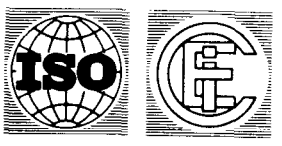
TECHNICAL REPORT

ISO/IEC TR 10034

First edition
1990-04-01

Guidelines for the preparation of conformity clauses in programming language standards

Lignes directrices pour la préparation de chapitres sur la conformité dans des normes concernant des langages de programmation



Reference number
ISO/IEC/TR 10034 : 1990 (E)

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

TABLE OF CONTENTS

		Page
FOREWORD		iii
1	INTRODUCTION	1
2	SCOPE	1
3	DEFINITIONS	2
3.1	CONFIGURATION	2
3.2	PROCESSOR	2
3.3	SUBSET	2
3.4	EXTENSION	2
3.5	IMPLEMENTATION DEFINED	2
3.6	ERROR	2
3.7	DEPRECATED LANGUAGE ELEMENT	2
3.8	CONFORMITY CLAUSE	2
3.9	CONFORMING PROGRAM	2
3.10	CONFORMING PROCESSOR	2
4	THE GUIDELINES	2
4.1	REQUIREMENTS OF A CONFORMING PROCESSOR	3
4.1.1	Documentation	3
4.1.2	Processor Dependencies	4
4.1.3	Errors	5
4.1.4	Extensions to the Language	5
4.1.5	Subsets of the Language	6
4.1.6	Deprecated Language Elements	6
4.2	REQUIREMENTS OF A CONFORMING PROGRAM	6
ANNEX A		7

© ISO/IEC 1990
All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) together form a system for worldwide standardization as a whole. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The main task of a technical committee is to prepare International Standards but in exceptional circumstances, the publication of a Technical Report of one of the following types may be proposed:

— type 1, when the required support within the technical committee cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when a technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

ISO/IEC/TR 10034, which is a Technical Report of type 3, was prepared by ISO/IEC JTC 1, *Information technology.*

This Technical Report is complementary to ISO/IEC/TR 10176, *Guidelines for the preparation of programming language standards*, which includes guidelines for the preparation of conformity clauses. This Technical Report provides more detailed information on the topic.

An important aspect of conformity with a standard is the methods of testing for it, and this is dealt with in ISO/IEC/TR 9547, *Programming language processors — Test methods — Guidelines for their development and acceptability.*

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Guidelines for the preparation of conformity clauses in programming language standards

1 INTRODUCTION

Conformity clauses are included within the language standard to aid the user of the standard in assessing conformity of processors and programs for adherence to the language standard. If conformity requirements are imprecise, testing for compliance can be difficult and potentially impossible for large portions of the language standard. Therefore, these guidelines seek to encourage the inclusion of conformity clauses in programming language standards, and recommend that the language standard precisely identify the criteria that must be met in order that a valid claim may be made that a processor or program conforms to the language standard.

2 SCOPE

Recognizing the dissimilarity of various language standards, the objective of this Technical Report is to provide guidelines for the preparation of conformity clauses for processors and conformity clauses for programs in language standards, together with an annex containing a checklist to aid in this preparation. It was not considered practical to provide model statements that would be suitable for inclusion in all language standards. Therefore, examples have been given to illustrate the type of issues that should be addressed and it is anticipated that these will be adapted, where appropriate, for inclusion in a particular language standard.

It should be borne in mind when reading this document that not all concepts will be applicable to all languages. As examples, language standards do not all specify subsets or permit extensions, and elements that are fully specified by one language standard may be dependent on the processor in another.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3 DEFINITIONS

For the purpose of this Technical Report the following definitions apply:

3.1 **configuration:** Host and target computers, any operating system(s) and software used to operate a language processor.

3.2 **processor:** Compiler, translator or interpreter working in combination with a configuration.

3.3 **subset:** Subset S of programming language L is a programming language such that every program in S
- is also a program in L and
- has the same meaning in S as it has in L.

3.4 **extension:** Facility in the processor that is not specified in the language standard but that does not cause any ambiguity or contradiction when added to the language standard (although, in some languages, it may serve to lift a restriction).

3.5 **implementation defined:** Dependent on the processor, but required by the language standard to be defined and documented by the implementer.

3.6 **error:** Incorrect program construct or incorrect functioning of a program, as defined by the language standard.

Note: For the purposes of this guideline, 'error' includes what may elsewhere be defined as 'error' or 'exception'.

