

INTERNATIONAL STANDARD

ISO/IEC
10031-1

First edition
1991-08-01

Information technology — Text and office systems — Distributed-office-applications model —

Part 1: General model

*Technologies de l'information — Bureautique — Modèle d'application
pour bureau distribué —*

Partie 1: Modèle général



Reference number
ISO/IEC 10031-1:1991 (E)
AVA Documents
Document Sharing Hub

ISO/IEC 10031-1:1991(E)

Contents	Page
Foreword	iii
Introduction	iv
1 Scope	1
2 Normative references	1
3 Definitions	2
4 Abbreviations	6
5 Model	7
6 Guidelines for the design of protocols	15
 Annexes	
A References, definitions, and abbreviations for informative annexes	21
B Relationship to other standards	24
C Requirements	26
D Basic concepts	28
E Identification considerations	45
F Security concepts	48
G Management	57
H Categories and relationship of applications	58
J Object model	67
K Standard set of operations	70

© ISO/IEC 1991

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

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. Draft International Standards adopted by the joint technical committee are circulated to national bodies for approval before their acceptance as International Standards. They are approved in accordance with procedures requiring at least 75 % approval by the national bodies voting.

International Standard ISO/IEC 10031-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

ISO/IEC 10031 consists of the following parts, under the general title: *Information technology – Text and office systems – Distributed-office-applications model –*

– *Part 1: General model*

– *Part 2: Distinguished-object-reference and associated procedures.*

Annexes A to K of this part of ISO/IEC 10031 are for information only.

Introduction

Standards for Open Systems Interconnection permit the functional components of an application to be distributed over a network. Some applications may be distributed in order to reduce costs, e.g. in the case of an office system connected by a high speed Local Area Network (LAN), where some expensive resources may be shared. Other applications may be distributed for administrative or functional reasons, e.g. in the case of a world wide electronic mail system. These are examples of distributed applications whose design differs from the traditional "single host" approach. In general, the increasing range of technological options permits a number of design approaches with quite a different distribution of costs and computational load on the various system elements, such as the desk top devices that populate a modern office and the "back room" devices with which these users are networked.

The purpose of ISO/IEC 10031 is to establish a general framework for distributed-office-applications based on the Remote Operations Service Element.

ISO/IEC 10031 is one of a series, relating to Open Systems Interconnection. Open Systems Interconnection standards are intended to facilitate homogeneous interconnection between heterogeneous information processing systems. ISO/IEC 10031 is within the framework for the coordination of standards for Open Systems Interconnection defined by ISO 7498.

A particular emphasis of ISO/IEC 10031 is to specify the homogeneous externally visible and verifiable characteristics needed for interconnection compatibility, while avoiding unnecessary constraints upon and changes to the heterogeneous internal design and implementation of the information processing systems to be interconnected. It does this by specifying a general model and a set of design principles that will ensure that different distributed-office-applications will function together in a coherent manner.

ISO/IEC 10031 defines the necessary common framework that will enable distributed-office-applications to be developed. Additionally, it provides the concepts and principles applicable to distributed-office-applications that will enable:

- a) the modular, simple and expandable development of related products;
- b) their implementation from the services of different vendors or service providers;
- c) their interworking;
- d) the optimization of development costs.

In the interest of effective standardization, ISO/IEC 10031 is oriented towards well understood needs. It is capable of modular extensions to cover future developments in technology and needs.

Although mainly intended for distributed-office-applications, the contents of ISO/IEC 10031 may be used in other information processing environments.

Information technology – Text and office systems – Distributed-office-applications model – Part 1: General model

1 Scope

ISO/IEC 10031 provides a framework for the purpose of the development of protocol standards for distributed-office-applications (DOAs). It applies to applications distributed over significant physical distances as well as “closely-coupled” office systems.

ISO/IEC 10031 describes a model. Distributed-office-applications to be standardized shall use the principles specified in it.

ISO/IEC 10031 provides guidelines for the design of protocols which allow access to the various applications and interactions between applications. The protocols for distributed applications are within the OSI Application Layer and conform to the Remote Operations defined in ISO/IEC 9072.

ISO/IEC 10031 embraces the intent that elements of a system conforming to some parts of it can be implemented from devices supplied by different vendors and by different service providers.

ISO/IEC 10031 does not define human-machine interfaces used with the distributed-applications. Also, it does not define the interface between the software that interacts directly with the user and the software of specific applications.

The content of ISO/IEC 10031 is comprised of two parts.

This part of ISO/IEC 10031 describes the general model of distributed-office-applications and is divided into two sections:

- a) model ;
- b) guidelines for the design of DOA protocols.

ISO/IEC 10031-2 describes the Distinguished-object-reference and associated procedures that may be used by all DOAs.

No requirement is made for conformance to this part of ISO/IEC 10031. Other parts of ISO/IEC 10031 may specify conformance for systems implementing procedures of those other parts.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 10031. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 10031 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 7498: 1984,

*Information processing systems - Open Systems Interconnection -
Basic Reference Model.*

ISO / IEC 10031-1:1991(E)

