



XML in LHCb Detector Description Framework

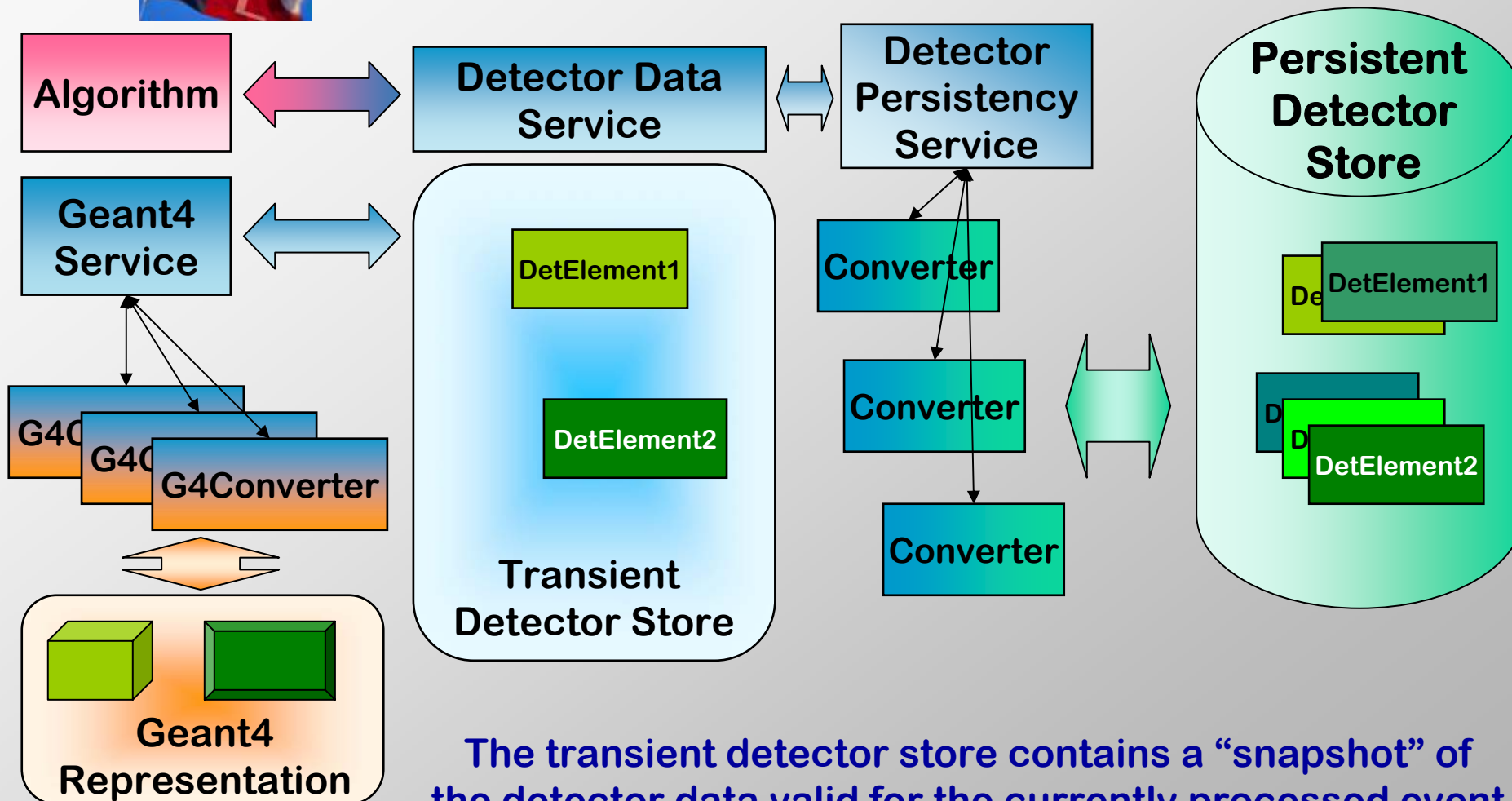
**Radovan Chytracsek
CERN**



- ⇒ Component of the Gaudi framework
- ⇒ Inherits the Gaudi architecture decisions
- ⇒ Detector description provides
 - ⇒ Single source of detector data for all clients
 - simulation, reconstruction, analysis, test beam
 - ⇒ It is not detector geometry only!
 - logical detector structure, geometry & positions, materials, mapping electronic channels to detector cells, detector control data needed for reconstruction, calibration and alignment data
 - ⇒ Versioning of all detector data based on event time, run #, etc.
- ⇒ Persistency of detector description in XML format
 - ⇒ XML may be stored in files on disk , on the Web or in a DBMS

