**ISO/IEC JTC 1/SC 22/WG 5**

**Fortran**

**Convenorship: ANSI (United States)**

**Document type:** Other document (Open)

**Title:** Committee Response to Comments on Second CD Draft for Fortran 2015

**Status:**

**Date of document:** 2017-10-20

**Expected action:** INFO

**Email of convenor:** steve@stevelionel.com

**Committee URL:** http://isotc.iso.org/livelink/livelink/open/jtc1sc22wg5

## Template for comments and secretariat observations

| Date:2017-09-08 | Document: | Project: |
|---|---|---|

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| US 001 | [27:11+5] | 04.01.1 | P1 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| DE 002 | 45:22 | 05.04.7 | | te | The wording appears not to be general enough to cover, for example, the case of an image selector (9.6) that contains a TEAM or TEAM_NUMBER specification. | Replace “that is in the same team” by “of a team in which it is established (5.4.8)”. | Paper 17-205 addressed this concern and was passed. |
| DE 003 | 50:10 | 06.02.1 | | te | The sequence of two periods should be explicitly listed as one of the exceptions in the referenced subclause, to prevent an assumed-rank object declaration of the form<br>real, dimension(. .) :: a<br>(with a blank between the periods). | After “;”, add “, ..“ | Paper 17-238r1 addressed this concern and was passed. |
| US 004 | [53:6] | 06.03.1 | P2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| US 005 | [53:19] | 06.03.2.2 | P1 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| US 006 | [54:9] | 06.03.2.4 | P1 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| US 007 | [55:11] | 06.03.3.3 | P2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| US 008 | [55:19] | 06.03.3.5 | P1 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| US 009 | [65:11] | 07.04.4.1 | P2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| GB 010 | 66:10 | 07.04.4.2 | Para 3 | ed | C724 is badly worded, because determining recursion is a runtime property not a compile-time one. | In C724, delete “recursive,” and append sentence “A function name declared with an asterisk *type-param-value* shall not have the RECURSIVE attribute.” in obsolescent font. | Paper 17-233 addressed this concern and was passed. |
| US 011 | [67:4] | 07.04.4.3 | P3 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occurrence" with "appearance". | The editor declined this suggestion. |
| US 012 | [86:5+3] | 07.05.7.2 | NOTE 7.52 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "Inaccessible entities occur" with "An entity may be inaccessible". | The editor declined this suggestion. |
| US 013 | [121:7] | 08.07 | P4 | te | Implicit type declaration of a variable declares only the type and kind, not any other attributes. 8.7p4 specifies that the effect of implicit type declaration of a variable in a BLOCK construct is as if the variable were explicitly declared within the outermost inclusive scope containing the BLOCK construct, and the variable is therefore not a construct entity. This clearly conflicts with explicit declaration of other attributes, for example, rank, within a BLOCK construct, the result of which causes the entity to be a construct entity, according to 19.4p1. Examples in 17-184. | Delete "inclusive". | Paper 17-232 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| US 014 | [121:7] | 08.07 | P4 | te | The use of “inclusive scope” in the specified text seems problematic.<br>An issue is described in paper 17-184 that the scope of an implicit declared variable should include the block scoping unit rather than its inclusive scope. | [121:7] Replace “inclusive scope” with “scoping unit” | Paper 17-232 addressed this issue and was passed. |
| GB 015 | 121:6-8 | 08.07 | Para 4 | te | The sentence “The data entity is treated as if it … declared in the host of that scope.” is not needed and is inconsistent with the rest of the paragraph. | Delete the sentence “The data entity is treated as if it … declared in the host of that scope.”. | Paper 17-232 addressed this issue and was passed. |
| DE 016 | 124: | 08.08 | NOTE 8.45 | ge | Opacity of the derived type discussed in the NOTE is not relevant for what the example intends to illustrate. | Replace “of an opaque type” by “of a derived type defined in the module” in line 2 of the NOTE, and delete the comment “! T is an opaque type” in the program text. | Suggestion declined; the opacity is relevant to this example, otherwise you could declare the type in a separate module. |
| US 017 | [125:15] | 08.09 | P2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "once for each occurrence" with "on output once for each appearance as a <namelist-group-object>" to correspond to "appear" (not "occur") on the previous line. | The editor declined this suggestion. |
