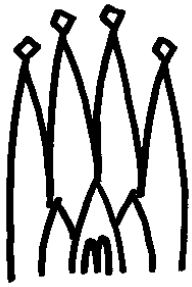
Detector Description in LHCb

Detector Description Workshop
22 May 2001
P. Mato / CERN



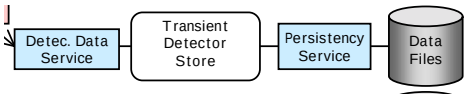
Contents

- Architecture
- Technical Choices
- Current Implementation Status
- Plans

Architecture

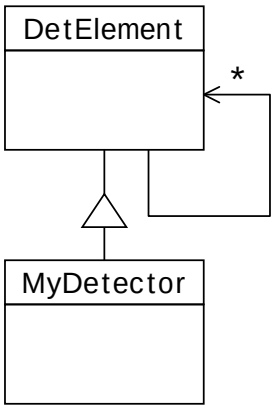
- Sub-Architecture of Gaudi
  - Same principles
  - Transient/Persistent representations
- Focus on the “Physics Algorithm”
- Coherent access to “all” detector data
  - Geometry, Calibration, Slow Control, etc.

Gaudi Architecture

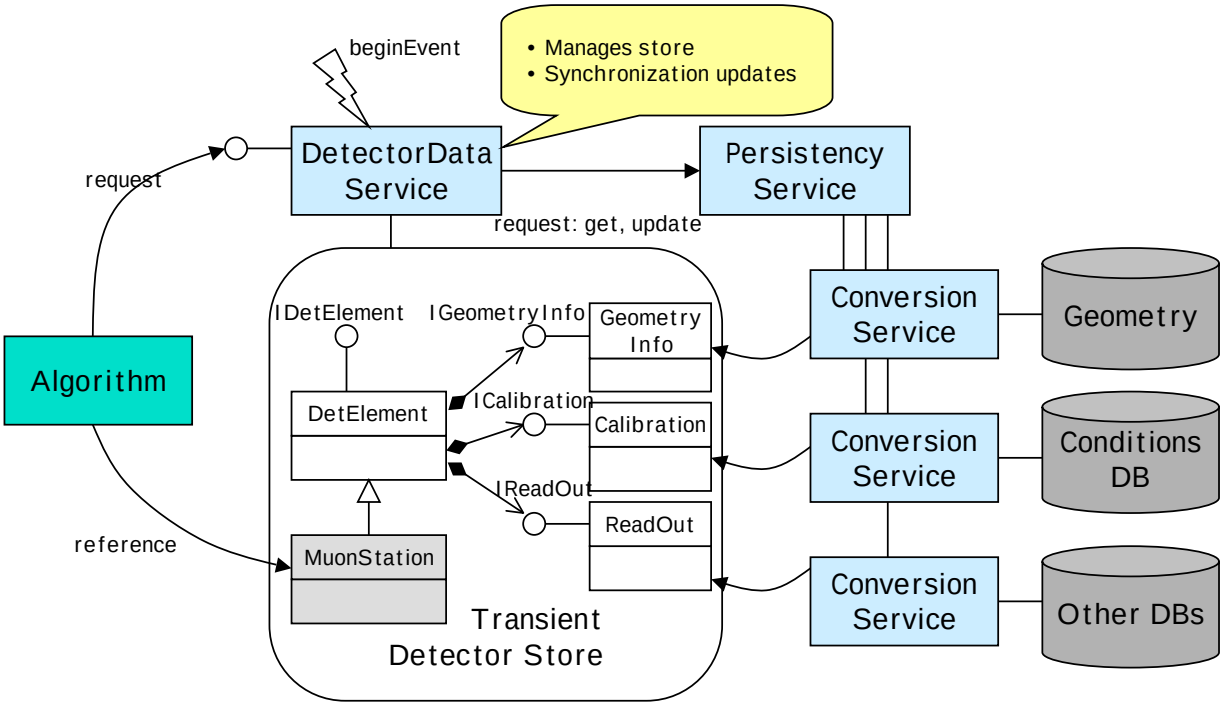


Logical Structure

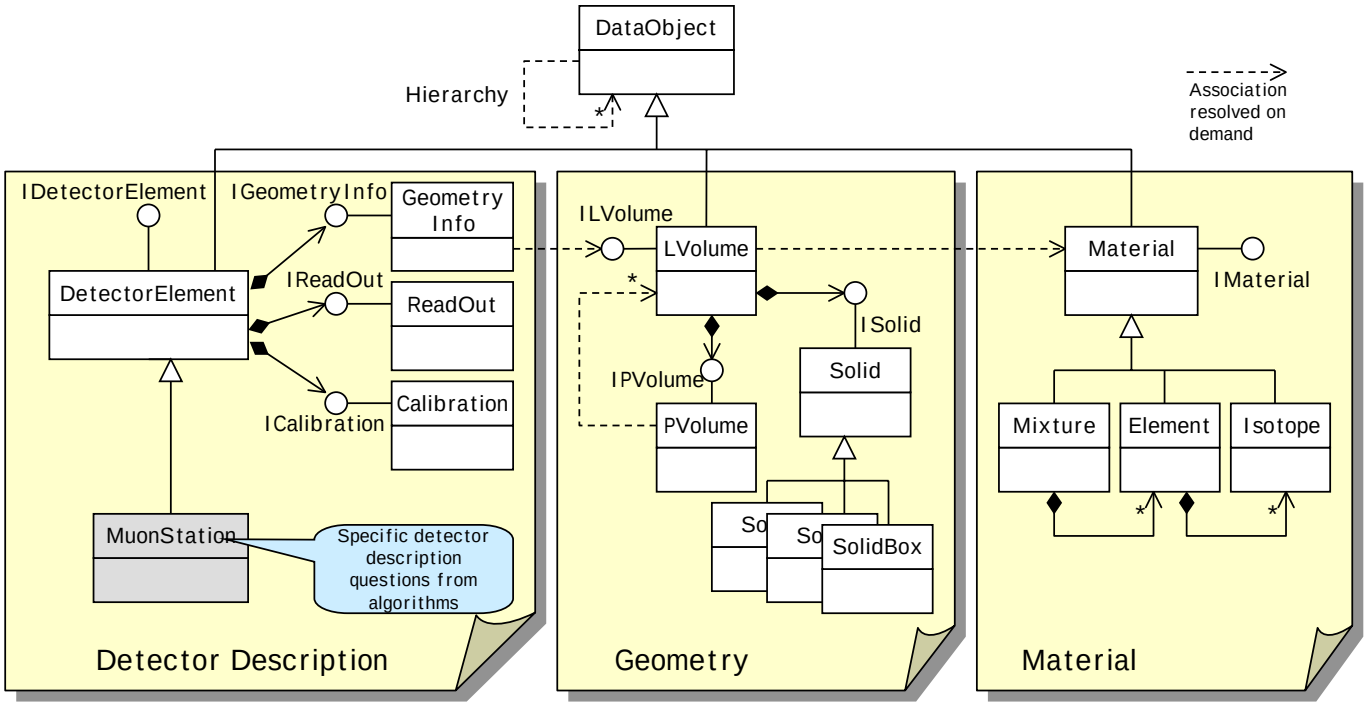
- The basic object is a Detector Element
  - Identification
  - Navigation (tree-like)
