

INTERNATIONAL
STANDARD

ISO/IEC
10164-9

First edition
1995-12-15

**Information technology — Open Systems
Interconnection — Systems Management:
Objects and attributes for access control**

*Technologies de l'information — Interconnexion de systèmes ouverts
(OSI) — Gestion de systèmes: Objets et attributs pour le contrôle d'accès*



Reference number
ISO/IEC 10164-9:1995(E)

Contents

	<i>Page</i>
1 Scope	1
2 Normative references	2
2.1 Identical Recommendations International Standards	2
2.2 Paired Recommendations International Standards equivalent in technical content	2
3 Definitions	3
3.1 Basic reference model definitions	4
3.2 Security architecture definitions	4
3.3 Management framework definitions	4
3.4 Security frameworks overview definitions	4
3.5 Access control framework definitions	4
3.6 Systems management overview definitions	5
3.7 Management information model definitions	5
3.8 Implementation conformance statement proforma definitions	5
3.9 Event report management definitions	5
3.10 OSI conformance testing definitions	5
3.11 Additional definitions	6
4 Symbols and abbreviations	6
5 Conventions	6
6 Requirements	6
7 Interpretation of the Access Control Model	7
7.1 Overview	7
7.2 Access control policies	8
7.3 Access control information	8
7.4 Access control procedures	9
7.5 Representation of access control rules	13
8 Generic definitions	14
8.1 Managed objects	14
8.2 Parameters	23
8.3 Name bindings	23
8.4 Attributes	23
8.5 Imported generic definitions	24
8.6 Compliance	24
9 Service definition	24
9.1 Introduction	24
9.2 Access control management service	24
9.3 Targets administration service	25
9.4 Initiators administration service	25
9.5 Operations administration service	25
9.6 Label administration service	26
9.7 Access control notification service	26

© ISO/IEC 1995

All rights reserved. Unless otherwise specified, 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

	<i>Page</i>
10 Functional units	26
11 Protocol	26
11.1 Elements of procedure	26
11.2 Abstract syntax	26
11.3 Negotiation of access control functional unit	27
12 Relationship with other functions	27
13 Conformance	29
13.1 Static conformance	29
13.2 Dynamic conformance	29
13.3 Management information conformance requirements	30
Annex A – Definition of management information	31
Annex B – MCS proforma	49
Annex C – MICS proforma	57
Annex D – MOCS proforma	61
Annex E – MRCS proforma for name binding	102
Annex F – MIDS (Parameter) proforma	104
Annex G – CMIP Access Control Parameter	105
Annex H – Relationship to ITU-T Rec. X.812 ISO/IEC 10181-3: Security Frameworks in Open Systems – Access Control	106

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

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. 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.

International Standard ISO/IEC 10164-9 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 21, *Open systems interconnection, data management and open distributed processing*, in collaboration with ITU-T. The identical text is published as ITU-T Recommendation X.741.

ISO/IEC 10164 consists of the following parts, under the general title *Information technology — Open Systems Interconnection — Systems Management*:

- *Part 1: Object management function*
- *Part 2: State management function*
- *Part 3: Attributes for representing relationships*
- *Part 4: Alarm reporting function*
- *Part 5: Event report management function*
- *Part 6: Log control function*
- *Part 7: Security alarm reporting function*
- *Part 8: Security audit trail function*
- *Part 9: Objects and attributes for access control*
- *Part 10: Accounting metering function*
- *Part 11: Metric objects and attributes*
- *Part 12: Test management function*
- *Part 13: Summarization function*
- *Part 14: Confidence and diagnostic test categories*
- *Part 15: Scheduling function*
- *Part 16: Management knowledge management function*
- *Part 17: Change over function*
- *Part 18: Software management function*
- *Part 19: Management domain and Management policy management function*

Annexes A to F form an integral part of this part of ISO/IEC 10164. Annexes G and H are for information only.

Introduction

This Recommendation | International Standard specifies an Access Control Security Model and the management information necessary for creating and administering access control associated with OSI Systems Management. Security policy adopted for any instance of use is not specified and is left as an implementation choice. This Specification is of generic application and is applicable to the security management of many types of application. It is expected to be adopted for TMN use.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

This page intentionally left blank

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

This page intentionally left blank

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SYSTEMS MANAGEMENT: OBJECTS AND ATTRIBUTES FOR ACCESS CONTROL

1 Scope

The specifications contained herein are applicable to the provision of access control for applications that use OSI management services and protocols.

This Recommendation | International Standard

- establishes user requirements for the provision of access control for applications that use OSI management services and protocols;
- interprets and applies the general model of access control defined in ITU-T Rec. X.812 | ISO/IEC 10181-3 for use with management applications that use OSI management services and protocols;
- defines procedures for the imposition of access control rules in conjunction with the use of OSI management services and protocols;
- defines managed object classes and attribute types that
 - a) represent some of the access control information that may be used in the provision of access control; and
 - b) are only for use when the management of the access control information is to be achieved using systems management;
- specifies the protocol that is necessary to exchange the access control information defined in this Recommendation | International Standard, when the exchange is achieved using OSI systems management;
- specifies conformance requirements for open systems that claim to support access control for applications that use OSI management services and protocols;
- specifies conformance requirements for open systems that claim to support the management of the access control information defined in this Recommendation | International Standard.

The access control information identified by this Recommendation | International Standard may be used in support of access control schemes based on access control lists, capabilities, security labels, and contextual constraints.

This Recommendation | International Standard does not

- define an access control policy for applications that use OSI management services and protocols;
- define security (or management) domains in which an access control policy may be imposed;
- define how the components of an access control function be implemented, nor where those components be located;
- specify the form of any access control information that is temporarily or permanently stored in an open system;
- specify any access control mechanisms, nor mandate the use of any particular access control mechanism;
- mandate that access control information be managed, and if it is to be managed, that management be achieved using OSI systems management;
- describe how communicating management application entities act to make access control decisions on behalf of, or for the benefit of any third party;
- specify any conformance requirement for the access control parameter defined in this Recommendation | International Standard.

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 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 the ITU maintains a list of currently valid ITU-T Recommendations.

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*.
- ITU-T Recommendation X.509 (1993) | ISO/IEC 9594-8:1995, *Information technology – Open Systems Interconnection – The Directory: Authentication framework*.
- CCITT Recommendation X.701 (1992)¹⁾ | ISO/IEC 10040:1992¹⁾, *Information technology – Open Systems Interconnection – Systems management overview*.
- CCITT Recommendation X.720 (1992) | ISO/IEC 10165-1:1993, *Information technology – Open Systems Interconnection – Structure of management information: Management information model*.
- CCITT Recommendation X.721 (1992) | ISO/IEC 10165-2:1992, *Information technology – Open Systems Interconnection – Structure of management information: Definition of management information*.
- CCITT Recommendation X.722 (1992) | ISO/IEC 10165-4:1992, *Information technology – Open Systems Interconnection – Structure of management information: Guidelines for the definition of managed objects*.
- ITU-T Recommendation X.724 (1993) | ISO/IEC 10165-6:1994, *Information technology – Open Systems Interconnection – Structure of management information: Requirements and guidelines for implementation conformance statement proformas associated with OSI management*.
- CCITT Recommendation X.730 (1992) | ISO/IEC 10164-1:1993, *Information technology – Open Systems Interconnection – Systems management: Object management function*.
- CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2:1993, *Information technology – Open Systems Interconnection – Systems management: State management function*.
- CCITT Recommendation X.732 (1992) | ISO/IEC 10164-3:1993, *Information technology – Open Systems Interconnection – Systems management: Attributes for representing relationships*.
- CCITT Recommendation X.734 (1992) | ISO/IEC 10164-5:1993, *Information technology – Open Systems Interconnection – Systems management: Event report management function*.
- CCITT Recommendation X.736 (1992) | ISO/IEC 10164-7:1992, *Information technology – Open Systems Interconnection – Systems management: Security alarm reporting function*.
- CCITT Recommendation X.740 (1992) | ISO/IEC 10164-8:1993, *Information technology – Open Systems Interconnection – Systems management: Security audit trail function*.
- ITU-T Recommendation X.810²⁾ | ISO/IEC 10181-1...²⁾, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Security frameworks overview*.
- ITU-T Recommendation X.812²⁾ | ISO/IEC 10181-3...²⁾, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Access control framework*.

2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1)*.
ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*.

¹⁾ As amended by ITU-T Rec. X.701/Cor.2 | ISO/IEC 10040/Cor.2.

²⁾ Presently at the stage of draft.

- CCITT Recommendation X.209 (1988), *Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*.
ISO/IEC 8825:1990, *Information technology – Open systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*.
- CCITT Recommendation X.217 (1992), *Service definition for the Association Control Service Element*.
ISO 8649:1988³⁾, *Information processing systems – Open Systems Interconnection – Service definition for the Association Control Service Element*.
- CCITT Recommendation X.227 (1992), *Connection-oriented protocol specification for the Association Control Service Element*.
ISO 8650:1988⁴⁾, *Information processing systems – Open Systems Interconnection – Protocol specification for the Association Control Service Element*.
- CCITT Recommendation X.290 (1992), *OSI conformance testing methodology and framework for protocol Recommendations for CCITT applications – General concepts*.
ISO/IEC 9646-1:1994, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts*.
- CCITT Recommendation X.291 (1992), *OSI conformance testing methodology and framework for protocol Recommendations for CCITT applications – Abstract test suite specification*.
ISO/IEC 9646-2:1994, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 2: Abstract Test Suite specification*.
- ITU-T Recommendation X.296⁵⁾, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Implementation conformance statements*.
ISO/IEC 9646-7:1995, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 7: Implementation conformance statements*.
- CCITT Recommendation X.700 (1992), *Management framework for Open Systems Interconnection for CCITT applications*.
ISO/IEC 7498-4:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework*.
- CCITT Recommendation X.710 (1991), *Common management information service definition for CCITT applications*.
ISO/IEC 9595:1991, *Information technology – Open Systems Interconnection – Common management information service definition*.
- CCITT Recommendation X.711 (1991), *Common management information protocol specification for CCITT applications*.
ISO/IEC 9596-1:1991, *Information technology – Open Systems Interconnection – Common management information protocol – Part 1: Specification*.
- CCITT Recommendation X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications*.
ISO 7498-2:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security architecture*.

3 Definitions

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

³⁾ As amended by ISO/IEC 8649:1988/Amd.1:1990.

⁴⁾ As amended by ISO/IEC 8650:1988/Amd.1:1990.

⁵⁾ Presently at the stage of draft

3.1 Basic reference model definitions

This Recommendation | International Standard makes use of the following term defined in ITU-T Rec. X.200 | ISO/IEC 7498-1:

- open system.

3.2 Security architecture definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.800 | ISO 7498-2:

- a) access control;
- b) access control list;
- c) authentication;
- d) capability;
- e) security label;
- f) security policy.

3.3 Management framework definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.700 | ISO/IEC 7498-4:

- a) managed object;
- b) systems management application entity.

3.4 Security frameworks overview definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.810 | ISO/IEC 10181-1:

- a) security certificate;
- b) security domain;
- c) security token.

3.5 Access control framework definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.812 | ISO/IEC 10181-3:

- a) access control certificate;
- b) Access Control Decision Information (ADI);
- c) Access Control Decision Function (ADF);
- d) Access Control Enforcement Function (AEF);
- e) Access Control Information (ACI);
- f) access control policy;
- g) contextual information;
- h) initiator;
- i) Initiator Access Control Decision Information (initiator ADI);
- j) Initiator Access Control Information (initiator ACI);
- k) Initiator-bound Access Control Information (initiator-bound ACI);
- l) Operand Access Control Decision Information (operand ACI);
- m) Operand-bound Access Control Information (operand-bound ACI);

- n) retained ADI;
- o) target;
- p) Target Access Control Decision Information (target ADI);
- q) Target Access Control Information (target ACI);
- r) Target-Bound Access Control Information (target-bound ACI).

3.6 Systems management overview definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.701 | ISO/IEC 10040:

- a) generic definitions;
- b) managed object class;
- c) Managed Object Conformance Statement (MOCS);
- d) Management Information Conformance Statement (MICS);
- e) management operation;
- f) MICS proforma;
- g) MOCS proforma;
- h) notification.

3.7 Management information model definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.720 | ISO/IEC 10165-1:

- a) action;
- b) characteristic.

3.8 Implementation conformance statement proforma definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.724 | ISO/IEC 10165-6:

- a) Managed Relationship Conformance Statement (MRCS);
- b) Management Conformance Summary (MCS);
- c) Management Information Definition Statement (MIDS) proforma;
- d) MCS proforma;
- e) MRCS proforma.

3.9 Event report management definitions

This Recommendation | International Standard makes use of the following term defined in CCITT Rec. X.734 | ISO/IEC 10164-5:

- event forwarding discriminator.

3.10 OSI conformance testing definitions

This Recommendation | International Standard makes use of the following term defined in CCITT Rec. X.290 | ISO/IEC 9646-1:

- a) PICS proforma;
- b) protocol implementation conformance statement;
- c) system conformance statement.

3.11 Additional definitions

- 3.11.1 security domain authority:** An entity responsible for the implementation of a security policy.
- 3.11.2 operation type:** The effective action on a managed object as a result of a management request.

4 Symbols and abbreviations

ACC	Access Control Certificate
ADI	Access Control Decision Information
ACI	Access Control Information
ACL	Access Control List
ADF	Access Control Decision Function
AEF	Access Control Enforcement Function
CMIS	Common Management Information Service
CMIP	Common Management Information Protocol
ICS	Implementation Conformance Statement
MAPDU	Management Application Protocol Data Unit
MCS	Management Information Conformance Statement
MICS	Management Information Conformance Statement
MIDS	Management Information Definition Statement
MOCS	Managed Object Conformance Statement
MRCS	Managed Relationship Conformance Statement
PAC	Privilege Attribute Certificate
PICS	Protocol Implementation Conformance Statement
SMAE	Systems Management Application Entity

5 Conventions

This Recommendation | International Standard makes use of the notational techniques for defining managed objects and attributes specified in CCITT Rec. X.722 | ISO/IEC 10165-4.

6 Requirements

An OSI Management user requires that unauthorized access to management applications and management information be prevented by the use of one or more access control mechanisms.

Control of access to management information is required in each of the following cases:

- to protect management information from unauthorized creation, deletion, modification or disclosure by means of OSI Management operations;
- to ensure that initiators are only able to use the management operations for which access rights were granted during application association establishment; and
- to prevent management information from being transmitted to unauthorized recipients by means of confirmed or non-confirmed event reports.

NOTE – For completeness there is also a requirement for the control of access to associations. This subject is for further study.

Various levels of access control may be required. For example, some users may be given read and write access to specific attributes, whilst others may have only read access or no access. Some users may be granted rights to access only specific managed objects whilst other users may have access to a different set of managed objects. For management operations, access restrictions need to cater for managed objects, individual attributes of managed objects, values of individual attributes, context of the access and actions associated with the managed object.

There is a requirement for the provision of an access control parameter that may be used in management exchanges which use CMIS.

There is a requirement for open systems which support access control for management applications which use OSI management services and protocols to be observed to exhibit the same access control behaviour when imposing the same set of access control rules.

It is required that the use of access control mechanisms not identified in this Recommendation | International Standard shall not be prevented by the provisions contained herein.

In order that the management of access control information be achieved using OSI systems management, it is required that:

- the information be modelled as managed objects so that it may be created, deleted, modified and read;
- it be possible to determine which targets may be accessed by an initiator; and
- it be possible to determine which initiators may have access to a target.

It is required that the discovery of which managed objects are contained within a management information sub-tree by use of a scoped get may be prevented.

It is required that the general application of a scoped management operation (such as delete) may be prevented.

It is required that unique security labels may be assigned to specific targets.

7 Interpretation of the Access Control Model

7.1 Overview

Access control for OSI management is based on the model for access control defined in ITU-T Rec. X.812 | ISO/IEC 10181-3. In the base model, the access control decision and enforcement functions are interposed between the initiator and the target. This Recommendation | International Standard illustrates how that model applies to the provision of access control for applications that use OSI Management services and protocols.

The access control decision function requires information, generically termed Access Control Information (ACI), for use in the decision making process. Access control information may be modelled as management information and documented using the notational techniques specified in CCITT Rec. X.722 | ISO/IEC 10165-4. This Recommendation | International Standard defines some access control information as attributes of managed object classes, so that OSI systems management may be used to exchange that information between open systems.

NOTE – This Recommendation | International Standard does not specify the actual form or structure of any access control information that is temporarily or permanently stored in an open system. It does specify the abstract syntax of those elements of access control information that may be exchanged between open systems using OSI systems management.

All access control schemes identified in ITU-T Rec. X.812 | ISO/IEC 10181-3 – ACL schemes, capability schemes, context based schemes and label based schemes – are applicable to the provision of access control for applications that use OSI Management services and protocols. Access control schemes may be used singly or in combination in order that access control may be supported in accordance with an access control policy appropriate to the security domain.

Access control is controlled by an access control policy. Where a specific access control policy is imposed on a group of elements, the combination of elements plus access control policy are encompassed by a specific security domain. Within a security domain a single access control policy is enforced at any instant of time.

Authentication of the OSI Management user and the peer authentication of SMAEs are outside the scope of this Recommendation | International Standard. However, the access control procedures defined herein require the use of authentication procedures at the appropriate time. Possible authentication procedures are described in CCITT Rec. X.217 | ISO 8649, CCITT Rec. X.227 | ISO 8650 and ITU-T Rec. X.509 | ISO/IEC 9594-8.

Access control for OSI Management is specified as a collection of denials and permissions to perform management operations. The user of the management application that invokes the management operation is the initiator, and the elements of management information, for example managed objects and attributes, identified by the parameters of the management operation combine to form the target(s).

Access control of event reports is effected by applying access control to management operations upon event forwarding discriminators.

7.2 Access control policies

An access control policy incorporates one or more sets of rules. It is the responsibility of the access control policy implementor to ensure that the access control rules accurately represent an instantiation of the access control policy.

An access control policy is a specific management policy that may be the subject of management policy administration.

NOTE – This Recommendation | International Standard does not specify the management of management policies and in particular does not specify any means to check the integrity of access control information.

An access control policy is enforced by the use of one or more access control schemes. Access control schemes include:

- ACL schemes;
- Capability schemes;
- Context based schemes; and
- Label based schemes.

If some characteristics of a managed object (for instance attributes) are under the jurisdiction of different security policies, then the managed object may be in multiple security domains. When multiple access control policies apply to a managed object, the enforced access control policy is one that is associated with both the initiator and the target.

7.3 Access control information

Access control information includes:

- access control rules;
- the identity of the initiator of the access request (Initiator ACI);
- capabilities and security clearances associated with the initiator (Initiator ACI);
- information pertaining to the authentication of the initiator (Retained ADI);
- the management information identities (targets) to which access has been requested (Target ACI);
- capabilities and security clearances associated with the target (Target ACI);
- the permitted operations that may be performed on the management information (Initiator ACI, Target ACI);
- information retained by the access control decision function for subsequent use (Retained ADI); and
- contextual information.

7.3.1 access control rules: The access control information that represents the permitted operations and the conditions upon their execution in a security domain. There are five classifications of access, control rule which are to be applied by the access decision function:

- **Global deny rules:** Access control rules that deny access to all targets. If a global rule denies access, then no other rule shall apply. If a global rule does not deny access, then the item deny rules are imposed.
- **Item deny rules:** Access control rules that deny access to particular targets. If an item deny rule denies access, then no other rule shall apply. If an item deny rule does not deny access, then the global grant rules are applied.
- **Global grant rules:** Access control rules that grant access to all targets. If a global rule grants access, then no other rule shall apply. If a global rule does not grant access, then the item grant rules are imposed.
- **Item grant rules:** Access control rules that grant access to particular targets. If an item grant rule grants access, then no other rule shall apply. If an item grant rule does not grant access, then the default rules are applied.
- **Default rules:** The access control rules to be applied when no other rule has specifically granted or denied access. The default rules shall grant or deny access.

7.3.2 action-bound ACI: The access control information (such as a security label) that is associated with the management information carried in management operations and event reports. Action-bound ACI may also be used to create and/or modify access control information associated with a target.

7.3.3 initiator-bound ACI: The access control information provided by, or otherwise associated with, the initiator of a management request. This information may be:

- the identity of the initiator of the management operation;
- information conveyed in the access control parameter of management operations (e.g. CMIS access control parameter, CMIP user information access control parameter);
- information assigned by a security domain authority; or
- a combination of the above.

The access control parameter conveyed by CMIS may take the form of security certificate or a security token.

NOTE – The identity of the initiator may be passed from authentication mechanisms using local procedures outside the scope of this Recommendation | International Standard.

7.3.4 target-bound ACI: The access control information that identifies the management information on which operations are to be performed.

7.3.5 contextual information: The access control information associated with context (for example, time of day, authentication level, location, resource limitation, participation in a relationship).

7.3.6 ADI: Action ADI, Initiator ADI, and Target ADI are derived from Action-bound ACI, Initiator-bound ACI and Target-bound ACI respectively, for the decision purpose.

7.3.7 retained ADI: The access control information that is retained by the access control decision function. According to the access control policy, some of this information may be retained for periods of time longer than that of the life of an association. Retained ADI may be used by the ADF to evaluate access privileges.

