

Information technology – Programming languages – Fortran

TECHNICAL CORRIGENDUM 4

Technical Corrigendum 4 to International Standard ISO/IEC 1539-1:2004 (E) was prepared by Joint Technical Committee ISO/IEC JTC1, Information Technology.

Subclause 5.1.2.16

In the third paragraph of the subclause replace “association status and array bounds” by “association status, dynamic type and type parameters, and array bounds”.

Subclause 6

Delete rule R605, which defines *default-logical-variable*, and constraint C604.

Subclause 6.2.1

Replace the first paragraph of the subclause (A **whole array** ... to the name.) by the following:

A **whole array** is a named array or a structure component whose final *part-ref* is an array component name; no subscript list is appended.

In the third paragraph, replace both occurrences of “whole array name” by “whole array designator”.

Subclause 6.3.1.1

At the end of the last sentence in the last paragraph of the subclause insert “unless the SOURCE= specifier appears and the corresponding component of the *source-expr* is allocated”.

Subclause 9.5.1

In rule R913, replace “ID = *scalar-int-variable*” by “ID = *id-variable*”.

Following rule R913, insert new rule and new constraint:

R913a *id-variable* **is** *scalar-int-variable*

C908a (R913a) The *scalar-int-variable* shall have a decimal range no smaller than that of default integer.

Subclause 9.5.1.8

In the second sentence of the first paragraph of the subclause, replace “This value” by “If this value is zero, the data transfer operation has been completed. A nonzero value”.

After the second sentence of the first paragraph, insert the following sentence, “This identifier is different from the identifier of any other pending data transfer operation for this unit.”

Subclause 9.5.3

After the last paragraph of the subclause, insert the following new paragraph:

If execution of the program is terminated during execution of a WRITE or PRINT statement, the contents of the file become undefined.

Subclause 9.6.1

In the first sentence of the third paragraph of the subclause, replace “the identifier” by “zero or the identifier”.

In the second sentence of the same paragraph, after “transfer operation” insert “, if any,”.

Subclause 9.9.1

In rule R930, replace:

“or EXIST = *scalar-default-logical variable*” by “or EXIST = *scalar-logical variable*”,
“or NAMED = *scalar-default-logical variable*” by “or NAMED = *scalar-logical variable*”,
“or OPENED = *scalar-default-logical variable*” by “or OPENED = *scalar-logical variable*”, and
“or PENDING = *scalar-default-logical variable*” by “or PENDING = *scalar-logical variable*”.

Subclause 9.9.1.10

In the first sentence of the subclause, replace “*scalar-default-logical variable*” by “*scalar-logical-variable*”.

Subclause 9.9.1.15

Replace “*scalar-default-logical variable*” by “*scalar-logical-variable*”.

Subclause 9.9.1.18

In both the first and second sentences of the subclause, replace “*scalar-default-logical variable*” by “*scalar-logical-variable*”.

Subclause 9.11

Delete the last paragraph of the subclause, viz. “A STOP statement shall not be executed during execution of an input/output statement”.

Subclause 10.9.2

Add the following sentence at the end of the first paragraph of the subclause and before the new text that was added in Technical Corrigendum 3:

Two undelimited character sequences are considered adjacent when both were written using list-directed input/output, no intervening data transfer or input/output file positioning operations on that unit occurred, and both were written either by a single data transfer statement, or during the execution of a parent data transfer statement along with its child data transfer statements.

Subclause 12.4

Following constraint C1224, insert the paragraph

The *data-ref* in a *procedure-designator* shall not be an unallocated allocatable variable or a pointer that is not associated.

Subclause 12.4.1.2

After “of type default character” in the second paragraph of the subclause, add “or of type character with the C character kind (15.1.1)”.

After “of type default character” in the first sentence of the third paragraph, add “or of type character with the C character kind”.

After “of type default character” in the third sentence of the third paragraph, add “or of type character with the C character kind”.

Subclause 12.4.1.6

Following list item 10 of the subclause, insert:

- (11) It shall not be supplied as the *data-ref* in a *procedure-designator*.
- (12) It shall not be supplied as the *variable* in a *proc-component-ref*.

Subclause 13.7.60

In *Case (i)* of the Result Value paragraph of the subclause, after “a whole array” delete “or array structure component”.

Subclause 13.7.124

In *Case (i)* of the Result Value paragraph of the subclause, after “a whole array” delete “or array structure component”.

Subclause 15.2.3

In the first sentence of the second paragraph, replace “have types and type parameters that are interoperable with the types of the corresponding components of the struct type” with “would interoperate with corresponding components of the C struct type as described in 15.2.4 and 15.2.5 if the components were variables”.

Subclause 16.4.2.1.4

After the second paragraph of the subclause, add the following new paragraph:

The association status of a pointer object with the VOLATILE attribute might change by means not specified by the program.

Subclause 16.5.5

In list item 19 of the subclause, after “default-initialized subcomponent”, insert “, except by an ALLOCATE statement with a SOURCE= specifier,”

Following list item 19, insert:

- (19a) Successful execution of an ALLOCATE statement with a SOURCE= specifier causes a subobject of the allocated object to become defined if the corresponding subobject of the SOURCE= expression is defined.

In list item 26, replace “becomes” by “might become”.

Subclause 16.5.6

Replace list item 11 of the subclause by the following:

- (11) Successful execution of an ALLOCATE statement with no SOURCE= specifier causes a subcomponent of an allocated object to become undefined if default initialization has not been specified for that subcomponent.

Following list item 11, insert:

- (11a) Successful execution of an ALLOCATE statement with a SOURCE= specifier causes a subobject of the allocated object to become undefined if the corresponding subobject of the SOURCE= expression is undefined.

Following list item 18, insert:

- (19) An object with the VOLATILE attribute (5.1.2.16) might become undefined by means not specified by the program.

Annex A

In the **whole array** paragraph of the subclause, after “named array” insert “or an array component of a structure, with no subscript list”.