| US 018 | [125:16] | 08.09 | P2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear" to correspond to "appearance" (not “occurrence”) on the next line. | The editor declined this suggestion. |
| US 019 | [125:20-22] | 08.09 | P5 | ed | The term “earlier” is usually but not always used to express a temporal relationship, e.g., “earlier standards.” When applied to the relative appearance of syntax terms, “previously” is usually used, and should be used in several places where “earlier” is used. See 17-185. | Replace the first sentence, viz. "A namelist group object ... scoping unit" with "The {declared type}, kind type parameters of the {declared type}, and {rank} of a namelist group object shall have been previously declared, or implied by implicit typing rules in effect for the scoping unit." {…} means hyperlink. | The editor declined this suggestion. |
| US 020 | [127:5] | 08.10.01.5 | P1 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the | Replace "occur" with "appear". | The editor declined this suggestion. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | latter where “appear” ought to be used. See 17-186. | | |
| US 021 | [127:30] | 08.10.02.1 | P4 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear" to correspond to "appearance" on the next line. | The editor declined this suggestion. |
| GB 022 | 133:14 | 09.04.2 | Para 1 | te | The effect of C917 is to be less restrictive to a *data-ref* of the form `coarray[1]%pointer%allocpoly` than to one of the form `coarray[1]%allocatable%allocpoly`. We believe that it was intended to be equally restrictive to both. | Delete "subobject of a". | Paper 17-248 addressed this issue and was passed. |
| US 023 | [139:29] | 09.06 | P3 | te | It was expected that the team values on the images of a team would differ from image to image to enable each image to access any other image of the team efficiently. It is therefore important that when the value is referenced by an image, it is the value that was defined by that image. Edits are needed to specify that a team value identifies an image as well as a team. | After “identify” add “the executing image and”. | Paper 17-250r2 addressed this issue and was passed. |
| JP 024 | 36 | 09.07.1.2 | Page 142 | ed | The following statement should be inserted after "the same number of times in this team" in the same way as the description of SYNC ALL statement:<br>"The segments that executed before the ALLOCATE statement on an active image of the current team precede the segments that execute after the ALLOCATE statement on another active image of the current team." | | Paper 17-209r1 addressed this issue and was passed. |
| JP 025 | 8 | 09.07.3.2 | Page 146 | ed | The following statement should be inserted after "the same number of times in this team" in the same way as the description of SYNC ALL statement: | | Paper 17-209r1 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | "The segments that executed before the statement on an active image of the current team precede the segments that execute after the statement on another active image of the current team." | | |
| US 026 | [170:17-25] | 10.02.1.3 | P3 | te | The portion of 10.2.1.3 paragraph 3 describing intrinsic assignment to an allocatable polymorphic variable does not address the possibility of a mismatched kind type parameter. 10.2.1.2 paragraph 1 item (7) precludes such a mismatch in any kind type parameter that is part of the declared type of the variable, but if the dynamic type is different, there exists the possibility of kind type parameters added in the extension. It does not appear that this failure to address was intentional: Such kind type parameter mismatches are definitely not allowed in the nonpolymorphic case. Although a plausible interpretation can be made for how to handle the mismatch in the most common cases, the cost of implementing that interpretation is significant. For many of the less common cases, no such plausible interpretation appears to exist. | In 10.2.1.3, paragraph 3, sentence 2, after "the dynamic type", insert "or any of the corresponding kind type parameter values". | Paper 17-236r1 addressed this issue and was passed. |
| US 027 | [179:14] | 10.02.3.2 | P9 | ed | The term "occur" is used mostly to refer to events, and "appear" is used with respect to syntax entities. But sometimes "occur" is used for the latter where "appear" ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| US 028 | [187] Note 11.4 | 11.01.4 | Note 11.4 | ed | In the code, the first appearance of 'Z" is in the block scoping unit. It seems suggest that the implicit declaration of "Z" is actually in the inclusive scope as 8.7p4 states.<br>It will become confusing after the fix of 8.7p4 to replace "inclusive scope" with "scoping unit". | [187] Note 11.4: Add "Z = 1" in the scoping unit of subroutine "S". | Paper 17-232 addressed this issue and was passed. |
