

Mokka and LCDG4

Ties Behnke, DESY and SLAC

- MOKKA: european (france) developed GEANT4 based simulation frame
- LCDG4: US developed GEANT4 based simulation frame
 - LCDroot: root flavour
 - LCDG4: plain “vanilla” GEANT4 flavour

- The “holy grail”:

Develop a common simulation frame for all LC studies which is shared between different regions.

- Boundary conditions: conserve (some) level of compatibility with existing frameworks like BRAHMS

Geometries

- Defining geometries is one of the most difficult tasks
 - flexible system is needed, to make realisation of different geometries easy
 - System has to support full set of GEANT solids and operations
 - System should allow the definition of sensitive detectors
 - System must allow interface to reconstruction

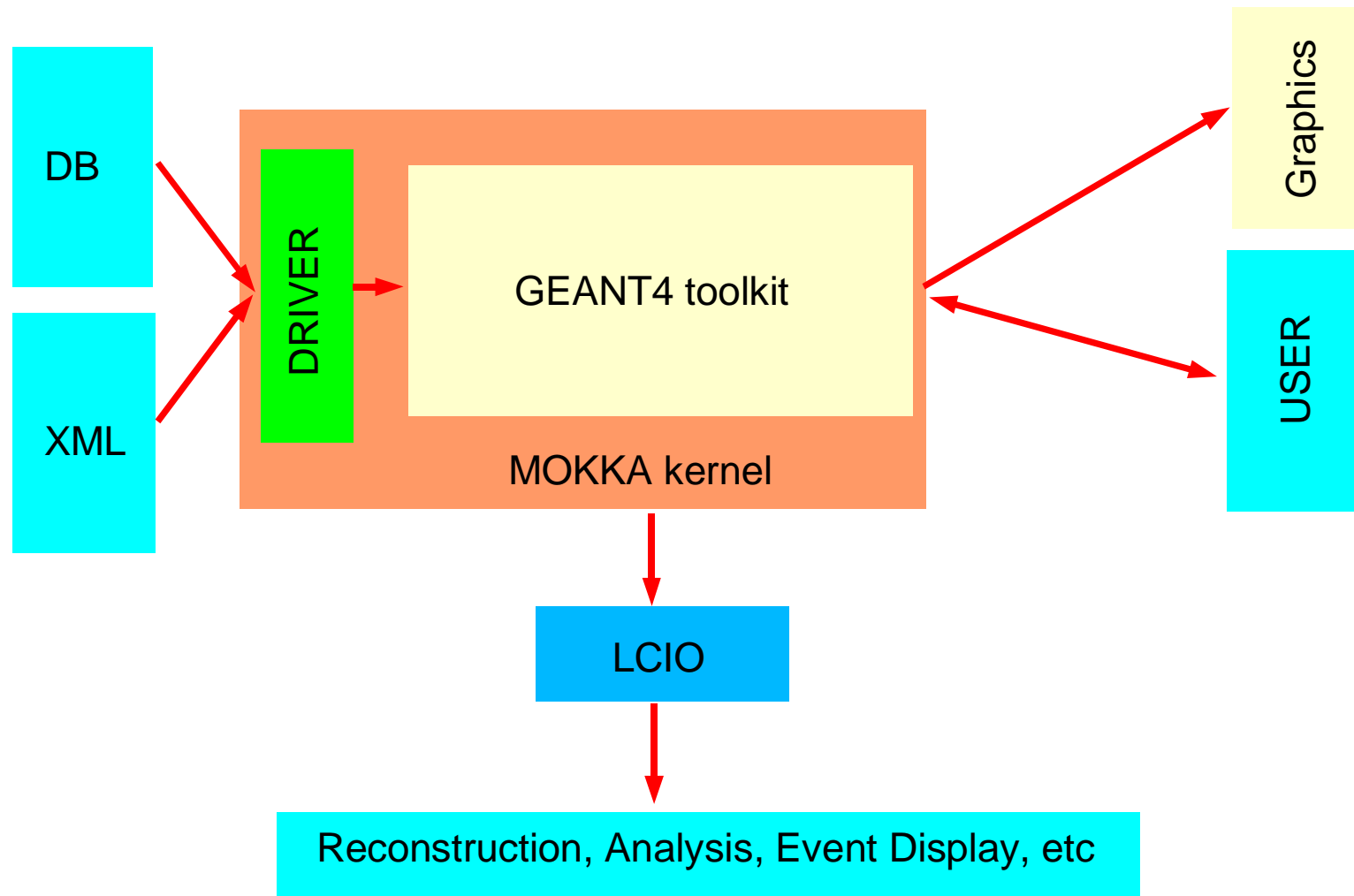
Currently no such system does exist!

