

Minutes of Electronic ARG Meeting 62S

10 November 2022

Attendees: Steve Baird, John Barnes, Randy Brukardt, Jeff Cousins (left at 13:30), Gary Dismukes, Bob Duff, Edward Fish, Brad Moore, Jean-Pierre Rosen, Ed Schonberg, Justin Squirek (joined 12:00), Tucker Taft, Tullio Vardanega, Richard Wai (joined 11:30).

Observers: None.

Meeting Summary

The meeting convened on Thursday, 10 November 2022 at 10:37 hours EST and adjourned at 13:34 hours EST. The meeting was held using Zoom. The meeting covered some of the agenda.

AI Summary

The following AI was approved:

AI22-0050-1/01 Ada.Unchecked_Deallocation_Subpool should be preelaborated (14-0-0)

The following AIs were approved with editorial changes:

AI22-0009-1/05 Nonvolatile views of volatile objects (14-0-0)

AI22-0029-1/05 External_Tag collisions (13-0-1)

The intention of the following AIs were approved but they require a rewrite:

AI22-0036-1/04 Attributes in the expression of Default_Value (13-0-0)

AI22-0051-1/01 Preelaborable_Initialization and contract aspects (12-0-2)

AI22-0053-1/01 An unintended consequence of AI12-0101-1 (13-0-0)

The intention of the following AI was voted, but then was discussed again later in the meeting (the final results are above):

AI22-0009-1/04 Nonvolatile views of volatile objects (12-0-0)

Detailed Minutes

Welcome

Steve welcomes everyone.

Apologies

Richard Wai will be late. Arnaud Charlet has a conflict.

Previous Meeting Minutes

There were no comments on the minutes: Approve minutes: 12-0-0.

Date and Venue of the Next Meeting

Randy proposes Thursday, January 19, 2023 for our next meeting. This is slightly longer than our usual 10 weeks in order to avoid the holidays for homework and agenda deadlines (with a January 12th date, the deadline would be the week of the New Year's Day Holiday).

There are no objections to the January 19th date; so we adopt that date. The meeting time and duration remains the same.

There will be hybrid meeting in Lisbon, June 16-18, 2023, co-located with the Ada-Europe conference.

FDIS Progress Report

Randy gives a short report. The FDIS came back with a bunch of comments. This time, most are related to wording of various sorts (no formatting issues). We need to decide how to handle these.

[The letters following match that of the written report sent prior to the meeting.]

[A] Randy suggests setting up an editing group to actually handle the changes. Then a very short review period (mostly to catch gross errors). We'd then submit and ballot the changes at the same time.

He notes that most of the changes are in non-normative text, so any problems can be fixed up "next time" (either in another draft of the FDIS, or in a future revision or corrigendum). He also notes that we do have a time limit for getting the FDIS published, so we want to make as much progress as possible as quickly as possible.

The group agrees with this plan. The editing group will be Tucker, Randy, and Steve.

[B] Many notes still need rewriting (the previous version wasn't acceptable). We'll try to rewrite them without losing the meaning, but if that isn't possible, we should just remove them from the ISO version (they're just notes, after all). Randy asks how they should be handled in relation to the RM. Tucker suggests that we want to do the least extra work for this ISO version. Thus, he suggests using option [3C], just leave the notes in the RM as is. Randy notes that we probably should revert them to their more readable Ada 2012 version in that case (but that would be a case-by-case judgement call). Tucker reiterates his "least work" mantra. No one disagrees with these conclusions.

[C] The ISO editors suggested adding some verbal form text to the Introduction, as their policy is to not allow such text elsewhere. But we only have a couple of non-critical sentences about this, and the suggested summary is an over simplification (it doesn't mention allowed alternative forms or examples).

We agree it is best to just delete the offending text from the ISO version, and make no other changes.

[D] We're not allowed to reference obsolete standards. 6 of our references are obsolete. Randy notes that we have to check that the new standards still contain the material we reference (in particular, the clauses of 10646). Jeff and Tucker volunteer to figure out the updates needed (if any).

[E] Various issues with the Terms and Definitions.

