ISO/IEC/JTC1/SC22/WG5-N1893

DRAFT TECHNICAL CORRIGENDUM 1

Notes for WG5:
Edits are included in this document from interpretations:
f03/0048, f03/0085, f03/0123, f03/0124, f08/0001, f08/0002, f08/0003, f08/0009, f08/0010, f08/0011, f08/0013, f08/0014, f08/0015, f08/0016, f08/0017/18, f08/0019, f08/0020, f08/0021, f08/0022, f08/0023, f08/0024, f08/0025, f08/0026, f08/0027, f08/0028, f08/0030, f08/0033, f08/0034, f08/0036, f08/0037, f08/0039, f08/0046, f08/0049, f08/0050, f08/0051, f08/0052, f08/0053.

Interpretations approved without generating edits:
f03/0091, f03/0105, f03/0110, f03/0128, f08/0006, f08/0007, f08/0035, f08/0044, f08/0047.

Page and line numbers on the right refer to 10-007r1. They, the interpretation references and notes in italics are for WG5 use only and will be deleted before the document is submitted to ISO.

Paragraph numbers are now printed in the ISO version of the standard so it is not totally absurd to have to refer to the twenty-fifth paragraph of a subclause.

[xv] *f08/0046*

Introduction

At the end of the fourth item in the bulleted list (Data declaration), append the sentence: "The VALUE attribute is permitted for an array, and for an object with a nonconstant length type parameter.".

[xvi] *f08/0051*

In the last item in the main bulleted list (Programs and procedures), before "An impure" insert the new sentence: "An argument to a pure procedure may have default INTENT if it has the VALUE attribute.".

[24:9] *f08/0011*

Subclause 1.6.2

In the first sentence of the first paragraph of the subclause change "This" to "Except as identified in this subclause, this".

[24:10] *f08/0033*

In the second sentence of the first paragraph change "Any" to "Except as identified in this subclause, any".

[24:11+] *f08/0011 and f08/0033*

Following the first paragraph of the subclause, add new paragraphs:

> Fortran 2003 specified that array constructors and structure constructors of finalizable type are finalized. This part of ISO/IEC 1539 specifies that these constructors are not finalized.
>
> Fortran 2003 permitted an INTENT(OUT) argument of a pure subroutine to be polymorphic. This is not permitted by this part of ISO/IEC 1539.

[76:10-, 25-26] *f08/0013*

Subclause 4.5.6.3

Move paragraph 9 of the subclause to precede paragraph 1. In addition, edit the paragraph by changing "the variable is" to "if the variable is not an unallocated allocatable variable, it is" and by appending a new sentence at the end of the paragraph: “If the variable is an allocated allocatable that would be deallocated by intrinsic assignment, the finalization occurs before the deallocation.”.

[76:10] *f08/0013*

In paragraph 1 of the subclause, after “it is finalized” insert “unless it is the variable in an intrinsic assignment (7.2.1.3) or a component thereof”

[76:17-18,21-22] *f08/0011*

Delete paragraphs 5 and 7 of the subclause.

{Note: It is assumed that f08/0034 refers to the new paragraph 8 introduced by f03/0085}

[76:23-24] *f03/0085 and f08/0034*

Replace paragraph 8 of the subclause with:

> When a procedure is invoked, an object that becomes argument associated with a nonpointer, nonallocatable INTENT(OUT) dummy argument of that procedure is finalized. The finalization caused by INTENT(OUT) is considered to occur within the invoked procedure; so for elemental procedures, an INTENT(OUT) argument will be finalized only if a scalar or elemental final subroutine is available, regardless of the rank of the actual argument.

[78:4] *f08/0052*

Subclause 4.5.7.3

In the first paragraph of the subclause, change “as a type-bound” to “as an accessible type-bound”.

[109:21-23] *f03/0123 and f08/0015*

Subclause 5.5

In the fourth paragraph of the subclause, delete the sentence “The mapping may ... scoping unit.” and replace “in the outermost inclusive scope in which it appears” by “; if the outermost inclusive scope in which it appears is not a type definition, it is declared in that scope, otherwise it is declared in the host of that scope”.

[111:19-20] *f08/0002*

Subclause 5.6

In the first sentence of the fifth paragraph, replace “type parameters, and shape” by “kind type parameters, and rank”.

