EDM Toolkit - PODIO

and potential topics for AIDA++

F. Gaede, DESY

WP3 Phone Meeting, May 7, 2019
PODIO:
- milestones and deliverables
- new features
- pLCIO
- Next Steps

potential continuation of projects for AIDA++
- PODIO
- advanced tracking ACTS
- MarlinMT
### Status: Milestones and Deliverables

<table>
<thead>
<tr>
<th>Name</th>
<th>What</th>
<th>When</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS19</td>
<td>Design document for EDM Toolkit</td>
<td>M14</td>
</tr>
<tr>
<td>MS90</td>
<td>Application of EDM Toolkit to LC</td>
<td>M44</td>
</tr>
<tr>
<td>D3.4</td>
<td>Event Data Model Toolkit</td>
<td>M40</td>
</tr>
</tbody>
</table>

- **status**
  - all Milestones and Deliverables **reached on time**
  - plan to continue improving PODIO, nevertheless
  - some minor feature development done in context of FCC
Recent Developments in PODIO

- moved Github repository to: https://github.com/aidasoft/podio
- added some standard templates from iLCSoft for release notes, issues, etc.
- addressed a few issues needed for MS90
  - add CollectionBase::size() method
  - implement vector member streaming
- some minor bugs and issues fixed

v00-09

- 2018-12-20 Frank Gaede (PR#39)
  - add some fixes and improvements
    - fix forward declarations in Object template when using a namespace for the EDM
    - fix array getter names when using the get/set syntax
    - add missing treatment for include statements in component's header files
    - handle array members in ostream operators
  - add CollectionBase::size() member function
    - allows to access collection size w/o knowing the concrete type
    - method is already generated in implementation classes

- 2018-12-06 Frank Gaede (PR#38)
  - add code generation for I/O of vector members - vector members are treated analogous to the reference vectors,i.e. streamed as one large vector per collection
  - updated tests/datamodel accordingly (using clang-format)

- 2018-11-30 Frank Gaede (PR#37)
  - handle references and vector members in collection's ostream operators

- 2018-11-30 Frank Gaede (PR#36)
  - add github templates for release notes, issues and contribution guidelines
  - add ReleaseNotes.md - contains all commit logs so far (v00-08)
Next Steps and Plans

implement direct binary I/O making use of array-of-POD - still pending

- use new ‘thread safe’ implementation of the SIO layer from LCIO (see talk R.Ete)
- like to benchmark the reading performance against current ROOT I/O
  - Note: plan to also benchmark pLCIO against existing LCIO

modify the treatment of constness

- current implementation has extra types for `const` objects, e.g. ConstMCParticle
- prototype implementation exists that ensures constness transparently (B.Hegner)
  - still just needs to be merged with master branch

have HDF5 I/O layer

- successfully applied for GSoC project under umbrella of HSF (G.Stewart)
potential continuation of projects in AIDA++
actively used by FCC

interest/ plans to move to PODIO by ILC, CLIC, CEPC

PODIO strong candidate for continuation in AIDA++!

potential work items:

- implementation of high performance I/O (SIO, Root, others)
- integration with application framework (Marlin, Gaudi, others)
- general improvements?
- . . .
advanced tracking - ACTS

- currently ‘used’ by FCC
- strong interest ILC, CLIC, Belle II, ATLAS, ...
- ACTS seems a vary natural candidate for AIDA++
- potential work items:
  - implementation of pattern recognition tools
  - integration in LC-framework
  - application to Belle-II
  - ...

F. Gaede, DESY
MarlinMT

- Marlin is used by ILC, CLIC, CEPC and Calice, LCTPC, EU Telescope, ... 
- recently started development of **MarlinMT** for parallel event processing
  - see presentation at annual meeting in Oxford
  - quite some progress since: implementation of a scheduler for full parallel event processing and started to look into porting actual processors to run in parallel
- MarlinMT also seems a good candidate for AIDA++
  - could be a light weight alternative to Gaudi for conceptual studies at future accelerators and for smaller experiments/test beams
- potential work items:
  - integration with PODIO (replacement of LCIO)
  - better integration with DD4hep:
    - access to conditions data and alignment
  - continuation of parallelization
  - ...