(1) The definitions should not start with "a" or "the". Should this be done in all versions, or just in the ISO version? Tucker suggests just making the change to the ISO style to all versions. He will look if any should be rewritten to make that style work better.

(2) Several terms need redoing, as the exact phrase defined does not appear in the body (which ISO is requiring). This task is given to the editing group.

(3) Several terms are "defined" in Notes to Entry, which is wrong. These should be given first class definitions. The editing group will also handle this.

(4) There is a cross-reference in the terms, which the ISO editors deleted. We'll just remove the cross-reference from the terms (Tucker notes that an appropriate cross-reference can be found in the index).

Hold AIs

Randy and Steve created a document with recommended handling of the 41 Ada 2012 Hold AIs.

The consequences of getting the recommendation wrong for a particular AI are not severe, worst case some extra work. The public can make requests that start issues for existing hold AIs at any time; AIs on our agenda can be voted Hold or No Action; issues can be turned into AIs or dropped as needed.

We ask if anyone wants to change the recommendation for a particular AI. After 30 seconds of silence, it is clear that the answer is No.

Approve Hold_AI recommendations as submitted: 12-0-0.

Unfinished Action Items

There are a few unfinished action items (Tucker Taft: OpenMP Technical Report; AI22-0034-1, announcement of the new UCI process and web sites; Edward Fish: AI22-0022-1). We did not discuss these. Additionally, three people still owe ACATS Tests: Ed Schonberg, Justin Squirek, and Tucker Taft. (Tucker is working with Jeff to get his finished.)

Current Action Items

The combined unfinished old action items and new action items from the meeting are shown below.

Steve Baird:

- AI22-0036-1
- Serve on the FDIS editing group (see FDIS report).

Randy Brukardt:

- Serve on the FDIS editing group (see FDIS report).

Editorial changes only:

- AI22-0009-1
- AI22-0029-1

Jeff Cousins:

- Determine what revisions are necessary for the standards referenced (with Tucker Taft, see FDIS report).

Edward Fish:

- AI22-0022-1
- AI22-0051-1 (with assistance from Tucker)

Ed Schonberg:

- ACATS C-Test(s) for filters

Justin Squirek:

- ACATS C-Test(s) for Wide versions of Command_Line, etc. (or choose another, untested area)

Tucker Taft:

- AI12-0346-1 – also, construct the Technical Report suggested by this AI.
- AI22-0034-1
- AI22-0051-1 (assist Edward Fish)
- Serve on the FDIS editing group (see FDIS report).
- Create an announcement of the new UCI process and web sites (see UCI WG report from meeting #62R).
- Determine what revisions are necessary for the standards referenced (with Jeff Cousins, see FDIS report).
- Check definitions to see if any should be rewritten to read better when in the correct style (see FDIS report).

Detailed Review

The minutes cover detailed review of Ada 2022 AIs (AI22s). The AI22s are presented in numeric order, which is not necessarily the order in which they were discussed. Votes are recorded as “for”-“against”-“abstentions”. For instance, a vote of 6-1-2 would have had six votes for, one vote against, and two abstentions.

If a paragraph number is identified as coming from the working Ada 2022 AARM, the number refers to the text in draft 34 of the Ada 2022 AARM (the submission draft). Paragraph numbers in other drafts may vary. Other paragraph numbers come from the final consolidated Ada 2012 AARM; again the paragraph numbers in the many drafts may vary.

Detailed Review of Ada 2022 AIs

AI22-0009-1/04 Nonvolatile views of volatile objects

Steve explains the change.

Tucker does not like the introduction of a new term, it seems like “both or neither are atomic” would work.

We discuss if something similar is needed for independent. Everything could be independent, and if the representation is the same as an independent object, there is no problem (and if not, the implementation will clearly choose to copy). On the other hand, volatile or atomic might change the code used to reference an object (as well as the representation), so that is still problematical even if the representation is unchanged.

Tucker will try to rewrite the paragraph.

Approve intent of AI: 12-0-0.

AI22-0009-1/05 Nonvolatile views of volatile objects

Tucker has proposed a rewording of the C.6(12.1/5) paragraph.

Much wordsmithing occurs (see the Google Docs version for the final result).

Some typos are fixed in the !ACATS section.