{Note: f08/0014 collected an individual NO vote on grounds of lack of clarity but is included in this document.}

[124:4-7] *f08/0014 and f08/0016*

Subclause 6.5.3.3.2

Replace the second paragraph of the subclause by:

> A vector-subscripted array section shall not be finalized by a nonelemental final subroutine.

[124:9] *f08/0039*

In the third paragraph of the subclause, replace “shall ... (16.6.7)” with “is not definable and shall not be defined or become undefined”.

[130:23] *f08/0010*

Subclause 6.7.3.2
Add the following sentence to the end of the first paragraph: "An allocatable variable shall not be deallocated if it or any subobject of it is argument associated with a dummy argument or construct associated with an associate name.".

[131:27] *f08/0010*

Subclause 6.7.3.3
Add the following sentence to the end of the first paragraph: "A pointer shall not be deallocated if its target or any subobject thereof is argument associated with a dummy argument or construct associated with an associate name.".

[151:13-15] *f08/0050*

Subclause 7.1.11
Replace the ninth paragraph of the subclause by:

> A generic entity referenced in a specification expression in the *specification-part* of a scoping unit shall have no specific procedures defined in that scoping unit, or its host scoping unit, subsequent to the specification expression.

[152:26-28] *f08/0050*

Subclause 7.1.12
Replace the third paragraph of the subclause by:

> A generic entity referenced in a constant expression in the *specification-part* of a scoping unit shall have no specific procedures defined in that scoping unit, or its host scoping unit, subsequent to the constant expression.

[177:28-29] *f08/0028*

Subclause 8.1.6.6.4
In the first paragraph of the subclause replace the fourth item in the bulleted list with the following:

- A branch occurs within the range of a DO construct and the branch target statement is neither the *end–do* nor within the range of the same DO construct.

[178:8-9] *f08/0023*

Subclause 8.1.6.7
In the first paragraph of the subclause, in the second item in the bulleted list replace the first sentence by: "A pointer that is used in an iteration other than as the pointer in pointer assignment, allocation, or nullification, either shall be previously pointer-assigned, allocated, or nullified in that iteration or shall not have its pointer association changed during any iteration.".

[178:13-14] *f08/0025*

In the third item in the bulleted list replace the second sentence by: "An allocatable object that is referenced, defined, deallocated, or has its allocation status, dynamic type, or a deferred type parameter value inquired about, in any iteration, either shall be previously allocated in that iteration or shall not be allocated or deallocated in any other iteration.".

[178:15-16] *f08/0022*

Replace the fourth item in the bulleted list ("An input/output ... iteration.") by:

- If data are written to a file record or position in one iteration, that record or position in that file shall not be read from or written to in a different iteration.

[178:17-18] *f08/0026*

Delete the fifth item in the bulleted list ("Records ... order.").

{Note: This adopts Corbett's suggestion of removing redundant text from the approved edit.}

[178:18+] *f08/0026*

At the end of the first paragraph, and before Note 8.9, add the new paragraph:

> The ordering between records written by different iterations is indeterminate.

[227:15] *f03/0048*

Subclause 9.6.4.8

In the twenty-fifth paragraph of the subclause, delete "record positioning".

[227:17-18] *f03/0048*

In the twenty-sixth paragraph, replace "A record positioning edit descriptor, such as TL and TR," by "The edit descriptors T and TL" and replace "record position" by "file position" twice.

[246:15] *f08/0030*

Subclause 10.3.1

After constraint C1002, add a new constraint:

> C1002A (R1005) An *unlimited-format-item* shall contain at least one data edit descriptor.

[249:11+] *f08/0030*

Subclause 10.4

After the seventh paragraph of the subclause, insert a new paragraph:

> If format control encounters the rightmost parenthesis of an unlimited format item, format control reverts to the leftmost parenthesis of that unlimited format item. This reversion of format control has no effect on the changeable modes (9.5.2).

[249:19-20] *f08/0030*

In the last sentence of the eighth paragraph of the subclause, change "If format control reverts ... , the" to "The".

[286:4] *f08/0001*

Subclause 12.4.3.4.5

In the third paragraph, in the third item in the bulleted list, after "the other has the POINTER attribute", insert "and not the INTENT(IN) attribute".