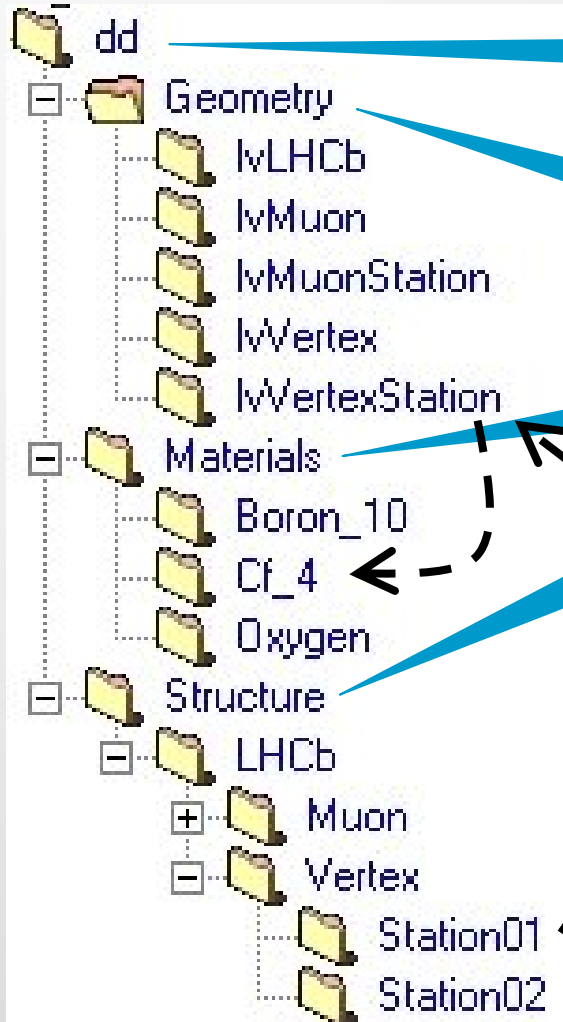


Detector Data Store





Transient Detector Store



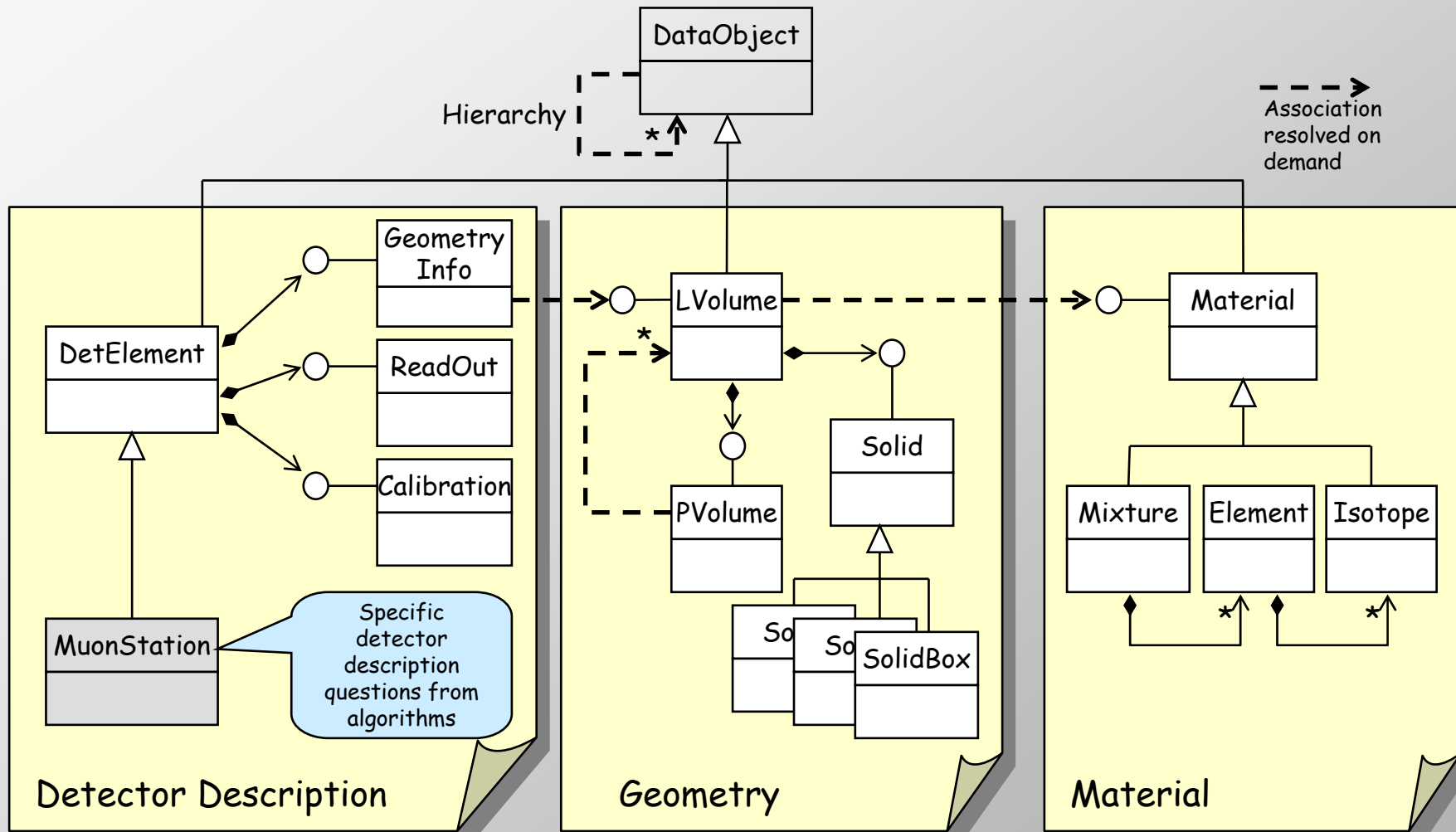
Root of the store

Top level catalogues

- Tree-like structure
- Items identified by a logical name
- Updated on demand
- Automatic update when a new event is loaded



Loading on demand



Choices made for XML

⇒ XML DOM is not used

⇒ memory expensive

⇒ we don't write data back to XML (yet)

⇒ we have already the transient hierarchical object model in Gaudi

⇒ SAX is the choice for implementation

⇒ easy to implement and extend

⇒ we loose structure of XML documents

⇒ IBM XML4C parser

- ⇒ Simple parser for evaluation of expressions
- ⇒ Expressions can be composed of
 - ⇒ integer and floating point numbers
 - 100 | 100. | .05 | 0.1 | 1.34-e12|-23
 - ⇒ operations: +, -, *, /, unary +|- , exponent ^
 - ⇒ parenthesized expressions: 1.4 * (23.4-e12 / 1.8)
- ⇒ Result is always evaluated to double value
- ⇒ Operator precedence from left to right:
 - ⇒ [()] [unary +|-] [^] [*|/] [+|-]
- ⇒ By default checks for units in expressions

CLHEP Units Compatibility

- ⇒ DTD for XML detector description defines units a la CLHEP
- ⇒ Units **MUST** be used where required
 - ⇒ XML converters assume the use of units
 - ⇒ In case the units are missing processing stops and an exception is thrown.
- ⇒ Use expressions parser where needed with check for units enabled
- ⇒ Examples
 - ⇒ `23*&cm; | 12*&volt; | 23.6*&g;/&cm3;`



C++ to XML

C++	XML
Class	XML Element
Primitive class data members	Element attributes
