UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T X.741

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

(04/95)

RÉSEAUX DE COMMUNICATION DE DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS

GESTION OSI

TECHNOLOGIES DE L'INFORMATION – INTERCONNEXION DES SYSTÈMES OUVERTS – GESTION-SYSTÈMES: OBJETS ET ATTRIBUTS DE CONTRÔLE D'ACCÈS

Recommandation UIT-T X.741

(Antérieurement «Recommandation du CCITT»)

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Au sein de l'UIT-T, qui est l'entité qui établit les normes mondiales (Recommandations) sur les télécommunications, participent quelque 179 pays membres, 84 exploitations de télécommunications reconnues, 145 organisations scientifiques et industrielles et 38 organisations internationales.

L'approbation des Recommandations par les membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution nº 1 de la Conférence mondiale de normalisation des télécommunications (CMNT) (Helsinki, 1993). De plus, la CMNT, qui se réunit tous les quatre ans, approuve les Recommandations qui lui sont soumises et établit le programme d'études pour la période suivante.

Dans certains secteurs de la technologie de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI. Le texte de la Recommandation X.741 de l'UIT-T a été approuvé le 10 avril 1995. Son texte est publié, sous forme identique, comme Norme internationale ISO/CEI 10164-9.

NOTE

Dans la présente Recommandation, l'expression «Administration» est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

RECOMMANDATIONS UIT-T DE LA SÉRIE X

RÉSEAUX DE COMMUNICATION DE DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS

(Février 1994)

ORGANISATION DES RECOMMANDATIONS DE LA SÉRIE X

Domaine	Recommandations
RÉSEAUX PUBLICS POUR DONNÉES	
Services et services complémentaires	X.1-X.19
Interfaces	X.20-X.49
Transmission, signalisation et commutation	X.50-X.89
Aspects réseau	X.90-X.149
Maintenance	X.150-X.179
Dispositions administratives	X.180-X.199
INTERCONNEXION DES SYSTÈMES OUVERTS	
Modèle et notation	X.200-X.209
Définition des services	X.210-X.219
Spécifications des protocoles en mode connexion	X.220-X.229
Spécifications des protocoles en mode sans connexion	X.230-X.239
Formulaires PICS	X.240-X.259
Identification des protocoles	X.260-X.269
Protocoles de sécurité	X.270-X.279
Objets gérés de couche	X.280-X.289
Test de conformité	X.290-X.299
INTERFONCTIONNEMENT DES RÉSEAUX	
Considérations générales	X.300-X.349
Systèmes mobiles de transmission de données	X.350-X.369
Gestion	X.370-X.399
SYSTÈMES DE MESSAGERIE	X.400-X.499
ANNUAIRE	X.500-X.599
RÉSEAUTAGE OSI ET ASPECTS DES SYSTÈMES	
Réseautage	X.600-X.649
Dénomination, adressage et enregistrement	X.650-X.679
Notation de syntaxe abstraite numéro un (ASN.1)	X.680-X.699
GESTION OSI	X.700-X.799
SÉCURITÉ	X.800-X.849
APPLICATIONS OSI	
Engagement, concomitance et rétablissement	X.850-X.859
Traitement des transactions	X.860-X.879
Opérations distantes	X.880-X.899
TRAITEMENT OUVERT RÉPARTI	X.900-X.999

TABLE DES MATIÈRES

Résumé

La présente Recommandation | Norme internationale spécifie un modèle de sécurité pour le contrôle d'accès ainsi que les informations de gestion qui sont nécessaires afin de créer et d'administrer le contrôle d'accès associé à la gestion des systèmes OSI. La politique de sécurité adoptée dans chaque instance d'utilisation n'est pas spécifiée et est laissée aux soins des réalisateurs. La présente Spécification est générique et applicable à la gestion de la sécurité de nombreux types d'application. Il est prévu qu'elle soit utilisée par les réseaux RGT.

NORME INTERNATIONALE

RECOMMANDATION UIT-T

TECHNOLOGIES DE L'INFORMATION – INTERCONNEXION DES SYSTÈMES OUVERTS – GESTION-SYSTÈMES: OBJETS ET ATTRIBUTS DE CONTRÔLE D'ACCÈS

1 Domaine d'application

Les spécifications contenues dans les articles ci-dessous sont applicables à la fourniture du contrôle d'accès dans les applications utilisant les services et protocoles de gestion OSI.

