



**INTERNATIONAL STANDARD ISO/IEC 1539-1:2004**  
**TECHNICAL CORRIGENDUM 1**

Published 2006-02-15

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 1**

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

*Partie 1: Langage de base*

*RECTIFICATIF TECHNIQUE 1*

Technical Corrigendum 1 to ISO 1539-1:2004 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*.

*In 4.5.3.3, Add the following sentence at the end of constraint C453:  
It shall not have the VALUE attribute.*

*In the fifth paragraph of 4.5.5.2, replace “first executable statement” by “executable constructs”.*

*In 4.5.5.2, add the following paragraph after the fifth paragraph:*

If a specification expression in a scoping unit references a structure constructor, the entity created by the structure constructor is finalized before execution of the executable constructs in the scoping unit.

*In 5.1, add the following sentence at the end of constraint C509:*

It shall not have the VALUE attribute.

*In 5.1.2.5.1, constraint C542, replace “a dummy argument, a function result, or an automatic array of a procedure” by “declared only in a subprogram or interface body”.*

*In 5.1.2.5.1, in the paragraph following constraint C542, after “subprogram” add “or interface body”.*

*In 6.3.1, constraint C625, after “unlimited polymorphic” add “or is of abstract type”.*

*In 6.3.3.1, in the second paragraph following Note 6.24, replace “first executable statement” by “executable constructs”.*

*In 7.1.6, add the following paragraph immediately before Note 7.10:*

If a specification expression in a module includes a reference to a generic, that generic shall have no specific procedures defined in the module subsequent to the specification expression.

*In 7.1.7, add the following paragraph immediately before Note 7.11:*

If an initialization expression in a module includes a reference to a generic, that generic shall have no specific procedures defined in the module subsequent to the initialization expression.

*In the first paragraph of 7.4.1.3, replace “the evaluation of all operations in expr and variable” with “the evaluation of expr and the evaluation of all expressions in variable”.*

*In 7.4.1.3, in list item (2) of the paragraph immediately following Note 7.39, insert “the value of” before each occurrence of “expr”.*

*In the first paragraph of 8.1.4.3, delete “TARGET”.*

*In the first paragraph of 8.1.4.3, after “the attribute.”, insert the following sentence:*

The associating entity has the TARGET attribute if and only if the selector is a variable and has either the TARGET or POINTER attribute.

*In the eighth paragraph of 9.5.3.4.2, replace “input item and its corresponding data edit descriptor” by “effective input item and its corresponding data edit descriptors”.*

*In list item (1) of 9.10.3, replace “input list item (9.5.3.4.2) and corresponding data edit descriptor that requires” by “effective input item (9.5.2) and its corresponding data edit descriptors that require”.*

*In 12.3.2.5, change “referenced” to “invoked”.*

*In 12.4, insert the following constraint after C1224:*

C1224a (R1219) If data-ref is an array, the referenced type-bound procedure shall have the PASS attribute.

*In 12.4.1.1, after “procedure”, insert “, or a procedure pointer component”.*

*In the first paragraph of 12.4.1.2, before “the declared type of the actual argument”, insert “either both the actual and dummy argument shall be unlimited polymorphic, or”.*