Complex class data members	Element embedded in Content Model
Reference	Element reference with hyperlink and class ID of a target object



DDDB Tag

```
<!ENTITY % elems "detelem | logvol | material">
<!ELEMENT DDDB (catalog | catalogref | %elems;)+>
<!ATTLIST DDDB version CDATA "3.0">
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DDDB SYSTEM "xml.db.dtd">
<DDDB>
  <catalog name="dd">
    <catalogref href="structure.xml#Structure"/>
    <catalogref href="materials/materials.xml#Materials"/>
    <catalogref href="geometry.xml#Geometry"/>
  </catalog>
</DDDB>
```



Detector Element Tag

```

<!ENTITY % detelemdata "author?,version,geometryinfo">
<!ENTITY % nmclid "name ID #REQUIRED classID CDATA #REQUIRED">
<!ELEMENT detelem (%detelemdata;, detelemref*, specific?)>
<!ATTLIST detelem %nmclid;>

```

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DDDb SYSTEM "../xml/db.dtd">
<DDDB>
  <detelem classID="2" name="LHCb">
    <author>RCH</author>
    <version>0.1</version>
    <geometryinfo>
      <lvname name="/dd/Geometry/LHCb/lvLHCb"/>
    </geometryinfo>
    <detelemref classID="9999" href="VertexRich1/Vertex/structure.xml#Vertex"/>
    <detelemref classID="2" href="VertexRich1/Rich1/structure.xml#Rich1"/>
    <detelemref classID="2" href="TrackerRich2/Tracker/structure.xml#Tracker"/>
    <detelemref classID="2" href="TrackerRich2/Rich2/structure.xml#RICH2"/>
    <detelemref classID="8900" href="CaloMuon/Calo/structure.xml#Ecal"/>
    <detelemref classID="2" href="CaloMuon/Muon/structure.xml#Muon"/>
  </detelem>
</DDDB>

```



Inheritance in XML