La présente Recommandation | Norme internationale:

- établit les besoins de l'utilisateur pour la fourniture du service de contrôle d'accès dans les applications utilisant les services et protocoles de gestion OSI;
- interprète et applique le modèle général de contrôle d'accès qui est défini dans la Rec. UIT-T X.812 | ISO/CEI 10181-3 et qui doit être utilisé dans les applications de gestion faisant appel aux services et protocoles de gestion OSI;
- définit des procédures pour l'application de règles de contrôle d'accès en liaison avec l'utilisation des services et protocoles de gestion OSI;
- définit des classes d'objets gérés et des types d'attribut:
  a) qui représentent certaines des informations de contrôle d'accès pouvant être utilisées pour fournir le service de contrôle d'accès; et
  b) qui ne peuvent être utilisés que lorsque la gestion des informations de contrôle d'accès doit être réalisée par la gestion des systèmes;
- spécifie le protocole nécessaire à l'échange des informations de contrôle d'accès définies dans la présente Recommandation | Norme internationale, lorsque l'échange est réalisé selon la gestion des systèmes OSI;
- spécifie les prescriptions de conformité pour les systèmes ouverts revendiquant la compatibilité avec le service de contrôle d'accès pour les applications faisant appel aux services et protocoles de gestion OSI;
- spécifie les prescriptions de conformité pour les systèmes ouverts revendiquant la compatibilité avec le service de contrôle d'accès défini dans la présente Recommandation | Norme internationale.

Les informations de contrôle d'accès figurant dans la présente Recommandation | Norme internationale pourront être utilisées pour mettre en œuvre des modes de contrôle d'accès fondés sur des listes de contrôle d'accès, sur des capacités, sur des étiquettes de sécurité et sur des contraintes contextuelles.

La présente Recommandation | Norme internationale:

- ne définit pas de politique de contrôle d'accès pour les applications qui font appel aux services et protocoles de gestion OSI;
- ne définit pas les domaines de sécurité (ou de gestion) dans lesquels une politique de contrôle d'accès peut être imposée;
- ne définit pas la manière dont les composantes d'une fonction de contrôle d'accès seront mises en œuvre ni à quel endroit ces composantes devront se trouver;
- ne spécifie pas la forme d'une quelconque information de contrôle d'accès qui est enregistrée à titre temporaire ou permanent dans un système ouvert;
- ne spécifie ni ne prescrit aucun mécanisme de contrôle d'accès particulier;
- ne prescrit pas de gérer les informations de contrôle d'accès ni, si elles doivent l'être, qu'elles le soient au moyen de la gestion des systèmes OSI;

- ne décrit pas la manière dont les entités d'application de gestion agissent, en cours de communication, pour prendre des décisions de contrôle d'accès au nom ou pour le compte d'une tierce partie quelconque;
- ne spécifie aucune exigence de conformité pour le paramètre de contrôle d'accès défini dans la présente Recommandation | Norme internationale.

2 Références normatives

Les Recommandations et les Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation | Norme internationale. Au moment de la publication, les éditions indiquées étaient en vigueur. Toute Recommandation ou Norme est sujette à révision et les parties prenantes aux accords fondés sur la présente Recommandation | Norme internationale sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur. Le Bureau de la normalisation des télécommunications de l'UIT tient à jour une liste des Recommandations UIT-T en vigueur.

2.1 Recommandations | Normes internationales identiques

- Recommandation UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Technologie de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base: Le modèle de référence de base.*
- Recommandation UIT-T X.509 (1993) | ISO/CEI 9594-8:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: Cadre d'authentification*.
- Recommandation X.701 du CCITT (1992)[1] | ISO/CEI 10400:1992[1], *Technologies de l'information – Interconnexion des systèmes ouverts – Aperçu général de la gestion des systèmes*.
- Recommandation X.720 du CCITT (1992) | ISO/CEI 10165-1:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: Modèle d'information de gestion*.
- Recommandation X.721 du CCITT (1992) | ISO/CEI 10165-2:1992, *Technologie de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: Définition des informations de gestion*.
- Recommandation X.722 du CCITT (1992) | ISO/CEI 10165-4:1992, *Technologie de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: Directives pour la définition des objets gérés*.
- Recommandation UIT-T X.724 (1993) | ISO/CEI 10165-6:1994, *Technologie de l'information – Interconnexion des systèmes ouverts – Structure de l'information de gestion: Spécifications et directives pour l'établissement des formulaires de déclaration de conformité d'instances associés à la gestion OSI*.
- Recommandation X.730 du CCITT (1992) | ISO/CEI 10164-1:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de gestion des objets*.
- Recommandation X.731 du CCITT (1992) | ISO/CEI 10164-2:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de gestion d'états*.
- Recommandation X.732 du CCITT (1992) | ISO/CEI 10164-3:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Attributs relationnels*.
- Recommandation X.734 du CCITT (1992) | ISO/CEI 10164-5:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de gestion des rapports d'événement*.
- Recommandation X.736 du CCITT (1992) | ISO/CEI 10164-7:1992, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de signalisation des alarmes de sécurité*.
- Recommandation X.740 du CCITT (1992) | ISO/CEI 10164-8:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de piste de vérification de sécurité*.

---

[1] Modifiée par Rec. UIT-T X.701/Cor.2 | ISO/CEI 10040/Cor.2.

– Recommandation UIT-T X.810[2] | ISO/CEI 10181-1 ...[2], *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité dans les systèmes ouverts: Vue d'ensemble*.

– Recommandation UIT-T X.812[2] | ISO/CEI 10181-3 ...[2], *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité dans les systèmes ouverts: Contrôle d'accès*.

