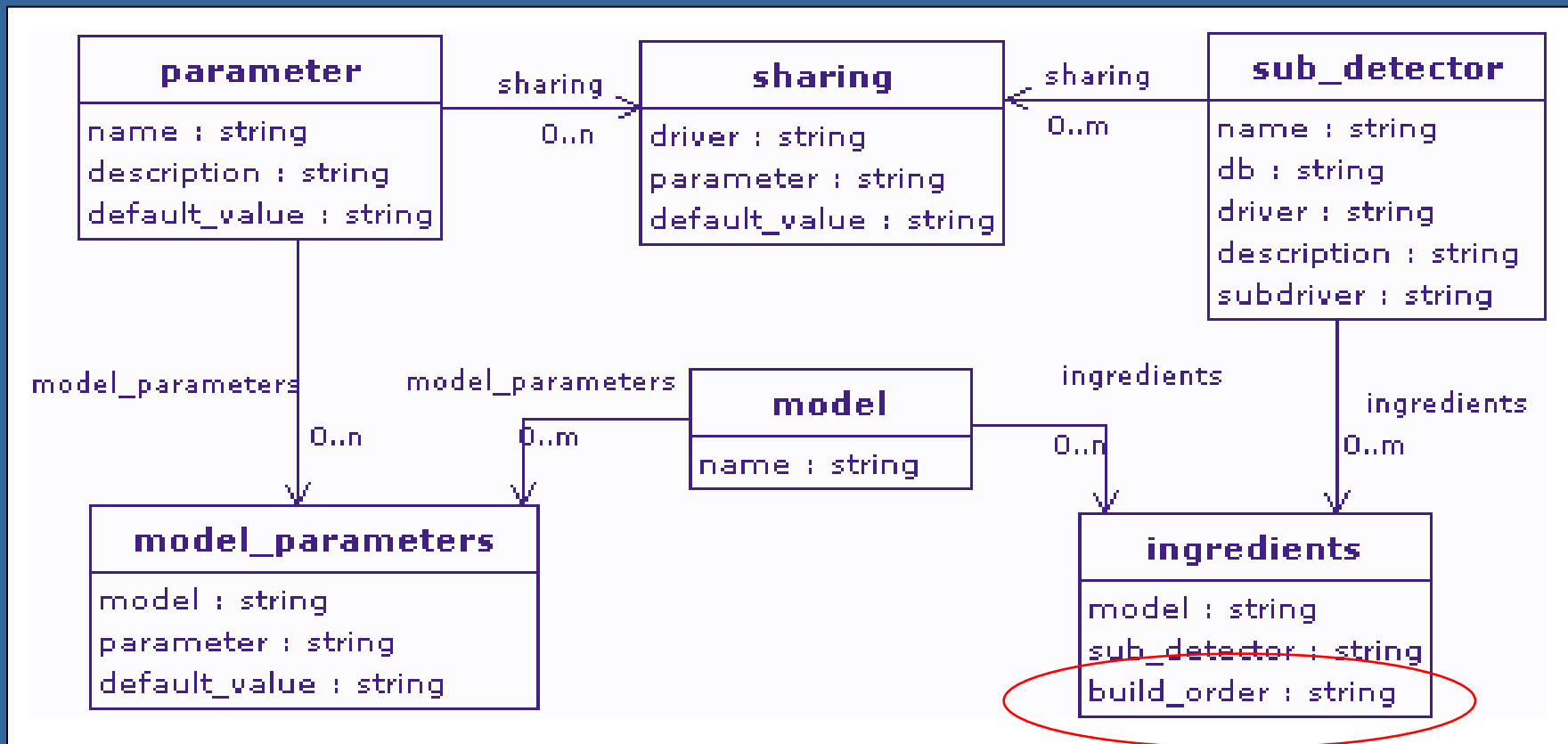


Cooking

- Example :
/Mokka/init/EditGeometry/rmSubDetector SHcal01

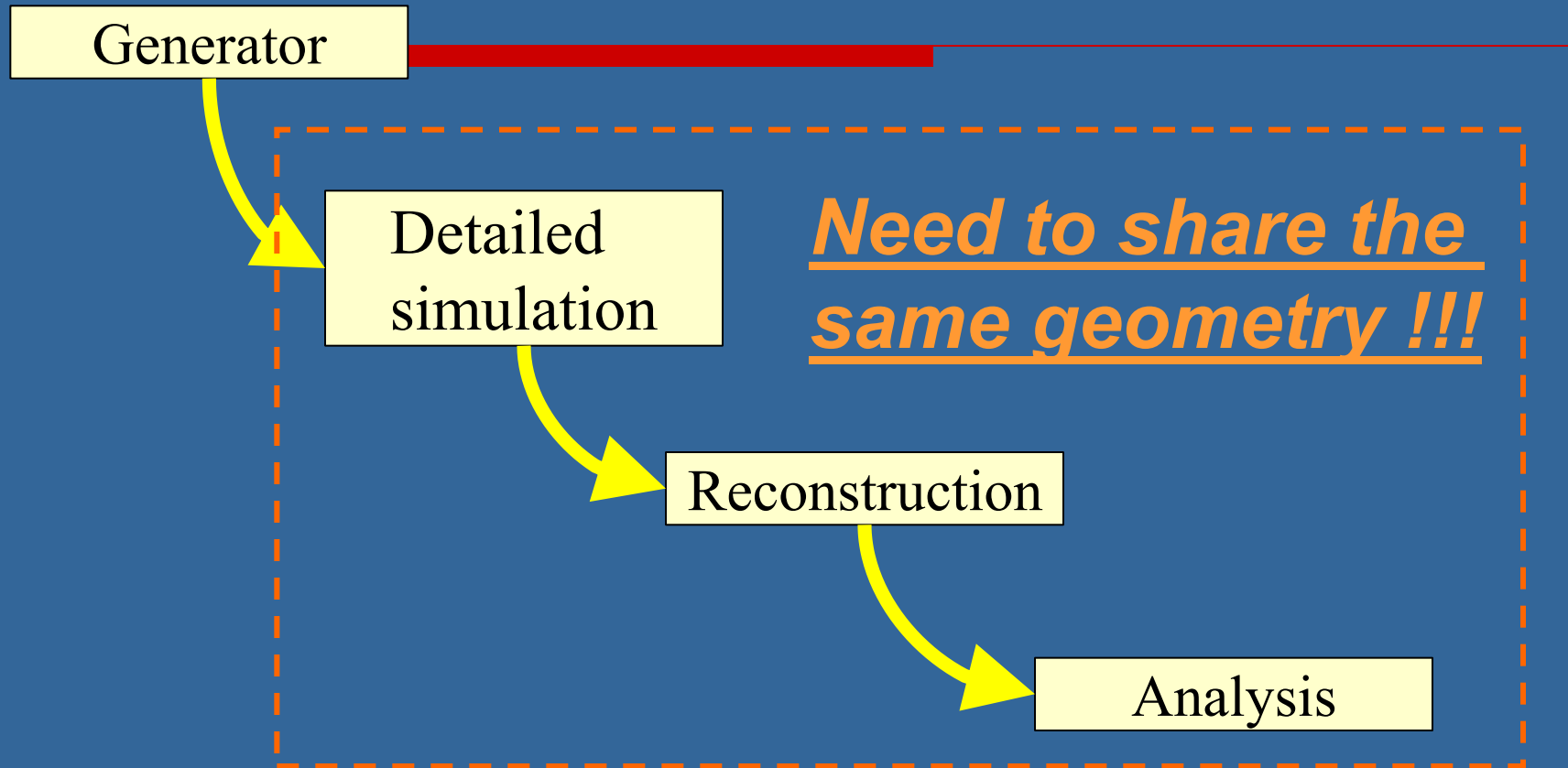


Parameters in Mokka DB



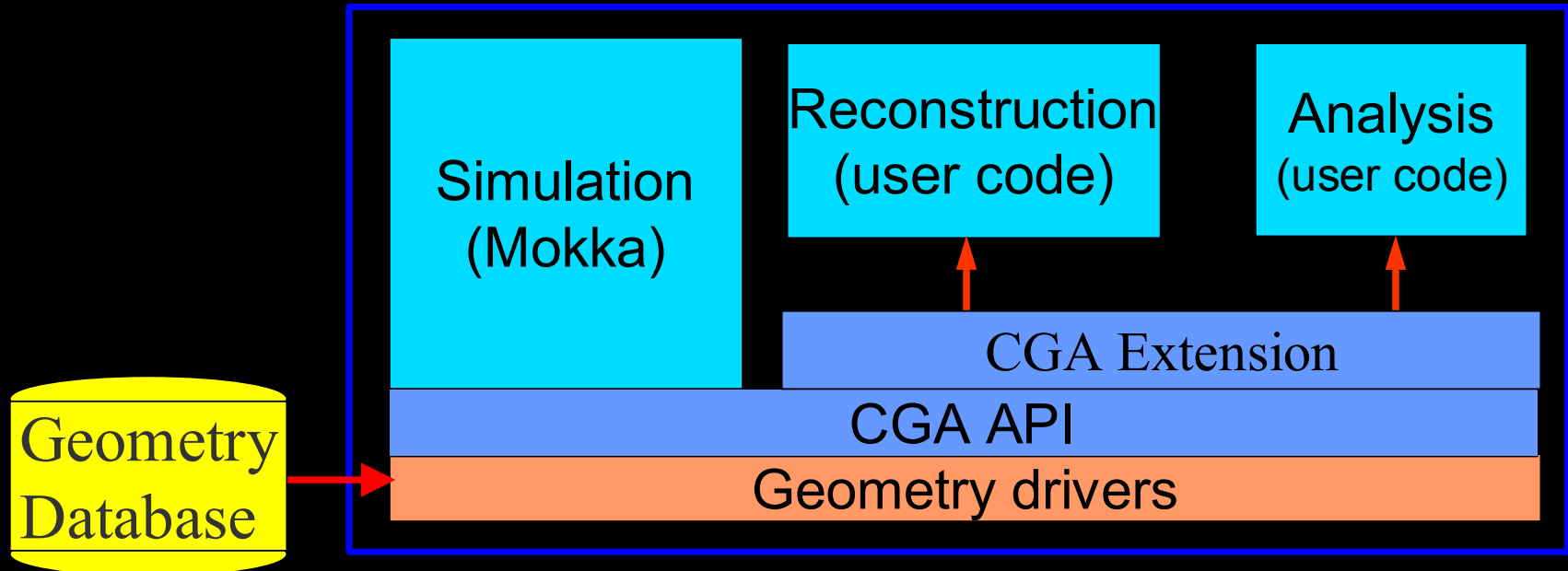
- ❑ Parameter values are overwritten by the `sub_detector` default, then the `model_parameter` default, then steering file value if any, and finally by the environment value if modified at run time by a previous driver.
- ❑ Scaling follow the model `build_order` in `ingredients`

Geometry has to be shared



And nobody never tried to retrieve geometrical information via SQL queries, except in the geometry drivers in MOKKA !!!

Mokka - Common Geometry Access API (F77, C++, C, Java)



- Implements some reconstruction utilities.

About CGA

□ Good features in CGA:

- Provides a full consistent geometry system for simulation, reconstruction and analyses.
- Provides a scan mechanism to explore the detector geometry and material proprieties (geantino tracking)