```

<!DOCTYPE DDDb SYSTEM "xmldb.dtd" [
  <!--Number of stations in Vertex detector-->
  <!ELEMENT SiTankRadius EMPTY>
  <!ATTLIST SiTankRadius n CDATA #REQUIRED>
  <!ELEMENT DiodePitch EMPTY>
  <!ATTLIST DiodePitch n CDATA #REQUIRED>
  <!ELEMENT ReadoutPitch EMPTY>
  <!ATTLIST ReadoutPitch n CDATA #REQUIRED>
]
<DDDB>
  <detelem classID="9999" name="Vertex">
    ...
    <geometryinfo>
      <lvname name="/dd/Geometry/lvVertex" />
      <support name="/dd/Structure/LHCb"> <npath value="pvVertex_0" /> </s
    </geometryinfo>
    <detelemref classID="2" href="#VStation01"/>
    <detelemref classID="2" href="#VStation02"/>
    <specific>
      <SiTankRadius n='17' /> <DiodePitch n='0.0025' /> <ReadoutPitch n='0.0050' />
    </specific>
  </detelem>
</DDDB>

```

Definition
of the user
XML tags

Detector
specific
data



Logical Volume Tag

```

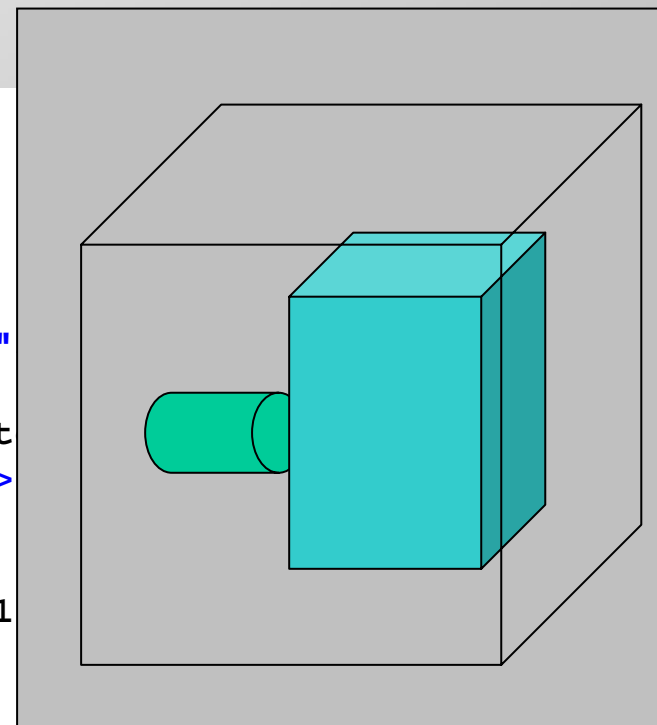
<!ENTITY % solid "(box | cons | sphere | tubs | trd |
                  union | subtraction | intersection)">
<!ELEMENT logvol (%solid;, (physvol | paramphysvol)*)>
<!ATTLIST logvol name ID #REQUIRED
                material CDATA #REQUIRED
                classID CDATA #FIXED "1100">

```

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DDDDB SYSTEM "../..//xmlldb.dtd">
<DDDB>
  <logvol name="lvVertexRich1" material="Vacuum">
    <box name="VertexRich1Box"
        sizeX="5000*&mm;" sizeY="5000*&mm;" sizeZ="
    <physvol name="pvVertex"
        logvol="/dd/Geometry/LHCb/Vertex/lvVert
    <posXYZ x="0*&mm;" y="0*&mm;" z="-1000*&mm;" />
  </physvol>
  <physvol name="pvRich1"
        logvol="/dd/Geometry/LHCb/Rich1/lvRich1
    <posXYZ x="0*&mm;" y="0*&mm;" z="2000*&mm;" />
  </physvol>
</logvol>
</DDDB>