3.7 **deprecated language element:** Element in the language standard which is intended to be deleted from the next revision of the language standard.

3.8 **conformity clause:** Statement that is not part of the language definition but that specifies requirements for compliance with the language standard.

3.9 **conforming program:** Program which is written in the language defined by the language standard and which obeys all the conformity clauses for programs in the language standard.

3.10 **conforming processor:** Processor which processes conforming programs and program units and which obeys all the conformity clauses for processors in the language standard.

4 THE GUIDELINES

The technical terms and meanings used in describing conformity clauses should be the same as those defined for describing the technical specifications in the language standard. When terms are used that are not defined in the language standard the terms

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

and definitions used in ISO 2382 - Data Processing - Vocabulary, should be used.

If the language standard does not fully define a feature of the language, the effect of attempting to use such a language feature may be unpredictable. Therefore, these guidelines recommend that, wherever possible, the standard should identify these areas and require an implementer to document the action to be taken by a processor. (Annex A contains a list of such possible features.)

4.1 REQUIREMENTS OF A CONFORMING PROCESSOR

The language standard should specify the rules for a conforming processor, possibly including one or more of the following examples :

> "A conforming processor shall correctly translate and execute all programs conforming to both the standard and the implementation defined features of the processor."
>
> "A conforming processor shall reject all program units that contain errors whose detection is required by the standard."
>
> "A conforming processor shall not implement any variation from the language standard except where the standard permits. Such permitted variations shall be implemented in the manner prescribed by the language standard and noted in the documentation accompanying the processor."

Note: A variation is an alternative to the handling of a given feature of the language standard.

4.1.1 Documentation

The technical specifications of the language standard may require a conforming processor to document its handling of certain features of the language. The language standard should also require a conforming processor to include the following in its accompanying documentation:

- a list of all definitions or values for the implementation defined features in the language standard;

- a list of all the features of the language standard which are dependent on the processor and not implemented by this processor due to non-support of a particular facility, where such non-support is permitted by the standard;

- a list of all the features of the language implemented by this processor which are extensions to the standard language;

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

- a statement of conformity, giving the complete reference of the language standard with which conformity is claimed, and, if appropriate, the subset of the language supported by this processor.

There should also be a requirement of a conforming processor about the claims made in its documentation, e.g. :

> "A conforming processor shall conform to its accompanying documentation where that documentation relates to the requirements of this language standard."

4.1.2 Processor Dependencies

The language standard should specify the criteria for determining conformity with regard to facilities which depend on the processor. Those cases in the language standard which pertain to specific facilities which depend on the processor should be identified, where known, in the language standard.

Separate conformity criteria should be specified where processor facilities which depend on the processor are available, as well as where the processor does not have a facility for supporting particular features of the language which depend on the processor, e.g. :

> "A conforming processor shall identify in its accompanying documentation the features of the language standard, supported by this processor, which depend on the processor."

> "Language elements that pertain to specific facilities that are dependent on the processor and for which support is not claimed need not be implemented. The absence of such facilities and pertaining language elements from an implementation must be documented."

Where the language standard imposes no limits at all upon an implementation defined value, the writer of portable programs must make certain assumptions about the values that are likely to be supported by all language processors, e.g., a minimum high value or a maximum low value. Similarly, the ability to test for conformity with the language standard may depend upon the tester making such assumptions when defining a test. For the purpose of determining processor conformity, consideration should be given to the specification of reasonable limits for implementation defined elements. These may take the form of recommendations, with a requirement that where they are not met this should be recorded in the documentation accompanying the processor.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

4.1.3 Errors

The language standard should specify how each error, or type of error, is to be treated by a conforming processor. Possible treatments are:

- there shall be a statement in an accompanying document that the error is not reported;

- the processor shall report the error during preparation of the program for execution;

- the processor shall report the error during execution of the program and continue execution;

- the processor shall report the error during the execution of the program and terminate execution.

The language standard should, where appropriate and practical, describe the recovery action to be taken by the processor on detection of an error.

4.1.4 Extensions to the Language

The language standard should address the question of extensions. If they are permitted, the language standard should require that such extensions be clearly described within the documentation accompanying the processor. In order that extensions do not restrict program portability the language standard may make some requirement about their implementation, e.g. :

> "A conforming processor shall offer a facility to report (or 'flag') the use of an extension which is statically determinable solely from inspection of a program statement, without execution."

> "A conforming processor shall offer a facility to reject the use of an extension in a program."

