

Subject: Comments on Section 16, Issues 319 and 336  
 From: Van Snyder

## 1 Edits – Organization

Edits refer to 01-007r3. 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 (before) the indicated line. Remarks are noted in the margin, or appear between [ and ] in the text.

The subdivision of the section according to [366:8-13] is not effective. Numerous entities are described in subclause 16.1 that are not names. Instead, classify according to whether entities are global etc.

[Editor: Delete “It ... called”.]	365:3
<ul style="list-style-type: none"> <li>• A <b>global entity</b> has a scope of a program;</li> <li>• A <b>local entity</b> has a scope of a scoping unit (2.2);</li> <li>• A <b>construct entity</b> has a scope of a construct;</li> <li>• A <b>statement entity</b> has a scope of a statement.</li> </ul>	4-6
<ol style="list-style-type: none"> <li>(1) A name (3.2.1),</li> <li>(2) A label (3.2.4),</li> <li>(3) An external input/output unit identifier (9.4),</li> <li>(4) A pending data transfer identifier (9.5.1.8, 9.6),</li> <li>(5) A generic identifier (12.3.2.1), or</li> <li>(6) A binding label (12.5.2.7, 15.2.7.1).</li> </ol>	365:8-13
[Editor: Delete unresolved issue 319.]	365:14-19
[Editor: Decapitate subclause 16.1. Raise subclauses within 16.1 one level.]	365:22-23
External input/output unit identifiers and pending data transfer identifiers are global entities.	366:11+
[Editor: Delete “and”.]	366:19
[Editor: Insert a comma at the end of the line.]	366:20
<ol style="list-style-type: none"> <li>(4) statement labels, and</li> <li>(5) generic identifiers</li> </ol>	366:20+
No two statements in the same scoping unit shall have the same label.	366:44+
[Common blocks are global entities.]	367:12
<b>16.2.1 Local entities that have the same names as common blocks</b>	
[16.1.2.4 has nothing to do with scope, association or definition. Editor: Move the entire subclause to [259:12+], making it 12.4.4.]	368:35ff

[Now covered elsewhere. Editor: Delete.]	372:34-41
[Now covered elsewhere. Editor: Delete, including unresolved issue note 336.]	373:1-13
[Editor: Move to [383:17+], to be near other stuff about allocatable.]	382:45-46

## 2 Edits – ordinary stuff

Edits refer to 01-007r3. 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 (before) the indicated line. Remarks are noted in the margin, or appear between [ and ] in the text.

[The subclause has nothing to do with procedure reference. Editor: Change the title to “ <b>Restrictions on generic procedure declarations</b> ”.]	367:24
[Simplification: Editor: “name ... assignment” ⇒ “generic identifier”.]	367:32-33
[Dummy procedures can now be part of generic interfaces. Editor: Delete “and” at [370:35]. Insert “, and if the name is not also a generic identifier”.]	370:35,36
[Dummy procedures can now be part of generic interfaces. Editor: Insert “that is not also a generic identifier” after “subprogram”.]	371:10
[Editor: Insert “derived” before “type” twice.]	371:19
A generic binding for which the <i>generic-spec</i> is not <i>generic-name</i> has a scope that is the scoping unit containing an object of the type.	371:26+ New ¶
[Editor: Delete the comma.]	371:28
[Editor: Insert “; it has no other attributes” after “function”.]	372:7
[Editor: “for” ⇒ “governing” (see [373:23]). Delete “the” twice (but not “The”).]	373:36
[Run-on sentence. Editor: “and” ⇒ “; it”.]	373:38
[Run-on sentence. Editor: second “and” ⇒ “; they”.]	374:1
[Editor: “or an” ⇒ “, <i>procedure-declaration-stmt</i> or”.]	374:6
[Run-on sentence. Editor: “and” ⇒ “;” twice.]	374:7,30
[Editor: Italicize all of “ <i>type-alias-stmt</i> ” (the “ty” isn’t currently italicized).]	374:25
[Editor: “with” ⇒ “to” (see [376:5]).]	376:8 376:18+

### NOTE 16.15<sup>1</sup>/<sub>2</sub>

The selector is also accessible within the construct.

If the pointer has deferred type parameters, their values are assumed from the target. If the pointer is polymorphic, its dynamic type is the dynamic type of the target.	376:24+ Same ¶
[The note is about pointer association status in particular, not pointer association in general. Editor: Move to [376:39+], after inserting “the association status of” after “considers” at [375:31].]	376:25-34
[Editor: “an object” ⇒ “a nonpointer object” (see [383:25-27]).]	377:6

[Editor: “a local” ⇒ “an unsaved local” (see [383:25-27]).]	377:11
[Editor: Insert “, default initialization is not specified for the component,” before “and”.]	377:25
[Association status is <i>not</i> determined by the target; it’s determined by the <i>target</i> . Editor: “target” ⇒ “ <i>data-target</i> or <i>proc-target</i> ”.]	377:41
[Editor: Delete “The ... blocks.” because it contradicts the next sentence, which is the correct one.]	379:29
[Run-on sentence. Editor: “and” ⇒ “; it”.]	373:38
[“either” implies that the following list encompasses all of the possibilities. Editor: Delete “, either”.]	380:35
[Number doesn’t agree. Editor: “that” ⇒ “those”.]	380:36
[Editor: “subcomponent” is an arcane term; reference this instance in the index.]	381:37
[Editor: Insert “have” after “to”.]	382:8
[Editor: “an” ⇒ “a nonpointer”.]	383:28
[Editor: “subcomponent” is an arcane term; reference this instance in the index.]	383:29
[ <i>namelist-group</i> is not a defined syntax term. Editor: “ <i>namelist-group</i> ” ⇒ “namelist group”.]	384:19
[Editor: Delete, because it’s just a procedure invocation, and that’s adequately covered elsewhere in the list.]	384:25-26
[Editor: Insert “that has a subcomponent” after “object” (because it’s the subcomponent, not the object that can have default initialization), and reference this instance in the index.]	384:42
[Editor: “the IOSTAT=” ⇒ “an IOMSG= or IOSTAT=”.]	384:45
[ <i>internal-file-unit</i> is no longer a syntax term. Editor: “ <i>internal-file-unit</i> ” ⇒ “ <i>internal-file-variable</i> ”.]	385:34

### 3 Don’t know what to do

Is an interface body an entity? If so, do we need to distinguish local vs. global? If so, several things might have the same identifier – an external procedure and all interface bodies that refer to it.	365:7-13
Does “different” mean “not necessarily the same” or “necessarily not the same”? If the latter, this is surely incorrect – because they might actually be the same.	378:22-23
Confusing example – to what is A(4:4) associated?	380:21-24
Are automatic or allocatable zero-size arrays always defined?	381:32
Are there any other events that cause associated variables of type C_PTR to become undefined?	385:19-23