INTERNATIONAL STANDARD ISO/IEC 1539-1:2010
TECHNICAL CORRIGENDUM 4

Published 2016-07-01

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Programming languages — Fortran —

Part 1: Base language

TECHNICAL CORRIGENDUM 4

Technologies de l'information — Langages de programmation — Fortran —

Partie 1: Langage de base

RECTIFICATIF TECHNIQUE 4

Technical Corrigendum 4 to ISO/IEC 1539-1:2010 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*.

ICS 35.060 **Ref. No. ISO/IEC 1539-1:2010/Cor.4:2016(E)**

Published in Switzerland

ISO/IEC 1539-1:2010 - TECHNICAL CORRIGENDUM 4

Introduction
In the second paragraph, append to the 'Data declaration' bullet point: "A *defined-operator* can be used in a specification expression.".

In the second paragraph, append two new sentences to the 'Data usage and computation' bullet point: "All transformational functions from the intrinsic modules IEEE_ARITHMETIC and IEEE_EXCEPTIONS can be used in constant expressions. All transformational functions from the intrinsic modules IEEE_ARITHMETIC, IEEE_EXCEPTIONS, and ISO_C_BINDING can be used in specification expressions."

In the second paragraph, in the 'Intrinsic modules' bullet point, before "The function C_SIZEOF" insert the new sentence:

> A contiguous array variable that is not interoperable but which has interoperable type and kind type parameter (if any), and a scalar character variable with length>1 and kind C_CHAR, can be used as the argument of the function C_LOC in the intrinsic module ISO_C_BINDING, provided the variable has the POINTER or TARGET attribute.

In the second paragraph, in the 'Programs and procedures' bullet point, replace the final sentence, "A line in the program is permitted to begin with a semicolon." with "A free form continuation line can begin with zero or more blanks followed by a semicolon.". Also, append the sentence, "The name of an external procedure that has a binding label is a local identifier and not a global identifier.".

Subclause 1.3
After the definition of **parent component** (1.3.33.2) insert a new term:

> **1.3.33.2a**
> **potential subobject component**
> nonpointer component, or potential subobject component of a nonpointer component

Subclause 1.6.2
After the six paragraphs added to the subclause in Technical Corrigendum 2, add the following new paragraphs:

> Fortran 2003 interpreted assignment to an allocatable variable from a nonconformable array as intrinsic assignment, even when an elemental defined assignment was in scope; this part of ISO/IEC 1539 does not permit assignment from a nonconformable array in this context.
>
> Fortran 2003 permitted a statement function to be of parameterized derived type; this part of ISO/IEC 1539 does not permit that.

Subclause 4.3.1.3
After the first paragraph, insert the following new paragraph:

> Where a data entity other than a component is declared explicitly using the CLASS specifier to be of derived type, the specified derived type shall have been defined previously. If the data entity is a function result, the derived type may be specified in the FUNCTION statement provided the derived type is defined within the body of the function or is accessible there by use or host association. If the derived type is specified in the FUNCTION statement and is defined within the body of the function, it is as if the function result variable were declared with that derived type immediately following its *derived-type-def*.

Subclause 4.5.4.6
In the first sentence of constraint C461, insert ", noncoindexed" after "nonallocatable" so that the sentence reads:

> C461 (R443) The designator shall designate a nonallocatable, noncoindexed variable that has the TARGET and SAVE attributes and does not have a vector subscript.

Subclause 5.3.4
In the first bullet point of the second paragraph, after "the variable" insert "is a dummy argument or" and insert a comma after "scoping unit".

Subclause 5.3.19
In constraint C560, after "for a coarray" insert ", or a variable with a coarray ultimate component,".

In constraint C561, after "for a coarray" insert ", or a variable with a coarray ultimate component,".

Subclause 5.6
Replace the second paragraph, by:

> The order in which the values appear on output is the same as the order of the *namelist-group-objects* in the namelist group object list; if a variable appears more than once as a *namelist-group-object* for the same namelist group, its value appears once for each occurrence.

Subclause 6.4.2
In constraint C617, replace "subcomponent" with "potential subobject component".

Subclause 6.7.1.1
In constraint C642, change "C_PTR," to "C_PTR or" and delete ", LOCK_TYPE ... LOCK_TYPE".