Approve AI with changes: 14-0-0.

AI22-0029-1/05 External_Tag collisions

We discuss the issues.

After much discussion, we add “user-specified” to the dynamic check (paragraph 75.1), so that one of the tags has to be user-specified. No other changes are made. This text was in the original check definition, and never got carried over (Randy thought that was intentional, but there was concern that instance bodies would not be properly handled in that case.)

Steve is worried that “different declarations” isn’t clear enough.

Add an AARM Discussion: “Each instance of a generic unit is considered a distinct declaration, as are declarations occurring within distinct instances of the generic unit.” (Added to AI by Tucker during the meeting).

Randy suggests changing “different” to “distinct” in paragraph 75.1, so that we use the same wording in both paragraphs 75.1 and 76.

Gary notes a typo in the discussion, but then Randy notes that we just changed the wording to make the whole thing incorrect. Tucker suggests just deleting it altogether, which was done in the Google Docs version.

Approve AI with changes: 13-0-1. Justin abstains.

AI22-0036-1/04 Attributes in the expression of Default_Value

[Editor's note: Fix paragraph references in the !standard headers, and then paragraph number for 13.14(10/5) – it's missing the /5.]

The 13.14(10/5) change is the only new part.

The wording only allows top-level attributes, shouldn't it allow attributes in subexpressions of the Default_Value? Yes, it should.

Tucker says that Val and Pos are calls, so this wording doesn't work.

Steve notes that we want to restrict the prefix to the type of the Default_Value; we don't want this to work for any random type. Randy notes that this will require implementations to jump through hoops, so we want it as narrow as possible (but no narrower). Someone notes that freezing does not cause problems for other types, so there is nothing added by including them. Note that it is not just the entity itself, but also subtypes of that entity that would be allowed. [Note that a subtype with explicit bounds would have already frozen the type, which would have made the aspect spec illegal. So there is no problem with S'First or S'Last.]

Gary notes that in the AARM note for 8.6, Enum_Pos should be Enum_Rep.

Gary also notes that in the discussion "we make the hole is small as possible" (is should be as).

Jean-Pierre asks why the current rule is so narrowly defined. Steve notes that for aspects, such uses would run afoul of the freezing rules. Randy notes that the problematic freezing rule only applies to representation aspects and some operational aspects; it does not apply to contract aspects. Thus, there could be a usage in (for instance) a predicate expression. But the example discussed (S'Size) is not covered by the proposed wording, since 'Size is defined for both subtypes and objects.

Tucker suggests that we should leave it as narrow as possible, and see if there is an implementation burden or user demand for a more expansive rule. It always easier to loosen rules than it is to tighten them.

Keep the rule to just items of the type, since other things can be frozen.

The AI is returned to Steve for rewording.

Approve intent of AI: 13-0-0.

AI22-0050-1/01 Ada.Unchecked_Deallocation_Subpool should be preelaborated

Randy explains that this seems to be an omission. The parent package is preelaborated, and the most similar thing (Unchecked_Deallocation) is preelaborated, so it is odd that Unchecked_Deallocation_Subpool is not preelaborated. Tucker notes that Binding Interpretations are easier to implement, since no versioning is required. We agree that the AI is correctly classified as a Binding Interpretation.

Approve AI: 14-0-0.

AI22-0051-1/01 Preelaborable_Initialization and contract aspects

We discuss this extensively. While suppressing the checks is uncomfortable, the alternative of making contracts and preelaboration incompatible with each other is more uncomfortable. And initialization checks are generally more to tell the reader (including tools) the intended initialization. Such checks should rarely fail.

Ed Fish will take the AI with help from Tucker.

The intent is that the checks are not performed in these cases. We want a permission for the implementation to make the checks if it can do so without violating the preelaboration restrictions (that is, either by making the checks at compile-time, or by delaying them until preelaboration is complete).

Approve intent: 12-0-2. Gary and Jeff abstained.

AI22-0053-1/01 An unintended consequence of AI12-0101-1

Tucker doesn't like the wording. We have run out of meeting time to wordsmith this. Therefore, Tucker will take the AI to rewrite the wording.

Approve intent of AI: 13-0-0.