INTERNATIONAL STANDARD

ISO/IEC 14769

First edition
2001-05-15

Information technology — Open Distributed Processing — Type Repository Function

Technologies de l'information — Traitement réparti ouvert — Fonction de répertoire de types

Reference number
ISO/IEC 14769:2001(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 14769 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software engineering*, in collaboration with ITU-T. The identical text is published as ITU-T Recommendation X.960.

Annex A forms a normative part of this International Standard. Annexes B and C are for information only.

Introduction

This Recommendation | International Standard prescribes the ODP Type Repository Function (subclause 14.4 of ITU-T Rec. X.903 | ISO/IEC 10746-3) to support the storage, retrieval and management of type descriptions within an identified framework for type descriptions.

ITU-T Rec. X.902 | ISO/IEC 10746-2 provides a general definition of type in subclause 9.7; this definition allows the description of types using any predicate. ITU-T Rec. X.903 | ISO/IEC 10746-3 introduces a number of target concepts specific to particular viewpoints. This Recommendation | International Standard supports the establishment of type definitions based on the concepts defined in the ODP family of Recommendations | International Standards.

This Recommendation | International Standard enables type descriptions for use by the ODP functions outlined in ISO/IEC 10746-3. Type descriptions can occur in specifications from any viewpoint, e.g. enterprise specification can introduce enterprise types. This Recommendation | International Standard specifically addresses the needs of the ODP computational and engineering viewpoint types, but is capable of supporting type descriptions coming from other viewpoint languages.

This Recommendation | International Standard permits the use of multiple type description languages. There are a number of widely used and standardized languages for type description, for example CORBA IDL, ASN.1, LOTOS, GDMO and SDL, which fulfil some of the requirements of type descriptions in ODP-RM. This Recommendation | International Standard does not define a single all-encompassing type language. Users can use either existing languages or languages defined within other ODP Recommendations | International Standards. Annex B is an informative annex outlining languages that support large sets of target concepts.

This Recommendation | International Standard supports type systems with a type Type (e.g. pass type as parameters as in the ODP computational language).

ITU-T Rec. X.903 | ISO/IEC 10746-3 defines a subtype relationship between computational operational interface signature types. This Recommendation | International Standards supports a wider variety of relationships between types, which might include the analysis of behaviour and environment contracts, but the definition of such relationships is not within the scope of this Recommendation | International Standard. Relationships between types can either be asserted or deduced. It is recognized that not all relationships (including equivalence) can always be automatically deduced. However, automatic deduction should be encouraged whenever applicable.

The type repository function supports the allocation of identifiers to types in order to allow the transmission of these "shorthand" representations across domains (i.e. between objects using different type repositories).

The type repository function addresses interworking and federation to support the distribution of the type repository function by clarifying the notion of type domains. This function supports both federation of type domains handling equivalent type systems and federation of type domains handling different type systems.

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN DISTRIBUTED PROCESSING – TYPE REPOSITORY FUNCTION

1 Scope

The concept of "type" is fundamental to ODP systems; the interaction model of ODP-RM involves strongly-typed interactions.

This Recommendation | International Standard:

– defines a framework for describing types of interest in ODP systems by determining what entities need to be typed and what needs to be said about the identified types. The primary focus of this work is the computational interface type system;

– identifies and characterizes type languages sufficient to describe the types identified above in an informative annex;

– provides enterprise, information, and computational specifications of a generic type repository function within the type description framework which can be specialized to select a specific type system or type notation. The type repository function provides:

- storage and retrieval of type descriptions;
- management of type descriptions;

- management of the relationship between types including matching of types;
- naming of types (in a manner consistent with ODP Naming Framework);
- interworking and federation of different type repositories.

This Recommendation | International Standard provides a standard method of accessing type descriptions used within open distributed processing systems, where the type descriptions can be in various concrete syntaxes and type languages used in these open distributed processing systems. This Recommendation | International Standard also facilitates the dynamic matching of types for interactions, binding and trading purposes.

2 Normative References

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and International Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of ITU maintains a list of the currently valid ITU-T Recommendations.

2.1 Identical Recommendations | International Standards

– ITU-T Recommendations X.680-series (1997) | ISO/IEC 8824 (all parts):1998, *Information technology – Abstract Syntax Notation One (ASN.1)*.

– ITU-T Recommendation X.725 (1995) | ISO/IEC 10165-7:1996, *Information technology – Open Systems Interconnection – Structure of management information: General relationship model*.

