Subject: Numerous changes in 12.4.1.2, hopefully including resolution of Issue 79
From: Van Snyder

1 Introduction

Issue 79 implicitly addresses numerous issues. The treatment of the interaction of pointer association status, allocation status and intent, and the treatment of polymorphism and type parameters, are fractured, dispersed and incomplete, although they might be completed elsewhere. I do not propose to re-write the section, but only to make a small dent in correcting, completing and reorganizing it.

The discussion of INTENT(OUT) is actually a bit of over-kill, because a conspiracy of 5.1.2.3 and 6.3 covers it all.

2 Edits

Edits refer to 99-007r2. 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 immediately following text, while a page and line number followed by + indicates that immediately following text is to be inserted after the indicated line. Remarks for the editor are noted in the margin, or appear between [and] in the text.

[Editor: Delete from "That is" to the end of the paragraph.] 77:22-24

The sentence repeats material at the immediately previously cited places. It is incomplete concerning type parameters, and making it complete would make it even more redundant.	*Note to J3*

[Editor: Replace from "If a dummy argument..." to the end of the penultimate sentence of the paragraph:] 275:6-8

If a dummy argument or its associated actual argument, or both, are polymorphic, and the dummy argument has INTENT(IN) or is neither allocatable nor a pointer, the declared type of the actual argument shall be an extension type (4.5.3) of the declared type of the dummy argument. If the dummy argument is polymorphic, is allocatable or a pointer, and has INTENT(OUT), the actual argument shall be polymorphic, and the declared type of the dummy argument shall be an extension type of the declared type of the actual argument. If the dummy argument is allocatable or a pointer and has neither INTENT(IN) nor INTENT(OUT), the declared type of the actual argument shall be the same as the declared type of the dummy argument.

The above rules guarantee that there can be no run-time type errors through argument association.	Note 12.21$\frac{1}{2}$

The values of assumed type parameters of a dummy argument are assumed from 275:20-21

If the dummy argument is allocatable and the associated actual argument is currently allocated, the initial values of deferred type parameters of the dummy argument are assumed from corresponding type parameters of its associated actual argument. When execution of the procedure completes, if a dummy argument is an associated pointer or a currently allocated allocatable variable, the values of the deferred type parameters of its associated actual argument become the same as the corresponding type parameters of the dummy argument. 275:23:42