7.4 Access control procedures

Access control rules specify the security criteria that have to be met in order to allow management information to be accessed. The rules may mandate that some or all of the following procedures be performed:

- validation of initiator-bound ACI;
- identification of the target;
- determination of the access decision;
- modification of retained ADI;
- modification of target-bound ACI; and
- enforcement of the decision.

Prior to the establishment of an association, access control information that represents access control rules for that access control domain may be generated and distributed by a security domain authority using mechanisms outside the scope of this Recommendation | International Standard.

The procedures which follow specify which access decisions have to be made, not where they are to be made. This Recommendation | International Standard does not specify whether the decisions are to be made in the managing system, managed system, both systems or elsewhere.

It is the responsibility of the initiator to provide access control information that is compatible with the access control mechanisms specified by the security policy.

The use of some or all of these procedures does not exclude the use of other procedures and other access control mechanisms, not specified by this Recommendation | International Standard.

7.4.1 Validation of initiator-bound ACI

Initiator-bound ACI may be supplied in the access control parameter of the CMIS service request for the management operation. The information may take the form of a security certificate, for example an Access Control Certificate (ACC), or a security token.

The security policy specifies which of the following shall be performed:

- the integrity of the information shall be validated using procedures outside the scope of this Recommendation | International Standard;
- the validity of the information shall be verified by checking that the origin of the information was a recognized security domain authority;

- the content of the information shall be verified by checking that the value of the information was within a permitted range.

7.4.2 Identification of the target

A target is an element of information that is to be protected by an access control scheme. Management information is contained in the management information tree. A sub-tree may be selected using the CMIS scope parameter. Where this is the case, then the entire selected sub-tree is considered a target. The selection may be further refined by use of the CMIS scope and filter parameters which select individual managed objects and their characteristics from the management information tree. In this case all the selected managed objects and their characteristics are the target. At the finest granularity it is possible to select individual managed objects from the management information tree. In this case, only the selected managed object and its characteristics are the target.

Any given management operation request identifies one or more targets. These targets are identified as follows:

- a) When the scope parameter is present in the request, the entire management sub-tree encompassed by the request is a target. That is, the combination of base managed object class, base managed object instance, scope, synchronization, operation type, attribute id, action id, attributeInfoArg value, actionInfoArg value parameters forms a target;
- b) When the scope and filter parameters are present in the request, the selected managed objects and characteristics of those managed objects is a target. That is, each distinct base managed object class, base managed object instance, attributeValue, operation type, attribute id, action id, attributeInfoArg value, actionInfoArg value) combination formed from
 - the managed objects selected by the base managed object class, base managed object instance and scope parameters; and
 - the filter item elements of the filter parameter
 forms a target; or
- c) Each distinct managed object and its characteristics selected by the operation form a target. That is, base managed object class, managed object class, base managed object instance, managed object instance, superior managed object instance, reference managed object instance, initial value managed object instance, operation type, attribute id, action id, attributeInfoArg value, actionInfoArg value combination for each managed object selected by the managed object selection parameters is a target.

NOTE – A target includes the operation(s) that is (are) requested for the managed object.

7.4.3 Determination of the access decision

The access control decision function shall perform all of the procedures specified by the security policy. Initiator ADI, Retained ADI, Target ADI, Operand ADI, and contextual information may be used during the procedures. The security policy may include some, or all, of the procedures in 7.4.3.1.

7.4.3.1 Access decision procedures

The procedure identified in 7.4.3.1.1 shall be applied first.

7.4.3.1.1 Identify the management information access rules that apply to the security domain of the initiator and the target.

7.4.3.1.2 For all global rules that deny access by initiators, perform the applicable tests in 7.4.3.2. If any test returns success, indicate to the access control enforcement function that the management operation request shall be denied and apply the procedures in 7.4.6, otherwise apply procedure in 7.4.3.1.3.

7.4.3.1.3 For all item rules that deny access to targets, perform the applicable tests in 7.4.3.2. If any test returns success, indicate to the access control enforcement function that the management operation request shall be denied and apply the procedures in 7.4.6, otherwise apply procedure in 7.4.3.1.4.

7.4.3.1.4 For all global rules that grant access by initiators, perform the applicable tests in 7.4.3.2. If any test returns success, indicate to the access control enforcement function that the management operation request shall be granted and apply the procedures in 7.4.4, 7.4.5 and 7.4.6, otherwise apply procedure in 7.4.3.1.5.

7.4.3.1.5 For all item rules that grant access to targets, perform the applicable tests in 7.4.3.2. If any test returns success, indicate to the access control enforcement function that the management operation request shall be granted and apply the procedures in 7.4.4, 7.4.5 and 7.4.6, otherwise apply procedure in 7.4.3.1.6.

7.4.3.1.6 If the policy has not specified any rule that specifically grants or denies access to the target, indicate to the access control enforcement function that the management operation request shall be granted or denied according to the default rule for that operation, and the default denial response shall apply. If the management operation request is to be denied, only the procedures in 7.4.6 apply otherwise apply the procedures in 7.4.4, 7.4.5 and 7.4.6.

7.4.3.2 Access decision tests

The following tests are available to the access decision function, in accordance with the security policy. Each test receives information associated with the initiator and a rule identifying operations on targets. Each test returns a boolean, the value of which is the truth or falsehood of the proposition that “the initiator satisfies the rule”.

Evaluating a requested operation on a specific target managed object may require information about a superior, reference or initial values managed object class or instance that will be disclosed to the initiator. If the initiator does not have access (GET permission) to the required information about the superior, reference or initial values managed object class or instance as determined appropriately by a) to e) below, then the rule shall evaluate to FALSE:

- a) When required by an ACL scheme, the identity, group or role of the initiator shall be compared with the identities of initiators that are associated with the rule. If an identical match is found and if the operation and target associated with the request are compatible with the operations and targets specified by the rule, then the rule shall evaluate to TRUE. If an identical match is not found or either the operation or target associated with the request is incompatible with the operations and targets specified by the rule, then the rule shall evaluate to FALSE.
- b) When required by a capability scheme, the identity associated with the initiator shall be compared with a list of initiator identities associated with the rule. If an identical match is found that allows the capability to be used and the operation and target identified in the request are compatible with those specified in the capability, then the rule shall evaluate to TRUE. If an identical match is not found or either the operation or target identified in the request is incompatible with those specified in the capability, then the rule shall evaluate to FALSE.
- c) When required by a context based scheme, the contextual conditions associated with the rule shall be checked. If all of the contextual conditions are satisfied, then the rule shall evaluate to TRUE, otherwise the rule shall evaluate to FALSE.
- d) When required by a label based scheme, the label associated with the initiator shall be validated against the label associated with the target. If the label associated with the initiator is determined by the label algorithm to be compatible with the label associated with the target, then the rule shall evaluate to TRUE, otherwise the rule shall evaluate to FALSE.
- e) When required by the security policy, perform any other test associated with the rule.

When evaluating the access control rules in a security domain which uses a combination of access control mechanisms, a single rule shall satisfy all the mechanisms associated with that rule.

NOTE – Some security policies may apply combinations of the access control schemes to the rule. In this case, the access control policy shall identify the precedence of the schemes.

7.4.4 Modification of retained ADI

If specified by the security policy, the ADI retained by the access control decision function may be modified using the initiator ACI that was supplied with the management operation request. Relevant information includes:

- ACI that is permanently associated with the initiator;
- ADI retained from previous associations;
- ACI supplied by the initiator;
- ACI obtained as a result of the authentication procedure; and
- contextual information.

NOTE – Mechanisms for managing, obtaining, storing, and retrieving retained ADI are outside the scope of this Recommendation | International Standard.

7.4.5 Modification of target ADI

If specified by the security policy, the ADI associated with the target may be modified using the Action ADI that was supplied with the management operation request according to the following procedures.

7.4.5.1 For the create operation, Action ADI may be used to create Target ADI specific to the newly created managed object. ADI associated with other targets may not be modified.

7.4.5.2 For the delete operation, Action ADI may be used to modify or remove Target ADI specific to the deleted managed object(s). ADI associated with other targets may not be modified.

7.4.5.3 For the replace attribute value, replace with default value, add member, and remove member operations, Action ADI may be used to modify Target ADI specific to the target attribute(s) being modified by the operation. ADI associated with other targets need not be modified.

NOTE – Independent of the management operations identified above, target ADI may be modified. ADI associated with targets may be created, deleted and modified by means outside the scope of this Recommendation | International Standard. For example, the creation of a managed object which represents a resource may also create target ADI as a direct consequence of the creation of the managed object. If management of ACI using OSI Systems Management is permitted, then target ADI may be modified according to clause 8.

7.4.6 Enforcement of the decision

The access control enforcement function shall be responsible for enforcing the policy decision indicated by the access control decision function.

The following subclauses describe:

- a) the meanings of the possible access denial responses that may be returned to the initiator as a result of action by the access control enforcement function;
- b) the procedure to be taken by the access control enforcement function as a result of a management operation request being received with invalid initiator-bound ACI;
- c) the procedure to be taken by the access control enforcement function as a result of a management operation being denied as a result of a global rule denying access;
- d) the procedure to be taken by the access control enforcement function as a result of a management operation being denied as a result of an item rule denying access or as a result of the default rule denying access; and
- e) the requirements for the recording and reporting of events significant to the support of an access control scheme.

7.4.6.1 Enforcement of access denial

Enforcement of access denied requires specification of the appropriate denial response to return to the initiator, and the specification of the specific targets to which access is denied.

One of the following denial response actions shall be specified:

- a denial response is given, the access control enforcement function shall ensure that the access denied error response is returned to the initiator if the management operation service was requested in the confirmed mode;
- no response is given, the access control enforcement function shall ensure that no response is returned to the initiator;
- the association is aborted, the access control enforcement function shall ensure that the ACSE A-ABORT procedure is invoked; or
- a false response is given, the access control enforcement function shall ensure that incorrect management information is returned to the initiator if the management operation service was requested in the confirmed mode.

If no denial response action is specified, the default denial response action is a matter of local policy.

The denial response shall have an associated denial granularity. The possible denial granularities include:

- A single denial at the level of the total management request. Denial of access to any element of management information will result in the total request being denied. No management operations on targets associated with the request shall be performed.
- A denial at the level of each managed object referenced in the request. Denial of access to any operations and attributes of that managed object will result in denial of access to that managed object, but not to other managed objects associated with the request. No management operations on targets associated with the managed object to which access is denied shall be performed.
- A denial at the level of each attribute within individual managed objects referenced by the request. A denial of access to a specific attribute within a managed object will result in access to that attribute being denied, but not to other attributes within the containing managed object, or attributes within other managed objects. No management operations on the specific attributes for which access is denied shall be performed.

If no denial granularity is specified, the default denial granularity is a matter of local policy.

NOTES

- 1 A default denial granularity at the level of the total request is recommended to satisfy the principle of least privilege.
- 2 A default enforcement action of no response or abort association is recommended to satisfy the principle of least privilege.

7.4.6.2 Denial as a result of invalid initiator-bound ACI

If the decision is to deny the request as a result of invalid initiator-bound ACI, then:

- no operation shall be performed on any of the targets specified in the request;
- the specified or default denial enforcement action shall be invoked, with the exception that a specified denial enforcement action of a false response be changed to abort the association;
- the specified granularity of the denial response, if any, shall be ignored, and the response given at the granularity of the total request.

7.4.6.3 Denial as a result of a global rule being satisfied

If the decision is to deny the management operation as a result of global rule, then:

- no operation shall be performed on any of the targets specified in the request;
- the specified or default denial enforcement action shall be invoked;
- the specified granularity of the denial response, if any, shall be ignored, and the response given at the granularity of the total request.

7.4.6.4 Denial as a result of an item rule or a default rule being satisfied

If the decision is to deny the management operation upon the identified target as a result of an item or default rule, then:

- if the target encompasses a sub-tree of the management information tree, i.e. the target was selected by a) of 7.4.2, then no operation shall be performed on any managed objects held within the selected sub-tree;
- if the target is a selection of managed objects [from b) of 7.4.2], then no operation shall be performed on those managed objects identified by the target;
- if the target is one of those identified in accordance with c) of 7.4.2, then no operation shall be performed on that target;
- the specified or default denial enforcement action shall be invoked;
- the granularity of the response shall be at the specified denial granularity, if is specified, or at the default denial granularity.

NOTES

- 1 In the case of multiple object selection, CMIS does not define a response suitable for use at the managed object level of denial. That is, a denial response shall be given for each attribute selected.
- 2 In the case of denial due to access being denied to an attribute in the filter, CMIS provides no means to indicate that the filter attribute is the cause. Therefore, if the target is one of those defined by b) of 7.4.2, and a denial response is called for at the managed object level, then the response need not provide an indication of the attribute that caused the denial.

7.4.6.5 Requirements for recording events significant to access control

If the security policy mandates that requests for management operations be recorded, the access control enforcement function shall ensure that the appropriate notification is generated as follows:

- if the access request is denied, the notification shall be a security alarm report; or
- if the access request is granted, the notification shall be a security audit trail report.

7.5 Representation of access control rules

An access control rule is a mapping from the combination of (initiator, target) pairs to the access permission (grant or deny).

The space of initiators comprises every possible user of managing applications. This space is divisible according to security policy (e.g. ACL schemes, capability schemes, label based schemes). There may be as many ways of dividing this space as there are security policies.

Groups of initiators may be represented by managed objects. The identification of initiators is dependent on the security policy.

The space of targets comprises every conceivable (management operation, argument) value pair. There are many ways of dividing this space:

- by operation;
- by managed object;
- by managed object class;
- by attribute;
- by action;
- by attribute/argument value;
- by scope parameter; and
- by synchronization parameter.

The finest granularity that may be achieved is managed object class, managed object instance, operation, action id, action value, attribute id, attribute value. Where multiple object selection is permitted, access control may be imposed on the managed object class, managed object instance, scope and synchronization combinations. Where scoping and filtering is permitted, access control may be imposed on the managed object class, managed object instance, attribute identifier and attribute value combination for each managed object within the scope.

Groups of targets may be represented by managed objects.

Access control is enforced within the context of a security domain. There are five hierarchies of access control rules within each domain:

- rules which deny specific initiators access to any of the targets in the domain;
- rules which deny specific initiators access to specific targets in the domain;
- rules which grant specific initiators access to all of the targets in the domain;
- rules which grant specific initiators access to specific targets in the domain; and
- rules which grant or deny access in the absence of any other rule which grants or denies access.

8 Generic definitions

The access control information and procedures described in clause 7 may be modelled as managed objects. Figure 1 depicts the inheritance hierarchy for the managed object classes defined in this Recommendation | International Standard.

Figure 2 shows the relationships between some of the managed objects identified in Figure 1.

NOTE – Not all the managed objects shown in Figure 1 are shown in Figure 2. The managed objects not included in Figure 2 (the access control managed object class and the assigned labels and derived managed object classes) are omitted for clarity.

8.1 Managed objects

8.1.1 Access control

This managed object class comprises the management information elements and behaviour which are common to all managed objects representing access control information. It is specified only to provide a single point of specialization for other managed object classes representing access control information.

8.1.1.1 Attributes of access control

The following mandatory attribute is defined for the access control managed object class.

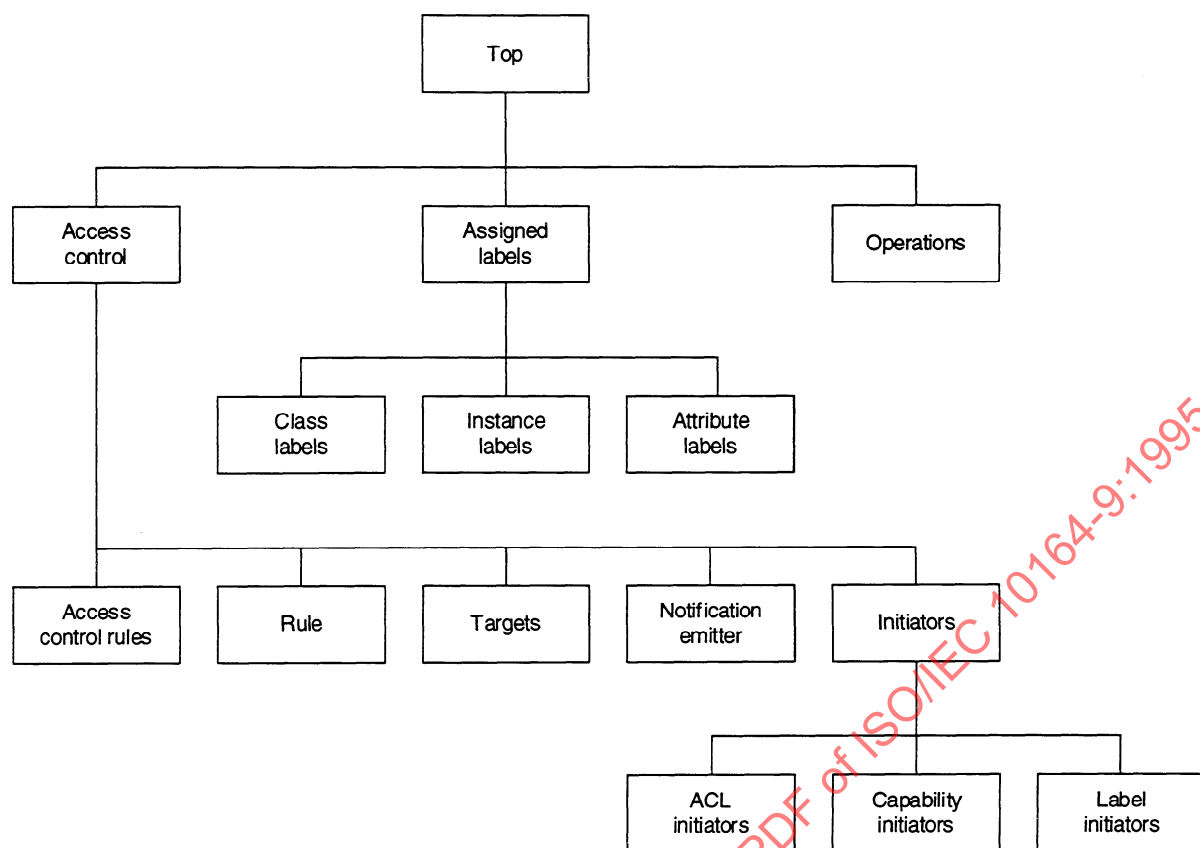
8.1.1.1.1 Access control object name

This attribute is used to identify instantiations of specializations of the access control managed object class.

8.1.1.2 Notifications of access control

The following notifications are defined for the access control managed object class:

- attribute value change;
- object creation; and
- object deletion.



TISO5400-95/d01

Figure 1 – Managed object class inheritance hierarchy

8.1.2 Access control rules

Access control rules managed objects represent the access control decision function for a security domain. Their attributes and contained rules managed objects identify the access control rules for the security domain. The access control rules managed object class is a sub-class of the access control managed object class.

The access control rules managed object contains the other managed objects that represent access control rules for the access control decision function.

8.1.2.1 Attributes of access control rules

The following mandatory attributes are defined for the access control rules managed object class.

8.1.2.1.1 Default access

The default access attribute identifies, in accordance with 7.4.3.1.6, the default access rights for each operation type.

8.1.2.1.2 Default denial response

The default denial response attribute identifies the default response returned to the initiator in the event that the ADF has denied access to the target on the basis of the default rule.

8.1.2.1.3 Domain identity

This attribute identifies the access control domain governing these access control rules.

8.1.2.1.4 Denial granularity

This attribute identifies the level at which denial of access shall be exhibited, if at all.