Following constraint C642, add a new constraint:

> C642a (R627) If SOURCE= appears, the declared type of *source-expr* shall not be LOCK_TYPE or have a potential subobject component of type LOCK_TYPE.

Instead of the edit in Technical Corrigendum 2, which replaced the entire fourth paragraph of the subclause, make the following change to this paragraph: replace "If *allocate-object* is" by "If an ALLOCATE statement has a SOURCE= specifier and an *allocate-object* that is". The edited paragraph thus reads:

> If an ALLOCATE statement has a SOURCE= specifier and an *allocate-object* that is a coarray, *source-expr* shall not have a dynamic type of C_PTR, C_FUNPTR, or LOCK_TYPE, or have a subcomponent whose dynamic type is LOCK_TYPE.

Subclause 6.7.1.2
In the second sentence of the fourth paragraph, change "On each image" to "If no error condition other than STAT_STOPPED_IMAGE occurs" and change "all other images" to "all non-stopped images". Append a new sentence so that the entire paragraph reads:

> When an ALLOCATE statement is executed for which an *allocate-object* is a coarray, there is an implicit synchronization of all images. If no error condition other than STAT_STOPPED_IMAGE occurs, execution of the segment (8.5.2) following the statement is delayed until all non-stopped images have executed the same statement the same number of times. The coarray shall not become allocated on an image unless it is successfully allocated on all non-stopped images.

Subclause 6.7.1.3
In the second bulleted item of the first paragraph, fourth sentence, insert "nonoptional" before "nonallocatable dummy argument".

Subclause 6.7.3.2
In the second sentence of the eleventh paragraph, change "On each image" to "If no error condition other than STAT_STOPPED_IMAGE occurs" and change "all other images" to "all non-stopped images". Append a new sentence so that the entire paragraph reads:

> When a DEALLOCATE statement is executed for which an *allocate-object* is a coarray, there is an implicit synchronization of all images. If no error condition other than STAT_STOPPED_IMAGE occurs, execution of the segment (8.5.2) following the statement is delayed until all non-stopped images have executed the same statement the same number of times. If the coarray is a dummy argument, its ultimate argument (12.5.2.3) shall be the same coarray on every image. The coarray shall not become deallocated on an image unless it is successfully deallocated on all non-stopped images.

Subclause 6.7.4
Append to the first paragraph the sentence: "The *stat-variable* shall not depend on the value of the *errmsg-variable*.".

Subclause 6.7.5
Append to the first paragraph the sentence: "The *errmsg-variable* shall not depend on the value of the *stat-variable*.".

Subclause 7.1.11
In the second paragraph after item (10), insert a new item:

> (10a) a reference to a transformational function from the intrinsic module IEEE_ARITHMETIC (14), IEEE_EXCEPTIONS (14), or ISO_C_BINDING (15.2), where each argument is a restricted expression,

In the eighth paragraph, replace "a type parameter or an array bound" with "a type parameter, array bound, or cobound" and replace "the type parameter, or array bound" with "the type parameter, array bound, or cobound".

Subclause 7.1.12
In the first paragraph, replace item (8) by:

> (8) a reference to a transformational function from the intrinsic module IEEE_ARITHMETIC or IEEE_EXCEPTIONS (14), where each argument is a constant expression,

Subclause 7.2.1.2
In the second paragraph after "coindexed object," delete "the variable". Before each "shall not" insert "the variable" and before "shall have" insert "of the variable" so that the second paragraph reads:

> If *variable* is a coindexed object,
> - the variable shall not be polymorphic,
> - the variable shall not have an allocatable ultimate component, and
> - each deferred length type parameter of the variable shall have the same value as the corresponding type parameter of *expr*.

Subclause 7.2.1.4
In item (5) (b) of the second paragraph, change "x_1 and x_2 are conformable" to "x_2 is scalar or has the same rank as x_1".

In the third paragraph of the subclause, append a new sentence:

> If the subroutine is elemental, x_2 shall have the same shape as x_1.

Subclause 8.1.3.1
In constraint C801, change "*associate-name* shall not appear" to "neither the *associate-name* nor any subobject thereof shall appear".

Subclause 8.1.3.3
In the second paragraph, change "the associate name shall not appear" to "neither the associate name nor any subobject thereof shall appear".

Subclause 8.1.4
Following the third paragraph, and before NOTE 8.5, insert a new paragraph:

It is permissible to branch to an *end-block-stmt* only from within its BLOCK construct.

