

X3J16/96-0142
WG21/N0960

ISO/IEC JTC 1/SC 24 N 1577

Liaison statement from SC 24/WG 6 to ISO/IEC JTC 1/SC22/WG21
on an extension of C++ toward multi-thread management

Introduction, background

SC24/WG6 is developing a new standard, referred to as PREMO (Programming Environments for Multimedia Objects). The goals of PREMO is to specify an object-oriented programming environment for incorporating multimedia functionality. More details on the project can be found on the public Web site of the project: <http://www.cwi.nl/Premo>.

PREMO has its own object model and object specification independent of any specific programming language. However, this general functionality is intended to be mapped to specific programming languages, especially those standardized by ISO. Being an object-oriented language, C++ is one of the natural candidates for this .

Initial studies have been done within the PREMO Rapporteur Group and OO Language Bindings study group to map the object model of PREMO to the one represented by C++. This study has shown that there is some functionality required by PREMO which is not available in the draft C++ Standard. Further, their realization would mean a significant burden on any implementation of PREMO. Because this functionality is of general use, i.e., not related to PREMO requirements only, SC24/WG6 feel it necessary to contact the relevant SC22 working group on these issues.

SC24/WG6 understands that a C++ DIS is planned for spring 1997, and that SC22 does not wish to delay publication of the C++ standard any further. Consequently, SC24/WG6 proposes to add an informative annex to the C++ Standard which would include necessary extensions for the purpose of multimedia programming. Such informative annex, or a modified version thereof, may later become part of a next revision of C++. Experts in the PREMO Rapporteur Group of SC24/WG6 are prepared to contribute an initial draft for those extensions in autumn 1996, and are also prepared to participate in the technical work finalizing this draft.

Technical content of the C++ extensions

The central concept within the PREMO object model is active objects. This means that in PREMO:

- each object may have its own thread of control, and can coexist concurrently with other objects;
- operation invocation on an object is equivalent to sending a message to another thread of control;
- an object may decide, based on its own state, which incoming operation invocations it is prepared to honour, and in which order they will be honoured.

Mapping these concepts to C++ may be done if an extension to C++ is defined, along the lines of:

- defining the concepts of active and mutex (i.e., mutual exclusion) classes; operation invocations on these classes have additional semantics related to concurrency management;
- additional language constructs, e.g., along the lines of Ada rendezvous constructs, should be added to the language for the control of the incoming operation invocations.

These concepts can be defined as a strict extension to C++, i.e., without influencing the semantics of the "core" C++ model and language. Such extensions have a

lready been defined and implemented by some experts of the SC24/WG6 PREMO Rapporteur Group, and the initial proposal would reflect their experience on that matter.

Note that language-level control over concurrency is a necessary tool in future, and not just for multimedia programming. Many operating systems provide kernel-based multithread facilities, but usually on a very low level. Consequently, an extension of C++ toward a proper control of concurrency would be beneficial to the C++ community at large.