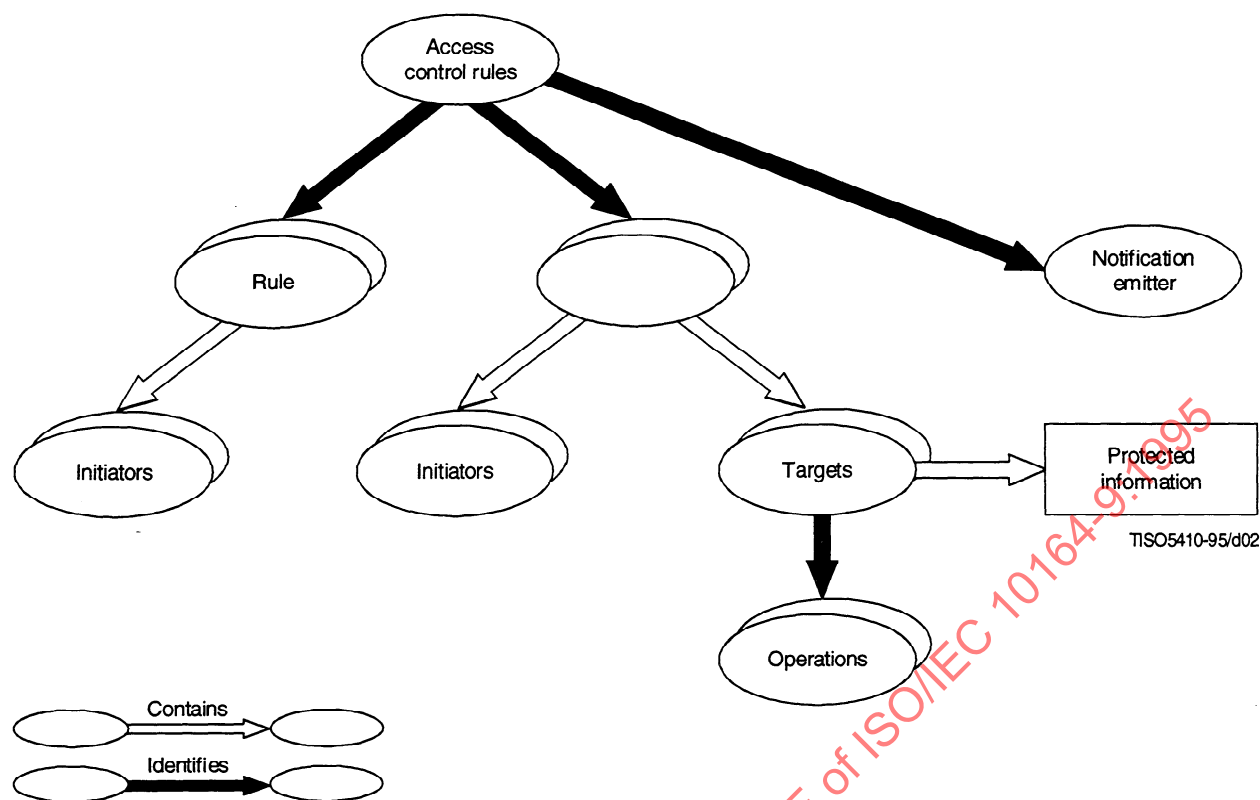


Figure 2 – Relationship between managed objects

8.1.3 Rule

The rule managed object class represents global and item rules. The rule managed object class is a sub-class of the access control managed object class.

8.1.3.1 Attributes of rule

The following mandatory attributes are defined for the rule managed object class.

8.1.3.1.1 Enforcement action

This attribute identifies the action to be taken by the access control enforcement function if the rule is satisfied.

8.1.3.1.2 Initiators list

This set-valued attribute identifies the sub-classes of initiators managed objects which specify the initiators to which the rule pertains.

8.1.3.1.3 Targets list

This set-valued attribute identifies the targets managed objects which themselves specify the targets to which the item rule pertains.

8.1.3.2 Scheduling packages

To accommodate various levels of complexity in scheduling rule activity periods, conditional packages that are related to scheduling are defined for the rule.

Scheduling packages enable rules managed objects to transit automatically between “in force” and “not in force” conditions, signalled by the availability status attribute taking the value {} and {offDuty} respectively.

8.1.3.2.1 Availability status package

This package shall be present if any of the other scheduling related packages are present. If this package is not present, then the rule managed object shall always be available.

8.1.3.2.2 Duration package

The duration package provides the ability to control automatically the times at which a managed object starts and stops functioning. This package shall be present if such functionality is required. If this package and one of the other scheduling packages are co-resident then the availability status shall take the value {offDuty} unless both packages signal activity, in which case the availability status shall be {}.

8.1.3.2.3 Daily scheduling package

The daily scheduling package provides the ability to schedule operation with a period of 24 hours. It shall be present if such functionality is required. It shall not be co-resident with either the weekly or external scheduler scheduling package.

8.1.3.2.4 Weekly scheduling package

The weekly scheduling package provides the ability to schedule operation with a period of one week. It shall be present if such functionality is required. It shall not be co-resident with either the daily or external scheduler scheduling package.

8.1.3.2.5 External scheduler scheduling package

The external scheduler scheduling package provides the capability of scheduling operation using a schedule defined in another managed object. It shall be present if such functionality is required. It shall not be co-resident with either the daily or weekly scheduling package.

8.1.3.3 State conditions package

The state conditions package enables a rule to operate within the context of the state of identified managed objects. It shall be present if such functionality is required.

8.1.3.3.1 Attributes of the state conditions package

The following mandatory attribute is defined for the state conditions package.

8.1.3.3.1.1 State conditions

This set-valued attribute identifies managed objects and associated filters upon the attributes of those managed objects.

8.1.3.4 Authentication context package

When present in a rule managed object the authentication context package specifies the authentication policy identity and authentication requirements that an initiator is required to satisfy.

8.1.3.4.1 Attributes of the authentication context package

The following mandatory attribute is defined for the authentication context package.

8.1.3.4.1.1 Authentication context

The authentication context attribute is a sequence of authentication policy identifier and the requirements identified thereby.

8.1.4 Notification emitter

This managed object class is used to enable notifications to be emitted that are applicable to the provision of access control for OSI management. The types of notification that are applicable include certain security alarms as defined in CCITT Rec. X.736 | ISO/IEC 10164-7, and certain security audit trail notifications as defined in CCITT Rec. X.740 | ISO/IEC 10164-8.

A single notification emitter managed object may be contained within an access control rules managed object.

8.1.4.1 Packages of notification emitter

This managed object class supports the following conditional packages to provide a flexible ability to transmit access control related notifications:

- security violation alarm package;
- time violation alarm package;
- operational violation alarm package;
- access control usage package; and
- access control service report package.

8.1.4.1.1 Security violation alarm package

This package enables a security alarm notification of type 'Security service or mechanism violation' and cause 'unauthorized access attempt' to be emitted if access control checks should fail.

8.1.4.1.2 Time violation alarm package

This package enables a security alarm notification of type 'Time domain violation' and causes 'Key expired' and 'out of hours activity' to be emitted if access control checks should fail. The cause 'key expired' shall be used when the key identified by the access control certificate seal is out of date. The 'out of hours activity' cause shall be used when contextual time checks fail.

8.1.4.1.3 Operational violation alarm package

This package enables a security alarm notification of type 'operational violation' and causes 'out of service' and 'unspecified reason' to be emitted if access control checks should fail. The cause 'out of service' shall be used when the access control mechanism identified is not available. The 'unspecified reason' cause shall be used in other cases.

8.1.4.1.4 Access control usage package

This package is used to count the number of valid and invalid access attempts and to enable usage reports containing this information to be sent to a security audit trail log. The usage report is sent at a time interval defined by the security policy. The additional information field is used to convey the counter values.

8.1.4.1.4.1 Attributes of the access control usage package

The following attributes are defined for the access control usage package.

8.1.4.1.4.1.1 Valid access attempts

This attribute is used to count the number of occasions that an access control decision function has authorized an access.

8.1.4.1.4.1.2 Invalid access attempts

This attribute is used to count the number of occasions that an access control decision function has not authorized an access.

8.1.4.1.5 Access control service report package

This package allows security audit trail notifications of type 'service report' to be emitted for possible inclusion in a security audit trail log.

8.1.5 Targets

A targets managed object identifies a collection of management information that is to be subject to access control. The targets managed object class is a sub-class of the access control managed object class.

8.1.5.1 Attributes of targets

The following mandatory attributes are defined for targets managed objects.

8.1.5.1.1 Managed object classes

This set-valued attribute identifies protected managed object classes and optional associated name bindings.

8.1.5.1.2 Managed object instances

This set-valued attribute identifies protected managed objects.

8.1.5.1.3 Scope

The scope attribute identifies a scope for the selection of protected managed objects.

8.1.5.1.4 Filter

This attribute identifies a filter to be applied to managed objects identified by the other attributes of the targets managed object to determine their inclusion as a protected managed object.

8.1.5.2 Operations list package

This package provides support for the operations type attribute as an alternative to the operations managed object. It may only be included in the targets managed object if it contains no instantiation of the operations managed object.

8.1.5.2.1 Attributes of the operations list package

The following mandatory attribute is defined for the operations type package.

8.1.5.2.1.1 Operations list

This set-valued attribute identifies the operation types that are subject to the rules identifying the targets managed object containing these operations.

8.1.6 Operations

The operations managed object identifies constraints on operation types for managed objects identified by the containing targets managed object.

Operation types are the operations defined in CCITT Rec. X.720 | ISO/IEC 10165-1, including:

- action;
- create;
- delete;
- get;
- replace;
- add member;
- remove member;
- replace with default;
- filter, and;
- multiple object selection.

There shall be only one operations managed object for a specific operations type contained within a targets managed object.

The operation type is used as the value of the naming attribute for the operations managed object class.

Conditional packages provide attributes to specify constraints upon the attributes and actions associated with the operation type. In addition, attributes are provided in conditional packages for the purpose of placing constraints on the value of the scope and synchronization parameter values that are allowed in an access request involving multiple object selection.

NOTE – It may be required to apply different constraints on the same operation type, or sub-operations for the same operation type, for the same managed object(s). For example, this requirement may exist when different constraints shall be applied to different specific actions for the same managed object(s). In such cases, a new targets managed object shall be created that contains the managed object to which the constraint applies, and which contains an operations managed object that specifies the new constraints to be applied for the operation type.

8.1.6.1 Attributes of operations

The following mandatory attribute is defined for the operations managed object class.

8.1.6.1.1 Operation type

This attribute identifies the operation type to which constraints apply, and is used for naming operations managed objects.

8.1.6.2 Notifications of operations

The following mandatory notifications are defined for the operations managed object class:

- a) object creation;
- b) object deletion;
- c) attribute value change.

8.1.6.3 Packages of operations

This managed object class supports the following conditional packages:

- attribute ids package;
- attribute modification package;
- actions package; and
- scope package.

8.1.6.3.1 Attribute ids package

The attribute ids package identifies attributes to which access is to be controlled. It shall be present if the operation type is get, replace, add value, remove value, replace with default, or filter and not present if the operation is any other type.

8.1.6.3.1.1 Attributes of the attribute ids package

The attribute identifier list attribute as specified in CCITT Rec. X.721 | ISO/IEC 10165-2 is included in this package.

8.1.6.3.2 Attribute modification package

The attribute modification package identifies constraints upon the modification of attribute values. It shall be present if the operation type is replace, add value, remove value, or create and not present if the operation is any other type.

8.1.6.3.2.1 Attributes of the attribute modification package

The attribute filter list attribute is included in this package.

8.1.6.3.2.1.1 Attribute filter list

This set-valued attribute identifies constraints upon the value of attributes in an operation request by means of a set of CMIS filters (one CMIS filter for each attribute for which constraints are specified).

8.1.6.3.3 Actions package

The actions package identifies constraints upon the values of action information. It shall be present if the operation type is action and not present if the operation is any other type.

8.1.6.3.3.1 Attributes of the actions package

The action filter list attribute is included in this package.

8.1.6.3.3.1.1 Action filter list

This set-valued attribute identifies actions and, optionally, constraints upon their argument values by means of a CMIS filter.

8.1.6.3.4 Scope package

The scope package identifies constraints upon the scope and synchronization parameters of management operations involving multiple object selection. It shall be present if the operation type is multiple object selection and not present if the operation is any other type.

NOTE – The scope package may be used, for example:

- to prevent discovery of, or access to, managed objects during a scoped get of an entire sub-tree – such as may be used to explore a system in an unauthorized manner;
- to protect given managed objects from deletion as part of a sub-tree, such that the managed object may be deleted by a certain initiator only if directly addressed, never as part of a scoped delete.

8.1.6.3.4.1 Attributes of the scope package

The following attributes are defined for the scope package.

8.1.6.3.4.1.1 Scope filter

For requests that select multiple managed objects the scope filter specifies constraints on the scope parameter of the request, and the scope attribute (see 8.1.5.1.3) identifier is used for all the filter items in the filter.

If the scope filter contains no filter items, then all possible values of the scope parameter shall be regarded as targets.

8.1.6.3.4.1.2 Synchronization filter

For requests that select multiple managed objects the synchronization filter specifies constraints on the synchronization parameter of the request and the synchronization attribute (see 8.4.2) identifier is used for all the filter items in the filter. If the synchronization filter contains no filter items, then all possible values of the synchronization parameter shall be regarded as targets.

8.1.7 Initiators

The Initiators managed object class identifies the permissible initiators of management operations.

8.1.7.1 Attributes of initiators

The following mandatory attribute is defined for the initiators managed object class.

8.1.7.1.1 Initiator ACI mandated

The attribute is used to indicate whether, to satisfy the access control scheme in use, initiator ACI is required with each individual management operation request.

8.1.8 ACL initiators

The ACL initiators managed object class contains a list of names or other identities that together form an access control list.

Multiple ACL initiators managed objects may be instantiated within a rule managed object.

8.1.8.1 Attributes of ACL initiators

The following mandatory attribute is defined for the ACL initiators managed object class.

8.1.8.1.1 Access control list

The access control list attribute is used to contain identities of initiators that are either specifically granted access to management information or specifically denied access to management information.

8.1.9 Capability initiators

The capability initiators managed object class contains a list of identities.

Multiple capability initiators managed objects may be instantiated within a rule managed object.

8.1.9.1 Attributes of capability initiators

The following mandatory attribute is defined for the capability initiators managed object class.

8.1.9.1.1 Capability identities list

The capability identities list attribute contains a set of identities.

The identities may be an individual name, group name, role name, or application name, each of which may be associated with an optional set of security domain authority name and operation type pairs; or, the identity may be of a form not specified within this Recommendation | International Standard.

8.1.10 Label initiators

The label initiators managed object may be used to specify constraints on management operations that are in addition to the constraints of requiring a compatibility match between the security label associated with the initiator and the security label associated with the target.

If a label initiators managed object is present within a rule managed object, the constraints defined by the rule managed object and by the targets managed objects contained within the same rule shall be in addition to the security label compatibility matching requirements of the label access control scheme.

If a label initiators managed object is not present within a rules managed object, then no additional constraints are imposed on the access beyond the security label compatibility matching requirements of the label access control scheme.

Multiple label initiators managed objects may be instantiated within a rule managed object.

8.1.10.1 Attributes of label initiators

The following mandatory attribute is defined for the label initiators managed object class.

8.1.10.1.1 Security label

The security label attribute contains a security label.

8.1.11 Assigned labels

This managed object is the root of the sub-tree that contains the label type managed objects which, in combination with precedence relationships, assign a single security label to targets. In addition to providing a container managed object for the different label type managed objects, this managed object provides a default security label to be assigned to management elements that are not specifically assigned a security label.

8.1.11.1 Attributes of assigned labels

The following mandatory attributes are defined for the assigned labels managed object class.

8.1.11.1.1 Label name

This attribute is used to identify instantiations and specializations of the assigned labels managed object class.

8.1.11.1.2 Security label

This attribute contains the security label to be assigned to the target.

8.1.12 Attribute label

This managed object is used to associate a single security label to targets that are specific attributes within a managed object.

8.1.12.1 Attributes of attribute label

The following mandatory attributes are defined for the attribute label managed object class.

8.1.12.1.1 Managed object instance

This attribute is used to identify a specific managed object.

8.1.12.1.2 Attribute identifier list

This attribute is used to identify specific attributes of the managed object identified by the managed object instance attribute.

8.1.13 Instance label

This managed object is used to associate a single security label to targets that are individual managed objects.

8.1.13.1 Attributes of instance label

The following mandatory attribute is defined for the instance label managed object class.

8.1.13.1.1 Managed object instances

This attribute is used to identify a list of specific targets by their managed object identifier.

8.1.14 Class label

This managed object is used to associate a single security label to targets that are managed object classes.

8.1.14.1 Attributes of class label

The following mandatory attribute is defined for the class label managed object class.

8.1.14.1.1 Managed object classes

This attribute is used to identify a list of specific managed object classes.

8.2 Parameters**8.2.1 Invalid access control filter**

This specific error reports an error in a proposed access control filter element. Its value shall be a sequence of an error id, taking one of the values duplicateId, heterogeneousId, or invalidId, and an optional CMIS Filter containing the filter in error.

8.3 Name bindings**8.3.1 Rule – Access control rule**

This name binding provides for rules and specializations to be contained within access control rules and specializations. The access control object name attribute shall be used for naming the rules. Rules managed objects may be created by management operation, with automatic instance naming and with reference object. Rules managed objects may be deleted by management operation.

8.3.2 Operations – Targets

This name binding provides for operations managed objects to be contained within targets managed objects. The operation type attribute shall be used for naming the operations. Operations managed objects may be created by management operation and with reference object. Operations managed objects may be deleted by management operation.

8.3.3 Notification emitter – Access control rule

This name binding provides for a single notification emitter managed object to be contained within an access control rules managed object. The notification emitter may be created, created with reference object and deleted by management operation. It may be named automatically.

8.3.4 Attribute label – Assigned labels

This name binding provides for attribute label managed objects to be contained within assigned label managed objects. Attribute label managed objects may be created and deleted by management operation.

8.3.5 Instance label – Assigned labels

This name binding provides for instance label managed objects to be contained within assigned label managed objects. Instance label managed objects may be created and deleted by management operation.

8.3.6 Class label – Assigned labels

This name binding provides for class label managed objects to be contained within assigned label managed objects. Class label managed objects may be created and deleted by management operation.

8.4 Attributes

The following attributes are not defined for any package or managed object class but are used for specifying other attributes and for filtering.

8.4.1 Access control filter

This set-valued attribute identifies constraints upon the value of parameters of management operations. Each element is a CMIS filter, addressed to a single element of management information. Each element of management information is addressed in at most one element of this attribute. An empty set indicates that all possible values are targeted.

8.4.2 Synchronization

This attribute value represents the synchronization parameter of management operations. It is used to represent filters upon this parameter.

8.5 Imported generic definitions

This Recommendation | International Standard makes use of the following generic definitions in CCITT Rec. X.730 | ISO/IEC 10164-1, CCITT Rec. X.731 | ISO/IEC 10164-2, CCITT Rec. X.732 | ISO/IEC 10164-3, CCITT Rec. X.734 | ISO/IEC 10164-5, CCITT Rec. X.736 | ISO/IEC 10164-7, and CCITT Rec. X.740 | ISO/IEC 10164-8:

- attribute identifier list;
- attribute value change notification;
- availability status package;
- counter;
- daily scheduling package;
- discriminator construct;
- duration package;
- external scheduler scheduling package;
- member;
- managed object instance;
- object creation notification;
- object deletion notification;
- operational violation;
- security service or mechanism violation;
- service report;
- time domain violation;
- usage report; and
- weekly scheduling package.

8.6 Compliance

Managed object class definitions support the functions defined in this Recommendation | International Standard by incorporating the specification of managed objects, attributes and notifications defined in this Recommendation | International Standard, in CCITT Rec. X.721 | ISO/IEC 10165-2, in CCITT Rec. X.736 | ISO/IEC 10164-7 and in CCITT Rec. X.740 | ISO/IEC 10164-8. The reference mechanism is defined in CCITT Rec. X.722 | ISO/IEC 10165-4.

9 Service definition

9.1 Introduction

Access control may be applied to management information. The access control applied may change with time or with changes to the access control policy. It is therefore necessary to provide a mechanism for administering the access control service.

9.2 Access control management service

This service provides for the administration of the access control rules utilized by a system.

The service enables managed objects from the following classes to be administered:

- access control rules; and
- rules.

9.2.1 Initiating access control rules

The PT-CREATE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system create instances of the managed object classes listed in 9.2.

9.2.2 Modifying access control rules

The PT-SET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system change the attribute values of attributes of the managed object classes listed in 9.2.

9.2.3 Terminating access control rules

The PT-DELETE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system delete instances of the managed object classes listed in 9.2.

9.3 Targets administration service

This service is used to administer the targets managed objects that identify the management information protected by access control.

9.3.1 Initiating access control targets

The PT-CREATE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system create instances of the targets managed object class.

9.3.2 Modifying access control targets

The PT-SET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system change the attribute values of attributes of the targets managed object class.

9.3.3 Terminating access control targets

The PT-DELETE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system delete instances of the targets managed object class.

9.4 Initiators administration service

This service is used to administer sub-classes of the initiators managed objects.

9.4.1 Initiating access control initiators

The PT-CREATE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system create instances of sub-classes of the initiators managed object class.

9.4.2 Modifying access control initiators

The PT-SET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system change the attribute values of attributes of sub-classes of the initiators managed object class.

9.4.3 Terminating access control initiators

The PT-DELETE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system delete sub-classes of instances of the initiators managed object class.

9.5 Operations administration service

This service is used to administer the operations managed objects.

9.5.1 Initiating access control operations

The PT-CREATE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system create instances of the operations managed object class.

9.5.2 Modifying access control operations

The PT-SET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system change the attribute values of attributes of the operations managed object class.

9.5.3 Terminating access control operations

The PT-DELETE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system delete instances of the operations managed object class.

9.6 Label administration service

This service is used to administer assigned labels, attribute label, instance label and class label managed objects.

9.6.1 Initiating label control operations

The PT-CREATE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system create instances of the assigned labels, attribute label, instance label and class label managed object classes.

9.6.2 Modifying access control operations

The PT-SET service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system change the attribute values of attributes of the assigned labels, attribute label, instance label and class label managed object classes.

9.6.3 Terminating access control operations

The PT-DELETE service defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to enable one open system to request that another system delete instances of the assigned labels, attribute label, instance label and object label managed object classes.

9.7 Access control notification service

This service enables reports of events relevant to the monitoring and administration of an access control service to be transmitted from one open system to another. The security alarm reporting function specified in CCITT Rec. X.736 | ISO/IEC 10164-7 is used to forward alarm reports and security audit trail records related to attempts to access management information illegally, whilst the security audit trail function specified in CCITT Rec. X.740 | ISO/IEC 10164-8 is used to forward reports related to the general use of the access control service.