| GB 029 | 186:8 | 11.01.4 | Para 1 | te | The change from Fortran 2008 of allowing an IMPLICIT statement in a BLOCK construct was not intended. | In R1109, delete line "[*implicit-part*]". | Paper 17-221r1 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| DE 030 | 187:10 | 11.01.5.1 | | te | R1113 appears to syntactically permit a coarray association of the form<br>CHANGE TEAM(t, b[:,:] => a)<br>This should be disallowed, because the associating entity is not intended to be potentially ALLOCATABLE. | [187:22+] add new constraint<br>“C1114a The *coarray-spec* in the *codimension-decl* shall be an *explicit-coshape-spec*”. | Declined; the suggested edit is redundant with existing constraint C827. |
| DE 031 | 188 | 11.01.5.2 | After NOTE 11.6 | ge | I suggest adding a NOTE that explains how coarrays established in an ancestor team are handled inside the block of a CHANGE TEAM construct.<br><br>It may be that additional clarification is also needed in the normative text, but do not have any explicit suggestions for this. | Add “NOTE 11.6+<br>A coarray that is established when a CHANGE TEAM statement is executed will retain its corank and cobounds inside the block construct. However, the mapping of cosubscripts to image indices will usually change, subject to the rules of section 9.6. If, depending on the way the team decomposition has been set up, it is desired to perform remote accesses based on corank or cobounds different from those of the original object, an associating coarray can be used. Appendix C.6.7 supplies an illustration of this.<br>If it is desired to perform accesses that retain the mapping of cosubscripts to image indices of an ancestor team (for example, the team in which the coarray is established), that team can be specified as the TEAM= argument in the image selector. This mechanism also permits addressing parts of the coarray hosted on an image that is not a member of the current team.” | Paper 17-195r3 addressed this issue and was passed. |
| US 032 | [188:5] | 11.01.5.2 | P1 | te | See the comment for 139:29. Also, it was intended that the CHANGE TEAM statement be allowed to use team values defined by the intrinsic GET_TEAM. | Change “defined by ... (11.6.9)” to “that identify those images and a single team”. | Paper 17-250r2 addressed this issue and was passed |
| JP 033 | 23 | 11.01.5.2 | Page 188 | ed | The following statement should be inserted after "the same number of times in the original team" in the same way as the description of SYNC ALL statement:<br>"The segments that executed before the CHANGE TEAM statement on an image of the new team precede the segments that execute after the | | Paper 17-209r1 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | CHANGE TEAM statement on another image of that team." | | |
| JP 034 | 27 | 11.01.5.2 | Page 188 | ed | The following statement should be inserted after "the same number of times in the original team" in the same way as the description of SYNC ALL statement:<br>"The segments that executed before the CHANGE TEAM statement on an active image of the new team precede the segments that execute after the CHANGE TEAM statement on another active image of that team." | | Paper 17-209r1 addressed this issue and was passed. |
| JP 035 | 31 | 11.01.5.2 | Page 188 | ed | The following statement should be inserted after "the same number of times in this team" in the same way as the description of SYNC ALL statement:<br>"The segments that executed before the END TEAM statement on an active image of the new team precede the segments that execute after the END TEAM statement on another active image of that team." | | Paper 17-209r1 addressed this issue and was passed. |
| US 036 | [191:27] | 11.01.7.2 | C1129 | te | C1129 prevents existence of statement or construct entities of statements or constructs within a DO CONCURRENT construct if DEFAULT ( NONE ) appears. See 17-183. | Insert ", is not a statement entity, is not a construct entity of a construct within that DO CONCURRENT construct," after "<block> of the construct". | Paper 17-242r1 addressed this issue and was passed. |
| GB 037 | 191:17 | 11.01.7.2 | Para 1 | ed | C1126 might be misunderstood to mean that a variable can only appear in one *locality-spec* in a whole scoping unit. | In C1126, after "in a *locality-spec*" insert "in a given *concurrent-locality*". | Paper 17-240 addressed this issue and was passed. |
