DD4hep Status

HEP detector description supporting the full experiment life cycle
• Motivation and Goals

=> Introduction / Reminders

• Concepts and Design

• Going to the 'real world'

• Summary
Motivation and Goal

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

- Description of a tree-like hierarchy of “detector elements”
  - Subdetectors or parts of subdetectors
- Detector Element describes
  - Geometry
  - Environmental conditions
  - Properties required to process event data
  - Optionally: experiment, sub-detector or activity specific data
• Motivation and Goals

• Concepts and Design

=> Reminder

• Going to the 'real world'

• Summary
DD4Hep - The Big Picture

Compact description xml

Detector constructors python c++

Geometry Display

Generic Detector Description Model Based on ROOT TGeo c++

Conditions DB

Alignment / Calibration

Extensions where required

GDML Converter xml

TGeo => G4 converters

Reconstruction Extensions

Analysis Extensions

SLIC [SiD Simulation]

Geant4 Program

Reconstruction Program

Analysis Program
- Motivation and Goals
- Concepts and Design
- Status of Ongoing Work
  - Simulation
  - Reconstruction
- Future work – next steps
- Summary
Simulation: Generic Geant 4 Gateway
(Markus Frank)

- **Simulation** = Geometry + Detector response + Physics
- **Attempt for formalization of Geant4**
  - Ideally: configuration without user code
  - Extensive usage of plugins
- **DDG4**
  - Bootstrap Geant4 from DD4hep in memory geometry
  - Configure using XML, python or Cint (ROOT 5)
  - Configure Geant4 actions, physics-list, processes, particle constructors, sensitive actions, I/O etc using module palette
Simulation: DDG4

(Markus Frank)

- **Concept**
  - Walk through the geometry starting from “world”
  - Convert the geometry from ROOT to Geant4
  - Instantiate sensitive detectors from palette [similar to palette of detector constructors]
  - Instantiate physics list, -constructors and -processes
  - Start simulating

- **Processing chain is implemented**
  - Validation in progress – time consuming process

- **Palette of sensitive detectors**
  - Is limited to some existing examples
  - Hope: palette gets populated by 'donations' of clients
Geant 4 Gateway using slic (1)
(Norman Graf, Jeremy McCormick)

- CERN/LCD follow suggestion to benefit from the 'slic' simulation framework (SiD)
  - Convert DD4hep geometry to LCDD notation (xml)
  - GDML: materials, solids, limit sets, regions logical-, placed volumes / physical volumes
    - Fields
    - Sensitive detector information

- Collaboration with SiD/SLAC (N.Graf, J.McCormick)
  - Introduce new segmentations, identification of deficiencies

- F.G. successfully simulated ILD example det.
Detector Segmentations
(Christian Grefe)

- Are the description of the sensitive detector regions
- Define encoding of the location of energy depositions (hits) in a simulation program
  - Encoding depends on the sensitive area(s) and detector technology
    Si Tracker: Side / Layer / Wafer / x-y local coordinates
  - Bi-directional
    volume ID in hit $<==>$ full resolution of
    - detector/component
    - local coordinate
  - But there are also less obvious segmentations mostly projective segmentations (e.g. calo towers)
Detector Segmentations

(Christian Grefe)

- Essential components to implement
  - Simulation programs
  - Digitization / Reconstruction applications
  - Bridge between the two worlds

- Shared, independent package
Reconstruction Interfaces
(Christian Grefe, Astrid Munich)

• Set of utilities to easy for users the retrieval of specialized geometrical questions
  – Work connected to segmentations
  – Transparently chain reoccurring call sequences
  – Precompute and cache information difficult or expensive to obtain but regularly needed
    [Implemented using extension mechanism]

• Astrid mimicked the GEAR-TPC model as in Marlin
  – Work done ~ year ago
  – Need to restart support for tracking detectors

• Christian was working on CALO interfaces
  – Layered detectors consisting of segmented active modules
Documentation

- http://aidasoft.web.cern.ch/DD4hep
- https://svnsrv.desy.de/basic/aidasoft/DD4hep/trunk
- In the svn doc area
  - DD4hepManual.pdf
    core API: 37 pages
  - DDG4Manual.pdf
    simulation: 25 pages
  - First issues
- Doxygen documentation
- Motivation and Goals
- Concepts and Design
- Implementation
- **Future work – next steps**
- Summary
Alignment and Detector Conditions

(Markus Frank)

- Less an issue during the experiment design phase
  - Less important for the communities designing detectors
  - Selling argument for existing (e.g. LHC) experiments
- Important topic to interpret event data from existing ('real') detectors
  - Necessity to deal with imperfections
    - Geometry => (Mis)Alignment
    - Anomalous conditions
      - Pressures, temperatures
        => Gains, refractive indices
        => Contractions, expansions
Other Upcoming Work [2014]