ISO 7498-2: 1989,	<i>Information processing systems - Open Systems Interconnection - Basic Reference Model - Part 2 : Security architecture.</i>
ISO 7498-3: 1989,	<i>Information processing systems - Open Systems Interconnection - Basic Reference Model - Part 3 : Naming and addressing.</i>
ISO 8649: 1988,	<i>Information processing systems - Open Systems Interconnection - Service definition for Association Control Service Element.</i>
ISO 8822:1988,	<i>Information processing systems - Open Systems Interconnection - Connection oriented presentation service definition.</i>
ISO/IEC 8824: 1990,	<i>Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1).</i>
ISO/IEC 9066-1: 1989,	<i>Information processing systems - Text communication - Reliable Transfer - Part 1: Model and service definition.</i>
ISO/IEC 9066-2: 1989,	<i>Information processing systems - Text communication - Reliable Transfer - Part 2: Protocol specification.</i>
ISO/IEC 9072-1: 1989,	<i>Information processing systems - Text communication - Remote Operations - Part 1: Model, notation and service definition.</i>
ISO/IEC 9072-2: 1989,	<i>Information processing systems - Text communication - Remote Operations - Part 2: Protocol specification.</i>
ISO/IEC 9594-2: 1990,	<i>Information processing systems - Open Systems Interconnection - The Directory - Part 2: Models.</i>
ISO/IEC 9594-3: 1990,	<i>Information processing systems - Open Systems Interconnection - The Directory - Part 3: Abstract service definition.</i>
ISO 9735: 1988,	<i>Electronic data interchange for administration, commerce and transport (EDIFACT) - Application level syntax rules.</i>
ISO/IEC 10021-2: 1990,	<i>Information processing systems - Text communication - Message-Oriented Text Interchange Systems (MOTIS) - Part 2: Overall architecture.</i>
ISO/IEC 10021-3: 1990,	<i>Information processing systems - Text communication - Message-Oriented Text Interchange Systems (MOTIS) - Part 3: Abstract Service Definition conventions.</i>
ISO/IEC 10021-5: 1990,	<i>Information processing systems - Text communication - Message-Oriented Text Interchange Systems (MOTIS) - Part 5: Message Store: Abstract service definition.</i>

3 Definitions

3.1 OSI basic reference model definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO 7498:

- a) Application Layer;
- b) application-entity;
- c) application-service-element;
- d) Presentation Layer;
- e) presentation-connection;
- f) protocol;

- g) service definition.

3.2 OSI basic reference model Security part definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO 7498-2:

- a) authentication;
- b) authorization;
- c) credentials;
- d) security policy.

3.3 Association control service element (ACSE) definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO 8649:

- a) application context;
- b) Association Control Service Element.

3.4 Presentation service definition

This part of ISO/IEC 10031 makes use of the following term defined in ISO 8822:

- a) abstract syntax.

3.5 Abstract syntax notation definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO 8824:

- a) ASN.1;
- b) external type;
- c) Generalized Time;
- d) macro;
- e) object identifier;
- f) UTC Time.

3.6 Reliable transfer service element (RTSE) definition

This part of ISO/IEC 10031 makes use of the following term defined in ISO/IEC 9066:

- a) Reliable Transfer Service Element.

3.7 Remote operations service element (ROSE) definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO/IEC 9072:

- a) argument;
- b) bind-operation;
- c) invoke;
- d) operation;
- e) perform;
- f) Remote Operations;
- g) Remote Operations Service Element;
- h) result;
- i) unbind-operation.

ISO/IEC 10031-1:1991(E)

3.8 Directory definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO/IEC 9594:

- a) attribute;
- b) attribute macro;
- c) attribute type;
- d) attribute value;
- e) filter.

3.9 EDIFACT definition

This part of ISO/IEC 10031 makes use of the following term defined in ISO 9735:

- a) EDIFACT.

3.10 Message oriented text interchange systems (MOTIS) definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO/IEC 10021-2:

- a) body part;
- b) IP-message;
- c) message.

3.11 Abstract service definition conventions definitions

This part of ISO/IEC 10031 makes use of the following terms defined in ISO/IEC 10021-3:

- a) abstract model;
- b) abstract operations;
- c) abstract service;
- d) abstract service macro;
- e) asymmetric;
- f) port;
- g) refinement;
- h) symmetric.

3.12 Distributed-office-applications model (DOAM) definitions

For the purposes of this part of ISO/IEC 10031, the following definitions apply:

3.12.1 accessee: An x-server which can assign distinguished-object-references (DORs) to objects which it manages upon requests from x-clients, and which can perform operations that designate objects by DORs which it assigned.

3.12.2 accessor: An x-server which can perform operations that designate objects by DORs, by accessing Accessees with the DORs.

3.12.3 control-attributes: Attributes, associated with a security-object that, when matched against the privilege-attributes of a security-subject, are used to grant or deny access to the security-object.

3.12.4 control-attribute-package: A collection of control-attributes.

- 3.12.5 consume-operation:** An operation invoked by an x-client to an Accessor that designates objects by DORs.
- 3.12.6 data-object:** An object that represents data.
- 3.12.7 data-object-value:** A value derived from a data object in accordance with a set of rules, or in the absence of such rules, the value of the entire object.
- 3.12.8 direct-value-access:** Data-object access by value, instead of a reference.
- 3.12.9 direct-value-transfer:** Direct transfer of data-object-value, instead of transfer of a reference.
- 3.12.10 distinguished-object-reference:** A unique reference to a real object in a DOA environment.
- 3.12.11 distributed-office-application:** A set of information processing resources distributed over one or more open systems which provides a well defined set of functionality to (human) users, to assist a given office task.
- 3.12.12 document:** A structured amount of information intended for either direct or indirect human perception that can be interchanged, stored, retrieved and processed by means of office applications.
- 3.12.13 initiator:** An x-client which invokes operations requesting DORs instead of data object value to an Accessee, and which invokes operations that designate objects by DORs to an Accessor.
- 3.12.14 office-data-object:** An object that represents office-information.
- 3.12.15 office-information:** Data used in the office environment.
- 3.12.16 privilege-attributes:** Attributes, associated with a security-subject that, when matched against control-attributes of a security-object, are used to grant or deny access to that security-object.
- 3.12.17 privilege-attribute-certificate:** A certificate using privilege attributes.
- 3.12.18 produce-operation:** A operation invoked by an x-client to an Accessee that requests an DOR instead of data object value.
- 3.12.19 qualified-attributes:** Attributes that have qualification for their usage.
- 3.12.20 referenced-object-access:** Access to objects by means of reference.
- 3.12.21 ROA-operation:** An operation invoked by an Accessor to an Accessee.
- 3.12.22 security-attributes:** A general term covering both privilege-attributes and control-attributes. The use of security-attributes is defined by a security policy.
- 3.12.23 security-object:** An entity in a passive role to which access is granted or denied according to an authorization-policy.
- 3.12.24 security-subject:** An entity in an active role that is granted or denied access to security-objects according to an authorization-policy.
- 3.12.25 user-application process:** An application-process that contains an OA-user and one or more clients of distributed(-office)-applications (e.g. x-client, y-client, etc.).
- 3.12.26 x-:** Generic placeholders for specific application names.
- 3.12.27 x-access:** The definition of the functionality of an x-application, as visible between a x-client and a x-server.
- 3.12.28 x-access-abstract-service:** The abstract service between an x-client and an x-server.