```





Physical Volume Tags

```
<!ELEMENT physvol ((posXYZ|posRPhiZ|posRThPhi), (rotXYZ|rotAxis)?)>
<!ATTLIST physvol name ID #REQUIRED logvol CDATA #REQUIRED>
```

```
<!ELEMENT paramphysvol (physvol, (posXYZ|posRPhiZ|posRThPhi))
<!ATTLIST paramphysvol number CDATA #REQUIRED>
```

```
<logvol name="lvPVStation" material="Vacuum">
  <tubs name="pvStationTubsPV" sizeZ="0.15*&mm;"
    outerRadius="60*&degree;" innerRad...
    startPhiAngle="-1.5*&degree;" endPhiAngle="181.5*&degree;" />
```

```
<paramphysvol number="6">
  <physvol name="waferPV" logvol="/dd/Geometry/LHCb/Vertex/lvWafer">
    <posXYZ x="0*&mm;" y="0*&mm;" z="-1.5*&mm;" />
    <rotXYZ rotX="0*&degree;" rotY="0*&degree;" rotZ="0*&degree;" />
  </physvol>
  <posXYZ x="0*&mm;" y="0*&mm;" z="0*&mm;" />
  <rotXYZ rotX="0*&degree;" rotY="0*&degree;" rotZ="53.25*&degree;" />
</paramphysvol>
```

```
<physvol name="pvwaferStrange" logvol="/dd/Geometry/LHCb/Vertex/lvWafer">
  <posXYZ x="0*&mm;" y="0*&mm;" z="-1.5*&mm;" />
  <rotXYZ rotX="0*&degree;" rotY="0*&degree;" rotZ="0*&degree;" />
</physvol>
```

```
</logvol>
```

Simple
physical
volume

Solid Tags

```

<!ENTITY % simplesolid "(box | cons | sphere | tubs | trd)">
<!ELEMENT box ((posXYZ | posRPhiZ | posRThPhi), (rotXYZ | rotAxis)?)?>
<!ATTLIST box name ID #REQUIRED
             sizeX CDATA #REQUIRED sizeY CDATA #REQUIRED sizeZ CDATA #REQUIRED>
<!ELEMENT union (%simplesolid;)+>
<!ATTLIST union name ID #REQUIRED>

```

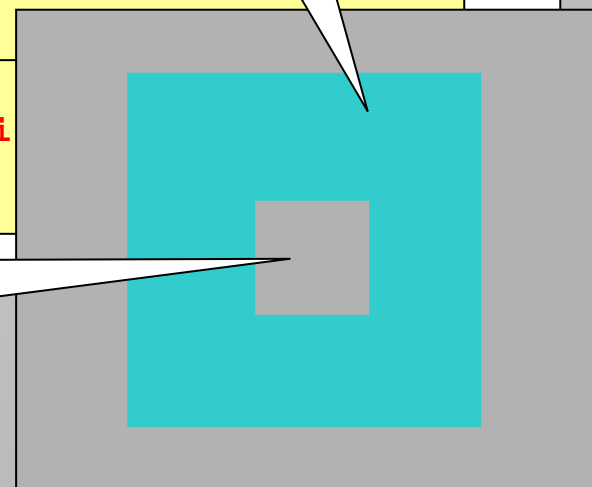
```

<logvol material="Vacuum" name="lvEcalOuter"
  <subtraction name="boxEcalOuter">
    <box name="boxEOMain"
      sizeZ="432*&mm;" sizeX="7933.44*&mm;" sizeY="6445.92*&mm;">
      <posXYZ z="0*&mm;" y="0*&mm;" x="0*&mm;" />
    </box>
    <box name="boxEOSubtracted"
      sizeZ="432*&mm;" sizeX="2479.20*&mm;" sizeY="2479.20*&mm;">
      <posXYZ z="0*&mm;" y="0*&mm;" x="0*&mm;" />
    </box>
  </subtraction>
</logvol>

```

**Main
or
reference
solid**

**Subtracted
solid
placed relative
to the main solid**



Position Tags

Cartesian coordinate system

```
<!ELEMENT posXYZ EMPTY>  
<!ATTLIST posXYZ x CDATA #REQUIRED y CDATA #REQUIRED z CDATA #REQUIRED>
```

Cylindrical coordinate system

```
<!ELEMENT posRPhiZ EMPTY>  
<!ATTLIST posRPhiZ r CDATA #REQUIRED phi CDATA #REQUIRED z CDATA #REQUIRED>
```

Spherical coordinate system

```
<!ELEMENT posRThPhi EMPTY>  
<!ATTLIST posRThPhi r CDATA #REQUIRED theta CDATA #REQUIRED phi CDATA #REQUIRED>
```


