TECHNICAL REPORT

ISO/IEC TR 10182

First edition
1993-12-15

Information technology — Programming languages, their environments and system software interfaces — Guidelines for language bindings

Technologies de l'information — Langages de programmation, leurs environnements et interfaces logicielles des systèmes — Techniques d'interface pour les normes de langages de programmation

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. 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 technical committees is to prepare International Standards, but in exceptional circumstances a technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support 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 Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.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 10182, which is a Technical Report of type 3, was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Sub-Committee SC 22, *Programming languages, their environments and system software interfaces*.

Information technology — Programming languages, their environments and system software interfaces — Guidelines for language bindings

1 INTRODUCTION

1.1 Status of the Document

This document is a compilation of the experience and knowledge gained by the members of ISO/IEC JTC1/SC22/WG11 (Techniques for Bindings) from the generation of programmers' interfaces to FUNCTIONAL INTERFACE STANDARDS. Although current experience was derived from the fields of computer graphics and database management, the problems discussed are thought to be generally applicable for mappings of other functional interface standards to programming languages. This document is intended

a) to identify the problems and conflicts which must be resolved;
b) to suggest guidelines for future use;
c) to provide scope and direction to required additional work, such as common procedural calling mechanisms and data types; and
d) as a historical record of past experiences and decisions.

This document is incomplete; the authors have concentrated on those areas where experience and expertise was readily available. The ideas and issues brought forward here emerged from more than ten years of work, and are represented in international Standards.

Section 2 of this document contains the results of a survey of current methods used for language binding development. Characteristics of each method are given, followed by reasons for the selection of the method.

Application of the methods has suggested some guidelines that are presented in Section 3. Sections 2 and 3 contain documentation of the current state of language binding efforts; Section 4 addresses future directions for language bindings.

Circulation of this document is necessary at this stage, as input and discussion from representatives of ISO/IEC JTC1/SC21 (functional specification standards developers), ISO/IEC JTC1/SC24 (computer graphics standards developers), and ISO/IEC JTC1/SC22 (language standards developers) is urgently sought. The document in its current form may be useful for those about to embark on language binding developments.

1.2 Scope

This document is based on experience gained in the standardization of two major areas in information processing. One area covers programming languages. The other area is composed of the services necessary to an application program to achieve its goal. The services are divided into coherent groups, each referred to as a SYSTEM FACILITY, that are accessed through a FUNCTIONAL INTERFACE. The specification of a system facility, referred to as a FUNCTIONAL SPECIFICATION, defines a collection of SYSTEM FUNCTIONS, each of which carries out some well-defined service.

Since in principle there is no reason why a particular system facility should not be used by a program, regardless of the language in which it is written, it is the practice of system facility specifiers to define an 'abstract' functional interface that is language independent. In this way, the concepts in a particular system facility may be refined by experts in that area without regard for language peculiarities. An internally coherent view of a particular system facility is defined, relating the system functions to each other in a consistent way and relating the system functions to other layers within the system facility, including protocols for communication with other objects in the total system.

However, if these two areas are standardized independently, it is not possible to guarantee that programs from one operating environment can be moved to another, even if the programs are written in a standard programming language and use only standard system facilities. A language binding of a system facility to a programming language provides language syntax that maps the system facility's functional interface. This allows a program written in the language to access the system functions constituting the system facility in a standard way. The purpose of a language binding is to achieve portability of a program that uses particular facilities in a particular language. Examples of system facilities that have had language bindings developed for them are GKS, NDL, and SQL (see Section 1.3, References). It is anticipated that further language binding development will be required. Some system facilities currently being standardized have no language bindings and additional system facilities will be standardized. There is a possibility of n x m language bindings, where n is the number of languages and m the number of system facilities.

The scope of this document is to classify language binding methods, reporting on particular instances in detail, and to produce suggested guidelines for future language binding standards.

Note that the language bindings and the abstract facility interfaces must have a compatible run time representation, but the abstract facility does not necessarily have to be written in the host language. For example, if the application program is using a Pascal language binding and the corresponding facility is written in FORTRAN, there must be a compatible run time representation in that operating environment. How this compatibility is achieved is outside the scope of these guidelines. This is generally a property of the operating environment defined by the implementor, and is reviewed briefly in this document.

1.3 References

ISO/IEC 1539:1991, *Information Technology - Programming languages - FORTRAN.*

ISO/IEC 1989:1985, *Programming Languages - COBOL.*

ISO/IEC 1989:1985/Amd.1:1992, *Programming Languages - COBOL - Amendment 1.*

ISO 6373:1984, *Data processing - Programming languages - Minimal BASIC.*

ISO 7185:1990, *Information technology - Programming languages - Pascal.*

ISO 7942:1985, *.Information processing systems - Computer graphics - Graphical Kernel System (GKS) functional description.*

ISO 8651-1:1988, *Information processing systems - Computer graphics - Graphical Kernel System (GKS) language bindings - Part 1: FORTRAN.*

ISO 8651-2:1988, *Information processing systems - Computer graphics - Graphical Kernel System (GKS) language bindings - Part 2: Pascal.*

ISO 8651-3:1988, *Information processing systems - Computer graphics - Graphical Kernel System (GKS) language bindings - Part 3: Ada.*

ISO 8651-4:1988, *Information technology - Computer graphics - Graphical Kernel System (GKS) language bindings - Part 4: C.*

ISO 8652:1987, *Programming Languages - Ada.*

ISO 8805:1988, *Information processing systems - Computer graphics - Graphical Kernel System for Three Dimensions (GKS-3D) functional description.*

ISO/IEC 8806-1:[1], *Information technology - Computer graphics - Graphical Kernel System for Three Dimensions (GKS-3D) language bindings - Part 1: FORTRAN.*

ISO 8907:1987, *Information processing systems - Database Languages - NDL.*

ISO/IEC 9075:1992, *Information technology - Database Languages - SQL.*

ISO/IEC 9592:1984, *Information processing systems - Computer Graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS).*

ISO/IEC 9593-1:1990, *Information processing systems - Computer graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) language bindings - Part 1: FORTRAN.*

ISO/IEC 9593-2:[1], *Information technology - Computer Graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) language bindings - Part 2: Extended Pascal.*

ISO/IEC 9593-3:1990, *Information technology - Computer Graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) language bindings - Part 3: Ada.*

ISO/IEC 9593-4:1991, *Information technology - Computer Graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) language bindings - Part 4: C.*

ISO/IEC 9899:1990, *Programming languages - C'.*

ISO/IEC 10206:1991, *Information Technology - Programming languages - Extended Pascal.*

C. Osland, *Case Study of GKS Development*, Eurographics Tutorials 1983, Springer.

J. R. Gallop, C. Osland, *Experience with Implementing GKS on a Perq and Other Computers*, Computer Graphics Forum 9(7), North-Holland 1985.

F. R. A. Hopgood, D. Duce, J. R. Gallop, D. Sutcliffe, *Introduction to the Graphical Kernel System (GKS)*, Academic Press, 1983.

M. Sparks, J. R. Gallop, *Language Bindings for Computer Graphics Standards*, IEEE Computer Graphics and Applications, Vol 6, Number 8, August 1986.

M. Sparks, *Graphics Language Bindings*, Computer Graphics Forum, Journal of the European Association for Computer Graphics, vol. 4(4) 1985.

M. Sparks, *Graphics Standards Bindings - What Are They and When Can We Use Them?*, Computer Graphics/November 1985.

---

[1] To be published.

3

Language Binding Generic Issues (document within ISO/IEC JTC1 SC24/WG4 and ISO/IEC JTC1 SC22/WG11)