10 Functional units

The objects and attributes for access control function constitutes a single systems management functional unit.

11 Protocol

11.1 Elements of procedure

This Recommendation | International Standard makes use of the elements of procedure defined in CCITT Rec. X.730 | ISO/IEC 10164-1, CCITT Rec. X.736 | ISO/IEC 10164-7, and CCITT Rec. X.740 | ISO/IEC 10164-8 for the services described in clause 9. There are no elements of procedure specific to this Recommendation | International Standard.

11.2 Abstract syntax

11.2.1 Managed objects

Table 1 identifies the relationship between access control managed objects and managed objects, whose abstract syntax is specified in Annex A.

11.2.2 Attributes

Table 2 identifies the relationship between access control attributes and management attributes whose abstract syntax is specified in Annex A.

11.2.3 Attribute groups

There are no attribute groups defined by this systems management function.

11.2.4 Actions

There are no actions defined by this systems management function.

11.2.5 Notifications

There are no notifications defined by this systems management function.

Table 1 – Managed objects

Managed object	Managed object class
Access control	accessControl
Access control rules	accessControlRules
ACL initiators	aclInitiators
Assigned labels	assignedLabels
Attribute label	attributeLabel
Capability initiators	capabilityInitiators
Class label	classLabel
Instance label	instanceLabel
Label initiators	labelInitiators
Notification emitter	notificationEmitter
Operations	operations
Rule	rule
Targets	targets

11.2.6 Parameters

Table 3 identifies the relationship between access control parameters and management parameters whose abstract syntax is specified in Annex A.

11.3 Negotiation of access control functional unit

This Specification assigns the object identifier

{ joint-iso-ccitt ms(9) function(2) part(9) functionalUnitPackage(1) }

as a value of the ASN.1 type **FunctionalUnitPackageId** defined in CCITT Rec. X.701 | ISO/IEC 10040 to use for negotiating the following functional unit:

0 access control functional unit

where the number identifies the bit position assigned to the functional unit, and the name references the functional unit as defined in clause 10.

Within the systems management application context, the mechanism for negotiating the access control functional unit is described by CCITT Rec. X.701 | ISO/IEC 10040.

NOTE – The requirement to negotiate functional units is specified by the application context.

12 Relationship with other functions

The object creation and object deletion notifications defined in CCITT Rec. X.730 | ISO/IEC 10164-1 are used to report the creation and deletion respectively, of instances of the managed object classes defined in this Recommendation | International Standard.

Table 2 – Management attributes

Access control attribute	Attribute name
Access control filter	accessControlFilter
Access control object name	accessControlObjectName
Action filter list	actionFilterList
Attribute filter list	attributeFilterList
Authentication context	authenticationContext
Capability identities list	capabilityIdentitiesList
Default access	defaultAccess
Default denial response	defaultDenialResponse
Denial granularity	denialGranularity
Domain identity	domainIdentity
Enforcement action	enforcementAction
Filter	filter
Initiator ACI mandated	initiatorACImandated
Initiators list	initiatorsList
Invalid access attempts	invalidAccessAttempts
Label name	labelName
Managed object classes	managedObjectClasses
Managed object instances	managedObjectInstances
Operation type	operationType
Operations list	operationsList
Scope	scope
Scope filter	scopeFilter
Security label	securityLabel
State conditions	stateConditions
Synchronization	synchronization
Synchronization filter	synchronizationFilter
Targets list	targetsList
Valid access attempts	validAccessAttempts

Table 3 – Management Parameters

Access Control Parameter	Parameter Name
Invalid access control filter	invalidAccessControlFilter

The attribute value change notification defined in CCITT Rec. X.730 | ISO/IEC 10164-1 is used to report changes in attribute values in instances of the managed objects defined in this Recommendation | International Standard.

The operational violation, security service or mechanism violation, and time domain violation notifications defined in CCITT Rec. X.736 | ISO/IEC 10164-7 are used to report security alarms associated with the operation of the access control mechanisms.

The service report and service usage notifications defined in CCITT Rec. X.740 | ISO/IEC 10164-8 are used to convey security audit trail reports associated with the use of access control services and mechanisms.

The management of access control information may make use of the following systems management services defined in CCITT Rec. X.730 | ISO/IEC 10164-1:

- PT-CREATE;
- PT-DELETE;
- PT-SET; and
- PT-GET.

The security policy may stipulate that these services be used over a secure association so that the ACI is protected from unwanted disclosure or modification.

13 Conformance

Implementations claiming to conform to this Recommendation | International Standard shall comply with the conformance requirements as defined in the following subclauses.

13.1 Static conformance

The implementation shall conform to the requirements of this Recommendation | International Standard in the manager role, the agent role, or both roles. A claim of conformance to at least one role shall be made in Table B.1.

If a claim of conformance is made for support in the manager role, the implementation shall support at least one management operation or notification of at least one of the managed objects specified by this Recommendation | International Standard. The conformance requirements in the manager role for those management operations are identified in Table B.3 and further tables referenced by Annex B.

If a claim of conformance is made for support in the agent role, the implementation shall support one or more instances of the access control rules managed object class identified in Table B.4.

The system shall support the transfer syntax derived from the encoding rules specified in CCITT Rec. X.209 | ISO/IEC 8825 named {joint-iso-ccitt asn1(1) basic encoding(1)} for the abstract data types referenced by the definitions for which support is claimed.

13.2 Dynamic conformance

Implementations claiming to conform to this Recommendation | International Standard shall support the elements of procedure and definitions of semantics corresponding to the definitions for which support is claimed.

13.3 Management information conformance requirements

Any MCS proforma, MICS proforma, MOCS proforma, MRCS proforma and MIDS proforma which conforms to this Recommendation | International Standard shall be technically identical to the proformas specified in Annexes B, C, D, E and F preserving table numbering and the index number of items, and differing only in pagination and page headers.

The supplier of an implementation which is claimed to conform to this Recommendation | International Standard shall complete a copy of the Management Conformance Summary (MCS) provided in Annex B as part of the conformance requirements, together with any other ICS proformas referenced as applicable from that MCS. An ICS which conforms to this Recommendation | International Standard shall:

- describe an implementation which conforms to this Recommendation | International Standard;
- have been completed in accordance with the instructions for completion given in ITU-T Rec. X.724 | ISO/IEC 10165-6;
- include the information necessary to uniquely identify both the supplier and the implementation.

Claims of conformance to the management information defined in this Recommendation | International Standard in managed object classes defined elsewhere shall include the requirements of the MIDS proforma in the MOCS proforma for the managed object class.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

Annex A

Definition of management information

(This annex forms an integral part of this Recommendation | International Standard)

A.1 Allocation of object identifiers

This Recommendation | International Standard allocates the following object identifiers

AccessControlDefinitions{ joint-iso-ccitt(2) ms(9) function(2) part9(9) asn1Module(2) 1 }

DEFINITIONS ::= BEGIN

```

accessControl-Object OBJECT IDENTIFIER ::=
    { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) }

accessControl-Package OBJECT IDENTIFIER ::=
    { joint-iso-ccitt(2) ms(9) function(2) part9(9) package (4) }

accessControl-Parameter OBJECT IDENTIFIER ::=
    { joint-iso-ccitt(2) ms(9) function(2) part9(9) parameter(5) }

accessControl-NameBinding OBJECT IDENTIFIER ::=
    { joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) }

accessControl-Attribute OBJECT IDENTIFIER ::=
    { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) }

```

END

A.2 Definition of access control managed object classes

A.2.1 Access control object class

The access control managed object class is used to provide a common naming attribute and attribute value change reporting for managed objects representing access control information. It is not intended to be instantiated.

accessControl MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": top;

CHARACTERIZED BY accessControlPackage PACKAGE

BEHAVIOUR accessControlBehaviour BEHAVIOUR

DEFINED AS

! The access control managed object class shall emit the object creation and object deletion notifications. Specializations of the access control managed object class shall define the conditions under which attribute value change notifications are to be emitted. ! ;;

ATTRIBUTES accessControlObjectName GET;

NOTIFICATIONS "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": attributeValueChange,
 "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": objectCreation,
 "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": objectDeletion;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) accessControl(1) };

A.2.2 Access control rules

The accessControlRules managed object class is used to define a representation of access control rules. One accessControlRules managed object is required per access decision function within a security domain.

accessControlRules MANAGED OBJECT CLASS

DERIVED FROM accessControl;

CHARACTERIZED BY accessControlRulesPackage PACKAGE

BEHAVIOUR accessControlRulesBehaviour BEHAVIOUR

DEFINED AS

! An access control rules managed object may contain rule managed objects, each of which represents a global or an item rule. It shall use those rules in the application of the procedures of 7.4 in accordance with the policy of the access control domain.

An attribute value change notification shall be emitted when any attribute of this object class is modified.

NOTE – An access control rules managed object may contain rule managed objects which are in conflict for a given initiator, target pair. The procedures of 7.4.3.1 ensure that the principle of least privilege applies.

!;;

ATTRIBUTES

defaultAccess	REPLACE-WITH-DEFAULT DEFAULT VALUE AccessControl-ASN1Module.denyAll
domainIdentity	GET-REPLACE,
denialGranularity	GET-REPLACE,
defaultDenialResponse	GET-REPLACE;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) accessControlRules(2) };

A.2.3 Rule

rule MANAGED OBJECT CLASS

DERIVED FROM accessControl;

CHARACTERIZED BY rulePackage PACKAGE

BEHAVIOUR ruleBehaviour BEHAVIOUR

DEFINED AS

! Each rule identifies its nature - to grant or deny access. In the case where the enforcement action attribute has a value of allow, then access is permitted, else the enforcement action attribute defines the type of denial response made to the initiator of the management operation.

A rule managed object may include characteristics to represent a context for the rule.

One such context is a scheduling capability. When included, the scheduling packages control the value of the availability status attribute which shall exhibit the value { off duty } when the schedule requires that the rule not be available and the value {} otherwise.

Another context is the state of other managed objects. When included, the state conditions package identifies managed objects and filters upon their attributes. This rule shall only pertain if the managed objects exist and the filters evaluate to TRUE.

The initiator list attribute identifies initiator managed objects which identify initiators within the context of one or more access control schemes. If the list is empty, the rule shall apply to all initiators.

The targets list attribute identifies the target managed objects which specify the targets to which the rule pertains. If the list is empty, the rule is a global rule otherwise it is an item rule.

The creation and deletion of rules shall be signalled by object creation and object deletion notifications respectively.

An attribute value change notification shall be emitted when any attribute of this object class is modified. !;;

ATTRIBUTES

enforcementAction	REPLACE-WITH-DEFAULT DEFAULT VALUE AccessControl-ASN1Module.denyAccess
initiatorsList	GET-REPLACE,
targetsList	GET-REPLACE ADD-REMOVE, GET-REPLACE ADD-REMOVE;;;

CONDITIONAL PACKAGES

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": availabilityStatusPackage

PRESENT IF ! Any of the scheduling packages (duration, daily, weekly, external) are present. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": duration

PRESENT IF ! The object is to be available from a specified start time, indefinitely or until a specified stop time. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": dailyScheduling

PRESENT IF ! Both the weekly scheduling package and external scheduler package are not present and daily scheduling is supported. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": weeklyScheduling

PRESENT IF ! Both the daily scheduling package and external scheduler package are not present and weekly scheduling is supported. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": externalScheduler

PRESENT IF ! Both the daily scheduling package and weekly scheduling package are not present and external scheduling is supported. !,

stateConditionsPackage PACKAGE

BEHAVIOUR stateConditionsBehaviour BEHAVIOUR

DEFINED AS

! When this package is present in a rule managed object, the filters identified by the state conditions attribute shall be evaluated for the managed objects identified by that attribute. If the managed objects are not available or the filters evaluates to FALSE then the rule shall evaluate to FALSE. If the filters evaluate to TRUE, then the rule shall evaluate to TRUE. ! ;;

ATTRIBUTES stateConditions GET-REPLACE ADD-REMOVE;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) stateConditionsPackage(1) };

PRESENT IF ! The state of another managed object provides a context for this rule. !,

authenticationContextPackage PACKAGE

BEHAVIOUR authenticationContextBehaviour BEHAVIOUR

DEFINED AS

! When this package is present in a rule managed object, then the authentication requirements specified by the authentication context attribute shall be satisfied before any further evaluation of the access rights of an initiator is performed.

If the authentication requirements are not satisfied, then the rule shall evaluate to FALSE.

!;;

ATTRIBUTES authenticationContext GET-REPLACE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) (2) };

PRESENT IF ! The authentication context is required. !;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) rule(3) };

A.2.4 Notification emitter

notificationEmitter MANAGED OBJECT CLASS

DERIVED FROM accessControl;

CHARACTERIZED BY accessControlNotificationEmitterPkg PACKAGE

BEHAVIOUR accessControlNotificationEmitterDefinition BEHAVIOUR

DEFINED AS

! This managed object class enables an access control scheme to report on potential or actual attacks on the security of management applications and management information. An instance of this managed object class shall support at least one of the conditional packages defined below. ! ;;;

CONDITIONAL PACKAGES

securityViolationAlarmPkg PACKAGE

BEHAVIOUR securityViolationAlarmBehaviour BEHAVIOUR

DEFINED AS

! This package enables a security alarm notification of type 'Security service or mechanism violation' and cause 'unauthorized access attempt' to be emitted if access control checks should fail. ! ;;

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2:1992": securityServiceOrMechanismViolation;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) securityViolationAlarmPkg(3) };

PRESENT IF ! the security policy requires that this security alarm type shall be emitted if the access control checks fail. !,

timeViolationAlarmPkg PACKAGE

BEHAVIOUR timeViolationAlarmBehaviour BEHAVIOUR

DEFINED AS

! This package enables a security alarm notification of type 'Time domain violation' and causes 'Key expired' and 'out of hours activity' to be emitted if access control checks should fail. The cause 'key expired' shall be used when the key identified by the access control certificate seal is out of date. The 'out of hours activity' cause shall be used when contextual time checks fail. ! ;;

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2:1992": timeDomainViolation;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) timeViolationAlarmPkg(4) };

PRESENT IF ! the security policy requires that this security alarm type shall be emitted when either out of hours activity is detected or an expired key has been used. !,

operationalViolationAlarmPkg PACKAGE

BEHAVIOUR operationalViolationAlarmBehaviour BEHAVIOUR

DEFINED AS

! This package enables a security alarm notification of type 'operational violation' and causes 'out of service' and 'unspecified reason' to be emitted if access control checks should fail. The cause 'out of service' shall be used when the access control mechanism identified is not available. The 'unspecified reason' cause shall be used in other cases. !;;

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2:1992": operationalViolation;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) operationalViolationAlarmPkg(5) };

PRESENT IF ! the security policy requires that this security alarm type shall be emitted when either the access control mechanism is unavailable or the security policy identifies further causes. !;

accessControlUsagePkg PACKAGE

BEHAVIOUR accessControlUsagePkgBehaviour BEHAVIOUR

DEFINED AS

! This package is used to count the number of valid and invalid access attempts and to enable usage reports containing this information to be sent to a security audit trail log. The usage report is sent at a time interval defined by the security policy. The additional information field is used to convey the counter values. !;;

ATTRIBUTES

validAccessAttempts,
invalidAccessAttempts;

NOTIFICATIONS

"Rec. X.740 | ISO/IEC 10164-8:1992": usageReport;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) accessControlUsagePkg(6) };

PRESENT IF ! the security policy requires that the number of valid and invalid access attempts are logged. !;

accessControlServiceReportPkg PACKAGE

BEHAVIOUR accessControlServiceReportPkgBehaviour BEHAVIOUR

DEFINED AS

! This package allows security audit trail notifications of type 'service report' to be emitted for possible inclusion in a security audit trail log. !;;

NOTIFICATIONS

"Rec. X.740 | ISO/IEC 10164-8:1992": serviceReport;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) accessControlServiceReportPkg(7) };

PRESENT IF ! the security policy requires that service reports are logged. !;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) notificationEmitter(4) };

A.2.5 Targets

The targets managed object class is used to identify managed objects to which access is controlled.

targets MANAGED OBJECT CLASS

DERIVED FROM accessControl;

CHARACTERIZED BY targetsPackage PACKAGE

BEHAVIOUR targetsBehaviour BEHAVIOUR

DEFINED AS

! Targets identify managed objects within the security domain. These managed objects are identified according to the following rules:

- all managed objects within the security domain and belonging to the managed object classes identified by the managed object classes attribute are identified with specified name bindings;
- all managed objects within the security domain identified explicitly by the managed object instances attribute are identified;
- each managed object selected according to a) and b) shall be regarded as a base managed object for selecting managed objects according to the scope and filter attributes; and
- all managed objects selected according to c) shall be regarded as the target managed objects.

Unless the targets managed object contains operations managed objects, the targets managed object identifies all operations upon the selected managed objects.

An attribute value change notification shall be emitted when any attribute of this managed object is modified. !;;

ATTRIBUTES

managedObjectClasses	GET-REPLACE ADD-REMOVE,
managedObjectInstances	GET-REPLACE ADD-REMOVE,
scope	GET-REPLACE,
filter	GET-REPLACE;;;

CONDITIONAL PACKAGES**operationsListPackage PACKAGE****BEHAVIOUR operationsListPackBehav BEHAVIOUR****DEFINED AS**

! This package provides support for the operations list attribute as an alternative to the operations managed object. It may only be included in the targets managed object if the targets managed object contains no instantiation of the operations managed object.

! ;;

ATTRIBUTES

operationsList GET-REPLACE ADD-REMOVE;;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) operationsListPackage(15) }

PRESENT IF ! No contained Operations object !**REGISTERED AS** { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) targets(5) };**A.2.6 Operations**

Instantiations of the operations managed object identify which operations may be carried out on the specified management information (identified by the targets managed object).

operations MANAGED OBJECT CLASS**DERIVED FROM** "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": top;**CHARACTERIZED BY** operationsPackage PACKAGE**BEHAVIOUR** operationsBehaviour BEHAVIOUR**DEFINED AS**

! The operations managed object identifies constraints on operation types for managed objects identified by the containing targets managed object.

The operation type is specified by the operation type attribute, which is also the naming attribute for the operations managed object class.

The constraints on the operation type, some of which are peculiar to the operation type, are specified by other attributes contained in conditional packages.

When a target managed object identifies the managed object specified in the access request, and contains one or more operations managed objects, then an access request shall satisfy the following conditions for the containing rule to be satisfied:

- the access request matches the operation type for one of the operations managed objects contained in the target; and
- the constraints specified for the operation type are satisfied.

The operations managed object shall emit the object creation notification when it is created and the object deletion notification when it is deleted. An attribute value change notification shall be emitted when any attribute of this managed object class is modified. !;;

ATTRIBUTES

operationType GET;

NOTIFICATIONS

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": attributeValueChange,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": objectCreation,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": objectDeletion;;;

CONDITIONAL PACKAGES**attributeIdsPackage PACKAGE****BEHAVIOUR** attributeIdsBehaviour BEHAVIOUR**DEFINED AS**

! The attributes identified by the attribute identifier list attribute shall be part of the target. If the attribute identifier list attribute is empty, then all attributes shall be part of the target for the identified operation for the managed objects identified by the containing targets managed object. ! ;;

ATTRIBUTES "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": attributeIdentifierList

GET-REPLACE ADD-REMOVE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) attributeIdsPackage(8) };**PRESENT IF** ! operation type is get, replace with default or filter !,

attributeModificationPackage PACKAGE

BEHAVIOUR attributeModificationBehaviour BEHAVIOUR

DEFINED AS

! The attribute values identified by the attribute filter list attribute shall be part of the target. If the attribute filter list attribute is empty, then all attributes and their values shall be part of the target for the identified operation for the managed objects identified by the containing targets managed object. If the attribute filter list attribute identifies an attribute without constraining its value, then all values of that attribute shall be part of the target for the identified operation for the managed objects identified by the containing targets managed object. ! ;;

ATTRIBUTES

attributeFilterList GET-REPLACE ADD-REMOVE;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) attributeModificationPackage(9) };

PRESENT IF ! operation type is replace, add, remove or create !,

actionsPackage PACKAGE

BEHAVIOUR actionsBehaviour BEHAVIOUR

DEFINED AS

! The action values identified by the action filter list attribute shall be part of the target. If the action filter list attribute is empty, then all actions and their information values shall be part of the target for the identified operation for the managed objects identified by the containing targets managed object. If the action filter list attribute identifies an action without constraining its information value, then all values of that action information shall be part of the target for the identified operation for the managed objects identified by the containing targets managed object.

NOTE – For the purposes of filtering, parameters of actions may be identified as attributes using the parameter template defined in CCITT Rec. X.722 | ISO/IEC 10165-4.