2.2 Paires de Recommandations | Normes internationales équivalentes par leur contenu technique

– Recommandation X.208 du CCITT (1988), *Spécification de la syntaxe abstraite numéro un (ASN.1)*.

ISO/CEI 8824:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – Spécification de la notation de syntaxe abstraite numéro un (ASN.1)*.

– Recommandation X.209 du CCITT (1988), *Spécification des règles de codage de base pour la notation de syntaxe abstraite numéro un (ASN.1)*.

ISO/CEI 8825:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – Spécification de règles de base pour coder la notation de la syntaxe abstraite numéro un (ASN.1)*.

– Recommandation X.217 du CCITT (1992), *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service applicable à l'élément de service de contrôle d'association*.

ISO 8649:1988[3], *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Définition du service pour l'élément de service de contrôle d'association.*

– Recommandation X.227 du CCITT (1992), *Technologies de l'information – Interconnexion des systèmes ouverts – Protocole en mode connexion applicable à l'élément de service de contrôle d'association: Spécification du protocole*.

ISO 8650:1988[4], *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Spécification du protocole pour l'élément de service de contrôle d'association*.

– Recommandation X.290 du CCITT (1992), *Cadre général et méthodologie des tests de conformité OSI pour les Recommandations sur les protocoles pour les applications du CCITT – Concepts généraux*.

ISO/CEI 9646-1:1994, *Technologies de l'information – Interconnexion de systèmes ouverts – Cadre général et méthodologie des tests de conformité – Partie 1: Concepts généraux*.

– Recommandation X.291 du CCITT (1992), *Cadre général et méthodologie des tests de conformité OSI pour les Recommandations sur les protocoles pour les applications du CCITT – Spécification de suite de tests abstraite.*

ISO/CEI 9646-2:1994, *Technologies de l'information – Interconnexion de systèmes ouverts – Cadre général et méthodologie des tests de conformité OSI – Partie 2: Spécification des suites de tests abstraites.*

– Recommandation UIT-T X.296[2], *Cadre général et méthodologie des tests de conformité OSI pour les Recommandations sur les protocoles pour les applications de l'UIT-T – Déclarations de conformité d'instance.*

ISO/CEI 9646-7:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Cadre général et méthodologie des tests de conformité OSI – Partie 7: Implementation conformance statements.*

– Recommandation X.700 du CCITT (1992), *Cadre de gestion pour l'interconnexion des systèmes ouverts pour les applications du CCITT*.

ISO/CEI 7498-4:1989, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Partie 4: Cadre général de gestion*.

– Recommandation X.710 du CCITT (1991), *Définition du service commun de transfert d'informations de gestion pour les applications du CCITT*.

ISO/CEI 9595:1991, *Technologies de l'information – Interconnexion de systèmes ouverts – Définition du service commun d'informations de gestion*.

– Recommandation X.711 du CCITT (1991), *Spécification du protocole commun de transfert d'informations de gestion pour les applications du CCITT*.

---

[2] Actuellement à l'état de projet.

[3] Modifiée par ISO/CEI 8649:1988/Amd.1:1990.

[4] Modifiée par ISO/CEI 8650:1988/Amd.1:1990.

ISO/CEI 9596-1:1991, *Technologies de l'information – Interconnexion de systèmes ouverts – Protocole commun d'information de gestion – Partie 1: Spécification*.

– Recommandation X.800 du CCITT (1991), *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT*.

ISO 7498-2:1989, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Partie 2: Architecture de sécurité*.

3 Définitions

Pour les besoins de la présente Recommandation | Norme internationale, les définitions suivantes s'appliquent.

3.1 Définitions relatives au modèle de référence

La présente Recommandation | Norme internationale utilise le terme suivant, qui est défini dans la Rec. UIT-T X.200 | ISO/CEI 7498-1:

– système ouvert.

3.2 Définitions relatives à l'architecture de sécurité

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. X.800 du CCITT | ISO 7498-2:

a) contrôle d'accès;

b) liste de contrôle d'accès;

c) authentification;

d) capacité;

e) étiquette de sécurité;

f) politique de sécurité.

3.3 Définitions relatives au cadre de gestion

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. X.700 du CCITT | ISO/CEI 7498-4:

a) objet géré;

b) entité d'application de gestion des systèmes.

3.4 Définitions relatives à la vue d'ensemble des cadres de sécurité

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. UIT-T X.810 | ISO/CEI 10181-1:

a) certificat de sécurité;

b) domaine de sécurité;

c) jeton de sécurité.

3.5 Définitions relatives au cadre de contrôle d'accès

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. UIT-T X.812 | ISO/CEI 10181-3:

a) certificat de contrôle d'accès;

b) information décisionnelle de contrôle d'accès (ADI);

c) fonction décisionnelle de contrôle d'accès (ADF);

d) fonction exécutive de contrôle d'accès (AEF);

e) information de contrôle d'accès (ACI);

f) politique de contrôle d'accès;

g) information contextuelle;

h) initiateur;

i) information décisionnelle de contrôle d'accès issue de l'initiateur (information ADI d'initiateur);

j) information de contrôle d'accès issue de l'initiateur (information ACI d'initiateur);

k) information de contrôle d'accès liée à l'initiateur (information ACI liée à l'initiateur);

l) information décisionnelle de contrôle d'accès issue de l'opérande (information ADI d'opérande);

m) information de contrôle d'accès liée à l'opérande (information ACI liée à l'opérande);

n) information ADI retenue;

o) cible;

p) information décisionnelle de contrôle d'accès issue de la cible (information ADI de cible);

q) information de contrôle d'accès issue de la cible (information ACI de cible);

r) information de contrôle d'accès liée à la cible (information ACI liée à la cible).