| GB 038 | 191:17+ | 11.01.7.2 | Para 1 | te | The syntax permits DEFAULT (NONE) to appear more than once in a single concurrent header in a DO CONCURRENT statement. | After C1126, insert new constraint:<br><br>"C1126a The DEFAULT ( NONE ) *locality-spec* shall not appear more than once in a given *concurrent-locality*.". | Paper 17-244r1 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| US 039 | [194:24+] | 11.01.7.5 | P1+ | ed | It is not obvious whether statement or construct entities of statements or constructs within DO CONCURRENT constructs are the same entity in every iteration, and therefore subject to the rules concerning unspecified locality, or they are different entities in every iteration. See 17-183. | Insert a note<br>" NOTE 11.10a<br>A statement entity of a statement within a DO CONCURRENT construct is a different entity in each iteration. A construct entity of a DO CONCURRENT construct, or a nonsaved construct entity of a BLOCK construct, within a DO CONCURRENT construct, is a different entity in each iteration of the containing DO CONCURRENT construct. A saved variable that is a construct entity of a BLOCK construct within a DO CONCURRENT construct is the same entity in every iteration and has unspecified locality." | Paper 17-242r1 addressed this issue and was passed. |
| US 040 | [194:25] | 11.01.7.5 | P2 | te | Ensure that a statement or construct entity of a statement or construct within a DO CONCURRENT, that has the same name as a variable with LOCAL locality, is not specified to become a construct entity of the DO CONCURRENT construct. See 17-183. | Replace "LOCAL or LOCAL_INIT locality is a construct entity with" with "LOCAL or LOCAL_INIT locality, and is not a construct or statement entity of a construct or statement within the DO CONCURRENT construct, is a construct entity of the DO CONCURRENT construct; it has". | Paper 17-242r1 addressed this issue and was passed. |
| JP 041 | 2 | 11.06.2 | Page 209 | ed | As it should be stated more clearly that segment ordering ensures memory operation order, in the same way as line 5 – 13, page 212, the following statements should be inserted after line 2, page 209:<br>"If the segment P_i on image P precedes the segment Q_j on image Q,<br>* a variable X on any image is defined, referenced, becomes undefined, or has its allocation status, pointer association status, array bounds, dynamic type, or type parameters changed or inquired about by execution of a statement of the segment P_i on image P, and<br>* X is defined, referenced, becomes undefined,<br>or has its allocation status, pointer association status, array bounds, dynamic type, or type parameters changed or inquired about by | | After careful consideration and a poll of J3 214 attendees, it was decided to not make the proposed edit. An explanation can be found in paper 17-215r1. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | execution of a statement of the segment Q_j on image Q,<br>then the action regarding X in the segment P_i on image P precedes the action regarding X in the segment Q_j on image Q." | | |
| US 042 | [213:3,4] | 11.06.6 | P1 | te | See the comment for 139:29. | After “identify” add “the executing image and”.<br>Delete the sentence “The executing …” | Paper 17-250r2 addressed this issue and was passed |
| US 043 | [214:34] | 11.06.9 | P2 | te | See the comment for 139:29. | After “identifies” add “that image and”. | Paper 17-250r2 addressed this issue and was passed |
| JP 044 | 8 | 11.06.9 | Page 215 | ed | The following statement should be inserted after "the same number of times in this team" in the same way as the description of SYNC ALL statement:<br>"The segments that executed before the FORM TEAM statement on an active image of the current team precede the segments that execute after the FORM TEAM statement on another active image of the current team." | | Paper 17-209r1 addressed this issue and was passed. |
| US 045 | [220:12] | 12.02.4 | P2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| US 046 | [243:24] | 12.06.4.5.1 | P4 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the latter where “appear” ought to be used. See 17-186. | Replace "occurrence" with "appearance". | The editor declined this suggestion. |
| US 047 | [268:0+6] | 13.02.2 | NOTE 13.2 | ed | The term "occur" is used mostly to refer to events, and “appear” is used with respect to syntax entities. But sometimes “occur” is used for the | Replace "occurs" with "appears". | The editor declined this suggestion. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | latter where “appear” ought to be used. See 17-186. | | |
| US 048 | [284:13] | 13.08.5 | P1 | ed | The scale factor P affects the editing done by the G edit descriptor for “numeric quantities” but the text fails to exclude integer formatting, for which scale factors are inapplicable. | In line 14, replace “numeric” with “real and complex” | Paper 17-241r1 addressed this issue and was passed. |
| US 049 | [297:1] | 14.02.2 | P2 | ed | An entity accessed by use association is specified to be previously defined in the scoping unit where it is accessed. This is correct for type definitions, but not for variable declarations. See 17-185. | Before "defined" insert "declared or". | Paper 17-185r1 addressed this issue and was passed. |