! ;;

ATTRIBUTES

actionFilterList GET-REPLACE ADD-REMOVE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) actionsPackage(10) };

PRESENT IF ! operation type is action !,

scopePackage PACKAGE

BEHAVIOUR scopeBehaviour BEHAVIOUR

DEFINED AS

! The scope and synchronization values identified by the scope and synchronization attributes shall be part of the target. ! ;;

ATTRIBUTES

scopeFilter GET-REPLACE,

synchronizationFilter GET-REPLACE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) scopePackage(11) };

PRESENT IF ! operation type is multiple object selection !;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) operations(6) };

A.2.7 Initiators

The initiators managed object class is used to identify a set of possible operation requestors. The precise means of identifying initiators depends upon the security policy. The initiators managed object class is not intended to be instantiated but is intended to be specialized to enable operation requestors to be identified in accordance with a given security policy. It is recommended that all packages which provide attributes to identify requestors be registered so that the packages attribute may be used to identify the policy.

initiators MANAGED OBJECT CLASS

DERIVED FROM accessControl;

CHARACTERIZED BY initiatorsPackage PACKAGE

BEHAVIOUR initiatorsBehaviour BEHAVIOUR

DEFINED AS

! Initiators identify individual requestors of management operations in accordance with the applicable access control schemes. The diversity of possible schemes prohibits a single representation of initiators. Specializations of the initiators managed object class provide attributes to identify requestors in accordance with given access control schemes.

Where a specialization identifies more than one access control scheme, it shall also contain behaviour to resolve conflicts of rights associated with the different schemes. ! ;;

ATTRIBUTES

initiatorACImandated **REPLACE-WITH-DEFAULT**
DEFAULT VALUE AccessControl-ASN1Module.false
GET-REPLACE;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) initiators(7) };

A.2.8 ACL initiators**aclInitiators MANAGED OBJECT CLASS**

DERIVED FROM initiators;
CHARACTERIZED BY aclPackage **PACKAGE**
BEHAVIOUR aclInitiatorsBehaviour **BEHAVIOUR**
DEFINED AS

! This managed object class is used to support an ACL based access control scheme.

The ACL initiators managed object class contains a list of names or other identities that together form an access control list. The identity of a management operation requestor shall be matched with the entries of an access control list to evaluate whether the requestor is an authorized initiator.

Multiple ACL initiators managed objects may be instantiated within a rule managed object.

An attribute value change notification shall be emitted when any attribute of this object class is modified. !;;

ATTRIBUTES

accessControlList **GET-REPLACE ADD-REMOVE;**
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) aclPackage(12) };;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) aclInitiators(8) };

A.2.9 Capability initiators**capabilityInitiators MANAGED OBJECT CLASS**

DERIVED FROM initiators;
CHARACTERIZED BY capabilityPackage **PACKAGE**
BEHAVIOUR capabilityInitiatorsBehaviour **BEHAVIOUR**
DEFINED AS

! The capability initiators managed object class contains a list of identities that are used to determine whether the security capability associated with the access request is allowed to be used by the initiator of the request.

The identity associated with the access request is matched with the contents of the capability identity list attribute to evaluate whether the security capability associated with the access request is allowed to be used by the initiator of the request.

The identities may be an individual name, group name, role name, or application name which may be associated with an optional set of security domain authority name and operation type pairs; or, the identity may be of a form unspecified within this Recommendation | International Standard.

NOTE – When a capability scheme is used, rule managed objects that specify deny permission are not required. The absence of the identity in the capability identities list attribute results in the capability not being valid. In addition, targets managed objects and associated operations managed objects are not required, unless further access constraints are required to enforce local security policy refinements of the containing security domain policy.

An attribute value change notification shall be emitted when any attribute of this object class is modified. !;;

ATTRIBUTES

capabilityIdentitiesList **GET-REPLACE;**
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) capabilityPackage(13) };;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) capabilityInitiators(9) };

A.2.10 Label initiators**labelInitiators MANAGED OBJECT CLASS**

DERIVED FROM initiators;
CHARACTERIZED BY labelPackage **PACKAGE**
BEHAVIOUR labelInitiatorsBehaviour **BEHAVIOUR**
DEFINED AS

! The labels initiators managed object may be used to specify constraints on management operations that are in addition to the constraint of requiring a compatibility match between the security label associated with the initiator and the security label associated with the target.

Access shall be granted or denied to an initiator in accordance with the containing rule only if the initiator's security label is a member of the set of security labels identified by the security label attribute, the operation on the target conforms to the conditions specified by the relevant targets managed object and operations managed objects associated with the rule, and the security label of the initiator is compatible with the security label assigned to the target.

NOTE – Association of a security label with a target must have occurred prior to the use of that label in the above procedure. Security labels are associated with targets using the assigned labels, attribute label, instance label, and class label managed objects and associated procedures described in 7.4.

An attribute value change notification shall be emitted when any attribute of this object class is modified. !;;

ATTRIBUTES

securityLabel GET-REPLACE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) labelPackage(14) };;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) labelInitiators(10) };

A.2.11 Assigned labels

assignedLabels MANAGED OBJECT CLASS

DERIVED FROM top;

CHARACTERIZED BY assignedLabelsPackage PACKAGE

BEHAVIOUR assignedLabelsPkgBehav BEHAVIOUR

DEFINED AS

! This managed object contains the attribute label, instance label and class label managed objects that, in combination with precedence relationships, assign a single security label to targets.

There shall be only one managed object of this class per access control decision function.

To assure association of a single security label with a target, a precedence relationship is specified between and within attribute label, instance label and class label managed objects classes as follows:

- Between class precedence relationships

Attribute label managed object > instance label managed object > object label managed object

- Within class precedence relationships.

All attribute label, instance label, and class label managed objects shall be considered to be ordered within their respective managed object class according to the value of the naming attribute for the managed object.

The value of the security label attribute within the attribute label, instance label, or class label managed object which references the target, either directly or indirectly, has the greatest class precedence, and is first in the lexicographical order within the class, shall be associated with the target.

If a security label is not associated with a target by an attribute label, instance label, or class label managed object, the default security label contained in the security label attribute of this managed object shall be associated with the target.

The assigned labels managed object class shall emit the object creation notification when a managed object of this class is created, and shall emit the object deletion notification when a managed object of this class is deleted. An attribute value change notification shall be emitted when any attribute of this managed object class is modified. !;;

ATTRIBUTES

labelName GET,

securityLabel GET;

NOTIFICATIONS

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": attributeValueChange,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": objectCreation,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": objectDeletion;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) assignedLabels(11) };

A.2.12 Attribute label

attributeLabel MANAGED OBJECT CLASS

DERIVED FROM assignedLabels;

CHARACTERIZED BY attributeLabelPackage PACKAGE

BEHAVIOUR attributeLabelPkgBehav BEHAVIOUR

DEFINED AS

! This managed object associates a security label with specific attributes within a managed object.

The security label is the value contained in the security label attribute.

The attributes are the values contained in the attribute identifier list attribute.

The managed object is the value contained in the managed object instance attribute.

There may be multiple managed objects of this class contained within an assigned labels managed object.

The behaviour of attribute label managed objects relative to others within its class, and managed objects within the instance label and class label managed object classes, shall be as defined in the assigned labels managed object behaviour. ! ;;

ATTRIBUTES

"CCITT Rec. X.721 | ISO 10165-2:1992": managedObjectInstance GET,

"CCITT Rec. X.721 | ISO 10165-2:1992": attributeIdentifierList GET;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) attributeLabel(12) };

A.2.13 Instance label

instanceLabel MANAGED OBJECT CLASS

DERIVED FROM assignedLabels;

CHARACTERIZED BY instanceLabelPackage PACKAGE

BEHAVIOUR instanceLabelPkgBehav BEHAVIOUR

DEFINED AS

! This managed object associates a security label with specific managed objects.

The security label is the value contained in the security label attribute.

The managed object identifiers are contained in the managed object instances attribute.

There may be multiple managed objects of this class contained within an assigned labels managed object.

The behaviour of instance label managed objects relative to others within its class, and managed objects within the attribute label and class label managed object classes, shall be as defined in the assigned labels managed object behaviour. ! ;;

ATTRIBUTES

managedObjectInstances GET;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) instanceLabel(13) };

A.2.14 Class label

classLabel MANAGED OBJECT CLASS

DERIVED FROM assignedLabels;

CHARACTERIZED BY classLabelPackage PACKAGE

BEHAVIOUR classLabelPkgBehav BEHAVIOUR

DEFINED AS

! This managed object associates a security label with specific managed object classes.

The security label is the value contained in the security label attribute.

The managed object class identifiers are contained in the managed object classes attribute.

There may be multiple managed objects of this class contained within an assigned labels managed object.

The behaviour of class label managed objects relative to others within its class, and managed objects within the attribute label and instance label managed object classes, shall be as defined in the assigned labels managed object behaviour. ! ;;

ATTRIBUTES

managedObjectClasses GET;;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) classLabel(14) };

A.3 Definition of name bindings

A.3.1 Rule – Access control rule

rule-accessControlRules NAME BINDING

SUBORDINATE OBJECT CLASS rule AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS accessControlRules AND SUBCLASSES;

WITH ATTRIBUTE accessControlObjectName;

CREATE WITH-AUTOMATIC-INSTANCE-NAMING, WITH-REFERENCE-OBJECT;
DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) rule-accessControlRules(1) };

A.3.2 Operations – Targets

operations-targets NAME BINDING

SUBORDINATE OBJECT CLASS operations AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS targets AND SUBCLASSES;

WITH ATTRIBUTE operationType;

CREATE WITH-REFERENCE-OBJECT;

DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) operations-targets(2) };

A.3.3 Notification emitter – Access control rules

notificationEmitter-accessControlRules NAME BINDING

SUBORDINATE OBJECT CLASS notificationEmitter AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS accessControlRules AND SUBCLASSES;

WITH ATTRIBUTE accessControlObjectName;

CREATE WITH-AUTOMATIC-INSTANCE-NAMING;

DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS

{ joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) notificationEmitter-accessControlRules(3) };

A.3.4 Attribute label – Assigned labels

attributeLabel-assignedLabels NAME BINDING

SUBORDINATE OBJECT CLASS attributeLabel AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS assignedLabels AND SUBCLASSES;

WITH ATTRIBUTE labelName;

CREATE;

DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) attributeLabel-assignedLabels(4) };

A.3.5 Instance label – Assigned labels

instanceLabel-assignedLabels NAME BINDING

SUBORDINATE OBJECT CLASS instanceLabel AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS assignedLabels AND SUBCLASSES;

WITH ATTRIBUTE labelName;

CREATE;

DELETE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) instanceLabel-assignedLabels(5) };

A.3.6 Class label – Assigned labels

classLabel-assignedLabels NAME BINDING

SUBORDINATE OBJECT CLASS classLabel AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS assignedLabels AND SUBCLASSES;

WITH ATTRIBUTE labelName;

CREATE;

DELETE;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) nameBinding(6) classLabel-assignedLabels(6) };

A.4 Definition of parameters

A.4.1 Invalid access control filter

invalidAccessControlFilter PARAMETER

CONTEXT SPECIFIC-ERROR;

WITH SYNTAX AccessControlDefinitions.InvalidAccessControlFilter;

BEHAVIOUR invalidAccessControlFilterBehaviour BEHAVIOUR

DEFINED AS

! This CMIS processing failure specific error reports an error in a proposed access control filter element. Its value shall be a sequence of an error id, taking one of the values duplicateId, heterogeneousId, or invalidId, and an optional CMIS Filter containing the filter in error. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) parameter(5) invalidAccessControlFilter(1) };

A.5 Definition of attributes**A.5.1 Access control list**

accessControlList ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.AccessControlList;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR aclBehaviour **BEHAVIOUR**

DEFINED AS

! This attribute is used to specify a list of initiators for use in an access control list based scheme. Initiators are identified by individual name, anonymous reference or by group name, roles or application entity titles. Initiators may be associated with specified applications. Individual group names may be used in conjunction with the OSI Directory.

The attribute enables either an initiator name or a proxy name to be used. The initiator name form may be syntactically either a distinguished name or an application entity title, whilst the proxy name takes the form of an object identifier and value.

The distinguished name form may be used either to identify a specific initiator, a group of initiators or a particular role.

The application entity title name form identifies the application entity title, and by reference the system that initiated the request.

The proxy name form is used when the name form is not a specific initiator, a group of initiators, a role or an application entity title. The proxy therefore allows the initiator to be anonymous. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) accessControlList(1) };

A.5.2 Access control filter

The following attribute is defined for inheritance purposes.

accessControlFilter ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.FilterList;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR accessControlFilterBehaviour **BEHAVIOUR**

DEFINED AS

! This set-valued attribute provides a set of CMIS filters for constraining the parameters of management operations. If the set is empty, the CMIS filter shall be regarded as identifying all possible targets identifiable by the derived attribute.

For any given CMIS filter of the set, every CMIS filter item shall identify the same attribute. Attempts to violate this constraint shall result in the invalid access control filter specific error with error identifier of heterogeneousIds.

No attribute shall be associated with more than one CMIS filter. Attempts to violate this constraint shall result in the invalid access control filter specific error with error identifier of duplicateIds.

All values of the attribute identifier fields of CMIS filter items shall identify management information that is valid for the given specialization of this attribute. Any violation shall result in the invalid access control filter specific error with the error identifier of invalid identifier. ! ;;

PARAMETERS invalidAccessControlFilter;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) accessControlFilter(2) };

A.5.3 Access control object name

accessControlObjectName ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.AccessControlObjectName;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR accessControlObjectNameBehaviour **BEHAVIOUR**

DEFINED AS

! This attribute is used to identify instantiations of specializations of the access control managed object class. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) accessControlObjectName(3) };

A.5.4 Action filter list

actionFilterList ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.ActionFilterList;
MATCHES FOR EQUALITY, SET-INTERSECTION, SET-COMPARISON;
BEHAVIOUR actionFilterlistBehaviour BEHAVIOUR
DEFINED AS

! This set-valued attribute identifies actions and, optionally, constraints upon their argument values by means of a CMIS filter.

For any given CMIS filter of the set, every CMIS filter item shall identify the same attribute. Attempts to violate this constraint shall result in the invalid access control filter specific error with error identifier of heterogeneousIds.

No attribute shall be associated with more than one CMIS filter. Attempts to violate this constraint shall result in the invalid access control filter specific error with error identifier of duplicateIds.

All values of the attribute identifier fields of CMIS filter items shall identify management information that is valid for the given specialization of this attribute. Any violation shall result in the invalid access control filter specific error with the error identifier of invalid identifier. !;;

PARAMETERS invalidAccesscontrolFilter;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) actionFilterList(4) };

A.5.5 Attribute filter list

attributeFilterList ATTRIBUTE

DERIVED FROM accessControlFilter;
BEHAVIOUR attributeFilterListBehaviour BEHAVIOUR
DEFINED AS

! This attribute identifies constraints upon the values of attributes.

If an attribute is identified without constraints upon its value e.g.

{ item : present : globalForm : accessControlList }

Then all values of the attribute are identified.

If the set is empty, then there are no constraints. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) attributeFilterList(5) };

A.5.6 Authentication context

authenticationContext ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.AuthenticationContext;
BEHAVIOUR authenticationContextPackageBehaviour BEHAVIOUR
DEFINED AS

! The authentication context attribute is a sequence of authentication policy identifier and the requirements identified thereby. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) authenticationContext(6) };

A.5.7 Capability identities list

capabilityIdentitiesList ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.CapabilityIdentitiesList;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR capabilityBehaviour BEHAVIOUR
DEFINED AS

! The capability identities list attribute contains a set of identities.

The identities may be an individual name, group name, role name, or application name, each of which may be associated with an optional set of security domain authority name and operation type pairs; or, the identity may be of a form unspecified within this Recommendation | International Standard. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) capabilityIdentitiesList(7) };

A.5.8 Default access

defaultAccess ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DefaultAccess;
MATCHES FOR EQUALITY;
BEHAVIOUR defaultAccessBehaviour BEHAVIOUR
DEFINED AS

! The default access attribute identifies, in accordance with 7.4.3.1.6, the default access rights for each operation type. Its value is a sequence enumerating the enforcement action for each operation type. The default value of the attribute shall be to deny all operations with the access denied response. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) defaultAccess(8) };

A.5.9 Default denial response

defaultDenialResponse ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DenialResponse;

MATCHES FOR EQUALITY;

BEHAVIOUR denialResponseBehaviour BEHAVIOUR

DEFINED AS

! This attribute defines the denial response to be returned in the event that the denial has been made as a result of the default rule having been satisfied. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) defaultDenialResponse(9) };

A.5.10 Denial granularity

denialGranularity ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DenialGranularity;

MATCHES FOR EQUALITY;

BEHAVIOUR denialGranularityBehaviour BEHAVIOUR

DEFINED AS

! This attribute identifies the level at which denial of access shall be exhibited, if at all. It shall take one of the values request, object, and attribute. If the value is request, then the entire request shall be denied if any target in that request is denied. If the value is object, then the request for that managed object shall be denied if any target within the request for that object is denied. If the value is attribute, then the request shall be denied at the attribute level. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) denialGranularity(10) };

A.5.11 Domain identity

domainIdentity ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DomainIdentity;

MATCHES FOR EQUALITY;

BEHAVIOUR domainNameBehaviour BEHAVIOUR

DEFINED AS

! This attribute identifies the access control domain governing these access control rules. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) domainIdentity(11) };

A.5.12 Enforcement action

enforcementAction ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.EnforcementAction;

MATCHES FOR EQUALITY;

BEHAVIOUR enforcementActionBehaviour BEHAVIOUR

DEFINED AS

! This attribute identifies the action to be taken by the enforcement function if the rule is satisfied. It shall take one of the values, deny with response (the default value), deny without response, abort association, deny with false response and allow. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) enforcementAction(12) };

A.5.13 Filter

filter ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": discriminatorConstruct;

BEHAVIOUR filterBehaviour BEHAVIOUR

DEFINED AS

! This attribute identifies a filter to be applied to managed objects identified by the other attributes of the targets managed object to determine their inclusion as a protected managed object. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) filter(13) };

A.5.14 Initiator ACI mandated

initiatorACImandated ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.Boolean;

MATCHES FOR EQUALITY;

BEHAVIOUR initiatorACImandatedBehaviour BEHAVIOUR

DEFINED AS

! The initiator ACI mandated attribute is of type boolean. The attribute is used to indicate whether, to satisfy the access control scheme in use, initiator ACI is required with each individual management operation request. An attribute value of TRUE indicates that initiator ACI is required in each management operation request, whilst a value of FALSE indicates that no initiator ACI is required. In

the event that the attribute has a value of TRUE and the management operation request does not contain initiator ACI, then access will be denied. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) initiatorACImandated(14) };

A.5.15 Initiators list

initiatorsList ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": member;

BEHAVIOUR initiatorsListBehaviour BEHAVIOUR

DEFINED AS

! This set-valued attribute identifies the sub-classes of initiator managed objects which specify the initiators to which the rule pertains. It shall be an error to attempt to include a value in the initiators list attribute that is not the name of an initiators managed object. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) initiatorsList(15) };

A.5.16 Invalid access attempts

invalidAccessAttempts ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992": counter;

BEHAVIOUR invalidAccessAttemptBehaviourPkg BEHAVIOUR

DEFINED AS

! This attribute is used to count the number of occasions that an access control decision function has not authorized the access. The attribute takes the form of a not-settable counter as defined by CCITT Rec. X.721 | ISO/IEC 10165-2. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) invalidAccessAttempts(16) };

A.5.17 Label name

labelName ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.LabelName;

MATCHES FOR EQUALITY, ORDERING;

BEHAVIOUR labelNameBehaviourPkg BEHAVIOUR

DEFINED AS

! This attribute assigns a name of type integer to security labels. This enables a check for ordering to take place. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) labelName(17) };

A.5.18 Managed object classes

managedObjectClasses ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.ObjectClassList;

MATCHES FOR EQUALITY SET-COMPARISON, SET-INTERSECTION;

BEHAVIOUR managedObjectClassesBehaviour BEHAVIOUR

DEFINED AS

! This set-valued attribute identifies protected managed object classes and optional associated name bindings.

Any attempt to include a value not known to be a managed object class within the domain shall result in the CMIS invalid attribute value error. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) managedObjectClasses(18) };

A.5.19 Managed object instances

managedObjectInstances ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": member;

BEHAVIOUR managedObjectInstancesBehaviourPkg BEHAVIOUR

DEFINED AS

! This set-valued attribute identifies protected managed objects. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) managedObjectInstances(19) };

A.5.20 Operation type

operationType ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.OperationType;

MATCHES FOR EQUALITY;

BEHAVIOUR operationTypeBehaviourPkg BEHAVIOUR

DEFINED AS

! This read-only attribute is used for naming operations managed objects. It may take one of the values: get, replace, add member, remove member, replace with default, multiple object selection, filter, create, delete, and action. ! ;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) operationType(20) };

A.5.21 Operations list**operationsList ATTRIBUTE**

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.OperationsList;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR operationsListBehaviourPkg **BEHAVIOUR**
DEFINED AS

! This set-valued attribute identifies operations that are to be granted or denied, according to permissions in the containing rule managed object, on targets identified by the targets managed object. Operations are identified by the operation type. This attribute may be used when no conditional constraints are imposed on the parameters of the operation. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) operationsList(21) };

A.5.22 Scope**scope ATTRIBUTE**

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.Scope;
MATCHES FOR EQUALITY;
BEHAVIOUR scopeBehaviourPkg **BEHAVIOUR**
DEFINED AS

! The scope attribute identifies a scope for the selection of protected managed objects. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) scope(22) };

A.5.23 Scope filter**scopeFilter ATTRIBUTE**

DERIVED FROM accessControlFilter;
BEHAVIOUR scopeFilterBehaviour **BEHAVIOUR**
DEFINED AS

! For requests that select multiple managed objects the scope filter specifies constraints on the scope parameter of the request, and the scope attribute identifier is used for all the filter items in the filter.