3.6 Définitions relatives à la vue d'ensemble de la gestion des systèmes

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. X.701 du CCITT | ISO/CEI 10040:

a) définitions génériques;

b) classe d'objets gérés;

c) formulaire de déclaration de conformité d'objet de gestion (MOCS);

d) formulaire de déclaration de conformité d'information de gestion (MICS);

e) opération de gestion;

f) formulaire MICS;

g) formulaire MOCS;

h) notification.

3.7 Définitions relatives au modèle d'informations de gestion

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. X.720 du CCITT | ISO/CEI 10165-1:

a) action;

b) caractéristique.

3.8 Définitions relatives au formulaire de déclaration de conformité d'instance

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. UIT-T X.724 | ISO/CEI 10165-6:

a) déclaration de conformité de relation gérée (MRCS);

b) récapitulatif de conformité de gestion (MCS);

c) formulaire de déclaration de définition d'information de gestion (MIDS);

d) formulaire MCS;

e) formulaire MRCS.

3.9 Définitions relatives à la gestion des rapports d'événement

La présente Recommandation | Norme internationale utilise le terme suivant, qui est défini dans la Rec. X.734 du CCITT | ISO/CEI 10164-5:

– discriminateur de retransmission (de rapport) d'événement.

3.10 Définitions relatives aux tests de conformité OSI

La présente Recommandation | Norme internationale utilise les termes suivants, qui sont définis dans la Rec. X.290 du CCITT | ISO/CEI 9646-1:

a) formulaire PICS;

b) déclaration de conformité d'instance de protocole;

c) déclaration de conformité d'un système.

3.11 Autres définitions

3.11.1 **autorité domaniale de sécurité**: Entité chargée de la mise en œuvre d'une politique de sécurité.

3.11.2 **type d'opération**: Action effectuée sur un objet géré à la suite d'une demande de gestion.

4 Symboles et abréviations

ACC Certificat de contrôle d'accès (*access control certificate*)

ADI Information décisionnelle de contrôle d'accès (*access control decision information*)

ACI Information de contrôle d'accès (*access control information*)

ACL Liste de contrôle d'accès (*access control list*)

ADF Fonction décisionnelle de contrôle d'accès (*access control decision function*)

AEF Fonction exécutive de contrôle d'accès (*access control enforcement function*)

CMIS Service commun d'information de gestion (*common management information service*)

CMIP Protocole commun d'information de gestion (*common management information protocol*)

ICS Déclaration de conformité d'instance (*implementation conformance statement*)

MAPDU Unité de données du protocole d'application de gestion (*management application protocol data unit*)

MCS Récapitulatif de conformité de gestion (*management conformance summary*)

MICS Déclaration de conformité d'information de gestion (*management information conformance statement*)

MIDS Déclaration de définition d'information de gestion (*management information definition statement*)

MOCS Déclaration de conformité d'objet de gestion (*managed object conformance statement*)

MRCS Déclaration de conformité de relation gérée (*managed relationship conformance statement*)

PAC Certificat d'accès privilégié (*privilege attribute certificate*)

PICS Déclaration de conformité d'instance de protocole (*protocol implementation conformance statement*)

SMAE Entité d'application de gestion des systèmes (*systems management application entity*)

5 Conventions

La présente Recommandation | Norme internationale fait appel aux techniques de notation spécifiées dans la Rec. X.722 du CCITT | ISO/CEI 10165-4 pour la définition des objets gérés et de leurs attributs.

6 Prescriptions

Un utilisateur des services de gestion OSI a besoin qu'un mécanisme de contrôle d'accès, unique ou multiple, empêche tout accès non autorisé aux applications de gestion et aux informations de gestion.

Le contrôle d'accès aux informations de gestion est requis dans chacun des trois cas suivants:

a) pour protéger les informations de gestion d'une création, suppression, modification ou divulgation non autorisée, due à une opération de gestion OSI;

b) pour faire en sorte que seuls les initiateurs soient en mesure d'utiliser les opérations de gestion pour lesquelles des droits d'accès ont été accordés au cours de l'établissement de l'association d'application; et

c) pour empêcher l'envoi d'informations de gestion à des destinataires non autorisés, en raison de rapports d'événement de type confirmé ou non confirmé.

NOTE – Pour être complet, il faut ajouter la nécessité du contrôle d'accès à des associations. Ce sujet fera l'objet d'une étude complémentaire.