Rotation Tags

Rotation around the axes X,Y and Z

```
<!ELEMENT rotXYZ EMPTY>
```

```
<!ATTLIST rotXYZ rotX CDATA #REQUIRED rotY CDATA #REQUIRED rotZ CDATA #REQUIRED>
```

Rotation using vector and angle

```
<!ELEMENT rotAxis EMPTY>
```

```
<!ATTLIST rotAxis axTheta CDATA #REQUIRED axPhi CDATA #REQUIRED  
angle CDATA #REQUIRED>
```



Material Tag

```

<!ENTITY % materialdata "a?,z?,n?,density,x0?,lambda?">
<!ELEMENT fraction (atoms | mass)>
<!ELEMENT atoms EMPTY>
<!ATTLIST atoms n CDATA #REQUIRED>
<!ELEMENT mass EMPTY>
<!ATTLIST mass n CDATA #REQUIRED>
<!ELEMENT material (%materialdata;, (materialref)*)>
<!ATTLIST material
    %nmclid;
    form (isotope | element | mixture) #IMPLIED
    temperature CDATA #IMPLIED
    pressure CDATA #IMPLIED
    state (solid | liquid | gas) #IMPLIED
>
<!ELEMENT a (#PCDATA)>
<!ELEMENT z (#PCDATA)>
<!ELEMENT n (#PCDATA)>
<!ELEMENT density (#PCDATA)>
<!ELEMENT x0 (#PCDATA)>
<!ELEMENT lambda (#PCDATA)>

```



Material Tag

```
<material name="Bor_10" classID="&isotope;">
  <a>10.000*&g;/&mole;</a>
  <z>5.000</z>
  <density>2.3400*&g;/&cm3;</density>
  <x0>0.0000e+00*&cm;</x0>
  <lambda>0.0000e+00*&cm;</lambda>
</material>
```

```
<material name="Water" classID="&mixture;">
  <density>1.0000*&g;/&cm3;</density>
  <materialref classID="&element;" href="elements.xml#Hydrogen">
    <fraction>
      <atoms n="2"/>
    </fraction>
  </materialref>
  <materialref classID="&element;" href="elements.xml#Oxygen">
    <fraction>
      <atoms n="1"/>
    </fraction>
  </materialref>
</material>
```

⇒ Status

- ⇒ XML based det. descr. is the integral part of Gaudi
 - Working on NT, Linux

⇒ Future steps

- ⇒ Versioning
- ⇒ Population of the XML DDDB from the LHCb detector description based on Geant3
- ⇒ Define the XML format for conditions
 - calibration, alignment, channel maps, slow control, ...
- ⇒ Implement the XML DDDB in the conjunction with a “real” DB (RD45 Conditions/DB, Oracle, ...)