- Validate the two simulation paths
  - Verify the translation mechanisms
  - Help new clients to use the infrastructure
- Extend, validate and support work on reconstruction interfaces
  - Currently concrete only for calorimetry
  - Tracking support starting (=> see talk of F.Gaede)
- Must come to gears with Mokka replacement
  - ILD simulation program: support will disappear
  - Test of concept done. Bulk driver translations missing
  - Item was on the list already last year
- Support for new clients
DD4hep Clients

- Linear Collider Detector community (ILD+SiD)
  - Work group established several months ago
    - M.Frank\(^{(1)}\), F.Gaede\(^{(2)}\), C.Graefe\(^{(1)}\), N.Graf\(^{(3)}\), J. McCormick\(^{(3)}\), N.Nikiforou\(^{(1)}\), C.Rosemann\(^{(2)}\), A.Sailer\(^{(1)}\)

- Clients evaluating DD4hep
  - LheC contact: P. Kotzka\(^{(2)}\)
  - FHC contact: C. Helsens\(^{(1)}\)
  - LHCb contact: M. Clemencic\(^{(1)}\)

\(^{(1)}\) CERN  \(^{(2)}\) Desy  \(^{(3)}\) SLAC
Summary

- The DD4hep core was consolidated
- On the track for simulation framework
  - 2 paths for ILD, generic framework else
- Support and developments event data processing beyond simulation ongoing
- We see interest from the HEP community
  - Clients want to leverage development effort to common infrastructure projects (LheC, FHC, LHCb)
Questions and Answers
Backup slides
Implementation: Geometry

Subdetector Hierarchy (Tree)

- Detectors
  - DetectorElement
    - PlacedVolume
      - [TGeoNode]
        - [TGeoMatrix]
        - [TGeoBox] [TGeoCone] [TGeoTube]

Subdetector status (conditions)

- Alignment
- Conditions
- Readout
- Visualization
- Segmentation

GDML content

Geometry
DDG4 Configuration Example (Incomplete)

```xml
<sequences>
  <sequence name="Geant4EventActionSequence/EventAction">
    <properties Control="true"/>
    <action name="Geant4Output2ROOT/RootOutput"/>
    <properties Control="true" Output="simple.root"/>
  </action>
</sequence>

<sequence name="Geant4GeneratorActionSequence/GeneratorAction">
  <action name="Geant4ParticleGun/Gun">
    <properties .... />
  </action>
</sequence>

<sequence sd="SiVertexBarrel" type="Geant4SensDetActionSequence">
  <properties Control="true"/>
  <filter name="GeantinoRejector"/>
  <filter name="EnergyDepositMinimumCut"/>
  <action name="Geant4SimpleTrackerAction/SiVertexBarrelHandler">
    <properties Control="true"/>
  </action>
</sequence>

....
</sequences>
```

**Geant4 event action setup**

**Geant4 generator action setup**

**Sensitive detector setup**

Instance type from palette

Instance name for reference
Client Extensions

- Provide flexible functionality to solve reconstruction and analysis problems
- Approach to deal with the “unforeseeable”
- Motivated by the fact that Different use cases require different functionality
  - Example: Optimization of coordinate transformations
    local TPC hit to experiment coordinates
    => specialized data required
    (cache of precomputed results)
  - Need to extend the detector element's data
Non Transparent Design Decisions

- Things which look of small importance => but have significant impact on users
- Units: TGeo: GeV/cm/sec [CKM] Geant4: MeV/mm/ns
  - Consequently apply units
    TGeoBBox(10*tgeo::mm, 10*tgeo::mm, 10*tgeo::mm)
    G4Box(10*CLHEP::mm, 10*CLHEP::mm, 10*CLHEP::mm)
  - To get raw number always divide (both TGeo, Geant4): g4Box->GetXHalfLength() / CLHEP::mm
- Transformations
  - CLHEP is a dead end (support ?)
  - Use ROOT::Math vectors & matrices to build geometries very similar (but not identical!)
    started from same code bases, then deviated
  - Used by most LHC experiments
Porting Mokka Drivers
(Frank Gaede, Andre Sailer, Shaojun Lu)

- **Aim is to investigate the translation of Mokka drivers 'with minimal effort'** (Model: ILD_o1_v05)
  - Create compact xml file from Mokka database
    - Serves as input to DD4hep driver
  - Translate G4 in driver calls to DD4hep calls
    - G4Shape, G4LogicalVolume, ... => Shape, Volume, ...
    - Created 'detector constructor' (~driver)
    - Leave as much unchanged as possible
  - **Experience: VXD, SIT, TPC, SET, beamcal and HCAL barrel**
    - Tracker driver simple, calorimeters much more complicated
    - Parameters change in Mokka at run-time,...
    - Automation without brain is difficult, and will be hard to maintain
    - Will need some policy how to avoid parameter anarchy
Porting existing Mokka Drivers