Divers niveaux de contrôle d'accès peuvent être requis. Par exemple, certains utilisateurs peuvent recevoir un accès en lecture et en écriture à des attributs spécifiques, alors que d'autres ne peuvent avoir qu'un accès en lecture ou aucun accès. Certains utilisateurs peuvent ne se voir accorder des droits d'accès qu'à certains objets de gestion spécifiques alors que d'autres utilisateurs peuvent accéder à un ensemble différent d'objets de gestion. Pour les opérations de gestion, des restrictions d'accès sont nécessaires pour protéger des objets gérés, des attributs particuliers d'objets gérés, des valeurs d'attributs particuliers, un contexte d'accès ou des actions associées à l'objet géré.

Il est prescrit de prévoir un paramètre de contrôle d'accès pouvant être utilisé dans les échanges d'informations de gestion au moyen du service CMIS.

Il est également prescrit que les systèmes ouverts assurant le contrôle d'accès à des applications de gestion OSI, ainsi que les protocoles à observer, manifestent le même comportement en termes de contrôle d'accès lors de l'application du même ensemble de règles de contrôle d'accès.

Il est prescrit que les dispositions de la présente Recommandation | Norme internationale n'empêchent pas l'utilisation de mécanismes de contrôle d'accès non identifiés dans son cadre.

De manière à réaliser la gestion des informations de contrôle d'accès au moyen de la gestion des systèmes OSI, il est nécessaire:

– que ces informations soient modélisées sous la forme d'objets gérés de façon qu'elles puissent être créées, supprimées, modifiées et lues;

– que l'on puisse déterminer les cibles auxquelles un initiateur peut avoir accès;

– et que l'on puisse déterminer les initiateurs qui peuvent avoir accès à une cible donnée.

Il est prescrit que l'on puisse empêcher la découverte – au moyen d'un analyseur-lecteur de signal – des objets gérés qui sont contenus dans un sous-arbre d'informations de gestion.

Il est prescrit que l'on puisse empêcher l'application générale d'une opération (comme une suppression) au moyen d'un analyseur-conditionneur de signal.

Il est prescrit que l'on puisse attribuer des étiquettes de sécurité uniques à des cibles spécifiques.

7 Interprétation du modèle de contrôle d'accès

7.1 Vue d'ensemble du modèle de contrôle d'accès

Le contrôle d'accès pour la gestion OSI est fondé sur le modèle défini à cette fin dans la Rec. UIT-T X.812 | ISO/CEI 10181-3. Dans ce modèle de base, les fonctions de décision et d'exécution du contrôle d'accès sont interposées entre l'initiateur et la cible. La présente Recommandation | Norme internationale montre comment ce modèle s'applique à la fourniture du contrôle d'accès à des applications utilisant les services et protocoles de gestion OSI.

La fonction décisionnelle de contrôle d'accès nécessite des informations placées sous l'appellation générique d'informations de contrôle d'accès (ACI), utilisées dans le processus de prise de décision. Les informations de contrôle d'accès peuvent être modélisées sous la forme d'informations de gestion et peuvent être décrites au moyen des techniques de notation spécifiées dans la Rec. X.722 du CCITT | ISO/CEI 10165-4. La présente Recommandation | Norme internationale définit certaines informations de contrôle d'accès sous la forme d'attributs de classes d'objets gérés, de manière que la gestion des systèmes OSI puisse servir à échanger ces informations entre des systèmes ouverts.

NOTE – La présente Recommandation | Norme internationale ne spécifie pas la forme ou la structure réelle d'une information de contrôle d'accès qui est stockée temporairement ou durablement dans un système ouvert. En revanche, elle spécifie la syntaxe abstraite des éléments d'information ACI qui peuvent être échangés par des systèmes ouverts au moyen de la gestion des systèmes OSI.

Tous les modes de contrôle d'accès identifiés dans la Rec. UIT-T X.812 | ISO/CEI 10181-3 – modes par liste ACL, par capacité, par contexte et par étiquette – sont applicables à la fourniture du service de contrôle d'accès aux applications utilisant les services et protocoles de gestion OSI. Ces modes de contrôle d'accès peuvent être utilisés isolément ou en combinaison, de manière que le contrôle d'accès puisse être assuré conformément à une politique de contrôle d'accès appropriée au domaine de sécurité.

Le contrôle d'accès est régi par une politique de contrôle d'accès. Lorsqu'une politique spécifique de contrôle d'accès est appliquée à un groupe d'éléments, la combinaison de ces derniers et de cette politique de contrôle d'accès délimite un domaine de sécurité spécifique. Une seule politique de contrôle d'accès est exécutée à un moment donné dans un domaine de sécurité donné.

L'authentification de l'utilisateur de la gestion OSI et l'authentification symétrique des entités SMAE sont hors du domaine d'application de la présente Recommandation | Norme internationale. Les procédures de contrôle d'accès qui y sont définies exigent toutefois l'emploi de procédures d'authentification au moment approprié. La Rec. X.217 du CCITT | ISO 8649, la Rec. X.227 du CCITT | ISO 8650 et la Rec. UIT-T X.509 | ISO/CEI 9594-8 décrivent des procédures d'authentification possibles.