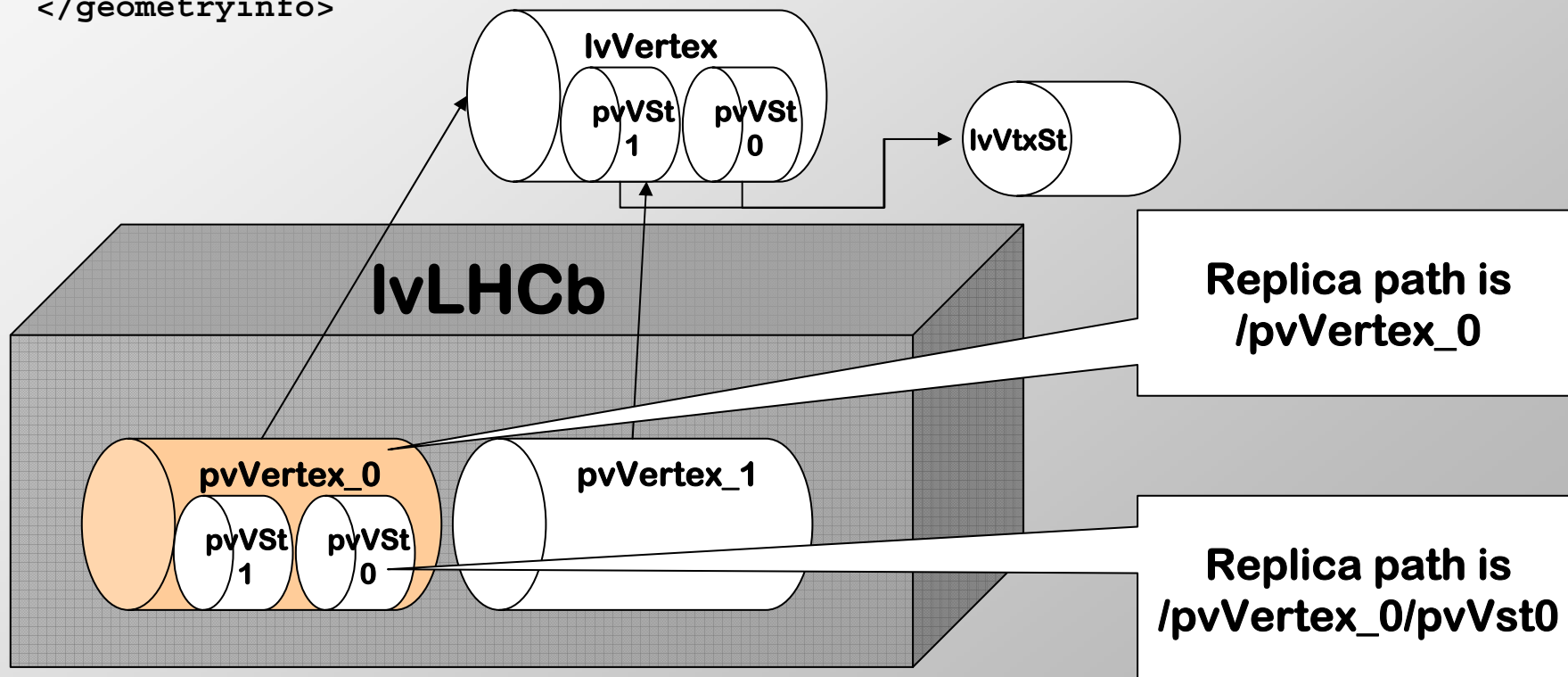


Geometry Info Tag

```

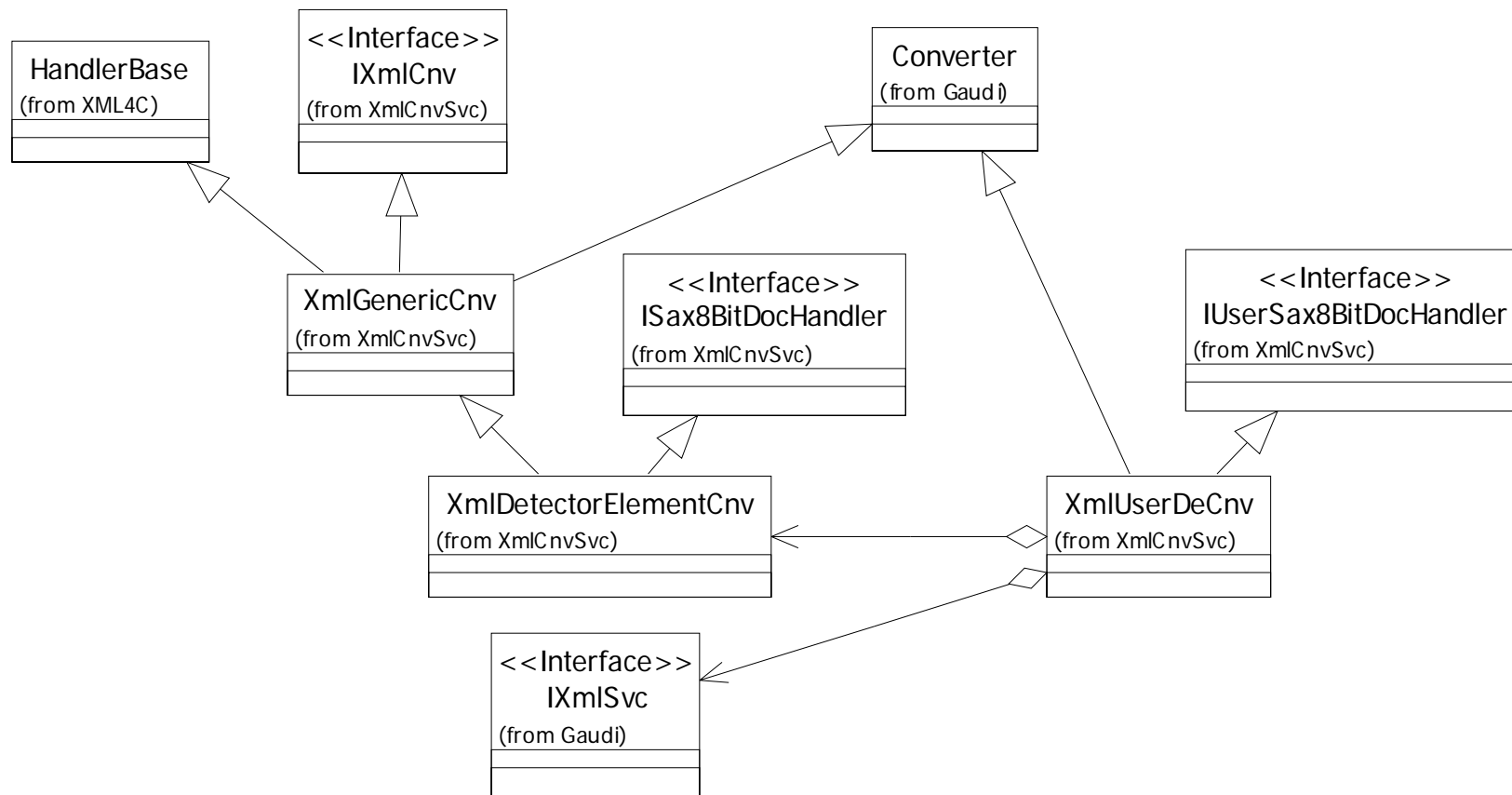
<geometryinfo>
  <lvname name="/dd/Geometry/lvVertex" />
  <support name="/dd/Structure/LHCb"> <npath value="pvVertex_0" /> </support>
</geometryinfo>

```

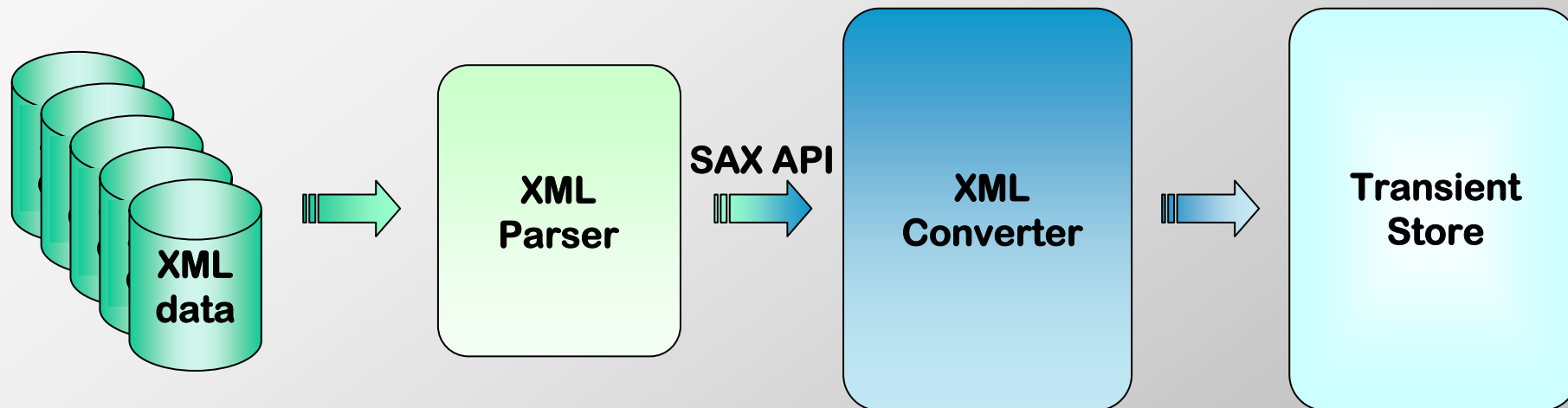




XML Converters Model



XML Conversion



- It can be any XML parser supporting SAX API for C++
- SAX - Simple API for XML



XML Briefly

⇒ XML is the

- ⇒ W3C Recommendation
 - XML 1.0
- ⇒ subset of SGML
 - 20% complexity
 - 80% flexibility of SGML
- ⇒ Application independent data interchange format

⇒ Safe investment in data and manpower

- ⇒ increasing support for XML in the industry
- ⇒ maintained outside CERN

⇒ XML features

- ⇒ User defined tags
- ⇒ Data self described
- ⇒ Hyperlinks
- ⇒ Properly nested tags forming a tree structure
- ⇒ Data separated from behavior

⇒ Problems with XML

- ⇒ C++ to XML mapping
- ⇒ Inheritance
- ⇒ Performance
- ⇒ Data Management



Persistent Detector Store

- ⇒ The Gaudi architecture shields end users from a persistent technology
 - ⇒ Allows co-existence of different persistency technologies
 - technology specific code only on well defined places
 - services and converters
 - ⇒ We don't know which technologies will be available in 2005

- ⇒ We use XML as our persistent data format
 - ⇒ XML can be stored in files, WWW, database