- MOKKA approach:
 - database for geometrical constants, drivers to realise the geometries
- LCDG4 approach:
 - XML description of geometries and sensitive detectors, drivers to realise the detectors in GEANT4

Both systems have their limitations, neither one is really generic

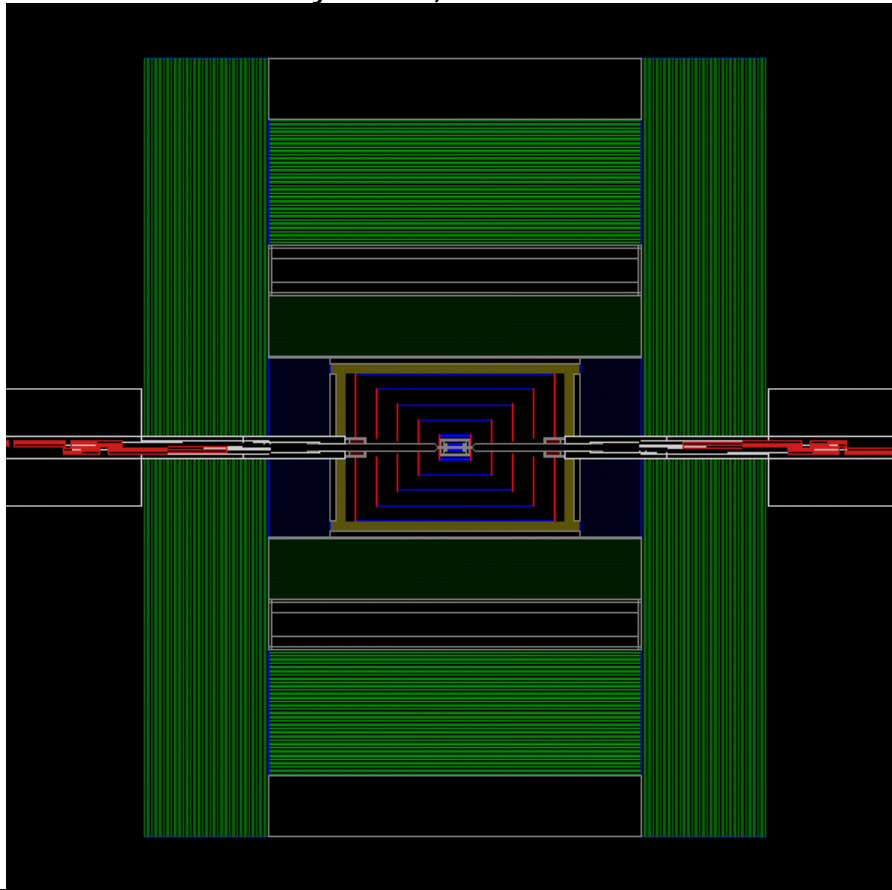
but both are being used, and are the best ones on the market at the moment

The MOKKA frame



Marriage of both

- Implement the US XML system as an additional driver in MOKKA:
 - capability to construct the XML based US detectors as MOKKA detectors
 - everything else is common for both: hits, hit formats, etc
 - additional advantage that it allows mixing: extends the capabilities of the XML based system, but retains backwards compatibility