[286:12-13] *f08/0053*

In the third paragraph of the subclause, in constraint C1214 replace "two ... identifier" by "if two procedures have the same generic identifier, their **`dtv`** arguments (9.6.4.8.3)".

[286:38] *f08/0053*

In the fifth paragraph of the subclause, replace "applies to" by "is consistent with".

[287:15+] *f08/0037*

Subclause 12.4.3.6

In rule R1213 in the first paragraph, following the line "**or** POINTER", add the new line

> **or** PROTECTED

[295:3] *f08/0014*

Subclause 12.5.2.4
In paragraph 18 of the subclause, after “If” insert “the procedure is nonelemental and”.

[312:23+] *f08/0033*

Subclause 12.7
{Note: It is assumed that Note 12.47 should continue to be immediately after C1278.}
Following constraint C1278 and Note 12.47, insert new constraint:

C1278a An INTENT(OUT) dummy argument of a pure procedure shall not be polymorphic.

[313:4+] *f08/0033*

Following constraint C1284, insert new constraint and new note:

C1284a A statement that might result in the deallocation of a polymorphic entity is not permitted in a pure procedure.

NOTE 12.48x
Apart from the DEALLOCATE statement, this includes intrinsic assignment if the variable has a polymorphic allocatable component at any level of component selection that does not involve a pointer component but which might involve one or more allocatable components.

[314:4-5] *f08/0049*

Subclause 12.8.1
In constraint C1290, delete “, and shall not ... constant expression”.

[314:5+] *f08/0024 and f08/0049*

Following constraint C1290 insert two new constraints:

C1290a The *specification-part* of an elemental subprogram shall specify the intents of all of its dummy arguments that do not have the VALUE attribute.

{Note: New constraint C1290a was added by f08/0024. That from f08/0049 is now shown as C1290b. Also ‘enquiry’ -> ‘inquiry’ as both words were used in the approved edit.}

C1290b In the *specification-expr* that specifies a type parameter value of the result of an elemental function, an object designator with a dummy argument of the function as the base object shall appear only as the subject of a specification inquiry, and that specification inquiry shall not depend on a property that is deferred.

[314:5+] *f08/0018*

At the end of the subclause, insert the new paragraph:

In a reference to an elemental procedure, if any argument is an array, all actual arguments that correspond to INTENT (OUT) or INTENT (INOUT) dummy arguments shall be arrays. All actual arguments shall be conformable.

[314:9-10] *f08/0018*

Subclause 12.8.2
In the first paragraph of the subclause delete the sentence “For those elemental ... conformable.”.

[314:14-17] *f08/0018*

Subclause 12.8.3

Delete the sentence “In a reference ... conformable with them.”.

[316:24-25] *f08/0003*

Subclause 13.2.4

In the second sentence of the first paragraph of the subclause, replace “an optional” by “a” and replace “, if present, specifies” by “can specify”.

[319, 322, 323] *f08/0003*

Subclause 13.5

In Table 13.1 replace

“ALL	(MASK [, DIM])”	by	“ALL	(MASK) or (MASK, DIM)”;
“ANY	(MASK [, DIM])”	by	“ANY	(MASK) or (MASK, DIM)”;
“NORM2	(X [, DIM])”	by	“NORM2	(X) or (X, DIM)”;
“PARITY	(MASK [, DIM])”	by	“PARITY	(MASK) or (MASK, DIM)”;
“THIS_IMAGE	(COARRAY[, DIM])”	by	“THIS_IMAGE	(COARRAY) or (COARRAY, DIM)”.

[328:2] *f08/0003*

Subclause 13.7.10

Replace the subclause heading by “**ALL (MASK, DIM) or ALL (MASK)**”.

[328:7] *f08/0003*

In the description of the DIM argument, delete “(optional)”.

[328:10] *f08/0003*

In the description of Result Characteristics, replace “is absent” by “does not appear”.

[329:6] *f08/0003*

Subclause 13.7.13

Replace the subclause heading by “**ANY (MASK, DIM) or ANY (MASK)**”.

[329:11] *f08/0003*

In the description of the DIM argument, delete “(optional)”.