- DetElement as information center
  - Be able to answer any detector related question
    - » E.g. global position of strip#, temperature of detector, absolute channel gain, etc.
  - Placeholder for specific code
    - » The specific answers will be coded by “Physicists”



Algorithm Accessing Detector Data



Simplified Diagram (simplified)

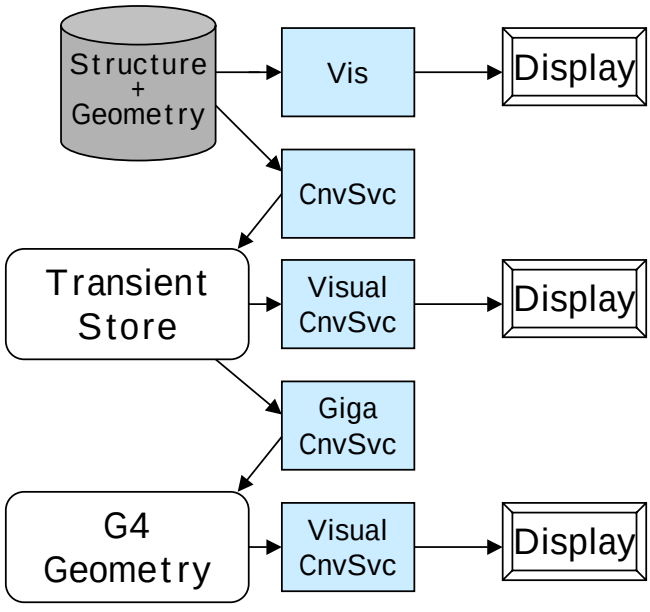


Interfacing with Geant4

- We integrate Gaudi with Geant4 by providing a number of “Gaudi Services” (GiGa)
- The GiGaGeomCnvSvc is able to convert transient objects (DetElem, LVolume, Surfaces, etc.) into G4 geometry objects
  - The conversion do not require “user” code
  - Flexibility in mapping Gaudi model to Geant4 model
- Single source of Geometry information