ISO / IEC 10031-1:1991(E)

- 3.12.29 x-access-protocol:** The protocol used between an x-client and an x-server.
- 3.12.30 x-application:** A distributed(-office)-application of a certain kind, e.g. an Electronic Mail Application or a Filing and Retrieval Application.
- 3.12.31 x-application-system:** The collection of x-clients and x-server-system that together provide x-user the functionality of x-application.
- 3.12.32 x-client:** That part of an x-application which is part of an application-process which contains an x-user.
- 3.12.33 x-server:** That part of an x-application which is part of a x-server-application-process and that provides the functionality specified by an x-access-abstract-service-definition.
- 3.12.34 x-server-system:** The collection of one or several x-servers.
- 3.12.35 x-system-abstract-service:** The abstract service between x-servers.
- 3.12.36 x-system protocol:** Protocol used between x-servers.
- 3.12.37 x-user:** Part of an application-process in the role assumed when using an x-application.

4 Abbreviations

ACSE	Association Control Service Element
ASN.1	Abstract Syntax Notation One
CAP	Control Attribute Package
DOA	Distributed Office Applications
DOAM	Distributed Office Applications Model
DOR	Distinguished Object Reference
EDIFACT	Electronic Data Interchange For Administration, Commerce, and Transport
OSI	Open Systems Interconnection
PAC	Privilege Attribute Certificate
QoS	Quality of Service
ROA	Referenced Object Access
ROSE	Remote Operations Service Element
RTSE	Reliable Transfer Service Element
UTC	Coordinated Universal Time

5 Model

NOTE – For tutorial background information about the concept used in this clause, see annex D.

5.1 DOA Abstract Model

5.1.1 Abstract Model for Access

Distributed-office-applications shall be developed aligning to the client-server abstract model shown in figure 1, using abstract service definition conventions defined by ISO/IEC 10021-3.