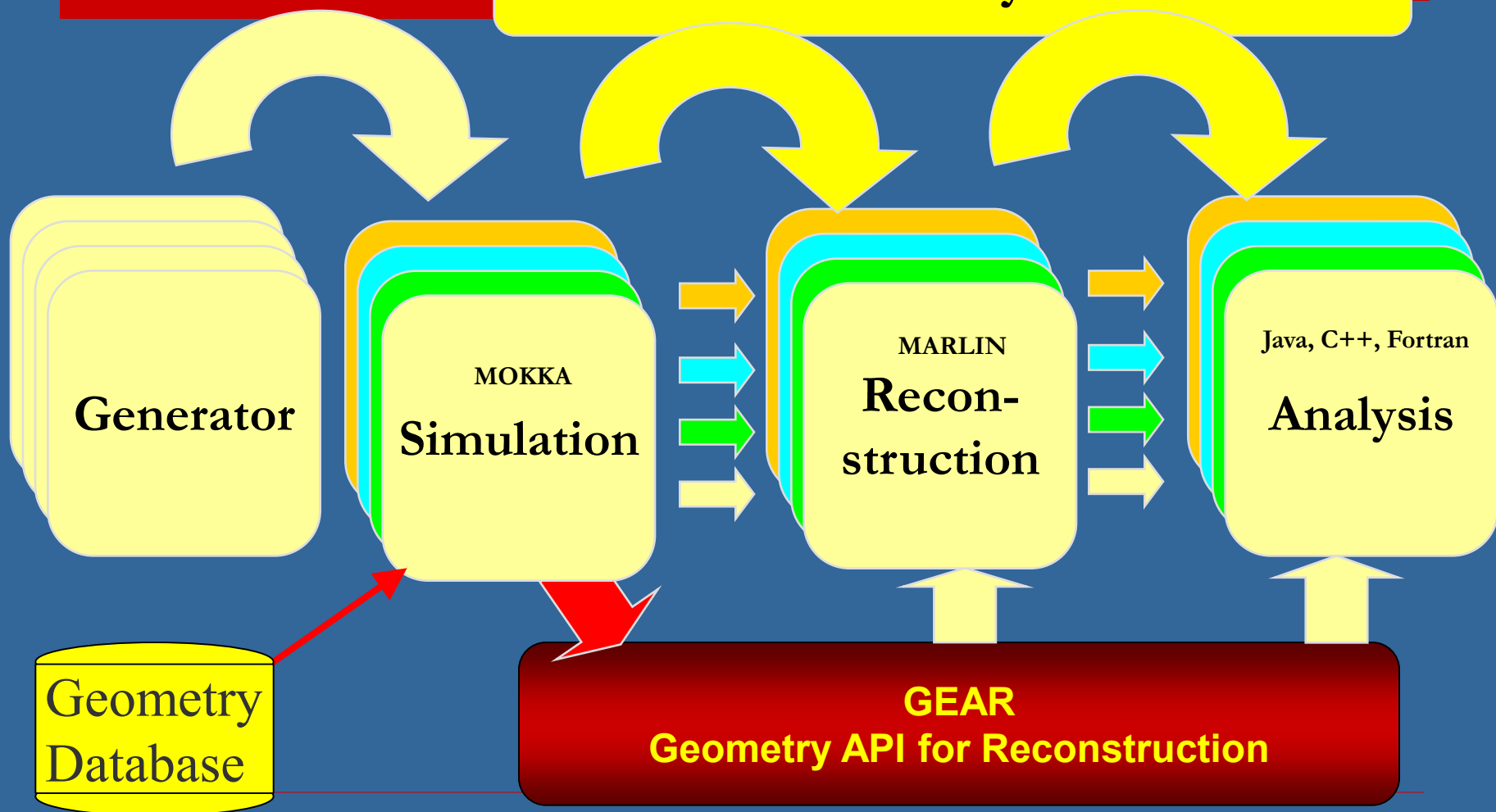
□ But...

- Links against Geant4 libraries
- Exports only low level geometry (G4 volumes level)
- a few users only

□ Available with Mokka since 2004, anyway

ILD Current framework

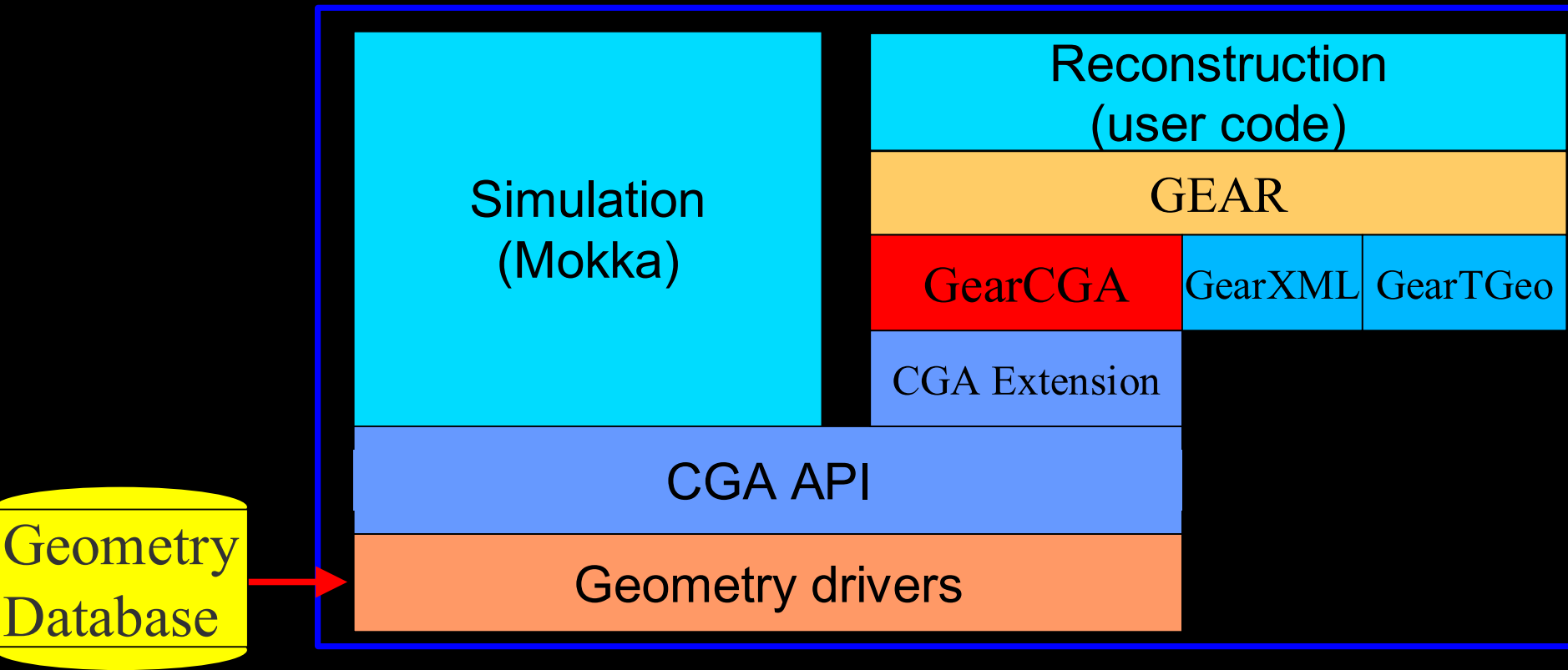
LCIO Persistency Framework



Improvements from GearCGA

- Usual GEAR properties (detector parameters) now available without having to previously run Mokka to generate the GEAR file
- together with the Point+Distance Properties
- information is now supplied by the CGA interface
- the libraries belonging to Mokka, Geant4, CLHEP, MySQL, GEAR, LCIO are put together in one library after linking Mokka

GearCGA



Towards a new geometry tool

- MOKKA history rises a set of user's requirements for the Geometry Toolkit
 - To be generic, flexible and friendly for the final user, etc.
- To adopt a CAD graphic interface:
 - No more script language neither XML edited by hand
 - See <http://www.opencascade.org>

Towards a new geometry tool

- Abstraction levels for the detector geometry:
 - To cope with the different views for simulation, reconstruction, analysis, event displays...
 - To use metadata to describe the abstraction levels, to keep the tool as generic as possible
- To provide a Data Manipulation Language (DML)?
- To define an abstract API (C++, Java, etc.)
- To provide at least one implementation for this API
- To provide at least one simulation/ reconstruction / event display chain compliant with it.