ATLAS Strategy for the Data Dictionary & Persistency

David R. Quarrie

Lawrence Berkeley National Laboratory

DRQuarrie@LBL.Gov

Background

Following CSG decision in Nov 2002 to terminate the ADL project we needed to develop an alternative strategy and workplan

Tentative plan proposed at that time was to use LCG Dictionary in conjunction with parsing C++ header files

- Already being investigated by CMS

Needed to understand options and timescales to see whether short-term interim solutions needed

Ad hoc working group setup

- Christian Arnault, Paolo Calafiura, Julius Hrivnac, David Malon, David Quarrie, SriniRajagopalan, RD Schaffer, Craig Tull

Timecales

- Release 7.0.0 (June 2003)
  - Full persistency available
  - Reworked Event Data Model
- Athens Workshop (May 2003)
  - Stated desire to have "some" persistency available
  - Working backwards, that implies end of March

Possible Implementation Strategies

- Extend AthenaROOT to add generic ROOTCINT support
  - Extends our current short-term persistency solution
  - An interim solution only
- Use POOL persistency but use ROOTCINT to parse C++ header files (i.e. bypass LCG Data Dictionary)
  - ROOT now supports parsing of "foreign" classes
  - Seen as an interim solution only
  - Packaged as much as possible like the final long-term solution
- Use POOL persistency & LCG Data Dictionary in combination with parsing C++ header files
  - Final long-term solution

Extend Athena ROOT

Currently converters are hand-written

Original intention was to have ADL generate them

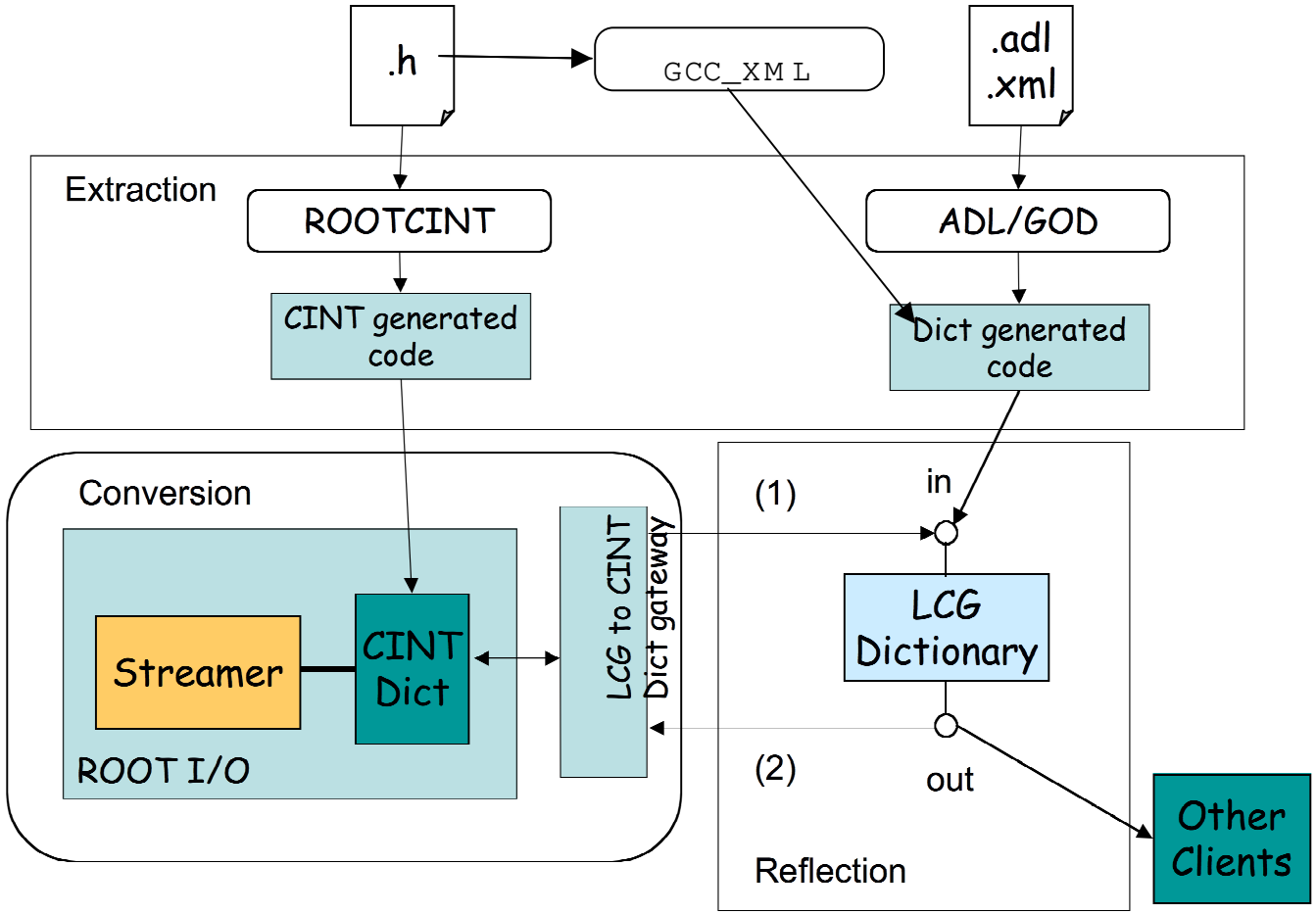
Could use ROOTCINT parsing of C++ header files in conjunction with a single generic converter