[329:14] *f08/0003*

In the description of Result Characteristics, replace “is absent” by “does not appear”.

[332:25] *f08/0027*

Subclause 13.7.21

In the fourth paragraph of the subclause, change “CALL ATOMIC_REF (I[3], VAL)” to
“CALL ATOMIC_REF (VAL, I[3])”.

[333:12-14] *f08/0019*

Subclause 13.7.24

In the third paragraph of the subclause, in the lines beginning N1 and N2, replace “of type integer and nonnegative” by “an integer scalar with a nonnegative value” and in the line beginning X, after “real” insert “; if the function is transformational, X shall be scalar”.

[334:12-14] *f08/0019*

Subclause 13.7.27

In the third paragraph of the subclause, in the lines beginning N1 and N2, replace “of type integer and nonnegative” by “an integer scalar with a nonnegative value” and in the line beginning X, after “real” insert “; if the function is transformational, X shall be scalar”.

[338:31] *f08/0003*

Subclause 13.7.41
In the description of the DIM argument, after “dummy argument” insert “, a disassociated pointer, or an unallocated allocatable”.

[347:31-32] *f08/0020*

Subclause 13.7.61
In the third paragraph of the subclause, for the VALUE argument, replace “for ... 7.1.5.5.2)” by “for the operator == or the operator .EQV.”.

[360:4] *f08/0003*

Subclause 13.7.90
In the description of the DIM argument, after “dummy argument” insert “, a disassociated pointer, or an unallocated allocatable”.

[360:25] *f08/0003*

Subclause 13.7.91
In the description of the DIM argument, after “dummy argument” insert “, a disassociated pointer, or an unallocated allocatable”.

[374:24] *f08/0003*

Subclause 13.7.123
Replace the subclause heading by “**NORM2 (X, DIM) or NORM2 (X)**”.

[374:29] *f08/0003*

In the description of the DIM argument, delete “(optional)”.

[374:31] *f08/0003*

In the description of Result Characteristics, replace “is absent” by “does not appear”.

[377:20] *f08/0003*

Subclause 13.7.128
Replace the subclause heading by “**PARITY (MASK, DIM) or PARITY (MASK)**”.

[377:25] *f08/0003*

In the description of the DIM argument, delete “(optional)”.

[377:28] *f08/0003*

In the description of Result Characteristics, replace “is absent” by “does not appear”.

[390:6] *f08/0021*

Subclause 13.7.160
In the third paragraph of the subclause, change “has any deferred type parameters” to “is unlimited polymorphic or has any deferred type parameters,”.

[392:6] *f08/0003*

Subclause 13.7.165
In the subclause heading replace “**or THIS_IMAGE (COARRAY[, DIM])**” by “**, THIS_IMAGE (COARRAY) or THIS_IMAGE (COARRAY, DIM)**”.

[392:11] *f08/0003*

In the description of the DIM argument, delete “(optional)”.

[394:27] *f08/0003*

Subclause 13.7.171
In the description of the DIM argument, after “dummy argument” insert “, a disassociated pointer, or an unallocated allocatable”.

[395:11] *f08/0003*

Subclause 13.7.172

In the description of the DIM argument, after “dummy argument” insert “, a disassociated pointer, or an unallocated allocatable”.

[406:15+] *f08/0009*

Subclause 14.9

In the first paragraph, add a new item after the second item of the bulleted list:

- the IEEE function abs shall be provided by the intrinsic function ABS,

{Note: The formatting of the new text has been changed so as to be consistent with that of item (9). This means that the second edit in f03/0124 is redundant.}

[455:4-10] *f03/0124*

Subclause 16.6.6

In the first paragraph replace item (1) entirely by:

(1) When a scalar variable of intrinsic type becomes defined, all totally associated variables of different type become undefined.
When a double precision scalar variable becomes defined, all partially associated scalar variables become undefined.
When a scalar variable becomes defined, all partially associated double precision scalar variables become undefined.

[487:28] *f03/0048*

Subclause C.6.2

In the third sentence of the first paragraph, delete “record positioning”.

[527:18] *f08/0036*

Subclause C.13.3.6

In the third paragraph of the subclause, replace “$|X_i|$” by “$|X_i|^2$”.