Le contrôle d'accès pour la gestion OSI est spécifié sous la forme d'un ensemble de refus et d'autorisations d'exécuter des opérations de gestion. L'utilisateur de l'application de gestion qui invoque l'opération de gestion est l'initiateur et les éléments d'information de gestion identifiés par les paramètres de l'opération de gestion, par exemple des objets gérés et des attributs, se combinent pour former la cible (ou les cibles).

Le contrôle d'accès des rapports d'événement s'effectue par application des procédures de contrôle d'accès aux opérations de gestion se rapportant aux discriminateurs de retransmission de rapport d'événement.

7.2 Politiques de contrôle d'accès

Une politique de contrôle d'accès comporte un ou plusieurs ensembles de règles. Il appartient au maître d'œuvre de la politique de contrôle d'accès de veiller à ce que les règles de contrôle d'accès donnent une image fidèle de l'instanciation de la politique de contrôle d'accès.

Une politique de contrôle d'accès est une politique de gestion spécifique qui peut être soumise à une administration de politique de gestion.

NOTE – La présente Recommandation | Norme internationale ne spécifie pas la gestion des politiques de gestion et, en particulier, ne spécifie aucun moyen de vérification de l'intégrité des informations de contrôle d'accès.

Une politique de contrôle d'accès est exécutée au moyen d'un ou de plusieurs des modes de contrôle d'accès suivants:

- modes par liste ACL;
- modes par capacités;
- modes par contexte; et
- modes par étiquette.

Si certaines caractéristiques d'un objet géré (par exemple ses attributs) sont placées sous la dépendance de différentes politiques de sécurité, cet objet géré peut appartenir à plusieurs domaines de sécurité. Lorsque plusieurs politiques de contrôle d'accès s'appliquent à un même objet géré, la politique de contrôle d'accès à exécuter est celle qui est associée à la fois à l'initiateur et à la cible.

7.3 Informations de contrôle d'accès

Les informations de contrôle d'accès (ACI) comprennent:

- les règles de contrôle d'accès;
- l'identité de l'initiateur de la demande d'accès (information ACI d'initiateur);
- les capacités et habilitations de sécurité associées à l'initiateur (information ACI d'initiateur);
- les informations relatives à l'authentification de l'initiateur (information ADI d'initiateur);
- les identités d'information de gestion (cibles) auxquelles l'accès a été demandé (information ACI de cible);
- les capacités et habilitations de sécurité associées à la cible (information ACI de cible);
- les opérations autorisées qui peuvent être exécutées sur les informations de gestion (information ACI d'initiateur, information ACI de cible);
- les informations retenues par la fonction décisionnelle de contrôle d'accès pour usage ultérieur (informations ADI retenues); et
- les informations contextuelles.

7.3.1 règles de contrôle d'accès: Informations de contrôle d'accès qui représentent les opérations permises et les conditions de leur exécution dans un domaine de sécurité donné. Il y a cinq classes de règles de contrôle d'accès, qui doivent être appliquées par la fonction ADF:

- **règles de refus globales** – Règles de contrôle d'accès qui refusent l'accès à toutes les cibles. Si une règle globale refuse l'accès, aucune autre règle ne doit s'appliquer. Si une règle globale ne refuse pas l'accès, les règles de refus d'item sont appliquées;
- **règles de refus d'item** – Règles de contrôle d'accès qui refusent l'accès à certaines cibles. Si une règle d'item refuse l'accès, aucune autre règle ne doit s'appliquer. Si une règle d'item ne refuse pas l'accès, les règles de refus globales sont appliquées;
- **règles d'autorisation globales** – Règles de contrôle d'accès qui autorisent l'accès à toutes les cibles. Si une règle globale autorise l'accès, aucune autre règle ne doit s'appliquer. Si une règle globale n'autorise pas l'accès, les règles d'autorisation d'item sont appliquées;
- **règles d'autorisation d'item** – Règles de contrôle d'accès qui autorisent l'accès à certaines cibles. Si une règle d'autorisation autorise l'accès, aucune autre règle ne doit s'appliquer. Si une règle d'item n'autorise pas l'accès, les règles par défaut sont appliquées;
- **règles par défaut** – Règles de contrôle d'accès qui doivent être appliquées lorsque aucune autre règle n'a expressément autorisé ou refusé l'accès. Les règles par défaut doivent autoriser ou refuser l'accès.

7.3.2 information ACI liée à l'action: Information de contrôle d'accès (par exemple une étiquette de sécurité) qui est associée à l'information de gestion acheminée dans les opérations de gestion et dans les rapports d'événement. Les informations ACI liées à l'action peuvent également servir à créer et/ou à modifier des informations de contrôle d'accès associées à une cible.

7.3.3 information ACI liée à l'initiateur: Information de contrôle d'accès fournie par une demande de gestion issue de l'initiateur ou associée d'une autre façon à une telle demande. Ces informations peuvent prendre l'une des formes suivantes:

- identité de l'initiateur de l'opération de gestion;
- information insérée dans le paramètre contrôle d'accès d'opérations de gestion (par exemple le paramètre de contrôle d'accès dans le service CMIS ou le paramètre de contrôle d'accès aux informations d'utilisateur dans le protocole CMIP);
- information désignée par une autorité de domaine de sécurité; ou
- une combinaison des informations ci-dessus.