This attribute identifies a filter upon the scope parameter of management operations. It shall have none or one element. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) scopeFilter(23) };

A.5.24 Security label**securityLabel ATTRIBUTE**

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.SecurityLabel;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR securityLabelBehaviour **BEHAVIOUR**
DEFINED AS

! The security label attribute contains a security label. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) securityLabel(24) };

A.5.25 State conditions**stateConditions ATTRIBUTE**

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.StateConditions;
MATCHES FOR EQUALITY;
BEHAVIOUR stateConditionsPackageBehaviour **BEHAVIOUR**
DEFINED AS

! This attribute identifies a managed object and a filter upon the attributes of that managed object. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) stateConditions(25) };

A.5.26 Synchronization

This attribute provides an attribute identifier and syntax for filtering upon the synchronization parameter of management operations.

synchronization ATTRIBUTE

WITH ATTRIBUTE SYNTAX AccessControlDefinitions.CMISSync;
BEHAVIOUR synchronizationBehaviour **BEHAVIOUR**
DEFINED AS

! This attribute value represents the synchronization parameter of management operations. It is used to represent filters upon this parameter. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) synchronization(26) };

A.5.27 Synchronization filter

synchronizationFilter ATTRIBUTE

DERIVED FROM accessControlFilter;

BEHAVIOUR synchronizationFilterBehaviour BEHAVIOUR

DEFINED AS

! For requests that select multiple managed objects the synchronization filter specifies constraints on the synchronization parameter of the request and the synchronization attribute identifier is used for all the filter items in the filter.

This attribute identifies a filter upon the synchronization parameter of management operations. It shall have none or one element. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) synchronizationFilter(27) };

A.5.28 Targets list

targetsList ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992": member;

BEHAVIOUR targetsListBehaviour BEHAVIOUR

DEFINED AS

! This set-valued attribute identifies the targets managed objects which themselves specify the targets to which the item rule pertains. It shall be an error to attempt to include a value which is not known to be the name of a targets managed object. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) targetsList(28) };

A.5.29 Valid access attempts

validAccessAttempts ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992": counter;

BEHAVIOUR validAccessAttemptBehaviourPkg BEHAVIOUR

DEFINED AS

! This attribute is used to count the number of occasions that an access control decision function has authorized the access. The attribute takes the form of a not-settable counter as defined by CCITT Rec. X.721 | ISO/IEC 10165-2. !;;

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) validAccessAttempts(29) };

A.6 Abstract syntax definitions

AccessControlDefinitions { joint-iso-ccitt ms(9) function(2) part9(9) asn1Module(2) 1 }

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS

AttributeId, CMISFilter, CMISSync, ObjectClass, ObjectInstance, Scope, ActionTypeId
FROM CMIP-1 { joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3) }

DistinguishedName

FROM InformationFramework { joint-iso-ccitt ds(5) modules(1) informationFramework(1) }

FunctionalUnitPackage

FROM SMASE-A-ASSOCIATE-Information { joint-iso-ccitt ms(9) smo(0) negotiationAbstractSyntax(1) version1(1) }

AETitle

FROM ACSE-1 { joint-iso-ccitt association-Control(2) abstractSyntax(1) apdus(0) version(1) }

DiscriminatorConstruct

FROM Attribute-ASN1Module { joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1 };

AccessControlList ::= SET OF CHOICE {
 proxy [0] Proxy,
 initiatorName [1] InitiatorName }

InitiatorName CHOICE {
 individualName [1] IMPLICIT DistinguishedName,
 groupName [2] IMPLICIT DistinguishedName,
 role [3] IMPLICIT DistinguishedName,
 application [4] IMPLICIT AETitle }

Proxy ::= SEQUENCE {
 proxyId [0] IMPLICIT OBJECT IDENTIFIER,
 proxyValue [1] ANY DEFINED BY proxyId }

AccessControlObjectName ::= GraphicString

ActionFilterList ::= SET OF SEQUENCE {
 actionTypeId ActionTypeId,
 attributeFilterList FilterList OPTIONAL }

AuthenticationContext ::= SEQUENCE
 { authenticationPolicyId [0] IMPLICIT OBJECT IDENTIFIER,
 requirements [1] ANY DEFINED BY authenticationPolicyId }

Boolean ::= BOOLEAN
 false **Boolean** ::= FALSE

CapabilityIdentitiesList ::= SET OF CHOICE {
 knownForm [0] SEQUENCE {
 initiatorName InitiatorName,
 sdaList SdaList OPTIONAL },
 unknownForm [1] SEQUENCE {
 identifier IMPLICIT OBJECT IDENTIFIER,
 value ANY DEFINED BY identifier }

SdaList ::= SET OF SEQUENCE {
 securityDomainAuthorityName SecurityDomainAuthorityName,
 operationType OperationType }

DefaultAccess ::= SEQUENCE {
 action [0] IMPLICIT EnforcementAction DEFAULT deny,
 create [1] IMPLICIT EnforcementAction DEFAULT deny,
 delete [2] IMPLICIT EnforcementAction DEFAULT deny,
 get [3] IMPLICIT EnforcementAction DEFAULT deny,
 replace [4] IMPLICIT EnforcementAction DEFAULT deny,
 addMember [5] IMPLICIT EnforcementAction DEFAULT deny,
 removeMember [6] IMPLICIT EnforcementAction DEFAULT deny,
 replaceWithDefault [7] IMPLICIT EnforcementAction DEFAULT deny,
 multipleObjectSelection [8] IMPLICIT EnforcementAction DEFAULT deny,
 filter [9] IMPLICIT EnforcementAction DEFAULT deny }

denyAll DefaultAccess ::= {}

DenialResponse ::= EnforcementAction {
 ENUMERATED
 denyWithResponse (0),
 denyWithoutResponse (1),
 abortAssociation (2),
 denyWithFalseResponse (3) }

DenialGranularity ::= ENUMERATED {
 request(0),
 object(1),
 attribute(2) }

DomainIdentity ::= CHOICE {
 domainName DistinguishedName,
 privateName OCTET STRING }

EnforcementAction ::= ENUMERATED {
 denyWithResponse (0),
 denyWithoutResponse (1),
 abortAssociation (2),
 denyWithFalseResponse (3),
 allow (4) }

Deny EnforcementAction ::= denyWithResponse

FilterList ::= SET OF CMISFilter

InvalidAccessControlFilter ::= SEQUENCE
 { errorId {
 ENUMERATED
 duplicateId(0),
 heterogeneousId(1),
 invalidId(2) },
 filter CMISFilter OPTIONAL }

LabelName ::= INTEGER

ObjectClassList ::= SET OF SEQUENCE {
 objectClass [0] ObjectClass,
 nameBinding [1] OBJECT IDENTIFIER OPTIONAL }

OperationsList ::= SET OF OperationType

OperationType ::= INTEGER {
 action (0),
 create (1),
 delete (2),
 get (3),

replace (4),
 addMember (5),
 removeMember (6),
 replaceWithDefault (7),
 multipleObjectSelection (8),
 filter (9) }

SecurityLabel ::= SET OF CHOICE {
 initiatorLabel [1] IMPLICIT SEQUENCE {
 clearance CHOICE {
 localForm [0] IMPLICIT INTEGER,
 globalForm [1] IMPLICIT OBJECT IDENTIFIER },
 category [2] IMPLICIT BIT STRING OPTIONAL } }

SecurityDomainAuthorityName ::= CHOICE {
 domainAuthorityName [1] IMPLICIT DistinguishedName,
 alternativeAuthorityName [2] IMPLICIT Proxy }

StateConditions ::= SET OF SEQUENCE { conditionalObject ObjectInstance,
 state CMISFilter }

END

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

Annex B

MCS proforma⁶⁾

(This annex forms an integral part of this Recommendation | International Standard)

B.1 Introduction

B.1.1 Purpose and structure

The management conformance summary (MCS) is a statement by a supplier that identifies an implementation and provides information on whether the implementation claims conformance to any of the listed set of document that specify conformance requirements to OSI management.

The MCS proforma is a document, in the form of a questionnaire that when completed by the supplier of an implementation becomes the MCS.

B.1.2 Instructions for completing the MCS proforma to produce an MCS

The supplier of the implementation shall enter an explicit statement in each of the boxes provided. Specific instruction is provided in the text which precedes each table.

B.1.3 Symbols, abbreviations and terms

For all annexes of this Recommendation | International Standard, the following common notations, defined in CCITT Rec. X.291 | ISO/IEC 9646-2 and ITU-T Rec. X.296 | ISO/IEC 9646-7, are used for the Status column:

- m Mandatory;
- o Optional;
- c Conditional;
- x Prohibited;
- Not applicable or out of scope.

NOTES

- 1 'c', 'm', and 'o' are prefixed by "c:" when nested under a conditional or optional item of the same table;
- 2 'o' may be suffixed by ".N" (where N is a unique number) for selectable options among a set of status values.

Support of at least one of the choices (from the items with the same value of N) is required.

The following requirements are commonly used throughout this MCS proforma:

c1: if B.1/1 then m else o

For all annexes of this Recommendation | International Standard, the following common notations, defined in CCITT Rec. X.291 | ISO/IEC 9646-2 and ITU-T Rec. X.296 | ISO/IEC 9646-7, are used for the Support column:

- Y Implemented
- N Not implemented
- No answer required
- Ig The item is ignored (i.e. processed syntactically but not semantically).

B.1.4 Table format

Some of the tables in this Recommendation | International Standard have been split because the information is too wide to fit on the page. Where this occurs, the index number of the first block of columns are the index numbers of the corresponding rows of the remaining blocks of columns. A complete table reconstructed from the constituent parts should have the following layout:

Index	First block of columns	Second block of columns	Etc.
-------	------------------------	-------------------------	------

⁶⁾ Users of this Recommendation | International Standard may freely reproduce the MCS proforma in this annex so that it can be used for its intended purpose, and may further publish the completed MCS. Instructions for completing the MCS proforma are specified in ITU-T Rec. X. 724 | ISO/IEC 10165-6.

In this Recommendation | International Standard the constituent parts of the table appear consecutively, starting with the first block of columns.

When a table with sub-rows is too wide to fit on a page, the continuation table(s) have been constructed with index numbers identical to the index numbers in the corresponding rows of the first table, and with sub-index numbers corresponding to the sub-rows within each indexed row. For example, if Table X.1 has 2 rows and the continuation of Table X.1 has 2 sub-rows for each row, the tables are presented as follows:

Table X.1 – Title

					Support		
Index	A	B	C	D	E	F	G
1	a	b	–				
2	a	b	–				

Table X.1 (continued) – Title

Index	Sub-index	H	I	J	K	L
1	1.1	h	i	j		
	1.2	h	i	j		
2	2.1	h	i	j		
	2.2	h	i	j		

A complete table reconstructed from the constituent parts should have the following layout:

								Support					
Index	A	B	C	D	E	F	G	Sub-index	H	I	J	K	L
1	a	b	–					1.1	h	i	j		
								1.2	h	i	j		
2	a	b	–					2.1	h	i	j		
								2.2	h	i	j		

References made to cells within tables shall be interpreted as references within reconstructed tables. In the example above, the reference X.1/1d corresponds to the blank cell in column G for row with Index 1, and X.1/1.2b corresponds to the blank cell in column L for row with sub-index 1.2.

B.2 Identification of the implementation

B.2.1 Date of statement

The supplier of the implementation shall enter the date of this statement in the box below. Use the format DD-MM-YYYY.

Date of statement

B.2.2 Identification of the implementation

The supplier of the implementation shall enter information necessary to uniquely identify the implementation and the system(s) in which it may reside, in the box below.

--

B.2.3 Contact

The supplier of the implementation shall provide information on whom to contact if there are any queries concerning the contents of the MCS or any referenced implementation conformance statement, in the box below.

--

B.3 Identification of the Recommendations | International Standards in which the management information is defined

The supplier of the implementation shall enter the title, reference number and date of the publication of the Recommendations | International Standards which specifies the management information to which conformance is claimed, in the box below.

Recommendations International Standards to which conformance is claimed

B.3.1 Technical corrigenda implemented

The supplier of the implementation shall enter the reference numbers of implemented technical corrigenda which modify the identified Recommendations | International Standards, in the box below.

--

B.3.2 Amendments implemented

The supplier of the implementation shall state the titles and reference numbers of implemented amendments to the identified Recommendations | International Standards, in the box below.

--

B.4 Management conformance summary

The supplier of the implementation shall state the capabilities and features supported and provide summary of conformance claims to Recommendations | International Standards using the tables in this annex.

The supplier of the implementation shall specify the roles that are supported, in Table B.1.

Table B.1 – Roles

Index	Roles supported	Status	Support	Additional information
1	Manager role support	o.1		
2	Agent role support	o.1		

The supplier of the implementation shall specify support for the systems management functional unit, in Table B.2.

Table B.2 – Systems management functional unit

Index	Systems management functional unit name	Manager		Agent		Additional information
		Status	Support	Status	Support	
1	Access control functional unit	c1		c2		
c1: if B.1/1a then o else –. c2: if B.1/2a then o else –.						

The supplier of the implementation shall specify support for management information in the manager role, in Table B.3.

Table B.3 – Manager role minimum conformance requirement

Index	Item	Status	Support	Additional information
1	Operations on managed objects	c3		
2	Object creation notification for access control managed object	c4		
3	Object deletion notification for access control managed object	c4		
4	Attribute value change notification for access control managed object	c4		
c3: if B.2/1a then o else (if B.1/1a then o.2 else –). c4: if B.2/1a then m else (if B.2/2a then o else (if B.1/1a then o.2 else –)). NOTE – Manager role minimum conformance requires support for at least one of the items identified in this table. Support for the functional unit identified in Table B.2 mandates support for some of those items. Conditions c3 and c4 express both of these requirements.				

The supplier of the implementation shall specify support for management information in the agent role, in Table B.4.

Table B.4 – Agent role minimum conformance requirement

Index	Item	Status	Support	Table reference	Additional information
1	Access control rules managed object	c5			
2	Rule managed object	c6			
3	Notification emitter managed object	c6			
4	Targets managed object	c6			
5	Operations managed object	c6			
6	ACL initiators managed object	c6			
7	Capability initiators managed object	c6			
8	Label initiators managed object	c6			
9	Assigned labels managed object	c6			
10	Attribute label managed object	c6			
11	Instance label managed object	c6			
12	Class label managed object	c6			
13	Sub-classes of log records associated with notifications emitted by sub-classes of access control managed object class	c7			
c5: if B.1/2a then m else –. c6: if B.1/2a then o else –. c7: if B.1/2a and B.5/1a then m else –. NOTE – The Table reference column is the notification, attribute or managed object table reference of the MOCS supplied by the supplier of the managed object which claims to import the notification or attribute from this Recommendation International Standard.					

Table B.5 – Logging of event records

Index		Status	Support	Additional information
1	Does the implementation support logging of event records in the agent role?	c8		
c8: if B.1/2a then o else –.				

NOTE 1 – Conformance to this Recommendation | International Standard does not require conformance to CCITT Rec. X.735 | ISO/IEC 10164-6.

The supplier of the implementation shall provide information on claims of conformance to any of the Recommendations | International Standards summarized in Tables B.6 to B.9. For each Recommendation | International Standard that the supplier of the implementation claims conformance to, the corresponding conformance statement(s) shall be completed, or referenced by, the MCS. The supplier of the implementation shall complete the Support, Table numbers and Additional information columns.

In Tables B.6 to B.9 the Status column is used to indicate whether the supplier of the implementation is required to complete the referenced tables or referenced items. Conformance requirements are as specified in the referenced tables or referenced items and are not changed by the value of the MCS Status column. Similarly, the Support column is used by the supplier of the implementation to indicate completion of the referenced tables or referenced items.

Table B.6 – PICS support summary

Index	Identification of the document that includes the PICS proforma	Table numbers of PICS proforma	Description	Constraints and Values	Status	Support	Table numbers of PICS	Additional Information
1	CCITT Rec. X.730 ISO/IEC 10164-1	Annex E all tables	SM application context	OBJECT IDENTIFIER	m			

NOTE 2 – Conformance to the MAPDUs defined in this Recommendation | International Standard can be claimed by completing the corresponding tables in the MICS and MOCS annexes of the referenced standards.

Table B.7 – MOCS support summary

Index	Identification of the document that includes the MOCS proforma	Table numbers of MOCS proforma	Description	Constraints and Values	Status	Support	Table numbers of MOCS	Additional Information
1	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.1 to D.5	accessControlRules	–	m			
2	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.6 to D.10	rule	–	o			
3	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.11 to D.15	notificationEmitter	–	o			
4	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.16 to D.20	targets	–	o			
5	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.21 to D.26	operations	–	o			
6	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.27 to D.31	aclInitiators	–	o			
7	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.32 to D.36	capabilityInitiators	–	o			
8	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.37 to D.41	labelInitiators	–	o			
9	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.42 to D.46	assignedLabels	–	o			
10	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.47 to D.51	attributeLabel	–	o			
11	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.52 to D.56	instanceLabel	–	o			
12	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex D, Tables D.57 to D.61	classLabel	–	o			

Table B.7 (concluded) – MOCS support summary

Index	Identification of the document that includes the MOCS proforma	Table numbers of MOCS proforma	Description	Constraints and Values	Status	Support	Table numbers of MOCS	Additional Information
13	CCITT Rec. X.730 ISO/IEC 10164-1	Annex C, all tables	objectCreation, objectDeletion, and AttributeValue Change	–	c9			
14	CCITT Rec. X.736 ISO/IEC 10164-7	Annex C, all tables	securityAlarmrecord	–	c9			
15	CCITT Rec. X.740 ISO/IEC 10164-8	Annex D, all tables	securityAuditTrailrecord	–	c9			
c9: if B.4/13a then m else –.								

Table B.8 – MRCS support summary

Index	Identification of the document that includes the MRCS proforma	Table numbers of MRCS proforma	Description	Constraints and Values	Status	Support	Table numbers of MRCS	Additional Information
1	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex E, all tables	rule-access ControlRules name binding	–	c10			
2	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex E, all tables	operations-targets name binding	–	c11			
3	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex E, all tables	notification Emitter-access ControlRules name binding	–	c12			
4	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex E, all tables	attributeLabel-assignedLabels name binding	–	c13			
5	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex E, all tables	instanceLabel assignedLabels name binding	–	c14			
6	ITU-T Rec. X.741 ISO/IEC 10164-9	Annex E, all tables	classLabel-assignedLabels name binding	–	c15			
7	CCITT Rec. X.740 ISO/IEC 10164-8	Item D.1/1	logRecord-log name binding	–	c16			
c10: if B.4/2a then o else –. c11: if B.4/5a then o else –. c12: if B.4/3a then o else –. c13: if B.4/10a then o else –. c14: if B.4/11a then o else –. c15: if B.4/12a then o else –. c16: if B.5/1a then o else –.								

Table B.9 – MICS support summary

Index	Identification of the document that includes the MICS proforma	Table numbers of MICS proforma	Description	Constraints and Values	Status	Support	Table numbers of MICS	Additional Information
1	ITU-T Rec. X.741 ISO/IEC 10164-9	Tables C.1 and C.2	management operations	–	c17			
2	CCITT Rec. X.730 ISO/IEC 10164-1	Table B.1	objectCreation, objectDeletion and attributeValue Change notifications	–	c18			
c17: if B.3/1a then m else –. c18: if B.3/2a or B.3/3a or B.3/4a then m else –.								

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 10164-9:1995

Annex C

MICS proforma⁷⁾

(This annex forms an integral part of this Recommendation | International Standard)

C.1 Introduction

The purpose of this MICS proforma is to provide a mechanism for a supplier of an implementation which claims conformance in the manager role to management information specified in this Recommendation | International Standard, to provide conformance information in a standard form.

C.2 Instructions for completing the MICS proforma to produce a MICS

The MICS proforma contained in this annex is comprised of information in tabular form, in accordance with ITU-T Rec. X.724 | ISO/IEC 10165-6. In addition to the general guidance given in ITU-T Rec. X.724 | ISO/IEC 10165-6, the Additional information column shall be used to identify the object classes for which the management operations are supported. The supplier of the implementation shall state which items are supported in the tables below and if necessary, provide additional information.

C.3 Symbols, abbreviations and terms

The following abbreviations are used throughout the MICS proforma:

dmi-att **joint-iso-ccitt ms(9) smi(3) part2(2) attribute(7)**

ac-att **joint-iso-ccitt ms(9) function(2) part9(9) attribute(7)**

The notations used for the Status and Support columns are specified in B.1.3.

C.4 Statement of conformance to the management information

C.4.1 Attributes

The specifier of a manager role implementation that claims to support management operations on the attributes specified in this Recommendation | International Standard shall import a copy of Table C.1 and complete it.

C.4.2 Create and delete management operations

The specifier of a manager role implementation that claims to support the create or delete management operations on the managed objects specified in this Recommendation | International Standard shall import a copy of Table C.2 and complete it.

⁷⁾ Users of this Recommendation | International Standard may freely reproduce the MICS proforma in this annex so that it can be used for its intended purpose, and may further publish the completed MICS.