| GB 050 | 303:7 | 15.04.1 | Para 1 | te | The use of the word “recursive” near the end of the paragraph is wrong because an interface cannot determine whether a procedure invokes itself. | Near the end of the paragraph, change “whether it is recursive” to “whether it has the NON_RECURSIVE attribute”. | Paper 17-239 addressed this issue and was passed. |
| US 051 | [310:15] | 15.04.3.4.5 | C1514(4) | ed | The term “earlier” is usually used for a temporal relationship. For the positional relationship of syntax entities, “previously” or “before” should be used. See 17-185. | Replace "earlier in the argument list than" with "appear in the argument list before". Since this doesn't refer to a definition, this edit might not be necessary. | The editor declined this suggestion. |
| US 052 | [310:14] | 15.04.3.4.5 | C1514(4) | ed | The term “occur” usually applies to events. The term “appear” should be used for syntax terms. See 17-186. | Replace "occur" with "appear". | The editor declined this suggestion. |
| GB 053 | 313:20 | 15.05.1 | Para 1 | te | The effect of C1528 is to allow a *data-ref* of the form `coarray[1]%pointer%allocpoly` but not to allow a *data-ref* of the form `coarray[1]%allocatable%allocpoly`. We believe that it was intended to disallow both. | Delete "subobject of a". | Paper 17-248 addressed this issue and was passed. |
| GB 054 | 320:30+ | 15.05.2.7 | Para 1 | ed | A note is needed to explain that C1542 does not constrain an intrinsic function such as ASSOCIATED. | At the end of the paragraph add “NOTE 15.26a Constraint C1542 does not apply to any intrinsic procedure because the intrinsic | Paper 17-237 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | | procedures are defined for their actual arguments." | |
| US 055 | [330:33-34] | 15.06.2.1 | P6 | ed | C1555 is covered by C1554. | Merge C1555 and C15554 and revise as 17-140 explains that these are not identical because there is a difference between "interoperable" and "of interoperable type". | The editor believes that the text is correct as-is. However, see paper 17-235 which modified C1555. |
| GB 056 | 329:28 | 15.06.2.1 | Para 2 | ed | An interface cannot specify that a procedure will call itself but it can specify that it will not do so. | Change "recursive" to "nonrecursive". | Paper 17-239 addressed this issue and was passed. |
| GB 057 | 330:16 | 15.06.2.1 | Para 3 | ed | The standard should state that a procedure declared with an asterisk *type-param-value* cannot be invoked recursively. | Append in obsolescent font: "A function name declared with an asterisk *type-param-value* shall not directly or indirectly invoke itself or any other procedure defined by the subprogram." | Paper 17-233 addressed this issue and was passed. |
| GB 058 | 330:34 | 15.06.2.1 | Para 6 | ed | C1555 does not allow for a type being non-interoperable because of a kind type parameter value. | At the end of C1555, add "and kind type parameters". | Paper 17-235 addressed this issue and was passed. |
| US 059 | [331:31,33] | 15.06.2.2 | P4 | ed | The term "occur" is used mostly to refer to events, and "appear" is used with respect to syntax entities. But sometimes "occur" is used for the latter where "appear" ought to be used. See 17-186. | Replace "occurrences" with "appearances" twice. | The editor declined this suggestion. |
| US 060 | [333:32-33] | 15.06.2.5 | P3 | ed | The description of recursive separate module subprograms can be simplified because the default is RECURSIVE. | Replace beginning of second sentence "It is recursive if and only if it is declared to be recursive" with "It is recursive unless it is declared to be nonrecursive" 17-187 | Paper 17-239 addressed this issue and was passed. |
| GB 061 | 333:33 | 15.06.2.5 | Para 3 | ed | A declaration cannot specify that a procedure will call itself but it can specify that it will not do so. | Change "It is recursive if and only if it is declared to be recursive by the interface body" to "It has the NON_RECURSIVE attribute if and only if it was declared to have that attribute by the interface body". | Paper 17-239 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| GB 062 | 353:6 | 16.09.1 | Para 2 | te | The third sentence of the paragraph, “The result characteristics are sometimes specified in terms of the characteristics of dummy arguments.” is not helpful because the detailed specifications are for the actual arguments, see the second sentence. | Delete the third sentence, “The result characteristics are sometimes specified in terms of the characteristics of dummy arguments.” | Paper 17-245r1 addressed this issue and was passed. |