Le paramètre de contrôle d'accès acheminé par le service CMIS peut prendre la forme d'un certificat de sécurité ou d'un jeton de sécurité.

NOTE – L'identité de l'initiateur peut être communiquée par des mécanismes d'authentification au moyen de procédures locales qui sont hors du domaine d'application de la présente Recommandation | Norme internationale.

7.3.4 information ACI liée à la cible: Information de contrôle d'accès qui désigne l'information de gestion sur laquelle des opérations doivent être exécutées.

7.3.5 information contextuelle: Information de contrôle d'accès associée au contexte (par exemple l'heure du jour, le niveau d'authentification, le lieu, les ressources disponibles, la participation à une relation).

7.3.6 information ADI: Information décisionnelle de contrôle d'accès pour une action, pour un initiateur ou pour une cible, déduite de l'information ACI liée à l'action, à l'initiateur ou à la cible selon le cas, en vue de la prise d'une décision.

7.3.7 information ADI retenue: Information de contrôle d'accès qui est retenue par la fonction décisionnelle de contrôle d'accès. Conformément à la politique de contrôle d'accès, une partie de cette information peut être retenue pendant des périodes plus longues que la durée de vie d'une association. Les informations ADI retenues peuvent être utilisées par la fonction ADF pour évaluer des privilèges d'accès.

7.4 Procédures de contrôle d'accès

Les règles de contrôle d'accès spécifient les critères de sécurité qui doivent être satisfaits pour que l'on autorise l'accès à des informations de gestion. Ces règles peuvent prescrire l'exécution d'une partie ou de la totalité des procédures suivantes:

- validation des informations ACI liées à l'initiateur;
- identification de la cible;

- détermination de la décision d'accès;
- modification des informations ADI retenues;
- modification des informations ACI liées à la cible; et
- exécution de la décision.

Avant l'établissement d'une association, l'information de contrôle d'accès qui représente les règles de contrôle d'accès pour ce domaine de contrôle d'accès peut être produite et diffusée par une autorité de domaine de sécurité utilisant des mécanismes qui sont hors du domaine d'application de la présente Recommandation | Norme internationale.

Les procédures qui suivent spécifient les décisions d'accès qui doivent être prises et non pas le lieu où elles doivent l'être. La présente Recommandation | Norme internationale ne spécifie pas si les décisions doivent être prises dans le système gestionnaire, dans le système géré, dans ces deux systèmes ou ailleurs.

Il appartient à l'initiateur de fournir des informations de contrôle d'accès qui soient compatibles avec les mécanismes de contrôle d'accès spécifiés par la politique de sécurité.

L'emploi de tout ou partie de ces procédures ne préjuge pas celui d'autres procédures et d'autres mécanismes de contrôle, non spécifiés par la présente Recommandation | Norme internationale.

7.4.1 Validation des informations ACI liées à l'initiateur

Les informations ACI liées à l'initiateur peuvent être insérées dans le paramètre contrôle d'accès de la primitive de demande de service CMIS pour l'opération de gestion. Ces informations peuvent prendre la forme d'un certificat de sécurité (comme un certificat ACC de contrôle d'accès) ou d'un jeton de sécurité.

La politique de sécurité spécifie l'exécution de l'une des actions suivantes:

- l'intégrité des informations doit être validée au moyen de procédures qui sont hors du domaine d'application de la présente Recommandation | Norme internationale;
- la validité des informations doit être vérifiée au moyen d'une vérification du fait que leur origine est une autorité de domaine de sécurité reconnue;
- le contenu des informations doit être vérifié au moyen d'une vérification du fait que leur valeur est contenue dans une étendue autorisée.

7.4.2 Identification de la cible

Une cible est un élément d'information qui doit être protégé par un mode de contrôle d'accès. L'information de gestion est contenue dans l'arbre des informations de gestion. Un sous-arbre peut être choisi au moyen du paramètre visibilité du service CMIS. Si c'est le cas, l'ensemble du sous-arbre choisi est considéré comme une cible. La sélection peut être raffinée jusqu'à un degré de granularité supérieur au moyen des paramètres visibilité et filtre du service CMIS, qui permettent de choisir, dans l'arbre des informations de gestion, des objets gérés individuels et leurs attributs. Dans ce cas, tous les objets gérés sélectionnés forment, avec leurs attributs, la cible. Au degré de granularité le plus fin, il est possible de sélectionner des objets gérés individuels dans l'arbre des informations de gestion. Dans ce cas, la cible n'est constituée que de l'objet géré sélectionné et de ses caractéristiques.