Table C.1 – Attribute support

Index	Attribute template label	Value of object identifier for the attribute	Constraints and values	Set by create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	c19		o.3	
2	nameBinding	{dmi-att 63}	–	c19		o.3	
3	packages	{dmi-att 66}	–	c19		o.3	
4	allomorphs	{dmi-att 50}	–	c19		o.3	
5	availabilityStatus	{dmi-att 33}	–	–		o.3	
6	startTime	{dmi-att 68}	–	c19		o.3	
7	stopTime	{dmi-att 69}	–	c19		o.3	
8	intervalsOfDay	{dmi-att 57}	–	c19		o.3	
9	weekMask	{dmi-att 71}	–	c19		o.3	
10	schedulerName	{dmi-att 67}	–	c19		o.3	
11	attributeIdentifierList	{dmi-att 8}	–	c19		o.3	
12	managedObjectInstance	{dmi-att 61}	–	c19		o.3	
13	accessControlList	{ac-att 1}	–	c19		o.3	
14	accessControlFilter	{ac-att 2}	–	c19		o.3	
15	accessControlObjectName	{ac-att 3}	–	c19		o.3	
16	actionFilterList	{ac-att 4}	–	c19		o.3	
17	attributeFilterList	{ac-att 5}	–	c19		o.3	
18	authenticationContext	{ac-att 6}	–	c19		o.3	
19	capabilityIdentitiesList	{ac-att 7}	–	c19		o.3	
20	defaultAccess	{ac-att 8}	–	c19		o.3	
21	defaultDenialResponse	{ac-att 9}	–	c19		o.3	
22	denialGranularity	{ac-att 10}	–	c19		o.3	
23	domainIdentity	{ac-att 11}	–	c19		o.3	
24	enforcementAction	{ac-att 12}	–	c19		o.3	
25	filter	{ac-att 13}	–	c19		o.3	
26	initiatorACImandated	{ac-att 14}	–	c19		o.3	
27	initiatorsList	{ac-att 15}	–	c19		o.3	
28	invalidAccessAttempts	{ac-att 16}	–	c19		o.3	
29	labelName	{ac-att 17}	–	c19		o.3	
30	managedObjectClasses	{ac-att 18}	–	c19		o.3	
31	managedObjectInstances	{ac-att 19}	–	c19		o.3	
32	operationType	{ac-att 20}	–	c19		o.3	
33	operationsList	{ac-att 21}	–	c19		o.3	
34	scope	{ac-att 22}	–	c19		o.3	
35	scopeFilter	{ac-att 23}	–	c19		o.3	
36	securityLabel	{ac-att 24}	–	c19		o.3	
37	stateConditions	{ac-att 25}	–	c19		o.3	
38	synchronization	{ac-att 26}	–	c19		o.3	
39	synchronizationFilter	{ac-att 27}	–	c19		o.3	
40	targetsList	{ac-att 28}	–	c19		o.3	
41	validAccessAttempts	{ac-att 29}	–	c19		o.3	

Table C.1 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	o.3		–		–		–		
7	o.3		–		–		–		
8	o.3		o.3		o.3		o.3		
9	o.3		o.3		o.3		o.3		
10	–		–		–		–		
11	o.3		o.3		o.3		–		
12	o.3		–		–		–		
13	o.3		o.3		o.3		o.3		
14	–		–		–		–		
15	–		–		–		–		
16	o.3		o.3		o.3		–		
17	o.3		o.3		o.3		–		
18	o.3		–		–		–		
19	o.3		–		–		–		
20	o.3		–		–		o.3		
21	o.3		–		–		–		
22	o.3		–		–		–		
23	o.3		–		–		–		
24	o.3		–		–		o.3		
25	o.3		–		–		–		
26	o.3		–		–		o.3		
27	o.3		o.3		o.3		–		
28	–		–		–		–		
29	–		–		–		–		
30	o.3		o.3		o.3		–		
31	o.3		o.3		o.3		–		
32	–		–		–		–		
33	o.3		o.3		o.3		–		
34	o.3		–		–		–		
35	o.3		–		–		–		
36	o.3		–		–		o.3		
37	o.3		o.3		o.3		–		
38	–		–		–		–		
39	o.3		–		–		–		
40	o.3		o.3		o.3		–		
41	–		–		–		–		
c1: if C2/1a or C2/3a or C2/5a or C2/7a or C2/9a or C2/11a or C2/13a or C2/15a or C2/17a or C2/19a or C2/21a or c2/23a then o else –.									

Table C.2 – Create and delete support

Index	Operation	Constraints and values	Status	Support	Additional information
1	Create support	Access control rules managed object	o		
1.1	Create with reference object	Access control rules managed object	–		
2	Delete support	Access control rules managed object	o		
3	Create support	Rule managed object	o		
3.1	Create with reference object	Rule managed object	c:o		
4	Delete support	Rule managed object	o		
5	Create support	Notification emitter managed object	o		
5.1	Create with reference object	Notification emitter managed object	c:o		
6	Delete support	Notification emitter managed object	o		
7	Create support	Targets managed object	o		
7.1	Create with reference object	Targets managed object	–		
8	Delete support	Targets managed object	o		
9	Create support	Operations managed object	o		
9.1	Create with reference object	Operations managed object	c:o		
10	Delete support	Operations managed object	o		
11	Create support	ACL initiators managed object	o		
11.1	Create with reference object	ACL initiators managed object	–		
12	Delete support	ACL initiators managed object	o		
13	Create support	Capability initiators managed object	o		
13.1	Create with reference object	Capability initiators managed object	–		
14	Delete support	Capability initiators managed object	o		
15	Create support	Label initiators managed object	o		
15.1	Create with reference object	Label initiators managed object	–		
16	Delete support	Label initiators managed object	o		
17	Create support	Assigned labels managed object	o		
17.1	Create with reference object	Assigned labels managed object	–		
18	Delete support	Assigned labels managed object	o		
19	Create support	Attribute label managed object	o		
19.1	Create with reference object	Attribute label managed object	–		
20	Delete support	Attribute label managed object	o		
21	Create support	Class label managed object	o		
21.1	Create with reference object	Class label managed object	–		
22	Delete support	Class label managed object	o		
23	Create support	Instance label managed object	o		
23.1	Create with reference object	Instance label managed object	–		
24	Delete support	Instance label managed object	o		

Annex D

MOCS proforma⁸⁾

(This annex forms an integral part of this Recommendation | International Standard)

D.1 Introduction

The purpose of this MOCS proforma is to provide a mechanism for a supplier of an implementation which claims conformance to a managed object class, to provide conformance information in a standard form.

D.2 Instructions for completing the MOCS proforma to produce a MOCS

The MOCS proforma contained in this annex is comprised of information in tabular form, in accordance with ITU-T Rec. X.724 | ISO/IEC 10165-6. The supplier of the implementation shall state which items are supported in the tables below and if necessary, provide additional information.

D.3 Symbols, abbreviations and terms

The following abbreviations are used throughout the MOCS proforma:

dmi-att joint-iso-ccitt ms(9) smi(3) part2(2) attribute(7)
dmi-nb joint-iso-ccitt ms(9) smi(3) part2(2) nameBinding(6)
dmi-not joint-iso-ccitt ms(9) smi(3) part2(2) notification(10)
dmi-pkg joint-iso-ccitt ms(9) smi(3) part2(2) package(4)
ac-obj joint-iso-ccitt ms(9) function(2) part9(9) managedObjectClass(3)
ac-att joint-iso-ccitt ms(9) function(2) part9(9) attribute(7)
ac-nb joint-iso-ccitt ms(9) function(2) part9(9) nameBinding(6)
ac-par joint-iso-ccitt ms(9) function(2) part9(9) parameter(5)
ac-pkg joint-iso-ccitt ms(9) function(2) part9(9) package(4)
sat-att joint-iso-ccitt ms(9) function(2) part8(8) attribute(7)
sat-not joint-iso-ccitt ms(9) function(2) part8(8) notification(10)

The notations used for the Status and Support columns are specified in B.1.3.

D.4 Access control rules managed object class

D.4.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the access control rules managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.1.

Table D.1 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	accessControlRules	{ac-obj 2}		

⁸⁾ Users of this Recommendation | International Standard may freely reproduce the MOCS proforma in this annex so that it can be used for its intended purpose, and may further publish the completed MOCS. Instructions for completing the MOCS proforma are specified in ITU-T Rec. X.724 | ISO/IEC 10165-6.

If the answer to the actual class question in the managed object class support Table D.1 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.2.

Table D.2 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.4.2 Packages

See Table D.3.

Table D.3 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c20		
3	allomorphicPackage	{dmi-pkg 17}	–	c21		
4	accessControlPackage	–	–	m		
5	accessControlRulesPackage	–	–	m		
c20: if D.3/3 then m else –.						
c21: if D.1/1b then – else m.						

D.4.3 Attributes

See Table D.4.

Table D.4 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c22		c22	
4	allomorphs	{dmi-att 50}	–	c23		c23	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	defaultAccess	{ac-att 8}	–	m		m	
7	defaultDenialResponse	{ac-att 9}	–	m		m	
8	denialGranularity	{ac-att 10}	–	m		m	
9	domainIdentity	{ac-att 11}	–	m		m	
c22: if D.3/2 then m else –.							
c23: if D.3/3 then m else –.							

Table D.4 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	m		–		–		m		
7	m		–		–		–		
8	m		–		–		–		
9	m		–		–		–		

D.4.4 Notifications

See Table D.5.

Table D.5 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	attributeValueChange	{dmi-not 1}	–	m			
2	objectCreation	{dmi-not 6}	–	m			
3	objectDeletion	{dmi-not 7}	–	m			

Table D.5 (continued) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	sourceIndicator	{dmi-att 26}	–	o		
	1.2	attributeIdentifierList	{dmi-att 8}	–	o		
	1.3	attributeValueChangeDefinition	{dmi-att 10}	–	m		
	1.3.1	attributeId	–	–	m		
	1.3.2	oldAttributeValue	–	–	o		
	1.3.3	newAttributeValue	–	–	m		
	1.4	notificationIdentifier	{dmi-att 16}	–	c24		
	1.5	correlatedNotifications	{dmi-att 12}	–	o		
	1.5.1	correlatedNotification	–	–	c:m		
	1.5.2	sourceObjectInst	–	–	c:o		
	1.5.2.1	distinguishedName	–	–	c:o.4		
	1.5.2.2	nonSpecificForm	–	–	c:o.4		
	1.5.2.3	localDistinguishedName	–	–	c:o.4		
	1.6	additionalText	{dmi-att 7}	–	o		
	1.7	additionalInformation	{dmi-att 6}	–	–		

Table D.5 (concluded) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
2	2.1	sourceIndicator	{dmi-att 26}	–	o		
	2.2	attributeList	{dmi-att 9}	–	o		
	2.3	notificationIdentifier	{dmi-att 16}	–	c25		
	2.4	correlatedNotifications	{dmi-att 12}	–	o		
	2.4.1	correlatedNotification	–	–	c:m		
	2.4.2	sourceObjectInst	–	–	c:o		
	2.4.2.1	distinguishedName	–	–	c:o.5		
	2.4.2.2	nonSpecificForm	–	–	c:o.5		
	2.4.2.3	localDistinguishedName	–	–	c:o.5		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	–		
3	3.1	sourceIndicator	{dmi-att 26}	–	o		
	3.2	attributeList	{dmi-att 9}	–	o		
	3.3	notificationIdentifier	{dmi-att 16}	–	c26		
	3.4	correlatedNotifications	{dmi-att 12}	–	o		
	3.4.1	correlatedNotification	–	–	c:m		
	3.4.2	sourceObjectInst	–	–	c:o		
	3.4.2.1	distinguishedName	–	–	c:o.6		
	3.4.2.2	nonSpecificForm	–	–	c:o.6		
	3.4.2.3	localDistinguishedName	–	–	c:o.6		
	3.5	additionalText	{dmi-att 7}	–	o		
	3.6	additionalInformation	{dmi-att 6}	–	o		
c24: if D.5/1.5 then m else o. c25: if D.5/2.4 then m else o. c26: if D.5/3.4 then m else o.							

D.5 Rule managed object class

D.5.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the rule managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.6.

Table D.6 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	rule	{ac-obj 3}		

If the answer to the actual class question in the managed object class support Table D.6 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.7.

Table D.7 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.5.2 Packages

See Table D.8.

Table D.8 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c27		
3	allomorphicPackage	{dmi-pkg 17}	–	c28		
4	accessControlPackage	–	–	m		
5	rulePackage	–	–	m		
6	availabilityStatusPackage	{dmi-pkg 22}	–	o		
7	duration	{dmi-pkg 26}	–	o		
8	dailyScheduling	{dmi-pkg 25}	–	o		
9	weeklyScheduling	{dmi-pkg 29}	–	o		
10	externalScheduler	{dmi-pkg 27}	–	o		
11	stateConditionsPackage	{ac-pkg 1}	–	o		
12	authenticationContextPackage	{ac-pkg 2}	–	o		
c27: if D.8/3 or any of D.8/6 through D.8/12 then m else –.						
c28: if D.6/1.b then – else m.						

D.5.3 Attributes

See Table D.9.

Table D.9 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c29		c29	
4	allomorphs	{dmi-att 50}	–	c30		c30	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	enforcementAction	{ac-att 12}	–	m		m	
7	initiatorsList	{ac-att 15}	–	m		m	
8	targetsList	{ac-att 28}	–	m		m	
9	availabilityStatus	{dmi-att 33}	–	–		c31	

Table D.9 (continued) – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
10	startTime	{dmi-att 68}	DMI default	c32		c32	
11	stopTime	{dmi-att 69}	DMI default	c32		c32	
12	intervalsOfDay	{dmi-att 57}	DMI default	c33		c33	
13	weekMask	{dmi-att 71}	–	c34		c34	
14	schedulerName	{dmi-att 67}	–	c35		c35	
15	stateConditions	{ac-att 25}	–	c36		c36	
16	authenticationContext	{ac-att 6}	–	c37		c37	

Table D.9 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	m		–		–		m		
7	m		m		m		–		
8	m		m		m		–		
9	–		–		–		–		
10	c32		–		–		–		
11	c32		–		–		c38		
12	c33		–		–		c33		
13	c34		c34		c34		c34		
14	x		–		–		–		
15	c36		c36		c36		–		
16	c37		–		–		–		
c29: if D.8/2 then m else –. c30: if D.8/3 then m else –. c31: if D.8/6 then m else –. c32: if D.8/7 then m else –. c33: if D.8/8 then m else –. c34: if D.8/9 then m else –. c35: if D.8/10 then m else –. c36: if D.8/11 then m else –. c37: if D.8/12 then m else –. c38: if D.6/1b then x else –.									

D.5.4 Notifications

See Table D.10.

Table D.10 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	attributeValueChange	{dmi-not 1}	–	m			
2	objectCreation	{dmi-not 6}	–	m			
3	objectDeletion	{dmi-not 7}	–	m			

Table D.10 (continued) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	sourceIndicator	{dmi-att 26}	–	o		
	1.2	attributeIdentifierList	{dmi-att 8}	–	o		
	1.3	attributeValueChangeDefinition	{dmi-att 10}	–	m		
	1.3.1	attributeId	–	–	m		
	1.3.2	oldAttributeValue	–	–	o		
	1.3.3	newAttributeValue	–	–	m		
	1.4	notificationIdentifier	{dmi-att 16}	–	c39		
	1.5	correlatedNotifications	{dmi-att 12}	–	o		
	1.5.1	correlatedNotification	–	–	c:m		
	1.5.2	sourceObjectInst	–	–	c:o		
	1.5.2.1	distinguishedName	–	–	c:o.7		
	1.5.2.2	nonSpecificForm	–	–	c:o.7		
	1.5.2.3	localDistinguishedName	–	–	c:o.7		
	1.6	additionalText	{dmi-att 7}	–	o		
	1.7	additionalInformation	{dmi-att 6}	–	o		
2	2.1	sourceIndicator	{dmi-att 26}	–	o		
	2.2	attributeList	{dmi-att 9}	–	o		
	2.3	notificationIdentifier	{dmi-att 16}	–	c40		
	2.4	correlatedNotifications	{dmi-att 12}	–	o		
	2.4.1	correlatedNotification	–	–	c:m		
	2.4.2	sourceObjectInst	–	–	c:o		
	2.4.2.1	distinguishedName	–	–	c:o.8		
	2.4.2.2	nonSpecificForm	–	–	c:o.8		
	2.4.2.3	localDistinguishedName	–	–	c:o.8		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	o		
3	3.1	sourceIndicator	{dmi-att 26}	–	o		
	3.2	attributeList	{dmi-att 9}	–	o		
	3.3	notificationIdentifier	{dmi-att 16}	–	c41		

Table D.10 (concluded) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
3 (cont.)	3.4	correlatedNotifications	{dmi-att 12}	–	o		
	3.4.1	correlatedNotification	–	–	c:m		
	3.4.2	sourceObjectInst	–	–	c:o		
	3.4.2.1	distinguishedName	–	–	c:o.9		
	3.4.2.2	nonSpecificForm	–	–	c:o.9		
	3.4.2.3	localDistinguishedName	–	–	c:o.9		
	3.5	additionalText	{dmi-att 7}	–	o		
	3.6	additionalInformation	{dmi-att 6}	–	o		
c39: if D.10/1.5 then m else –. c40: if D.10/2.4 then m else –. c41: if D.10/3.4 then m else –.							

D.6 Notification emitter managed object class**D.6.1 Statement of conformance to the managed object class**

The supplier of the implementation shall state whether or not all mandatory features of the notification emitter managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.11.

Table D.11 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	notificationEmitter	{ac-obj 4}		

If the answer to the actual class question in the managed object class support Table D.11 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.12.

Table D.12 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.6.2 Packages

See Table D.13.

Table D.13 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c42		
3	allomorphicPackage	{dmi-pkg 17}	–	c43		
4	accessControlPackage	–	–	m		
5	accessControlNotificationEmitterPkg	–	–	m		
6	securityViolationAlarmPkg	{ac-pkg 3}	–	o		
7	timeViolationAlarmPkg	{ac-pkg 4}	–	o		
8	operationalViolationAlarmPkg	{ac-pkg 5}	–	o		
9	accessControlUsagePkg	{ac-pkg 6}	–	o		
10	accessControlServiceReportPkg	{ac-pkg 7}	–	o		
c42: if D.13/3 or D.13/6 through D.13/10 then m else –.						
c43: if D.11/1b then – else m.						

D.6.3 Attributes

See Table D.14.

Table D.14 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c44		c44	
4	allomorphs	{dmi-att 50}	–	c45		c45	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	invalidAccessAttempts	{ac-att 16}	–	c46		c46	
7	validAccessAttempts	{ac-att 29}	–	c46		c46	

Table D.14 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	–		–		–		–		
7	–		–		–		–		
c44: if D.13/2 then m else –. c45: if D.13/3 then m else –. c46: if D.13/9 then m else –.									

D.6.4 Notifications

See Table D.15.

Table D.15 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	attributeValueChange	{ dmi-not 1 }	–	m			
2	objectCreation	{ dmi-not 6 }	–	m			
3	objectDeletion	{ dmi-not 7 }	–	m			
4	securityServiceOrMechanism Violation	{ dmi-not 13 }	–	c47			
5	timeDomainViolation	{ dmi-not 15 }	–	c48			
6	operationalViolation	{ dmi-not 8 }	–	c49			
7	usageReport	{ sat-not 2 }	–	c50			
8	serviceReport	{ sat-not 1 }	–	c51			
c47: if D.13/6 then m else –. c48: if D.13/7 then m else –. c49: if D.13/8 then m else –. c50: if D.13/9 then m else –. c51: if D.13/10 then m else –.							

Table D.15 (continued) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	sourceIndicator	{ dmi-att 26 }	–	o		
	1.2	attributeIdentifierList	{ dmi-att 8 }	–	o		
	1.3	attributeValueChangeDefinition	{ dmi-att 10 }	–	m		
	1.3.1	attributeId	–	–	m		
	1.3.2	oldAttributeValue	–	–	o		
	1.3.3	newAttributeValue	–	–	m		
	1.4	notificationIdentifier	{ dmi-att 16 }	–	c52		
	1.5	correlatedNotifications	{ dmi-att 12 }	–	o		
	1.5.1	correlatedNotification	–	–	c:m		
	1.5.2	sourceObjectInst	–	–	c:o		
	1.5.2.1	distinguishedName	–	–	c:o.10		
	1.5.2.2	nonSpecificForm	–	–	c:o.10		
	1.5.2.3	localDistinguishedName	–	–	c:o.10		
	1.6	additionalText	{ dmi-att 7 }	–	o		
	1.7	additionalInformation	{ dmi-att 6 }	–	o		
2	2.1	sourceIndicator	{ dmi-att 26 }	–	o		
	2.2	attributeList	{ dmi-att 9 }	–	o		
	2.3	notificationIdentifier	{ dmi-att 16 }	–	c53		
	2.4	correlatedNotifications	{ dmi-att 12 }	–	o		
	2.4.1	correlatedNotification	–	–	c:m		
	2.4.2	sourceObjectInst	–	–	c:o		
	2.4.2.1	distinguishedName	–	–	c:o.11		
	2.4.2.2	nonSpecificForm	–	–	c:o.11		
	2.4.2.3	localDistinguishedName	–	–	c:o.11		
	2.5	additionalText	{ dmi-att 7 }	–	o		
	2.6	additionalInformation	{ dmi-att 6 }	–	o		
3	3.1	sourceIndicator	{ dmi-att 26 }	–	o		
	3.2	attributeList	{ dmi-att 9 }	–	o		
	3.3	notificationIdentifier	{ dmi-att 16 }	–	c54		
	3.4	correlatedNotifications	{ dmi-att 12 }	–	o		
	3.4.1	correlatedNotification	–	–	c:m		
	3.4.2	sourceObjectInst	–	–	c:o		
	3.4.2.1	distinguishedName	–	–	c:o.12		
	3.4.2.2	nonSpecificForm	–	–	c:o.12		
	3.4.2.3	localDistinguishedName	–	–	c:o.12		
	3.5	additionalText	{ dmi-att 7 }	–	o		
	3.6	additionalInformation	{ dmi-att 6 }	–	o		
4	4.1	securityAlarmCause	{ dmi-att 21 }	–	m		
	4.2	securityAlarmSeverity	{ dmi-att 23 }	–	m		
	4.3	securityAlarmDetector	{ dmi-att 22 }	–	m		
	4.3.1	mechanism	–	–	o		
	4.3.2	object	–	–	o		
	4.3.3	application	–	–	o		
	4.4	serviceUser	{ dmi-att 25 }	–	m		

Table D.15 (continued) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
4 (cont.)	4.4.1	identifier	–	–	o		
	4.4.2	details	–	–	o		
	4.5	serviceProvider	{ dmi-att 24 }	–	m		
	4.5.1	identifier	–	–	o		
	4.5.2	details	–	–	o		
	4.6	notificationIdentifier	{ dmi-att 16 }	–	c55		
	4.7	correlatedNotifications	{ dmi-att 12 }	–	o		
	4.7.1	correlatedNotification	–	–	c:m		
	4.7.2	sourceObjectInst	–	–	c:o		
	4.7.2.1	distinguishedName	–	–	c:o.13		
	4.7.2.2	nonSpecificForm	–	–	c:o.13		
	4.7.2.3	localDistinguishedName	–	–	c:o.13		
	4.8	additionalText	{ dmi-att 7 }	–	o		
	4.9	additionalInformation	{ dmi-att 6 }	–	o		
5	5.1	securityAlarmCause	{ dmi-att 21 }	–	m		
	5.2	securityAlarmSeverity	{ dmi-att 23 }	–	m		
	5.3	securityAlarmDetector	{ dmi-att 22 }	–	m		
	5.3.1	mechanism	–	–	o		
	5.3.2	object	–	–	o		
	5.3.3	application	–	–	o		
	5.4	serviceUser	{ dmi-att 25 }	–	m		
	5.4.1	identifier	–	–	o		
	5.4.2	details	–	–	o		
	5.5	serviceProvider	{ dmi-att 24 }	–	m		
	5.5.1	identifier	–	–	o		
	5.5.2	details	–	–	o		
	5.6	notificationIdentifier	{ dmi-att 16 }	–	c56		
	5.7	correlatedNotifications	{ dmi-att 12 }	–	o		
	5.7.1	correlatedNotification	–	–	c:m		
	5.7.2	sourceObjectInst	–	–	c:o		
	5.7.2.1	distinguishedName	–	–	c:o.14		
	5.7.2.2	nonSpecificForm	–	–	c:o.14		
	5.7.2.3	localDistinguishedName	–	–	c:o.14		
	5.8	additionalText	{ dmi-att 7 }	–	o		
	5.9	additionalInformation	{ dmi-att 6 }	–	o		
6	6.1	securityAlarmCause	{ dmi-att 21 }	–	m		
	6.2	securityAlarmSeverity	{ dmi-att 23 }	–	m		
	6.3	securityAlarmDetector	{ dmi-att 22 }	–	m		
	6.3.1	mechanism	–	–	o		
	6.3.2	object	–	–	o		
	6.3.3	application	–	–	o		
	6.4	serviceUser	{ dmi-att 25 }	–	m		
	6.4.1	identifier	–	–	o		
	6.4.2	details	–	–	o		
	6.5	serviceProvider	{ dmi-att 24 }	–	m		

Table D.15 (concluded) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
6 (cont.)	6.5.1	identifier	–	–	o		
	6.5.2	details	–	–	o		
	6.6	notificationIdentifier	{dmi-att 16}	–	c57		
	6.7	correlatedNotifications	{dmi-att 12}	–	o		
	6.7.1	correlatedNotification	–	–	c:m		
	6.7.2	sourceObjectInst	–	–	c:o		
	6.7.2.1	distinguishedName	–	–	c:o.15		
	6.7.2.2	nonSpecificForm	–	–	c:o.15		
	6.7.2.3	localDistinguishedName	–	–	c:o.15		
	6.8	additionalText	{dmi-att 7}	–	o		
	6.9	additionalInformation	{dmi-att 6}	–	o		
7	7.1	notificationIdentifier	{dmi-att 16}	–	c58		
	7.2	correlatedNotifications	{dmi-att 12}	–	o		
	7.2.1	correlatedNotification	–	–	c:m		
	7.2.2	sourceObjectInst	–	–	c:o		
	7.2.2.1	distinguishedName	–	–	c:o.16		
	7.2.2.2	nonSpecificForm	–	–	c:o.16		
	7.2.2.3	localDistinguishedName	–	–	c:o.16		
	7.3	additionalText	{dmi-att 7}	–	o		
	7.4	additionalInformation	{dmi-att 6}	–	o		
8	8.1	serviceReportCause	{at-att 1}	–	m		
	8.2	notificationIdentifier	{dmi-att 16}	–	c59		
	8.3	correlatedNotifications	{dmi-att 12}	–	o		
	8.3.1	correlatedNotification	–	–	c:m		
	8.3.2	sourceObjectInst	–	–	c:o		
	8.3.2.1	distinguishedName	–	–	c:o.17		
	8.3.2.2	nonSpecificForm	–	–	c:o.17		
	8.3.2.3	localDistinguishedName	–	–	c:o.17		
	8.4	additionalText	{dmi-att 7}	–	o		
	8.5	additionalInformation	{dmi-att 6}	–	o		
c52: if D.15/1.5 then m else –. c53: if D.15/2.4 then m else –. c54: if D.15/3.4 then m else –. c55: if D.15/4.7 then m else –. c56: if D.15/5.7 then m else –. c57: if D.15/6.7 then m else –. c58: if D.15/7.2 then m else –. c59: if D.15/8.3 then m else –.							

D.7 Targets managed object class**D.7.1 Statement of conformance to the managed object class**

The supplier of the implementation shall state whether or not all mandatory features of the targets managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.16.

Table D.16 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	targets	{ac-obj 5}		

If the answer to the actual class question in the managed object class support Table D.16 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.17.

Table D.17 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.7.2 Packages

See Table D.18.

Table D.18 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c60		
3	allomorphicPackage	{dmi-pkg 17}	–	c61		
4	accessControlPackage	–	–	m		
5	targetsPackage	–	–	m		
6	operationsListPackage	{ac-pkg 15}	–	o		
c60: if D.18/3 or D.18/6 then m else –.						
c61: if D.16/1b then – else m.						

D.7.3 Attributes

See Table D.19.

Table D.19 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c62		c62	
4	allomorphs	{dmi-att 50}	–	c63		c63	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	managedObjectClasses	{ac-att 18}	–	m		m	
7	managedObjectInstances	{ac-att 19}	–	m		m	
8	scope	{ac-att 22}	–	m		m	
9	filter	{ac-att 13}	–	m		m	
10	operationsList	{ac-att 21}	–	c64		c64	

Table D.19 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	m		m		m		–		
7	m		m		m		–		
8	m		m		–		–		
9	m		m		–		–		
10	c64		c64		c64		–		
c62: if D.18/2 then m else –. c63: if D.18/3 then m else –. c64: if D.18/6 then m else –.									

D.7.4 Notifications

See Table D.20.

Table D.20 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	attributeValueChange	{ dmi-not 1 }	–	m			
2	objectCreation	{ dmi-not 6 }	–	m			
3	objectDeletion	{ dmi-not 7 }	–	m			

Table D.20 (continued) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	sourceIndicator	{ dmi-att 26 }	–	o		
	1.2	attributeIdentifierList	{ dmi-att 8 }	–	o		
	1.3	attributeValueChangeDefinition	{ dmi-att 10 }	–	m		
	1.3.1	attributeId	–	–	m		
	1.3.2	oldAttributeValue	–	–	o		
	1.3.3	newAttributeValue	–	–	m		
	1.4	notificationIdentifier	{ dmi-att 16 }	–	c65		
	1.5	correlatedNotifications	{ dmi-att 12 }	–	o		
	1.5.1	correlatedNotification	–	–	c:m		
	1.5.2	sourceObjectInst	–	–	c:o		
	1.5.2.1	distinguishedName	–	–	c:o.18		
	1.5.2.2	nonSpecificForm	–	–	c:o.18		
	1.5.2.3	localDistinguishedName	–	–	c:o.18		
	1.6	additionalText	{ dmi-att 7 }	–	o		
	1.7	additionalInformation	{ dmi-att 6 }	–	o		
2	2.1	sourceIndicator	{ dmi-att 26 }	–	o		
	2.2	attributeList	{ dmi-att 9 }	–	o		
	2.3	notificationIdentifier	{ dmi-att 16 }	–	c66		
	2.4	correlatedNotifications	{ dmi-att 12 }	–	o		
	2.4.1	correlatedNotification	–	–	c:m		
	2.4.2	sourceObjectInst	–	–	c:o		
	2.4.2.1	distinguishedName	–	–	c:o.19		
	2.4.2.2	nonSpecificForm	–	–	c:o.19		
	2.4.2.3	localDistinguishedName	–	–	c:o.19		
	2.5	additionalText	{ dmi-att 7 }	–	o		
	2.6	additionalInformation	{ dmi-att 6 }	–	o		
3	3.1	sourceIndicator	{ dmi-att 26 }	–	o		
	3.2	attributeList	{ dmi-att 9 }	–	o		
	3.3	notificationIdentifier	{ dmi-att 16 }	–	c67		
	3.4	correlatedNotifications	{ dmi-att 12 }	–	o		

Table D.20 (concluded) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
3 (cont.)	3.4.1	correlatedNotification	–	–	c:m		
	3.4.2	sourceObjectInst	–	–	c:o		
	3.4.2.1	distinguishedName	–	–	c:o.20		
	3.4.2.2	nonSpecificForm	–	–	c:o.20		
	3.4.2.3	localDistinguishedName	–	–	c:o.20		
	3.5	additionalText	{dmi-att 7}	–	o		
	3.6	additionalInformation	{dmi-att 6}	–	o		
c65: if D.20/1.5 then m else –. c66: if D.20/2.4 then m else –. c67: if D.20/3.4 then m else –.							

D.8 Operations managed object class

D.8.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the operations managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.21.

Table D.21 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	operations	{ac-obj 6}		

If the answer to the actual class question in the managed object class support Table D.21 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.22.

Table D.22 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.8.2 Packages

See Table D.23.

Table D.23 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c68		
3	allomorphicPackage	{dmi-pkg 17}	–	c69		
4	accessControlPackage	–	–	m		
5	operationsPackage	–	–	m		
6	attributeIdsPackage	{ac-pkg 8}	–	o		
7	attributeModificationPackage	{ac-pkg 9}	–	o		
8	actionsPackage	{ac-pkg 10}	–	o		
9	scopePackage	{ac-pkg 11}	–	o		
c68: if D.23/3 or D.23/6 or D.23/7 or D.23/8 or D.23/9 then m else –.						
c69: if C.21/1b then – else m.						

D.8.3 Attributes

See Table D.24.

Table D.24 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c70		c70	
4	allomorphs	{dmi-att 50}	–	c71		c71	
5	operationType	{ac-att 20}	–	m		m	
6	attributeIdentifierList	{dmi-att 8}	–	c72		c72	
7	attributeFilterList	{ac-att 5}	–	c73		c73	
8	actionFilterList	{ac-att 4}	–	c74		c74	
9	scopeFilter	{ac-att 23}	–	c75		c75	
10	synchronizationFilter	{ac-att 27}	–	c75		c75	

Table D.24 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	–		–		–		–		
7	c73		c73		c73		–		
8	c74		c74		c74		–		
9	c75		–		–		–		
10	c75		–		–		–		
c70: if D.23/2 then m else –. c71: if D.23/3 then m else –. c72: if D.23/6 then m else –. c73: if D.23/7 then m else –. c74: if D.23/8 then m else –. c75: if D.23/9 then m else –.									

D.8.4 Notifications

See Table D.25.

Table D.25 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	attributeValueChange	{dmi-not 1}	–	m			
2	objectCreation	{dmi-not 6}	–	m			
3	objectDeletion	{dmi-not 7}	–	m			

Table D.25 (concluded) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	sourceIndicator	{ dmi-att 26 }	–	o		
	1.2	attributeIdentifierList	{ dmi-att 8 }	–	o		
	1.3	attributeValueChangeDefinition	{ dmi-att 10 }	–	m		
	1.3.1	attributeId	–	–	m		
	1.3.2	oldAttributeValue	–	–	o		
	1.3.3	newAttributeValue	–	–	m		
	1.4	notificationIdentifier	{ dmi-att 16 }	–	c76		
	1.5	correlatedNotifications	{ dmi-att 12 }	–	o		
	1.5.1	correlatedNotification	–	–	c:m		
	1.5.2	sourceObjectInst	–	–	c:o		
	1.5.2.1	distinguishedName	–	–	c:o.21		
	1.5.2.2	nonSpecificForm	–	–	c:o.21		
	1.5.2.3	localDistinguishedName	–	–	c:o.21		
	1.6	additionalText	{ dmi-att 7 }	–	o		
	1.7	additionalInformation	{ dmi-att 6 }	–	o		
2	2.1	sourceIndicator	{ dmi-att 26 }	–	o		
	2.2	attributeList	{ dmi-att 9 }	–	o		
	2.3	notificationIdentifier	{ dmi-att 16 }	–	c77		
	2.4	correlatedNotifications	{ dmi-att 12 }	–	o		
	2.4.1	correlatedNotification	–	–	c:m		
	2.4.2	sourceObjectInst	–	–	c:o		
	2.4.2.1	distinguishedName	–	–	c:o.22		
	2.4.2.2	nonSpecificForm	–	–	c:o.22		
	2.4.2.3	localDistinguishedName	–	–	c:o.22		
	2.5	additionalText	{ dmi-att 7 }	–	o		
	2.6	additionalInformation	{ dmi-att 6 }	–	o		
3	3.1	sourceIndicator	{ dmi-att 26 }	–	o		
	3.2	attributeList	{ dmi-att 9 }	–	o		
	3.3	notificationIdentifier	{ dmi-att 16 }	–	c78		
	3.4	correlatedNotifications	{ dmi-att 12 }	–	o		
	3.4.1	correlatedNotification	–	–	c:m		
	3.4.2	sourceObjectInst	–	–	c:o		
	3.4.2.1	distinguishedName	–	–	c:o.23		
	3.4.2.2	nonSpecificForm	–	–	c:o.23		
	3.4.2.3	localDistinguishedName	–	–	c:o.23		
	3.5	additionalText	{ dmi-att 7 }	–	o		
	3.6	additionalInformation	{ dmi-att 6 }	–	o		
c76: if D.25/1.5 then m else –. c77: if D.25/2.4 then m else –. c78: if D.25/3.4 then m else –.							

D.8.5 Parameters

The supplier of the implementation shall state which items are supported in Table D.26 and if necessary provide additional information.

Table D.26 – Parameter support

Index	Parameter template label	Value of parameter identifier	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	invalidAccessControlFilter	{ac-par 1}	–	c79			
c79: if D.23/7 or D.23/9 then o else –.							

D.9 ACL initiators managed object class

D.9.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the ACL initiators managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.27.

Table D.27 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	aclInitiators	{ac-obj 8}		

If the answer to the actual class question in the managed object class support Table D.27 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.28.

Table D.28 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.9.2 Packages

See Table D.29.

Table D.29 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c80		
3	allomorphicPackage	{dmi-pkg 17}	–	c81		
4	accessControlPackage	–	–	m		
5	initiatorsPackage	–	–	m		
6	aclPackage	{ac-pkg 12}	–	m		
c80: if D.29/3 then m else –.						
c81: if D.26/1b then – else m.						

D.9.3 Attributes

See Table D.30.

Table D.30 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c82		c82	
4	allomorphs	{dmi-att 50}	–	c83		c83	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	initiatorACImandated	{ac-att 14}	–	m		m	
7	accessControlList	{ac-att 1}	–	m		m	

Table D.30 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	m		–		–		m		
7	m		m		m		–		
c82: if D.30/2 then m else –.									
c83: if D.30/3 then m else –.									

D.9.4 Notifications

See Table D.31.

Table D.31 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non-confirmed	
1	attributeValueChange	{dmi-not 1}	–	m			
2	objectCreation	{dmi-not 6}	–	m			
3	objectDeletion	{dmi-not 7}	–	m			

Table D.31 (continued) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	sourceIndicator	{dmi-att 26}	–	o		
	1.2	attributeIdentifierList	{dmi-att 8}	–	o		
	1.3	attributeValueChangeDefinition	{dmi-att 10}	–	m		
	1.3.1	attributeId	–	–	m		
	1.3.2	oldAttributeValue	–	–	o		
	1.3.3	newAttributeValue	–	–	m		
	1.4	notificationIdentifier	{dmi-att 16}	–	c84		
	1.5	correlatedNotifications	{dmi-att 12}	–	o		
	1.5.1	correlatedNotification	–	–	c:m		
	1.5.2	sourceObjectInst	–	–	c:o		
	1.5.2.1	distinguishedName	–	–	c:o.24		
	1.5.2.2	nonSpecificForm	–	–	c:o.24		
	1.5.2.3	localDistinguishedName	–	–	c:o.24		
	1.6	additionalText	{dmi-att 7}	–	o		
	1.7	additionalInformation	{dmi-att 6}	–	o		
2	2.1	sourceIndicator	{dmi-att 26}	–	o		
	2.2	attributeList	{dmi-att 9}	–	o		
	2.3	notificationIdentifier	{dmi-att 16}	–	c85		
	2.4	correlatedNotifications	{dmi-att 12}	–	o		
	2.4.1	correlatedNotification	–	–	c:m		
	2.4.2	sourceObjectInst	–	–	c:o		
	2.4.2.1	distinguishedName	–	–	c:o.25		
	2.4.2.2	nonSpecificForm	–	–	c:o.25		
	2.4.2.3	localDistinguishedName	–	–	c:o.25		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	o		
	2.7	additionalText	{dmi-att 7}	–	o		
3	3.1	sourceIndicator	{dmi-att 26}	–	o		
	3.2	attributeList	{dmi-att 9}	–	o		
	3.3	notificationIdentifier	{dmi-att 16}	–	c86		
	3.4	correlatedNotifications	{dmi-att 12}	–	o		

Table D.31 (concluded) – Notification support

Index	Sub-index	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
3 (cont.)	3.4.1	correlatedNotification	–	–	c:m		
	3.4.2	sourceObjectInst	–	–	c:o		
	3.4.2.1	distinguishedName	–	–	c:o.26		
	3.4.2.2	nonSpecificForm	–	–	c:o.26		
	3.4.2.3	localDistinguishedName	–	–	c:o.26		
	3.5	additionalText	{ dmi-att 7 }	–	o		
	3.6	additionalInformation	{ dmi-att 6 }	–	o		
c84: if D.31/1.5 then m else –. c85: if D.31/2.4 then m else –. c86: if D.31/3.4 then m else –.							

D.10 Capability initiators managed object class

D.10.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the capability initiators managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed, in Table D.32.

Table D.32 – Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	capabilityInitiators	{ ac-obj 9 }		

If the answer to the actual class question in the managed object class support Table D.32 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.33.

Table D.33 – Actual class support

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.10.2 Packages

See Table D.34.

Table D.34 – Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	topPackage	–	–	m		
2	packagesPackage	{dmi-pkg 16}	–	c87		
3	allomorphicPackage	{dmi-pkg 17}	–	c88		
4	accessControlPackage	–	–	m		
5	initiatorsPackage	–	–	m		
6	capabilityPackage	{ac-pkg 13}	–	m		
c87: if D.34/3 then m else –. c88: if D.31/1b then – else m.						

D.10.3 Attributes

See Table D.35.

Table D.35 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by Create		Get	
				Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c89		c89	
4	allomorphs	{dmi-att 50}	–	c90		c90	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	initiatorACImandated	{ac-att 14}	–	m		m	
7	capabilityIdentitiesList	{ac-att 7}	–	m		m	

Table D.35 (concluded) – Attribute support

Index	Replace		Add		Remove		Set to Default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	m		–		–		m		
7	m		–		–		–		
c89: if D.34/2 then m else –.									
c90: if D.34/3 then m else –.									

D.10.4 Notifications

See Table D.36.

Table D.36 – Notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Confirmed	Non- confirmed	
1	attributeValueChange	{dmi-not 1}	–	m			
2	objectCreation	{dmi-not 6}	–	m			
3	objectDeletion	{dmi-not 7}	–	m			