Subject: Follow-up to 03-196r1 and 03-224/N1568
From: Van Snyder

1 Edits – Follow-up to 03-196r1

Edits refer to 03-007. Page and line numbers are displayed in the margin. Absent other instructions, a page and line number or line number range implies all of the indicated text is to be replaced by associated text, while a page and line number followed by + (-) indicates that associated text is to be inserted after (before) the indicated line. Remarks are noted in the margin, or appear between [and] in the text.

[The present text overlooks host association, at least. Editor: "an argument ... block" ⇒ "a variable". By simply saying "variable" we catch them all – including the arguments. By the way, 16.4.2.1.3 doesn't say anything about pointer association status becoming undefined in the same way as this says variables become undefined, and we also don't say anything about allocation status, or the open/closed status of a unit, or ..., going Schrödinger, but we're not going to tackle those problems now.] 423:47-48

[Editor: Delete "argument or".] 424:2

[We don't want variables accessed by use association to become undefined or to be finalized. One could say that "variables in" a module include those accessed by use association. Editor: "variables in" ⇒ "local variables of" twice.] 424:8,10

2 Edits – Follow-up to 03-224/N1568

Data subgroup was asked to look at the edits for [59:15] and [255:11] from 03-224, which began life as N1546, which later became N1568.

[Editor: After "name." insert "It has the accessibility of the parent type."] 59:15

While reviewing [255:11-14], it was noticed that it specifically allows a list of statements, some of which are attribute specification statements, and then explicitly prohibits certain attribute specifiers. We noticed that ASYNCHRONOUS and VOLATILE are in neither list, so one is prohibited from specifying them with a statement but allowed to specify them with an attribute specification. Further, it was noticed that one cannot specify a binding label for a common block. Both are "serious flaws" (to quote the WG5 resolutions).

It's clear that the BIND statement has to be allowed. There are already enough restrictions on it that it could only be used in a block data program unit to specify the binding label for a common block.

Although specifying ASYNCHRONOUS and VOLATILE in a block data program unit might seem fatuous, subgroup decided to allow rather than prohibit them, so that those who incorporate common block declarations throughout their programs by using include lines could do so without needing to put relevant ASYNCHRONOUS and VOLATILE declarations in a different include file from the remainder of the declarations of the common variables.

C1117 (R1116) A *block-data specification-part* shall contain only derived-type definitions and ASYNCHRONOUS, BIND, COMMON, DATA, DIMENSION, EQUIVALENCE, IMPLICIT, INTRINSIC, PARAMETER, POINTER, SAVE, TARGET, USE, VOLATILE, and type declaration statements. 255:11-14

Although C532 prohibits a *language-binding-spec* from appearing anywhere other than in the specification part of a module, subgroup decided not to remove the BIND attribute from the list of prohibited attribute specifiers at [256:1].