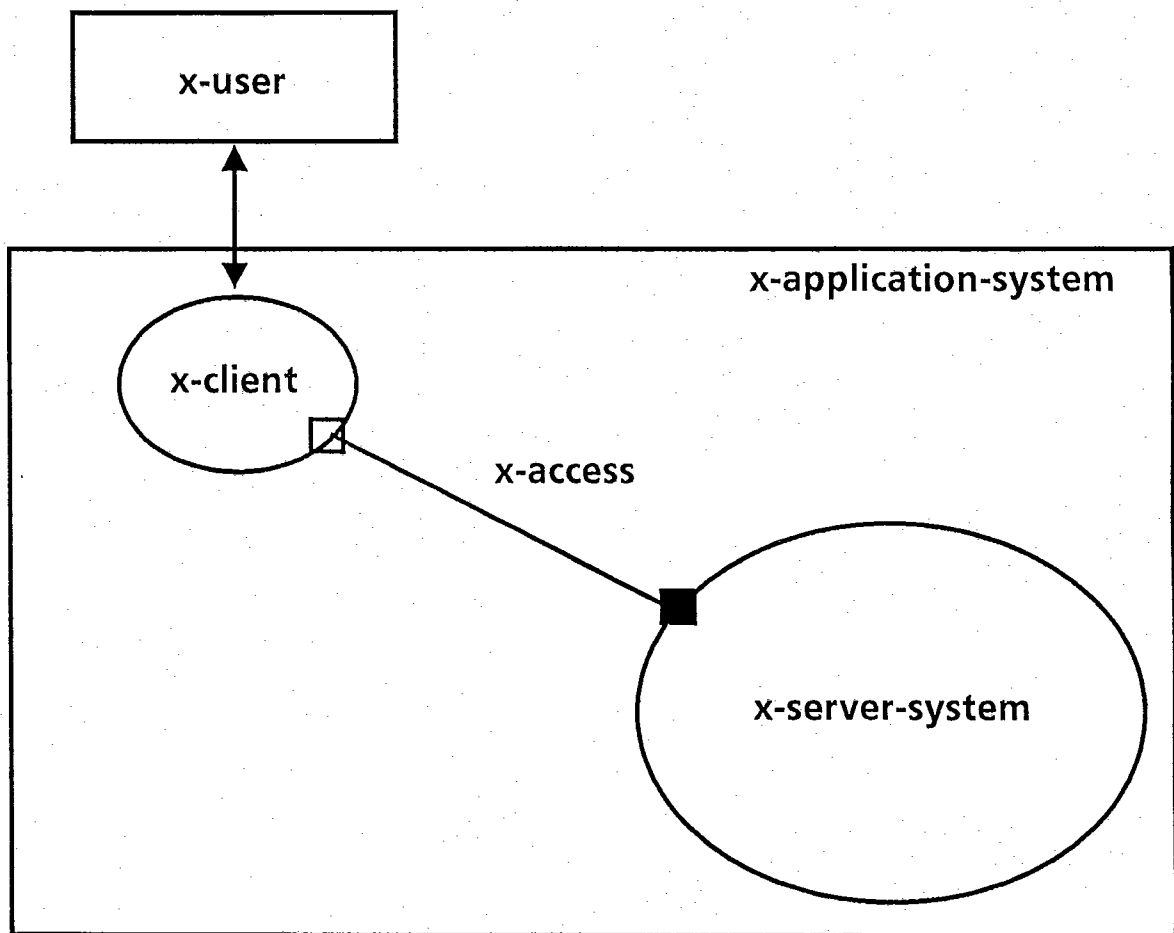


Figure 1 – Distributed-office-applications Abstract Model for Access

In figure 1, the x-user is a user of an x-application, which is provided by the x-application-system. The x-user interacts with x-client to use the service provided by the x-application. The x-client accesses to the x-server via the x-access. The x-server-system may be divided and distributed to more than one x-servers. Details of the internal structure of x-server-system is defined in 5.1.2.

ISO /IEC 10031-1:1991(E)

One or more ports may be defined between the x-client and the x-server. For every port, port type shall be asymmetric.

NOTE 1 – Services provided by symmetric ports for access are outside the scope of this part of ISO/IEC 10031.

Information exchanged between the x-client and the x-server-system shall be Office-information. Office-information is data used in the office environment, for example:

- a) documents;
- b) messages;
- c) EDIFACT data;
- d) attributes of documents;
- e) time;
- f) information relating to messages;
- g) information to file documents;
- h) information to print documents (including fonts);
- i) management information for servers.

This information is viewed as a collection of office data objects, which can be accessed and manipulated individually or in groups.

NOTE 2 – Exchange of information other than Office-information, is, and will be, defined by other International Standards.

5.1.2 Abstract Model for Server-Systems

The x-server-system in figure 1 may be refined to distribute the x-server-system by defining an abstract service between servers, as ISO/IEC 10021-3 suggests. Figure 2 shows a refinement of an x-server-system.

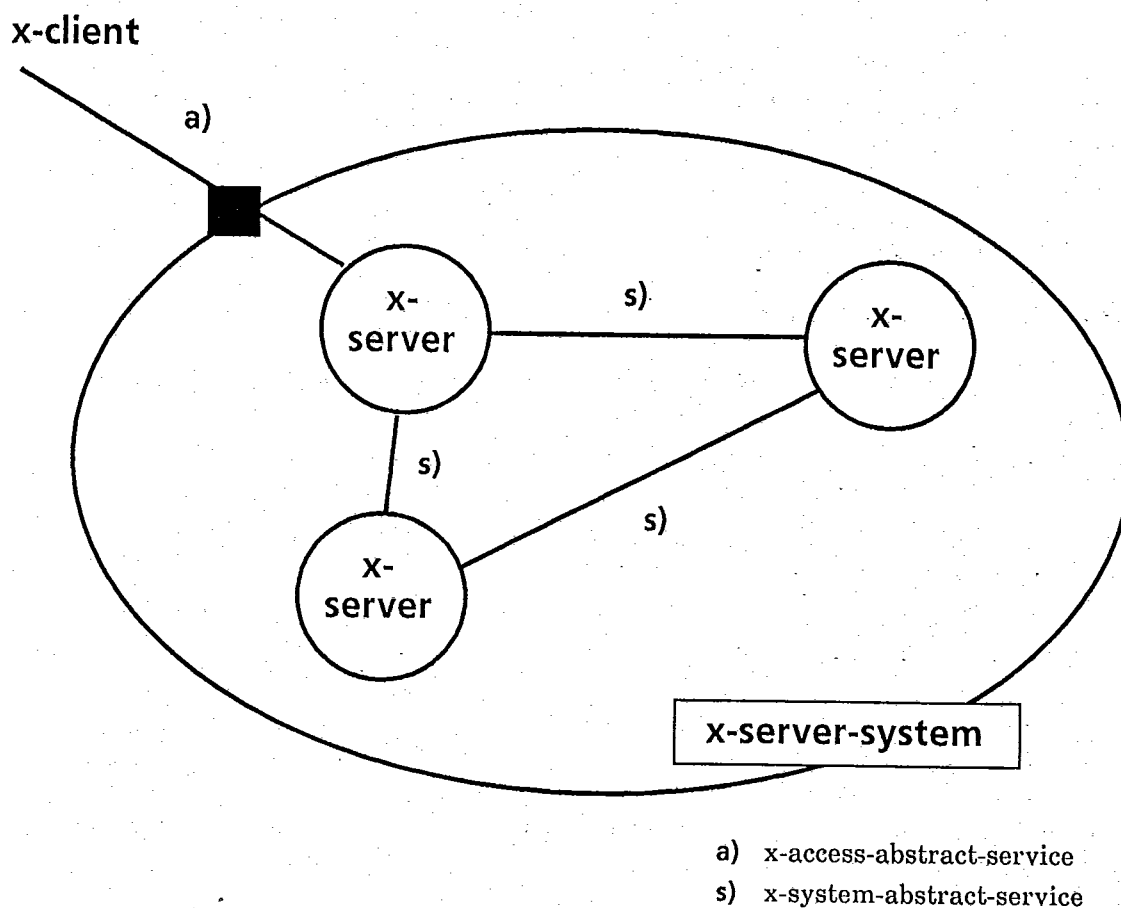


Figure 2 – Refinement of x-server-system

In figure 2, an x-client accesses an x-server-system via the x-access-abstract-service (a)). In the x-server-system, an x-server responds to the access. The x-server may interact with other x-servers via the x-system-abstract-service (s)) to perform the service requested by the x-client.

An x-server-system may comprise different types of x-servers.

One or more ports may be defined between x-servers. Any type of port may be used.