| US 063 | [383:20]<br>[425:6] | 16.09.76<br>16.9.165 | | te | The same discussion that led me to examine 10.2.1.3 also led me to look at the specifications of the intrinsic functions SAME_TYPE_AS and EXTENDS_TYPE_OF. The issues I see there are less definitive, but I include them here in case circumstances prove favorable for addressing them:<br>1. Taken by itself, the second sentence of NOTE 16.26 is vaguely mysterious. It is only by looking elsewhere in the standard that I conclude that this was intended to convey that if either argument to SAME_TYPE_AS is an unlimited polymorphic that is disassociated or unallocated, SAME_TYPE_AS should return false. I suggest that be made explicit, either in the specification or the note.<br>2. I am troubled by the aspects of these functions that are processor dependent. In the absence of any way for a program to know whether the results it receives are from a well-specified or processor-defined case, there is no way to attach any meaning to those results. I suggest adding a note to both functions encouraging a policy for these cases of “when in doubt, return false”. Such a policy would allow a program to attach meaning to true results.<br>3. I would prefer it if the results for intrinsic dynamic types were consistent with those for extensible derived types, rather than processor dependent. However, I | My preference would be to “fix” the specifications to require matching corresponding type parameter values for a true result. Alternatively, the functions could be marked now as obsolescent, since no replacement should be necessary for features which have no useful functionality. Simply ignoring the problem does not strike me as a reasonable response. | Paper 17-247r1 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | recognize that "processor dependent" would allow processors to implement that way now and allow a future revision to impose that requirement, so I am not proposing any change in this regard unless changes here are deemed necessary for some other reason.<br>4. I am of the opinion that a serious error of omission took place with respect to these functions during the creation of F2003 and that this error has been propagated to subsequent revisions. These functions were precursors of the TYPE IS and CLASS IS type guards in the SELECT TYPE construct and were expected to produce equivalent results. When parameterized derived types were added to F2003, the type guards were modified to require kind type parameter value matching, but an equivalent change was not made in these two functions. Unfortunately, the result of this omission was not an incomplete or broken specification, just one sometimes giving the "wrong" answer. In the general case, there is nothing more that can be done with two objects known to be of the same type, but not necessarily the same kind type parameter values, than can be done with objects of different types. | | |
| US 064 | [389:14-16] | 16.09.85 | P5 | te | See the comment for 139:29. | After "identifies" add "the executing image and", thrice. | Paper 17-250r2 addressed this issue and was passed |
| DE 065 | 492 | 18.03.6 | NOTE 18.15 | ge | The NOTE needs an update to account for the extended interop features added to the language. | After "never interoperable", add " with a C array."<br>After "assumed-size array.", add "If it is a dummy argument, it might interoperate with a C descriptor (18.4, 18.3.7)." | We disagree; such variables are not themselves interoperable in that they do not "correspond to a formal |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | | | parameter of the prototype", but they can be compatible with a dummy argument declaration in a procedure interface. See 18.3.7p2(5) |
| DE 066 | 500: 22-23 | 18.05.5.2 | | te | The text excludes the ability to obtain a valid address if the C descriptor is for an assumed-size array. | After "of the C descriptor", add ", except that if the C descriptor is for an assumed-size array, the restriction on the value of subscripts[n-1] is reduced to that it must be larger than or equal to the lower bound of dimension n-1." | Paper 17-211r2 addressed this issue and was passed. |
| DE 067 | 504: 18-29 | 18.05.5.7 | | ed | This appears to be the description of the functions' effect, but is misplaced. | Move the complete text block to 504:40+ and prepend the "Description." header. | Paper 17-199r1 addressed this issue and was passed. |
| DE 068 | 505: 41-42 | 18.05.5.8 | | te | The *source* parameter must not be a C descriptor for an assumed-size array, because as specified currently *result* would usually end up being both assumed-size and non-contiguous. | After "nonallocatable nonpointer array", add "that is not assumed-size". | Paper 17-214r2 addressed this issue and was passed. |