Geometry Visualization

- Visualization is essential for developing the geometry
  - Applicable at the different data representations
- Generic geometry information conversion to 3D graphics data
- GaudiLab (Onx)



Conditions DB

- Accessing detector conditions data (calibration, slow control, alignment, etc.) should be the same as geometry data
  - Time validity period
  - Versioning
- Detector geometry may simply be a concrete type of conditions data
  - Store it using the same DB implementation

Persistency based on XML files

- XML is used as persistent representation of the Structure, Geometry and Materials (eventually also Conditions)
- Mapping each C++ class into an XML element
  - Inheritance emulation (Generic and Specific Detector Element)
  - Relationships using “Links” and symbolic names
- Allow math expressions with parameters and physical units
  - Using expression evaluator (available in CLHEP)

XML Converters

- Capable of converting (one way for the time being) XML into C++ objects
  - Originally using SAX interface, re-implemented with DOM (Xerces-C)
  - Specific converters for specific “DetElement”
- Available Converters
  - Structure: Catalog, DetElement
  - Geometry: LVolume, Surface, Solids (various shapes, boolean), PVolumes (parametric)
  - Materials: Isotope, Element, Mixture, TabulatedProperty

XML Files

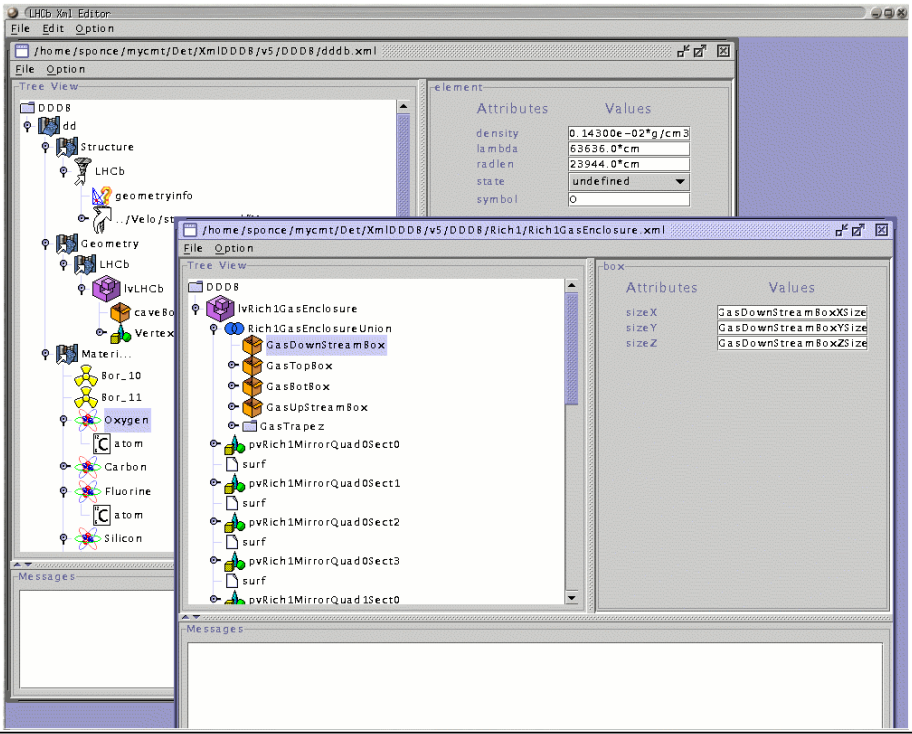
- Separated XML files
  - By sub-detector and data type (structure, geometry, material)
  - Low coupling of developments
- Versioning done using CVS
- Possible migration to the “Conditions DB”