5.2 Realization of DOA Abstract Model

5.2.1 Realization of Abstract Model for Access

To realize the Abstract Model for access, ROSE defined by ISO/IEC 9072 and its OSI mapping shall be used. The Layer Model is shown in figure 3. Further information about how x-clients and x-servers are identified, see 6.4.4.

5.2.2 Realization of Abstract Model for server-systems

There is no restriction to realize the Abstract model for server-systems. Examples are shown in annex D.

5.3 Referenced-object-access

5.3.1 Classes of Data Access

The access of data object values conceptually involves three parties:

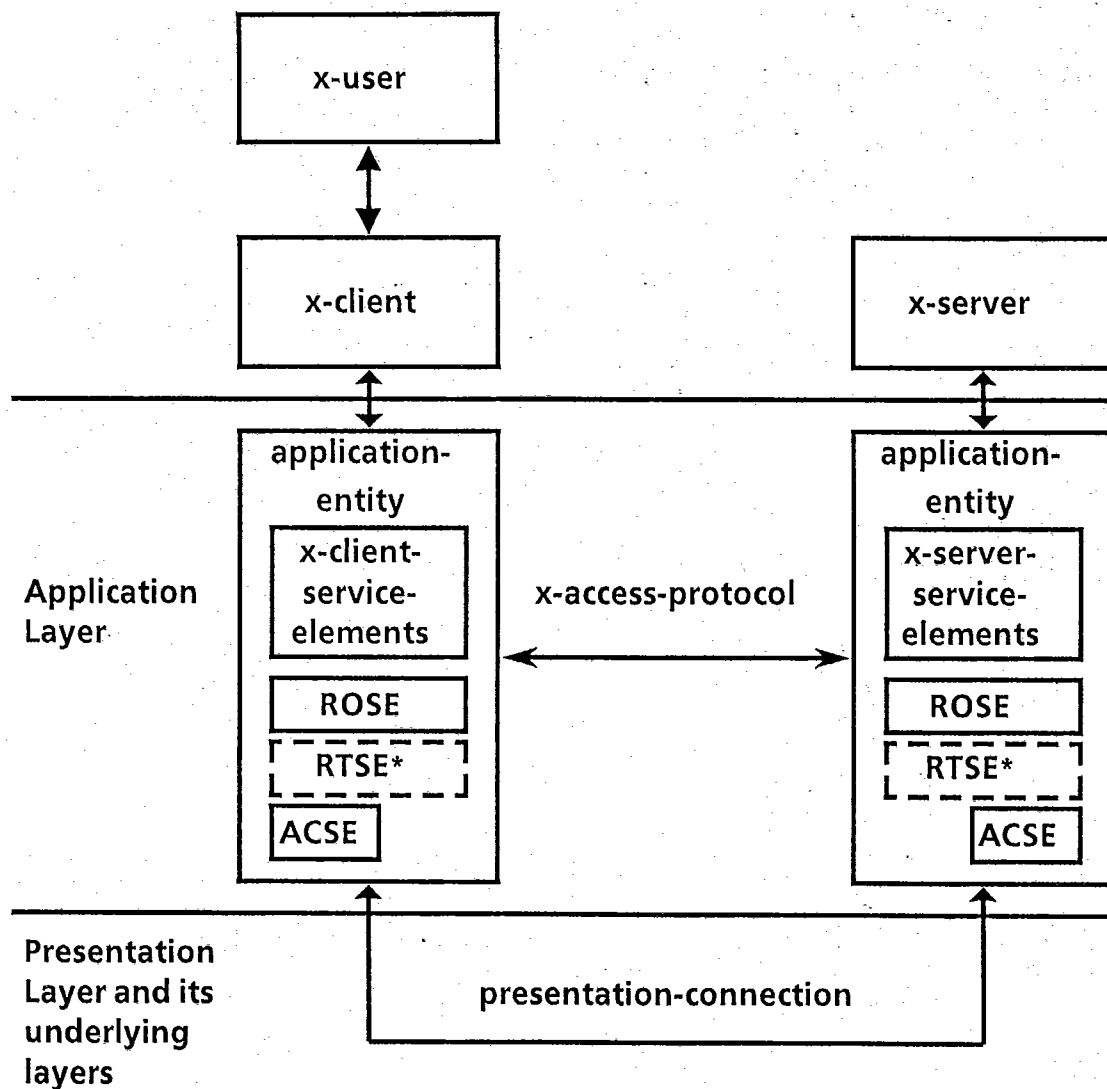
- a) an Initiator, which requests the access,
- b) an Accessee, which stores and produces the data object value,
- c) an Accessor, which consumes or modifies the data object value.

Within distributed-office-applications, there will be applications that will act as an Accessee or Accessor for data objects, for example files, documents or body parts.

When the Initiator is co-located with either the Accessor or the Accessee, data access takes place as part of the access request. This is known as *direct value* access.

In cases where the Initiator is separated from both Accessor and Accessee either physically or by a period of time, using direct value access may involve two data transfers (a "read" and a "write" operation). Alternatively, to use network capacity more efficiently, the Initiator may ask an Accessee to return a *reference* to the data object, rather than its actual value. This reference can then given by the Initiator to the Accessor, which can contact the Accessee directly to access the data object value with a single transfer.

Using high-level programming languages as an analogy, one can consider that the argument or result is passed "by value" when using direct access, and "by name" when using referenced data access.



*use of RTSE is optional

NOTES

- 1 This figure is an example and does not restrict mappings.
- 2 An x-client and an x-server may have several application entities and vice versa.
- 3 An application entity may serve different server types.

Figure 3 – Layer Model for Realization of DOA Abstract Model for Access

5.3.2 Functional Model for Referenced-object-access

5.3.2.1 Functional Model

The referenced-object-access (ROA) functional model is applicable when an Initiator, Accessee and Accessor are separated either spatially or temporally. Figure 4 illustrates this functional model. For example, the Initiator, Accessee and Accessor may be running on three different systems, or the Initiator system may at some later point in time act as the Accessee or Accessor.