Subclause 8.1.5
Following the third paragraph, and before NOTE 8.6, insert a new paragraph:

It is permissible to branch to an *end-critical-stmt* only from within its CRITICAL construct.

Subclause 8.1.6.7
In the first paragraph, after the fourth bullet point, add new bullet point:

- A DO CONCURRENT construct shall not contain an input/output statement that has an ADVANCE= specifier.

Subclause 8.1.9.1
In constraint C836, change "*associate-name* shall not appear" to "neither the *associate-name* nor any subobject thereof shall appear".

Subclause 8.5.3
Following constraint C851, add new constraint:

C851a (R859) A *stat-variable* or *errmsg-variable* in a *sync-stat* shall not be a coindexed object.

Subclause 8.5.4
Following constraint C852 and before the first paragraph, insert a new paragraph:

The value of *image-set* shall not depend on the value of *stat-variable* or *errmsg-variable*.

Subclause 8.5.6
Following constraint C853 and before the first paragraph, insert a new paragraph:

The *lock-variable* shall not depend on the value of *stat-variable*, *errmsg-variable*, or the *scalar-logical-variable* in the ACQUIRED_LOCK= specifier. The *scalar-logical-variable* shall not depend on the value of the *lock-variable*, *stat-variable*, or *errmsg-variable*.

Subclause 8.5.7
Before the first paragraph, insert a new paragraph:

The *stat-variable* shall not depend on the value of the *errmsg-variable*, *lock-variable*, or the *scalar-logical-variable* in the ACQUIRED_LOCK= specifier. The *errmsg-variable* shall not depend on the value of the *stat-variable*, *lock-variable*, or the *scalar-logical-variable* in the ACQUIRED_LOCK= specifier.

Subclause 11.2.3
In constraint C1113, after "shall be the name of a nonintrinsic module" insert "that declares a separate module procedure".

Subclause 12.4.3.2
At the end of the first sentence of the fifth paragraph, change “or a dummy procedure” to “, dummy procedure, or procedure pointer”.

In the second sentence of the same paragraph, after “interface body, the procedure is a dummy procedure” change “; otherwise” to “. If the procedure has the POINTER attribute, it is a procedure pointer. If it is not a dummy procedure or procedure pointer”.

This makes that whole paragraph read:

> An interface body in a generic or specific interface block specifies the EXTERNAL attribute and an explicit specific interface for an external procedure, dummy procedure, or procedure pointer. If the name of the declared procedure is that of a dummy argument in the subprogram containing the interface body, the procedure is a dummy procedure. If the procedure has the POINTER attribute, it is a procedure pointer. If it is not a dummy procedure or procedure pointer, it is an external procedure.

Subclause 12.4.3.3
In the first paragraph, after “imported in this manner and is” change “defined” to “declared”.

In the second paragraph, after “is accessed by host association and is” change “defined” to “declared”.

Subclause 12.5.2.4
In the eighteenth paragraph, after applying the changes in Technical Corrigendum 1, between “is nonelemental” and “and the actual argument”, insert “, the dummy argument does not have the VALUE attribute,”. This makes the whole sentence read:

> If the procedure is nonelemental, the dummy argument does not have the VALUE attribute, and the actual argument is an array section having a vector subscript, the dummy argument is not definable and shall not have the ASYNCHRONOUS, INTENT (OUT), INTENT (INOUT), or VOLATILE attributes.

After the eighteenth paragraph, add the following new paragraph before NOTE 12.24:

> If the dummy argument has a coarray ultimate component, the corresponding actual argument shall have the VOLATILE attribute if and only if the dummy argument has the VOLATILE attribute.

In constraint C1238, append at the end of the sentence: “, unless the dummy argument has the VALUE attribute”.

In constraint C1239, after “ASYNCHRONOUS attribute” insert: “, but does not have the VALUE attribute”.

In constraint C1240, after “ASYNCHRONOUS attribute” insert: “, but does not have the VALUE attribute”.

Subclause 12.5.2.13
In item (3)(b) of the first paragraph, after “target other than” insert “a coindexed object or”.

In item (4)(b) of the first paragraph, after “target other than” insert “a coindexed object or”.

Subclause 12.6.4
In the first paragraph, following constraint C1275 add new constraint:

> C1275a A statement function shall not be of a parameterized derived type.