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE DDDB (View Source for full doctype...)>
<DDDB version="3.1">
  <detelem name="LHCb" type="passive" classID="2">
    <author>Radovan Chytracek</author>
    <version>0.1</version>
    <geometryinfo lvname="/dd/Geometry/LHCb/lvLHCb" />
    <detelemref href="../Velo/structure.xml#Velo" />
    <detelemref href="../Rich1/structure.xml#Rich1" />
    <detelemref href="../Shield/structure.xml#Shield" />
    <detelemref href="../Magnet/structure.xml#Magnet" />
    <detelemref href="../Tracker/structure.xml#Tracker"/>
    <detelemref href="../Rich2/structure.xml#Rich2" />
    <detelemref href="../Spd/structure.xml#Spd" />
    <detelemref href="../Prs/structure.xml#Prs" />
    <detelemref href="../Ecal/structure.xml#Ecal" />
    <detelemref href="../Hcal/structure.xml#Hcal" />
    <detelemref href="../Muon/structure.xml#Muon" />
  </detelem>
</DDDB>
```

XML Detector Description Editor

- Developed an graphical editor to “hide” XML to the end-users (physicists)
  - It understands our model (DTD)
  - It understands “links” and allow us to edit a web of XML files
  - Generic (possible to use another DTD)
  - Implemented in Java (portable)

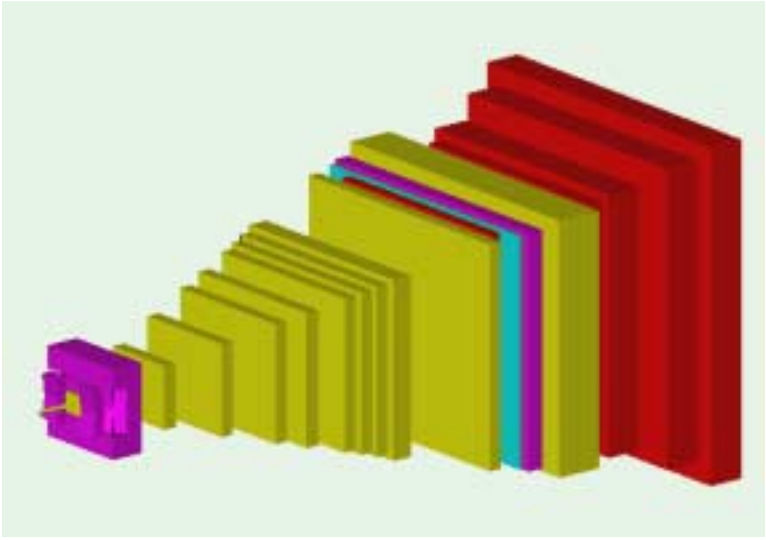
XMLEditor



Status of LHCb Detector Description

- The DetDesc framework is functional (transient classes, XML DTD, XML converters, editor, etc)
- Several sub-detectors are already describing their structure and geometry using the provided framework
  - Calorimeters (HCAL, ECAL,...), RICH, ...
- Other sub-detectors (+Magnet) coming soon
- Visualization based on OnX (GaudiLab)
- Conversion to Geant4 exists
  - Ready to start tests with G4 in Gaudi

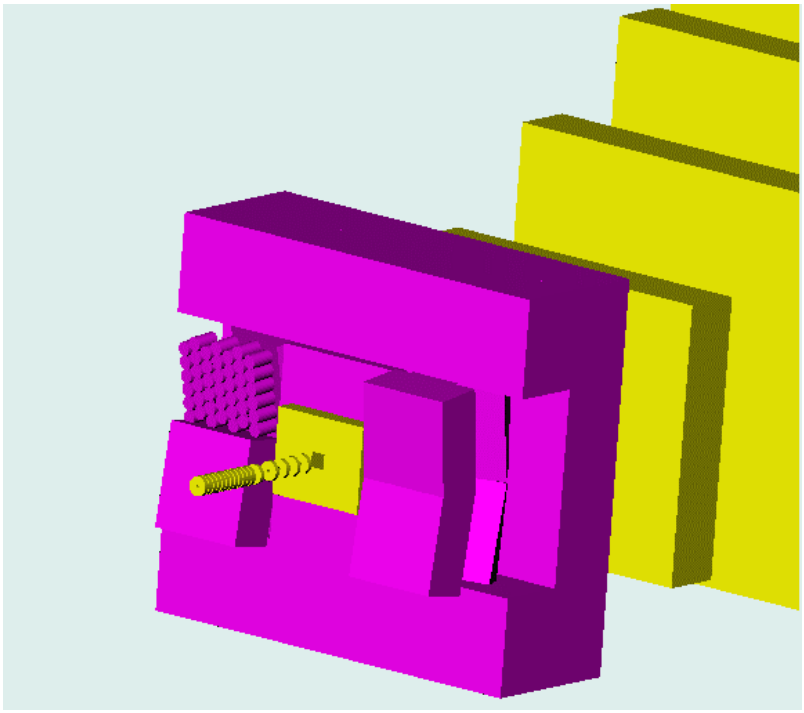
Example



- The complete LHCb detector geometry

Example (2)

- Detailed view of the VELO and RICH I



Plans

- Continue with the consolidation and deployment of the DetDesc framework
  - Most of sub-detector code still needs to be developed. We will discover new use-cases.
- Study the possibility of generating XML converters using data dictionary services
- Integration with Conditions DB
  - Uniform access from Algorithms
  - Wish to format conditions data in XML