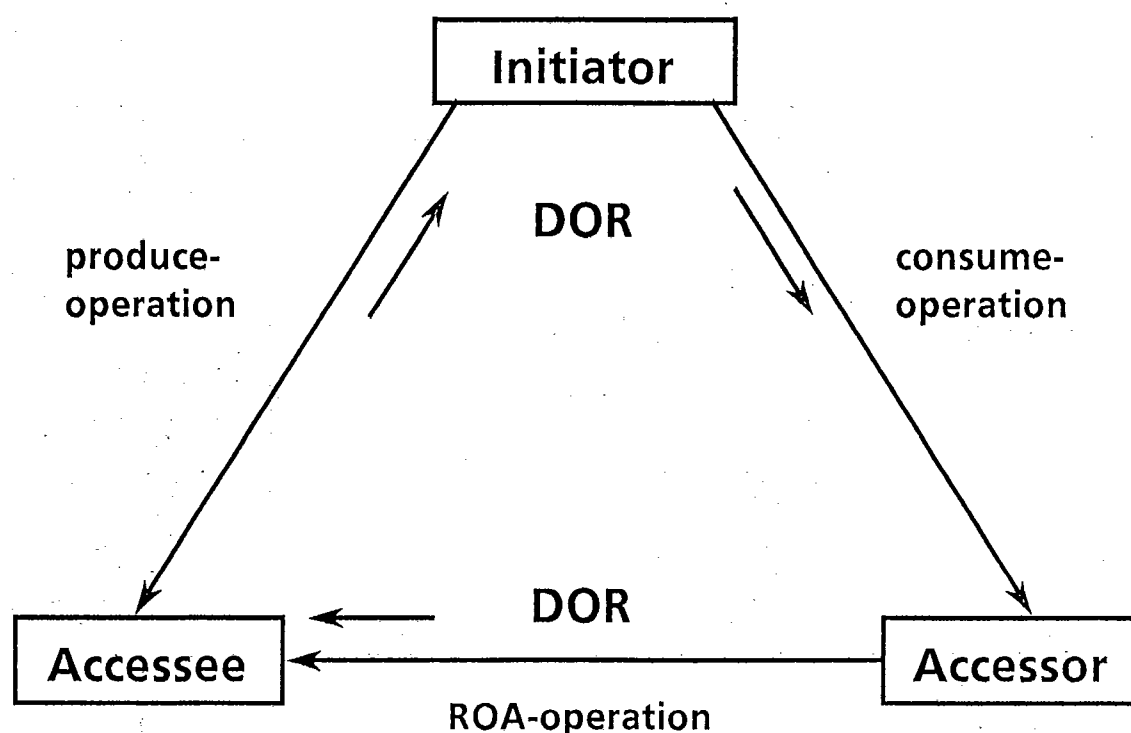


Figure 4 – Referenced-object-access model

In ROA, a reference to the data object value (known as a Distinguished-object-reference, or DOR) is returned to the Initiator in the result of a produce-operation and passed to the Accessor in the argument of a consume-operation. The Accessor then invokes an ROA-operation. In the case of a write operation, a new value, or instructions for modifying the data value, can be passed with the access request. In the case of a read operation, the actual value of the referenced data object is returned in the result of the access operation.

The value accessed when an ROA-operation is performed is related to the data known to the Accessee by application-specific rules that are associated with the DOR during the produce-operation. For example, a DOR could be defined to reference the first Body part of a particular MOTIS IP-message.

ROA-operations are not necessarily constrained to access a fixed or permanent data object.

In figure 4, a DOR is generated by the Accessee in response to a produce-operation invoked by the Initiator. The DOR is offered to the Accessor by the Initiator as a parameter in an consume-operation. The Accessor can then use the DOR to interact with the Accessee.

5.3.2.2 Produce-Operations

In some data operations, the Initiator uses an application-specific protocol to select a data object value (the complete data object, or some subset or derivative of the data object) from an Accessee. This class of operation is termed "produce-operation".

In direct value transfer, the Accessee returns the data object value to the Initiator, and the Initiator takes the role of the Accessor. In the indirect case, on the other hand, the Initiator requests a reference to the data object, rather than the data object value, and the Accessee returns the DOR to the Initiator. DORs identify their associated data value uniquely.

Where DORs are supported by a produce protocol element, a parameter in the Invoke is required to specify whether a direct data value or a DOR is to be returned. The result will accordingly contain a data value or a DOR.

5.3.2.3 Consume-Operations

The Initiator can also use an application-specific protocol to cause a data object value to be accessed by an Accessor. This class of operation is termed "consume-operation". In direct value transfer the Initiator also acts as the Accessee and provides the data object value in the protocol. In the indirect case, the Initiator provides a DOR (previously obtained from an Accessee) to the Accessor. The Accessor uses this DOR to perform an access operation on the Accessee to read or write the referenced data object value.

Where DORs are supported by a consume protocol element, the supplied data may be either a data object value or a DOR. The result will have identical semantics in both cases, but if a DOR is used, the Accessor may need to wait for the result of the ROA-operation before returning the result of the consume operation.

5.3.2.4 ROA-operations

The distributed-office-applications model defines a special class of protocols which always use DORs to interact with Accessee Application-objects, and which provide a generalized set of operations. This class is called "Referenced-object-access protocols (ROA-protocols)".

5.3.2.5 Implications of supporting DORs

DORs require additional functionality to be built into the Accessee and Accessor:

- a) The Accessee needs to be able to provide a DOR rather than a data object value in response to a produce-operation;
- b) The Accessor needs to be able to accept a DOR rather than a data object value in a consume operation;
- c) The Accessor needs to be able to invoke an access-operation;
- d) The Accessee needs to be able to perform an access-operation.

