Subject: Comments on Clause 12
From: Van Snyder

1 Edits — and comments without editorial suggestions

Edits refer to 06-007. 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.

[Editor: "or" ⇒ comma, insert ", the appearance of an object processed by user-defined derived-type input/output (9.5.3.7) in an input/output list, or finalization (4.5.6)" at the end of the sentence.] 297:21-22

[Internal subprograms are now allowed to be actual arguments. Editor: Delete ", the internal procedure name shall not be argument associated with a dummy procedure (12.5.1.6)".] 298:3-4

[Delete because "generic identifier" implies "explicit interface".] 300:1-3

[Editor: For consistency with [302:12-14] "An explicit ... way" ⇒ "If an external procedure does not exist in the program, an interface body for it may be used to specify an explicit specific interface but the procedure shall not be used in any other way".] 302:27-29

[The note seems to say that dummy arguments specified in a procedure definition or an interface body might be the same as other dummy arguments in the same definition or interface body. Editor: "may be different" ⇒ "in an interface body may be different from the corresponding dummy argument names in the procedure definition".] 302:30+2

[Subclause 16.3.4 doesn't have anything to do with scope of local identifiers, which is the topic of Subclause 16.3. It more logically belongs here anyway. Editor: Move Subclause 16.3.4 here. Do we thereby need to convert some of the ordinary normative text in it to constraints?] 306:4+

[Editor: "initial-proc-target" ⇒ "*initial-proc-target*".] 307:11

[Editor: "initialization target" ⇒ "**initialization target**". `\tdef` will add it to the index, which is needed, since the only one there refers to [65:26] and has to do only with variables.] 307:44

[Editor: Insert ", by user-defined derived-type input/output (9.5.3.7), or by finalization (4.5.6)" at the end of the sentence.] 310:1

[The requirements aren't (can't be) posed as constraints. Editor: "constraints" ⇒ "requirements". At [311:13-16] one can conclude that no more than one actual argument can correspond to a dummy argument. Editor: "actual arguments that correspond" ⇒ "an actual argument that corresponds".] 317:5+2

[Doesn't work for optional dummy arguments that don't correspond to actual arguments. Editor: "the" ⇒ "any corresponding".] 318:20

[Doesn't work for optional dummy arguments that don't correspond to actual arguments. Editor: "the" ⇒ "any corresponding".] 318:22

[Editor: "associated ... argument" ⇒ "present".] 319:40

[Editor: Insert "of" after "value".] 321:4

[Editor: Delete "or" at [322:16], insert ", or finalization of an object" after "item".] 322:16-17

[Editor: for consistency with [323:11], delete second "if".] 323:21

[Editor: Simplify the paragraph by replacing it:] 327:28-32

The RECURSIVE *prefix-spec* shall appear if any procedure defined by the subprogram directly or indirectly invokes itself or any other procedure defined by the subprogram.

NOTE 12.40a

Each ENTRY statement in the subprogram defines an additional function.

[But don't do both this and 06-285, which deletes this pararaph.]

[Editor: Simplify and correct: "the pointer . . . disassociated" ⇒ "on return the pointer association status of the result variable shall not be undefined".] 327:41-42

[Editor: Simplify the paragraph by replacing it:] 329:19-23

The RECURSIVE *prefix-spec* shall appear if any procedure defined by the subprogram directly or indirectly invokes itself or any other procedure defined by the subprogram.

NOTE 12.43a

Each ENTRY statement in the subprogram defines an additional subroutine.

[But don't do both this and 06-285, which deletes this pararaph.]

[Editor: "function or subroutine" ⇒ "procedure".] 329:29

When execution of an instance completes it ceases to exist. 329:38+ New ¶

[Editor: "is only accessible by use association" ⇒ "can be accessed by use association only".] 330:22+2

[We usually use "executable construct" instead of "executable statement". Editor: "executable statement" ⇒ "executable construct" twice.] 331:31-32

[Editor: Insert "or another procedure defined by the subprogram" after "itself".] 332:6

[Editor: Insert "of the referenced procedure" after "list".] 332:19

[Editor: Delete second "an", insert "function" after the second "intrinsic".] 333:7

[Editor: "attributes" ⇒ "parameters".] 333:28

[Wouldn't it be simpler to say "a pure intrinsic procdure"? Compare to [335:12]. If it works there, why not here? Editor: Replace by the following:] 333:33-34

(1) a pure intrinsic procedure (13.1),

[Editor: Insert ", or within the *specification-part* of a BLOCK construct within a pure subprogram," before "shall". In light of C1281, delete "or *internal-subprogram-part*".] 334:7

[Editor: "assignment" ⇒ "defined assignment, user-defined derived-type input/output". Yes, DTIO can happen, so long as the unit is an internal file.] 335:2

[Repairing the note to account for IMPURE would result in it saying "the constraints on pure procedures apply to pure procedures." Editor: Delete Note 12.51.] 336:8+1-3