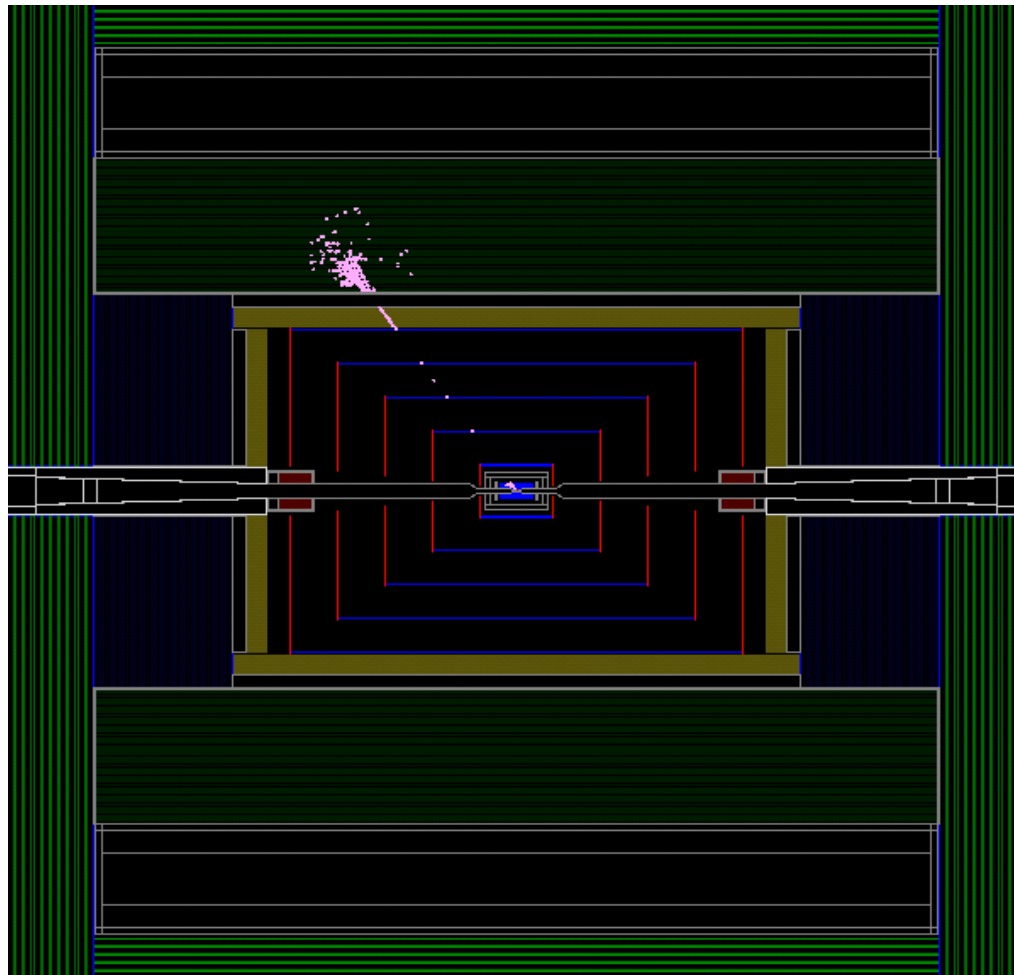


sideview of the US SD
detector in the MOKKA
frame.