Seen as a totally throw-away solution

- None of the development would be applicable to the final desired long term solution

Solution rejected

LCG Data Dictionary



Use GuineaEvent package as testbed

- Developed for ADL validation/testing
- Covers most of ATLAS requirements
  - IdentifiableContainer, STL containers, DataLink<T>, etc.
- Identify the problems with parsing of GuineaEvent header files by candidate parsers
  - ROOTCINT
  - Gcc_xml
  - Doxygen
  - JavaCC

GuineaEvent & ROOTCINT

- Problems with GaudiKernel/Kernel.h
  - Fails on a "long long" datatype
- Problems with CLHEP classes
  - Fails on HEPSTL::vector instead of std::vector
- Problems with Boost
  - Heavily templated classes
- Probably localizable to StoreGate classes
- ROOT team working on fixes to some of these
- Not a simple "out of the box" solution

GuineaEvent & gcc_xml

- Gcc_xml is an OpenSource project to provide a XML back-end to GCC
  - Small but growing community of users & developers
  - Requires gcc 3.x
- Prototype available in contrib. area of POOL from Zhen Xie (CMS)
- Ran GuineaEvent through gcc_xml
  - Successfully created XML for 220 (!) classes
  - Didn't go as far as loading the LCG Dictionary

GuineaEvent & Doxygen/JavaCC

- Doxygen
  - Can generate XML as well as HTML
  - Evaluation showed XML less suitable than gcc_xml for loading data dictionary
- JavaCC
  - Used for ADL project
  - Is a "compiler-compiler" that can cope with many grammars
  - Intended to try the GuineaEvent classes in conjunction with the C++ grammar
    - Didn't get to this

Summary of Parsers

- Problems with ROOTCINT parser
  - Concerns about using it for an interim solution
- Gcc_xml looks like the best XML generator
- Prototype LCG Data Dictionary code needed to be reworked
  - Orsay offered to do this
  - Timescale – a couple of weeks
- The situation looks good from the perspective of loading of the LCG Dictionary
  - Available by mid-Feb.

Data Dictionary coupling to POOL

POOL 0.3 release in December 2002 had severe restrictions

- Essentially full support for ROOT TObject classes
- Otherwise simple, untemplated, classes with no pointers

However, in talking to Torre, Dirk, Pere and Markus Frank it seems that good progress has been made in the interim

- Foreign class support now includes std::vector, std::list

Reasonable optimism for good coverage in Feb POOL release

High optimism for full ATLAS capability in May POOL/SEAL release

Strategy Decision

- Aim for full long-term solution
  - POOL persistency with LCG Data Dictionary loaded by gcc_xmlparser for C++ header files
- Identify several work areas
  - XML parser & LCG Data Dictionary Loader
    - Orsay (Christian et al.)
  - Athena Data Dictionary Service Integration
    - LBNL (Paolo et al.)
  - Athena POOL Conversion Service
    - BNL (David Adams et al.)
  - CMT pattern integration
    - SIT (Christian et al.)
  - POOL Integration
    - Try to augment effort with BNL people?

Other Work Areas

- Revisit Physics EDM
  - Task force being setup to revisit reconstruction/HLT Algorithm granularity and associated EDM
  - Goal is to incorporate design feedback for Release 7.0.0
- Document constrained set of C++ that will be supported by the ATLAS EDM
  - The GuineaEvent classes embody perhaps 90% of this
- Understand how to customize the dictionary (e.g. to indicate data members that should not be persistified)
  - Proposal to use XML "strategy" files
  - Still need to look into this in detail

Athens Workshop

I still can't give a positive response to the request that there be "some persistency" support available in time for Athens

When I asked Fabiola & David Rousseau whether "4 or 40 classes" were desired, the answer was 20-40

- This is clearly not feasible using hand written AthenaROOT converters

The POOL/LCG solution doesn't look like it'll be ready in time

Summary

- We now have a new strategy in place
  - Aim for full long-term solution
    - POOL persistency with LCG Data Dictionary loaded by gcc_xmlparser for C++ header files
- We have identified work areas
- We'll revisit the strategy following Feb POOL release
- Inevitably some schedule slippage following ADL cancellation
- Goal is complete EDM support for Release 7.0.0
- Still unable to commit to satisfying request for Athens Workshop