- ITU-T Recommendation X.902 (1995) | ISO/IEC 10746-2:1996, *Information technology – Open Distributed Processing – Reference Model: Foundations*.
- ITU-T Recommendation X.903 (1995) | ISO/IEC 10746-3:1996, *Information technology – Open Distributed Processing – Reference Model: Architecture*.
- ITU-T Recommendation X.910 (1998) | ISO/IEC 14771:1999, *Information technology – Open Distributed Processing – Naming framework*.
- ITU-T Recommendation X.920 (1997) | ISO/IEC 14750:1999, *Information technology – Open Distributed Processing – Interface definition language*.
- ITU-T Recommendation X.930 (1998) | ISO/IEC 14753:1999, *Information technology – Open Distributed Processing – Interface references and binding*.
- ITU-T Recommendation X.950 (1997) | ISO/IEC 13235-1:1998, *Information technology – Open Distributed Processing – Trading function: Specification*.

2.2 Additional References

- ISO/IEC 10027:1990, *Information technology – Information Resource Dictionary System (IRDS) framework.*
- ISO/IEC 13719 (all parts):1998, *Information technology – Portable Common Tool Environment (PCTE).*
- ISO/IEC 15474 (all parts):...[1], *Information technology – CDIF Framework.*

2.3 Specifications of the Object Management Group

This Recommendation | International Standard makes references to the following specifications:

- Object Management Group, ad/97-08-14 and ad/97-08-15, *Meta-Object Facility*, 1997.
- Object Management Group, ad/97-08-02 through ad/98-08-09, *Unified Modelling Language*, 1997.

Annex C identifies the clauses of this Recommendation | International Standard that reference text in the Meta-Object Facility.

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply, except for where the text is described as being common with the Meta-Object Facility.

3.1 Terms defined in other International Standards

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.902 | ISO/IEC 10746-2 Open Distributed Processing – Reference Model: Foundations):

- action;
- behaviour;
- binding;
- client object;
- consumer object;
- creation (of an <X>);
- data type;
- deletion (of an <X>);
- domain;
- environment contract type;
- identifier;
- information;
- instance (of a type);

[1] To be published.

- instantiation (of an <X>);
- interface;
- interface signature;
- name;
- obligation;
- ODP standard;
- ODP system;
- object;
- policy;
- role;
- subtype;
- supertype;
- state (of an object);
- <X> template;
- trading;
- type;
- viewpoint.

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.903 | ISO/IEC 10746-3 (Open Distributed Processing – Reference Model: Architecture):

- announcement signature;
- binding object;
- community;
- compound binding;
- computational interface;
- computational interface signature;
- computational interface type;
- computational object signature;
- computational specification;
- computational viewpoint;

  NOTE – Throughout this Recommendation | International Standard, the qualifier "computational" is frequently omitted.
- dynamic schema;
- engineering interface reference;
- enterprise specification;
- <X> federation;
- flow signature type;
- flow type;
- information specification;
- interrogation signature;
- invariant schema;
- invocation;
- operation;
- operation signature type;
- operational interface signature;
- primitive binding;
- primitive signal binding type;

- primitive stream binding type;
- service offer;
- signal interface signature type;
- signal signature type;
- static schema;
- stream interface signature type;
- termination signature type;
- trading function;
- type repository function.

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.950 | ISO/IEC 13235-1 (Open Distributed Processing – Trading Function: Specification):

- service type;
- property type.

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.930 | ISO/IEC 14753 (Open Distributed Processing – Interface References and Binding):

- additional information;
- behaviour;
- causality information;
- channel class;
- direct reference type;
- flow description type;
- flow type;
- group information;
- interface description type;
- interface reference;
- interface type;
- location information;
- non-interpreted reference type;
- null reference type;
- opaque information;
- operation description type;
- operation type;
- operational interface type;
- quality-of-service statement type;
- relocation information;
- security information;
- stream interface type.

3.2 Terms defined in this Recommendation | International Standard

This Recommendation | International Standard defines the following terms:

3.2.1 **relationship:** A predicate involving two or more roles with assigned values.

3.2.2 **relationship type:** A type of relationship which expresses the number and type of the roles.

3.2.3 **relation:** A set of relationships of the same relationship type.

3.3 Terms defined in the OMG Meta-Object Facility

In text described as common with the Meta-Object Facility, this Recommendation | International Standard uses the following definition.

3.3.1 Meta-Object Facility: a specification of the Object Management Group for repositories of type information for arbitrary type systems.

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply.

IDL	Interface Definition Language
MOF	Meta-Object Facility
ODP	Open Distributed Processing
ODP-RM	Open Distributed Processing: Reference Model
OMG	Object Management Group
TR	ODP Type Repository

The following additional abbreviations occur in sections of the OMG Meta-Object Facility specification which are incorporated by reference in this Recommendation | International Standard:

CORBA	Common Object Request Broker Architecture
MODL	Meta-Object Definition Language
OCL	Object Constraint Language
UML	Unified Modelling Language

5 Overview and Motivation

Open distributed processing in multi-organizational environments requires that various kinds of meta-level information are available at run-time in each interoperating system.