Extensions to the language could imply extra reserved words, The language standard may impose a restriction on the form of these words, e.g. :

> "Any words that are defined as reserved for a particular processor and are in addition to those defined as reserved in this standard shall be outside the range of identifiers permitted in a program, thus ensuring that a conforming program will still be translated by a conforming processor in the manne prescribed by the language standard."

If conformity may be claimed for a subset of the language, the language standard should address the question of extensions to the subset, e.g. :

> "Extensions to the subset shall not conflict with the requirements of the full language standard."

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

4.1.5 Subsets of the Language

Where subsets of the language are permitted within the language standard, the language standard should specify the rules for conformity to a subset, e.g. :

> "A conforming processor shall fully support all the language elements of the subset with which conformity is claimed."

The language standard should specify whether or not a conforming processor is permitted to support language elements of a higher subset, and may require that every processor conforming to a subset of the language provide a facility to report the use of language elements outside that subset, i.e. treat them as extensions.

4.1.6 Deprecated Language Elements

Where appropriate, the language standard should address the use of deprecated language elements, e.g. :

> "A conforming processor shall provide a warning mechanism that may be invoked to indicate the use of deprecated elements in a program."

4.2 REQUIREMENTS OF A CONFORMING PROGRAM

The language standard should specify the rules for a conforming program, e.g. :

> "A conforming program shall use only the constructs specified in the language standard."
>
> "A conforming program shall not depend on extensions implemented by a language processor."
>
> "A conforming program shall not use any extensions implemented by a language processor."
>
> "A program claiming conformity to a subset of a language shall not use any facility which is outside that subset."

The language standard should warn that use of deprecated language elements in a program could cause that program to be non-conforming with future revisions of the language standard.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

ANNEX A (informative)

CHECKLIST OF POTENTIAL LANGUAGE FEATURES THAT MAY DEPEND ON PROCESSOR FACILITIES

The following list describes features that may be dependent on the processor. The list is not to be considered either as exhaustive or as prescriptive.

In some cases a language standard which specifically mentions a particular feature has been shown in parantheses as an example. Some items are generalizations drawn from one or more language standards.

A.1 the size or complexity of a program and its data that will exceed the capability of the processor of a particular configuration (Ada, ISO 8652-1987; FORTRAN, ISO 1539-1980)

A.2 the results of attempting to process or run a program where that program or its data will exceed the capability of the processor of a particular configuration

A.3 the maximum or minimum allowable number of statements in a program

A.4 the maximum or minimum allowable number of operands in an expression, statement or program

A.5 the maximum or minimum length in characters of a line of source text

A.6 the maximum or minimum number of files permitted in a particular program run

A.7 the maximum number of files which can be opened simultaneously

A.8 the representation of external identifiers and titles

A.9 the character set (PL/I, ISO 6160-1979)

A.10 the collating sequence (COBOL, ISO 1989-1985; PL/I, ISO 6160-1979)

A.11 graphic representations (PL/I, ISO 6160-1979)

A.12 extralingual characters (PL/I, ISO 6160-1979)

A.13 the limit on the number of unique statement labels

A.14 the maximum or minimum length of identifiers

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

A.15 the maximum length of character strings

A.16 the default values for uninitialized variables

A.17 the maximum or minimum value of integers

A.18 the internal precision of real numbers

A.19 the maximum or minimum value of real numbers

A.20 the method of rounding or truncating numeric results

A.21 the results of numeric conversions (PL/I, ISO 6160-1979)

A.22 the representation of fixed or floating point numbers where it is necessary to explain the results of operations upon them (Ada, ISO 8652-1987)

A.23 the maximum or minimum length of a record (PL/I, ISO 6160-1979)

A.24 the maximum or minimum length of keys in keyed files (PL/I, ISO 6160-1979)

A.25 the maximum allowable number of dimensions in an array and the maximum allowable number of array elements

A.26 the maximum depth of nesting

A.27 the maximum levels of recursion

A.28 the order of evaluation of variables or expressions or operands or parameters where an implementer may vary the order or impose an order

A.29 the way in which non-valid programs are rejected and how this is reported

A.30 the reporting of errors or exceptions where more than one error condition could arise (COBOL, ISO 1989-1985)

A.31 the system action following certain error conditions (COBOL, ISO 1989-1985).

UDC 681.3.06 : 800.92

Descriptors : data processing, programming (computers), computer programs, programming languages.

Price based on 4 pages

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale