Session Summary: New Core Language Features

Session Chair: Joyce L. Tokar
Rapporteur: Brian Dobbing

Abstract

The first session in the 12th International Real-Time Ada Workshop was a summary of the new core language features that are proposed for the Amendment to Ada that is scheduled for 2005.

1 Introduction

This session focussed on bringing the workshop delegates up-to-date on the proposals under consideration by the Ada Rapporteur Group (ARG) for the amendment to the Ada language ISO standard in 2005. Originally, this session was to be presented by Stephen Michell. Unfortunately, Stephen was called away on an emergency and Joyce Tokar and John Barnes gave the presentation.

2 ISO Standards Process

The ISO requires a five-year review cycle for the Ada standard. In 2000, the Technical Corrigendum was issued as a delta to the Ada95 standard to correct minor errors. For the 2005 amendment, the schedule is to complete the proposed text of the amendment by Spring 2005. This amendment will be based on those Ada Issues (AIs), covering both new capabilities and modifications to existing features (including defect correction), that are recommended by the Ada Rapporteur Group, and approved by the ISO WG9 delegated bodies. The proposed text will then be subjected to a series of ballots, by WG9, SC22 and finally JTC1 committees, leading to the final approval of the amendment at the end of 2005.

3 Ada Issues

All major features of Ada95 are affected by the current set of AIs. This includes:

- Real-time capabilities
- High integrity capabilities
- Object oriented capabilities
- Interfacing capabilities
- General language capabilities

The AIs have varying status within the current ARG discussions, including:

- WG9-approved
- ARG-approved,
- Under discussion,
- Undiscussed, and
- Rejected.

Prior to the workshop, there were nine AIs that relate to real-time capability, as follows:

- 249 Ravenscar Profile (WG9-approved)
- 265 Partition Elaboration Policy (WG9-approved)
- 266 Task Termination Procedure (Under discussion)
- 297 Timing Events (ARG-approved)
- 298 Non-Preemptive Dispatching (ARG-approved)
- 305 New Pragma and Additional Restrictions identifiers (WG9-approved)
- 307 Execution Time Clocks (ARG-approved)
- 321 Definition of Dispatching Policy (ARG-approved)
- 327 Dynamic Ceiling Priorities (Under discussion)

The workshop would be discussing in future sessions how to advance the real-time AIs that have not yet been approved by WG9, and whether additional real-time AIs are required.

Joyce and John briefly listed all the non real-time AIs that are being considered, and they presented details of a few key ones, as follows.

AI 217 addresses the cyclic dependency of types declared in separate package specifications by proposing a `limited with P;` construct to import a (tagged) type from `package P` that implies only a minimal syntax check of `package P`.

AI 251 introduces new functionality to support abstract interfaces, similar to those provided in Java. This concept has also been recently proposed for task and protected types (in AI 345) to provide a solution for extendable protected and task types, that does not present the same challenges as earlier proposals, such as the one presented in AI 250 (which is now dormant). Tucker Taft has also written a section on inheritance of interfaces for protected and task types (see [1]).

AI 262 proposes the ability for a specification to “with” a private package and have the with-ed package apply to the private part only, to give more flexibility in constructing subsystems.

The amendment is also likely to include some new standard packages in the following areas:

- Maths packages
- Database interfaces
- Network interfaces

There is also AI 248 that addresses a standard set of operations on directory structures.

Finally it was pointed out that there were several new AI numbers that had been allocated immediately prior to the commencement of the IRTAW, the contents of which were largely unknown. Hence it was not possible to consider these at the workshop, other than AI 345, which was assumed to be based on Tucker Taft's write-up of task and protected type interfaces for Ada User Journal [1].

The workshop attendees were very interested in the proposal for concurrency interfaces as this work had been originally submitted by members of the IRTAW as AI 250. The IRTAW gave overwhelming support for the new interfaces type as described in AI 345.

4 Ada Rationale

It was pointed out that there is no plan at present to update the Ada Rationale – hence there is an issue over how the rationale behind the intended use of the new features in Ada2005 will be captured.

It was concluded that the most likely avenue for this would be revisions of established textbooks on Ada.

References

[1] Section 7 of "Object-Oriented Programming Enhancements in Ada 200Y", Ada User Journal Vol 24, Number 2, June 2003.