Especially, information is needed about types and type systems, for determining:

– conformity of information presentation during compilation;

– similarity of offered and requested services in trading;

– conformity and substitutability of servers during service invocation;

– required configuration of objects for object binding;

Cooperation between autonomous systems requires knowledge about the relationships between types or type systems.

5.1 Type Repository

The type repository stores type definitions, type relations, and information about the type system itself.

A type system is structured by a set of target concepts. It should be noted that the set of target concepts may grow during the lifetime of the type system. For example, the ODP-RM computational type system includes the following target concepts: object, type, template, service, interface, operation, stream, flow, signal. Based on these target concepts, a banking application could define types applicable for banking, e.g. a BankAccount interface with operations deposit, withdraw, and balance.

No single type system or type language can be assumed. There are already a multiplicity of type systems in use (including many standardized ones) and often many type languages for each type system. For example, protocol data units can be described in ASN.1, data types in ACT-ONE, relational schemata in SQL, file formats in COBOL, interfaces in ODP IDL, pipes in ISO RPC IDL. A canonical type system or canonical type language would have to be a superset of all existing type systems and type languages. Furthermore, the set of target concepts is open-ended and so every new target concept would require the extension of the canonical type language. This is technically and politically infeasible.

Although no single type system or type language can be assumed, it is nonetheless possible to develop specific languages for interchange between type systems, e.g. the CDIF family of standards.

The set of relationships between types and type systems cannot be predetermined. The type repository depends on both external assertions of relationships in addition to its own ability to derive relationships through semantic analysis.

Some type descriptions can be used as templates. A template has sufficient detail to enable instantiation on a selected platform. On another platform, the same description may not be sufficient as a template.

NOTE – The architecture described conforms to the IRDS Framework.

5.2 Meta-Object Facility

This Recommendation | International Standard is technically aligned with the OMG Meta-Object Facility, a specification of a type repository system for models (types in ODP) and meta-models (type systems in ODP). The definition of a meta-model (type system) includes the definition of classes (target concepts in ODP) and associations (type relations in ODP).

The Meta-Object Facility can support multiple meta-models (type systems) and multiple models (types) within each meta-model (type system). The Meta-Object Facility unifies the handling of models and meta-models by developing a meta-meta-model (type system for describing type systems) for defining meta-models. Thus, all handling of information is performed relative to a nominated set of meta-information. A newly created Meta-Object Facility contains only the meta-meta-model, enabling the definition of meta-models (type systems), which in turn enable the definition of models (types).

The Meta-Object Facility is type-language neutral. It stores models, meta-models and its own meta-meta-model as graphs of linked CORBA objects; the mechanisms which translate to/from these graphs into particular syntaxes are outside the scope of the Meta-Object Facility specification.

The Meta-Object Facility was developed to support generic modelling needs which occur in such areas as information management, software development, and data warehousing. The overview of the Meta-Object Facility is given in Section 2: "Facility Purpose and Use" of the OMG Meta-Object Facility specification.

6 Enterprise Specification

The scope of an enterprise specification is defined in ITU-T Rec. X.903 | ISO/IEC 10746-3 and refined by ODP Enterprise Language. The enterprise specification identifies the objectives and the policy statements that govern the activities of the type repository function.

6.1 Objective

The objective of the type repository function is to manage a repository of type system descriptions, type descriptions and type relationships so that queries can be made on any stored type system description, type description or type relationship, whenever needed for the development, operation, and management of ODP systems.

6.2 Type Repository Community

A type repository community consists of objects that take on one or more roles within the community. The behaviour of each role and the behaviour of the community as a whole are governed by a set of repository policy rules. Members of the community are obliged to obey these policy rules.

6.2.1 Roles

Objects may take on the following roles within a type repository community as shown in Table 1 and as illustrated in Figure 1. A type repository is governed by a single TR type system description, but is intended to handle multiple type systems, each containing a set of types as representations for the target concepts of that type system (see examples in Annex A). This Recommendation | International Standard defines the TR type system.

NOTE 1 – Information about types and type systems includes the relationships between types and the relationships between type systems.

NOTE 2 – Examples of a type system include the Pascal programming language, SQL schemas for defining relational tables, and trader service types (see A.2.) Examples of a type include a Pascal function declaration, the definition of a payroll database in SQL, or a printer service type for the trader.

6.2.1.1 Cardinality of Roles

The purpose of this subclause is to define how many times each role can occur in the enterprise specification. It does not constrain the number of objects that can fill each role; that is the subject of 6.2.1.2.