Hits etal

- all detectors regardless of the origin share the same hit objects
(see Franks talk on LCIO)

SD detector
with hits in
MOKKA
simulation

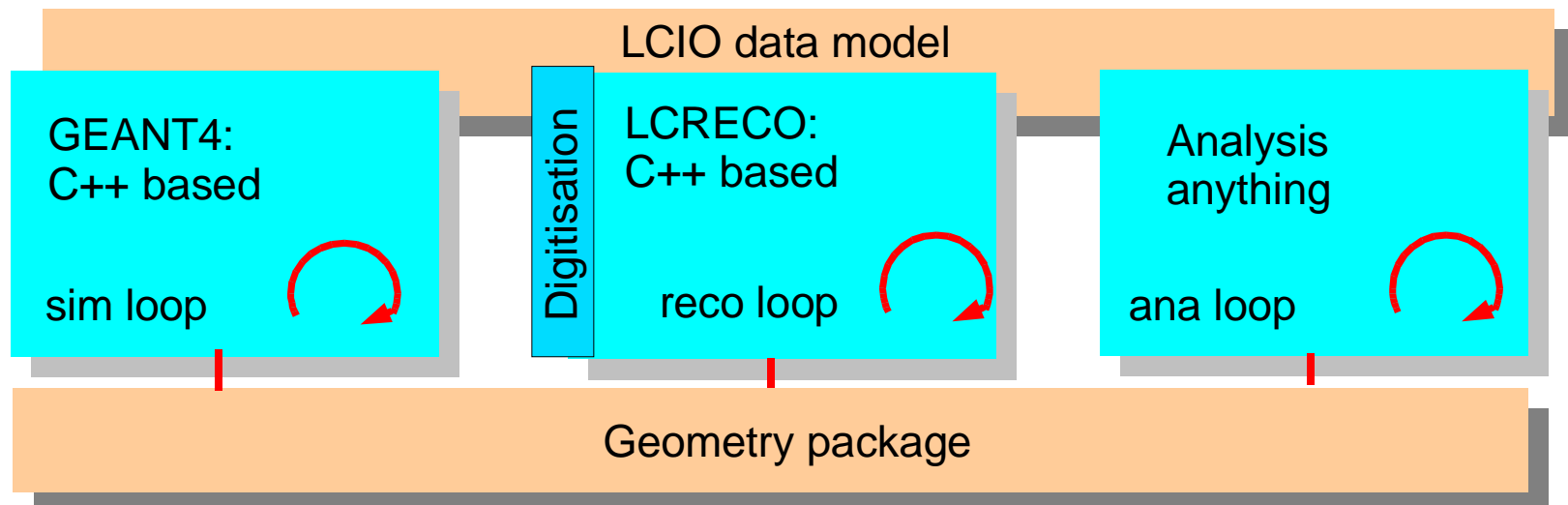


Whats next with MOKKA/LCDG4

- Clean up the interfaces and the code
- Bring things into the mainstream MOKKA release
- Implementation of LCIO will be a central point to have people accept one common Kernel
- Open Questions:
 - Many...
 - geometries:
 - ➔ hierarchies
 - ➔ links between detectors
 - ➔ materials
 - ➔ sensitive detectors
 - ➔ more generic drivers
 - graphics: quite limited at the moment
 - generic interface for histogramming: AIDA?
 - speed – at the moment MOKKA is significantly slower than BRAHMS for the same events

Beyond MOKKA

- Goal: provide a light-weight environment for reconstruction and analysis
 - simple
 - not tightly integrated with any particular package like root, JAS, etc
 - flexible
- Proposal: “plain C++” frame
 - knows about LCIO and LCIO datamodel
 - talks to reconstruction packages like tracking, energy flow etc
 - provides a simple user interface
 - allows the interaction with more fancy environment like root, JAS etc



Summary and Conclusion

- We are rapidly moving towards a common GEANT4 frame between US and EU
- MOKKA as a basis becomes more and more complete
- Major development in the medium term needed for geometry system
- Next big project is reconstruction frame
- Will develop a prototype in close collaboration with out US colleagues
- Expect some first results before the end of June this year

my personal summary and recommendation for people who actually want to do “analysis:

- at the moment use BRAHMS: its faster and more complete
- during the course of this years the tools to go to OO and C++ based packages should become available and usable

A Higgs event