Within application-specific protocols, standards may chose whether:

ISO / IEC 10031-1:1991(E)

- a) to allow DORs to be used in any protocol element where a reference to a data object is supplied or returned;
- b) to place additional restrictions on where DORs are allowed;
- c) to define special protocol elements to handle the ROA-produce and ROA-consume operations.

The first of these option is strongly preferred, and shall be used whenever possible.

If either Accesssee or Accessor in a particular proposed access does not support DOR, then the Initiator has no choice but to execute two consecutive direct-value transfers. In this case, a produce-operation invoked by the Initiator returns a data object value from the Accesssee to the Initiator, and the Initiator transfers that data object value to the Accessor in the argument of a consume-operation. (This description applies to a read operation. In the case of a write operation, the data will flow in the opposite direction.)

5.3.2.6 Quality of Service

The value of some data objects will change with time, or the object may be deleted. Individual protocols may chose whether DORs:

- a) refer to the value of the data object when the DOR was generated;
- b) refer to the current value of the data object;
- c) become undefined if the object is updated.

To assist in the control of references to dynamically changing objects, DOR may include a Quality-of-service (QoS) indication. The QoS describes the expected or required scope for the validity of both the DOR and the value of the associated data. X-access protocols may need to support protocol elements to update the QoS.

5.3.2.7 Structure of DORs

Details of the structure of the DOR and its associated procedures are defined in ISO/IEC 10031-2.

6 Guidelines for the design of protocols

6.1 Introduction

This clause shows the guidelines for the design of protocols to be adhered to by all distributed-office-applications standards.

6.2 Office-information

The main purpose of distributed-office-applications is to interchange, store, retrieve and process office-information.

To maintain the variety of existing or future concepts and types of office-information, the abstract syntax and the semantics of the office-data-object may typically be transparent to the protocols of distributed-office-applications. In this case, the office-data-object shall appear as an ASN.1 external type in the "direct-reference" variant (i.e. no presentation layer negotiation of encoding rules) in the abstract syntax of DOA protocols. The "direct-reference OBJECT IDENTIFIER" value of the external type references both the abstract syntax and the encoding of the object. This value shall be used in attributes identifying the object type.

6.3 Object model and remote operations

6.3.1 Use of remote operations

The Remote Operations defined in ISO/IEC 9072 provide a notation and a protocol specification for bind-operations, unbind-operations and operations (called type-operations in the object model). A standard set and naming guidelines for operations are described in the following subclauses.

All access protocols for distributed-office-applications shall conform to the Remote Operations as specified in ISO/IEC 9072. In more detail the access protocols shall use the notation and concepts of that International Standard and shall allow any mappings defined in ISO/IEC 9072-1 clause 11. Annex J gives a short introduction to these concepts in the context of an access protocol taking into consideration the application rules of 6.4.

For system protocols, it is encouraged to also use Remote Operations wherever possible.

6.3.2 Use of the abstract service technique for the x-service-definition

The abstract service technique is based on a number of ASN.1 macros which are used to describe the function and parameters of services. This description technique for services is closely related to the way Remote Operations are formally defined. The technique guarantees full consistency between service definitions and protocol specifications. It also saves duplication of work and documentation through the possibility to import definitions made for services into the formal protocol. It is also very easy to import definitions made in one DOA to another DOA without having to redefine or redocument them. Standards for MHS and Directory are already using this technique. All future DOAs shall use the same technique for the documentation of services.

Abstract service macros are defined in ISO/IEC 10021-3.

ISO / IEC 10031-1:1991(E)

6.4 Application rules

The following rules have been established in order to simplify the management of shared resources among a number of applications.

6.4.1 Concurrency and resource sharing**6.4.1.1 Concurrency**

There are established techniques in centralized systems to control concurrent access and to preserve the integrity of data. For distributed systems there are no economic general solutions to the general case of distributed data.

Applications should avoid the general case. Until a stronger requirement arises and a solution is worked out for a specific application, the guideline is to use a managed weak consistency - that is:

- a) allow inconsistent data;
- b) have one master copy for each item of data, and have one specific server responsible for updating it;
- c) have one sequence of propagating changes to copies of that data item and changes to related data items;
- d) minimize relationships between data items in different servers;
- e) provide administrative controls to regulate how long it takes for a change to propagate;
- f) design applications to be tolerant of or resilient to out of date data.

With this guideline, concurrency controls can be limited within a single x-server, or at most within an x-system. The impact on protocols is limited to the impact of resource sharing.

6.4.1.2 Resource sharing

Shared resources within a server are managed by the server (which in turn relies on the underlying operating system of the node).

The impact on protocols is limited to managing the consequences of a server not being able to respond to an interaction for a time. This can be manifested by a refusal to accept an interaction, a response indicating a delay or a deferred response. The entity acting as an x-user may impose time-out disciplines.

If required, an x-system may provide for management of resources shared between its x-servers. This will require to be expressed in the x-system-protocol, but will not impact the x-access-protocol except as described above.

6.4.2 Network transparency

To insulate users from the details of environmental network configuration, servers and clients shall be referred by name rather than by the Presentation Address. The Directory may be used to provide this translation.

6.4.3 Common definition of time

All protocols in the distributed-office-application environment shall express the time using the data type "GeneralizedTime" as defined by ISO 8824.

Time :: = GeneralizedTime

NOTE – ISO/IEC 9594 and ISO/IEC 10021 currently use "UTCTime" instead of "GeneralizedTime".
Use of GeneralizedTime in these standards, keeping backward compatibility, is planned.

6.4.4 Common definition of identifiers

All objects defined in DOA-standards shall have at least one globally unique name. A common perception of names and a common definition of identifiers are defined in ISO 7498-3, ISO 8824, and ISO/IEC 9594. They are introduced in annex E.

The ASN.1 encoding of names used by applications adhering to this Model is as defined in ISO/IEC 9594.