| DE 069 | 506:35 | 18.05.5.9 | | te | The *source* parameter must not be a C descriptor for an assumed-size array, because pointer assignment requires that the target have a shape. | After "nonallocatable nonpointer data object", add "that is not an assumed-size array" | Paper 17-214r2 addressed this issue and was passed. |
| DE 070 | 507:1 | 18.05.5.9 | | ed | This appears to be misplaced | Move the Result Value para to 507:10+. | Paper 17-198 addressed this issue and was passed. |
| DE 071 | 507:24+ | 18.06 | | te | An assumed-shape dummy argument cannot be associated with an assumed-size actual argument. 318:28-29 states this for Fortran, but it may be appropriate to reiterate this for C.<br>If it is decided that no normative text is needed, I suggest rewording the edit as a NOTE, adding a reference to subclause 15.5.2.4. | Add a new paragraph:<br>"If the address of a C descriptor is a C actual argument that corresponds to an assumed-shape Fortran dummy argument, that descriptor shall not be for an assumed-size array." | Paper 17-214r2 addressed this issue and was passed. |
| US 072 | [516:4] | 19.03.5 | P2 | ed | The term "occur" is used mostly to refer to events, and "appear" is used with respect to syntax entities. But sometimes "occur" is used for the latter where "appear" ought to be used. See 17-186. | Replace "occurs" with "appears". | The editor declined this suggestion. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| US 073 | [516:12] | 19.04 | P1 | te | Subclause 19.4 prohibits implicit type declaration of construct entities of BLOCK constructs. In light of 8.7p4 saying "as if … explicitly" this might not be necessary. See 17-184. | Replace "explicitly" with "explicitly or implicitly". | Paper 17-232 addressed this issue and was passed. |
| US 074 | [516:13] | 19.04 | P4 | te | A variable within a BLOCK construct that is typed implicitly and otherwise declared only in an ASYNCHRONOUS or VOLATILE statement is not a construct entity, but ought to be. See 17-184. | Replace "only in ASYNCHRONOUS and VOLATILE" with "one accessed by use or host association and declared only in ASYNCHRONOUS or VOLATILE". | The editor believes that the current text is correct. |
| US 075 | [518:1] | 19.05.1.4 | P1 | te | An entity accessed by use association is considered to be previously defined or declared within the scoping unit in which it is accessed. The same ought to be explicit for entities accessed by host association. See 17-185. | Before "In the case..." insert a sentence "A host-associated entity is considered to have been previously declared or defined." Compare to 14.2.2p2 at [297:1]. | Paper 17-185r1 addressed this issue and was passed. |
| DE 076 | 562:32 | C.6.07 | | te | The use of TEAM_NUMBER in the example is invalid with the rules as currently specified.<br>The associating entity is established in the current team, according to [46:5-6]. But [139:33] says that TEAM_NUMBER requires its coarray to be established in an ancestor team.<br>My take is that [139:33] is too restrictive; it should be allowed to use an associating entity in this context because the selector must be established in an ancestor team anyway. The edits are crafted such as to permit TEAM_NUMBER= but exclude TEAM= for this situation. | [139:33] After "ancestor of the current team", add ", or an associating coarray specified in an enclosing CHANGE TEAM construct."<br>[562:41+] Add explanatory text:<br>"Because the selector for a coarray association in a CHANGE TEAM must be established when the statement is executed (11.1.5.2), the associating coarray will have the same type, type parameters and bounds on any pair of images of the ancestor team that are members of different sibling teams. This allows the use of TEAM_NUMBER in an image selector inside the block construct to address the corresponding associating coarray in a sibling team." | Paper 17-213r2 addressed this issue and was passed. |
| GB 077 | xix | Introduction | Para 2 | ed | The text in the bullet item "Program units and procedures" states that by default any procedure can be invoked recursively. This is not the case for those declared with an asterisk *type-param-value.* | In the bullet item "Program units and procedures", change "Procedures, including elemental procedures, can be invoked recursively by default;" to | Paper 17-233 addressed this issue and was passed. |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

| MB/ NC[1] | Line number | Clause/ Subclause | Paragraph/ Figure/Table | Type of comment[2] | Comments | Proposed change | Observations of the secretariat |
|---|---|---|---|---|---|---|---|
| | | | | | | "Procedures that are not declared with an asterisk *type-param-value* can be invoked recursively by default;". | |

1 **MB** = Member body / **NC** = National Committee (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)
2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial