



Contribution ID: 427

Type: poster

Persistence for Analysis Objects

Thursday, 30 September 2004 10:00 (0 minutes)

There are two kinds of analysis objects with respect to their persistent requirements:

* Objects, which need direct access to the persistency service only for their IO operations (read/write/update/...): histograms, clouds, profiles, ...

All Persistency requirements for those objects can be implemented by standard Transient-Persistent Separation techniques like JDO, Serialization, etc.

* Objects, which need direct access to the persistency service for some of their standard operations: NTuples, Tags,.... It is not feasible to completely separate Transient and Persistent form of those objects. Their Persistency should be tightly interfaced with their transient form. One possibility is to directly implement a persistent extension of those objects for each persistency mechanism.

The SQLTuple has been developed to deliver efficient SQL persistency for AIDA standard NTuple objects. The implementation is based on FreeHEP AIDA implementation and is completely inter-operable with other FreeHEP components as well as with other AIDA implementations. SQLTuple dependency on SQL database implementation is handled at run-time by textual configuration. In principle all mainstream SQL databases are supported. The default mapping layer can be customized so that, for example, LCG Pool Tag databases can be transparently supported. This customization is used to implement higher level management utilities for Pool Tag databases - package ColMan. ColMan utilities are accessible also from the C++ environment and via standard Web Service.

The representation will cover both SQLTuple and ColMan packages and their inter-operability with other tools. Performance assessment of various available technologies will be covered as well.

Primary author: HRIVNAC, J. (LAL)

Presenter: HRIVNAC, J. (LAL)

Session Classification: Poster Session 3

Track Classification: Track 3 - Core Software