6.4.5 Use of attributes and filters

Many objects in the context of distributed-office-applications (e.g. represented in an information base) are characterized by attributes. An attribute consists of an attribute type, which identifies the class of information given by that attribute and the corresponding attribute value(s).

The attribute concept, a notation supporting the definition of attributes, and the abstract syntax of attributes are defined in ISO/IEC 9594-2. A subset is defined in ISO/IEC 10021-5. Standards for distributed-office-applications shall use attributes if appropriate and shall refer to ISO/IEC 9594-2.

If objects represented by entries in an information base are characterized by attributes, information retrieval (e.g. entry selection) may require filters. A filter applies a test that is either satisfied or not (e.g. by a particular entry). The filter is expressed in terms of assertions about the presence or value of certain attributes (e.g. of an entry).

The semantics and abstract syntax of filters are defined in ISO/IEC 9594-3, and a subset is defined in ISO/IEC 10021-5. Standards for distributed-office-applications shall use filters if appropriate and shall refer to ISO/IEC 9594-3.

Attribute Types defined for one application may be reused by another application as long as the definitions and the semantics are the same. The OBJECT IDENTIFIER defined in ISO 8824 is used as a tool to achieve this.

The Attribute Macro is defined in ISO/IEC 9594-2.

The technique of "qualified attributes" may be considered for use in distributed-office-applications. This allows for example a client to flag whether a particular attribute is compulsory in the sense that the corresponding server has to understand the semantics of the attribute and know how to respond, or if the server can ignore the attribute or substitute a default.

NOTE – For examples of the usage of this technique, see ISO/IEC 10175-1:1992¹⁾, *Information technology – Text and office systems – Document printing application – Part 1: Abstract service definition and procedures*.

¹ To be published.

ISO / IEC 10031-1:1991(E)

6.4.6 Referenced-object-access

Data object values and DOR in access protocols may typically appear as ASN.1 external types.

6.4.7 Application-service-elements and application context.

Application-service-elements implementing the functions required for several access protocols may be combined into a single application context. This requires distinct abstract syntaxes for each of these application-service-elements. (See also figure 3.)

6.5 Security rules**6.5.1 Introduction**

The following rules have been established in order to simplify the implementation and management of the security aspects of Access Control and Authentication.

The rules enable a wide range of security policies to be used without change to the distributed-office-applications protocols, including security policies that demand individual accountability of persons using the distributed-office-applications.

NOTE - For detailed tutorial introduction to the security concepts, see annex F.

6.5.2 Security subject

The security subject is often the individual accountable for the operations being performed. In some security policies, accountability may be vested in a group of people, or in a user. Distributed-office-applications may control access in terms of the security subject identity, or in terms of capabilities claimed by the security subject (there are techniques to validate such claims). A Privilege Attribute Certificate is a data structure that securely carries the attributes of security subject; the values present in the structure depend upon the security subject and the security policy in force.

Distributed-office-applications shall use a Privilege Attribute Certificate (PAC) to signify the authenticity and privileges of the security subject. Where the security policy in force requires individual accountability, the PAC shall include the necessary identification.

The PAC shall be passed on the BIND, and it shall apply to all subsequent operations in the association until UNBIND or ABORT terminate it. Each individual operation shall allow the client to pass another PAC; this shall supplement the BIND's PAC for that individual operation.

Where an operation causes processing subsequent to the operation, or subsequent to the UNBIND or ABORT, the the PAC(s) in force for the operation shall also apply to the subsequent processing.

6.5.3 Security objects

The security object is the object that is being protected in order that access by security subjects is regulated by the security policy in force. The security policy may require that checks are made on the basis of privileges claimed by the security subjects identity, or on the basis of privileges claimed by the security subject, or some composite of the two approaches. A Control Attribute

Package (CAP) is a data structure that securely carries the access control information; the values present depend upon the security policy in force and the individual object.

Distributed-office-applications shall use a Control Attribute Package (CAP) to convey the access control information of an office-data-object. This can occur when creating an object, modifying an object, or transferring an object with its access control information.

6.5.4 Access Control

The preceding security rules separate out the design of distributed-office-applications protocols from the design of security elements conveyed within the protocols. This facilitates the use of the same protocols in differing security environments.

The operation definitions and data structure models of distributed-office-applications shall permit a wide range of security policies to be used in conjunction with their protocols.

Any security policy assumptions, such as implicit copying of access control attributes from one security object to another, should be qualified by "subject to the constraints of the security policy in force".

6.5.5 Security Errors

Almost any operation may be wholly refused on the grounds of security.

Security refusals shall be reported as errors, taking precedence over other errors (except for protocol errors which prevent the determination of the security subject, the security object and the operation). Security error responses shall not be phrased in such a way, nor accompanied by, nor followed by other error responses that could imply or convey information that the security subject should not be given.

Operations that cause one access to a single security object shall have no effect on the object if that access is refused.

Operations that access several security objects may encounter security refusals on some of those objects. The resulting effects and responses of such operations shall be defined.

6.6 Standard set of operations

This subclause shows the standard set of abstract operations.

Its primary purpose is to harmonize sets of abstract operations and their names for various distributed office applications. This harmonization will reduce the time and effort needed to standardize specific distributed office applications, and will make it easier for implementors of DOA standards to become familiar with and apply operations of a specific application.

The developers of a specific distributed-office-application shall use the operations of this standard set to define the set of operations in their specific distributed-office-application where appropriate.

The following is the standard set of DOA abstract operations:

- a) List;
- b) Read;
- c) Modify;
- d) Copy;
- e) Move;
- f) Search;

ISO / IEC 10031-1:1991(E)

- g) Create;
- h) Delete;
- i) Reserve;
- j) Notify;
- k) Abandon.

An example of the details of those operations are shown in annex K.