Contribution ID: **222** Type: **oral presentation**

The SEAL C++ Reflection System

Monday 27 September 2004 14:40 (20 minutes)

The C++ programming language has very limited capabilities for reflection information about its objects. In this paper a new reflection system will be presented, which allows complete introspection of C++ objects and has been developed in the context of the CERN/LCG/SEAL project in collaboration with the ROOT project.

The reflection system consists of two different parts. The first part is a code generator that produces automatically reflection information from existing C++ classes. This generation of the reflection information is done in a non intrusive way, which means that the original C++ classes definition do not need to be changed or instrumented. The second part of the reflection system is able to load/build this information in memory and provides an API to the user.
The user can query reflection information from any C++ class and also interact generically with the objects, like invocation of functions, setting and getting data members or constructing and deleting objects. When designing the different packages it was taken care of having minimal dependencies on external software and a possibility to port the software to different platforms/compilers.

A quick overview of the current implementation in use by the LCG SEAL and POOL projects will be given. A more detailed description of the new model, which aims to reflect the complete C++ language and to be a common reflection system used also by the ROOT framework, will be given.

Authors: MATO, P. (CERN); ROISER, S. (CERN)

Presenter: ROISER, S. (CERN)

Session Classification: Core Software

Track Classification: Track 3 - Core Software