Edits - VOLATILE Fixups

To: J3
From: Craig Dedo
Date: February 29, 2000
Subject: Edits - VOLATILE Fixups

1. Rationale

This paper answers the questions about the VOLATILE attribute that Malcolm Cohen asked in paper J3 / 00-125. JOR considered all of the questions asked in the paper. Following are the responses to the questions and edits.

2. Responses to Questions and Edits

Q1. Editorial Notes.

These editorial changes are not needed due to edits in response to Q2.

Q2. What association rules of 14.6.1.1 are being referred to?

This question and its related questions motivated JOR to replace the entire bulleted list in 5.1.2.13 with a much broader specification.

[78:2-8] Replace the bulleted list with:
"An object shall have the VOLATILE attribute if there is a reference to or definition of the object, or the object becomes undefined, by means not specified in this standard."

[78:17+] Add a new paragraph.
"The interpretation of a program containing objects with the VOLATILE attribute is processor dependent."

Q3. What is meant by the "storage sequence of a different object" in item (3)?

This list item has been removed in favor of the broad qualification for the VOLATILE attribute constructed in answer to Q2.

Q4. Does pointer association interact with volatility if the target is volatile and the pointer is not?

No. Although legal, it is usually unsafe to associate a non-volatile pointer with a volatile target.

[78:17+] Add a note.

NOTE 5.21a

> If the value of the target of a pointer can change, by means outside of Fortran, while a pointer is associated with a target then the pointer must have the VOLATILE attribute. Usually, a pointer should have the VOLATILE attribute if its target has the VOLATILE attribute. Similarly, all members of an EQUIVALENCE group should have the VOLATILE attribute if one member has the VOLATILE attribute.

Q5. How does VOLATILE interact with INTENT(IN)? Should we make VOLATILE and INTENT(IN) incompatible?

JOR decided that VOLATILE is incompatible with INTENT(IN). Following are edits to resolve this incompatibility.

[65:7-8] Replace “or EXTERNAL” with “EXTERNAL, or INTENT(IN)”.

[65:10] Insert “VOLATILE, “ between “DIMENSION, “ and “INTENT(INOUT), “.

Q6. How does VOLATILE interact with Fortran 90 Interpretation 125?

JOR decided that a dummy argument should not have either the VOLATILE or ASYNCHRONOUS attribute if it is associated with an actual argument that is an array section with a vector subscript. The following edit makes this clear.

[259:36] Replace “INTENT(OUT) or INTENT(INOUT)” with “the INTENT(OUT), INTENT(INOUT), VOLATILE, or ASYNCHRONOUS attributes”.

A future paper will further clarify this issue.

Q7. Do we need language that makes it clear that references to VOLATILE dummy arguments that are associated with VOLATILE actual arguments need to reference the most recent definition of the actual argument?

No. We believe that note 5.21 [78:12-15] already makes this clear.

3. References

ISO/IEC 1539-1:1997(E) International Standard Programming Language Fortran 95, 5.1.2.13, 5.1, 12.4.1.2

J3 / 00-125, “VOLATILE Questions”

[End of J3 / 00-132]