Toute requête d'opération de gestion désigne une ou plusieurs cibles. Ces cibles sont identifiées comme suit:

a) lorsque le paramètre visibilité est présent dans la demande, l'ensemble du sous-arbre d'informations de gestion désigné par cette demande constitue une cible. C'est-à-dire que celle-ci est formée de la combinaison des paramètres suivants: classe des objets gérés de base, instance d'objet géré de base, visibilité, synchronisation, type d'opération, identificateur d'attribut, identificateur d'action, valeur de l'argument information d'attribut, valeur de l'argument information d'action;

b) lorsque les paramètres visibilité et filtre sont présents dans la demande, la cible est constituée des objets gérés choisis et de leurs caractéristiques. C'est-à-dire qu'une cible est constituée de chaque combinaison distincte (des paramètres classe des objets gérés de base, instance d'objet géré de base, valeur d'attribut, type d'opération, identificateur d'attribut, identificateur d'action, valeur de l'argument information d'attribut, valeur de l'argument information d'action) qui est formée à partir:

   - des objets gérés qui ont été sélectionnés par les paramètres classe d'objets gérés de base, instance d'objet géré de base, visibilité;
   - des éléments de l'item *filter* du paramètre filtre;

c) ou bien une cible est constituée par chaque objet géré distinct avec ses caractéristiques, sélectionné par l'opération. C'est-à-dire que, pour chaque objet géré qui a été sélectionné par les paramètres de sélection d'objet géré, une cible est formée par la combinaison des paramètres suivants: classe des objets gérés de base, classe des objets gérés, instance d'objet géré de base, instance d'objet géré, instance d'objet géré supérieur, instance d'objet géré de référence, instance d'objet géré valeur initiale, type d'opération, identificateur d'attribut, identificateur d'action, valeur de l'argument information d'attribut, valeur de l'argument information d'action.

NOTE – Une cible comporte l'opération (ou les opérations) nécessaire(s) pour l'objet géré.

7.4.3 Détermination de la décision d'accès

La fonction décisionnelle de contrôle d'accès doit exécuter toutes les procédures spécifiées par la politique de sécurité. Les informations ADI d'initiateur, les informations ADI retenues, les informations ADI de cible, les informations ADI d'opérande et les informations contextuelles pourront être utilisées au cours de ces procédures. La politique de sécurité peut comprendre tout ou partie des procédures indiquées au 7.4.3.1.

7.4.3.1 Procédures de décision d'accès

La procédure décrite au 7.4.3.1.1 doit être appliquée en premier.

7.4.3.1.1 Identifier les règles d'accès aux informations de gestion qui s'appliquent au domaine de sécurité de l'initiateur et de la cible.

7.4.3.1.2 Pour toutes les règles globales qui refusent l'accès par des initiateurs, effectuer les tests applicables qui sont décrits au 7.4.3.2. Si un de ces tests renvoie un message de succès, indiquer à la fonction exécutive de contrôle d'accès que la demande d'opération de gestion doit être rejetée puis appliquer les procédures décrites au 7.4.6 ou, si elles ne sont pas applicables, la procédure du 7.4.3.1.3.

7.4.3.1.3 Pour toutes les règles d'item qui refusent l'accès à des cibles, effectuer les tests applicables qui sont décrits au 7.4.3.2. Si un de ces tests renvoie un message de succès, indiquer à la fonction exécutive de contrôle d'accès que la demande d'opération de gestion doit être rejetée puis appliquer les procédures décrites au 7.4.6 ou, si elles ne sont pas applicables, la procédure du 7.4.3.1.4.

7.4.3.1.4 Pour toutes les règles globales qui autorisent l'accès par des initiateurs, effectuer les tests applicables qui sont décrits au 7.4.3.2. Si un de ces tests renvoie un message de succès, indiquer à la fonction exécutive de contrôle d'accès que la demande d'opération de gestion doit être acceptée puis appliquer les procédures décrites aux 7.4.4, 7.4.5 et 7.4.6 ou, si elles ne sont pas applicables, la procédure du 7.4.3.1.5.

7.4.3.1.5 Pour toutes les règles d'item qui autorisent l'accès à des cibles, effectuer les tests applicables qui sont décrits au 7.4.3.2. Si un de ces tests renvoie un message de succès, indiquer à la fonction exécutive de contrôle d'accès que la demande d'opération de gestion doit être acceptée puis appliquer les procédures décrites aux 7.4.4, 7.4.5 et 7.4.6 ou, si elles ne sont pas applicables, la procédure du 7.4.3.1.6.

7.4.3.1.6 Si la politique de sécurité n'a pas spécifié de règle autorisant ou refusant expressément l'accès à la cible, indiquer à la fonction exécutive de contrôle d'accès que la demande d'opération de gestion doit être acceptée ou refusée selon la règle par défaut pour cette opération, puis appliquer la réponse de refus par défaut. Si la demande d'opération de gestion doit être refusée, seules les procédures du 7.4.6 s'appliquent; si ce n'est pas le cas, appliquer les procédures des 7.4.4, 7.4.5 et 7.4.6.

7.4.3.2 Tests de décision d'accès

Les tests suivants sont à la disposition de la fonction décisionnelle de contrôle d'accès, en accord avec la politique de sécurité. Chaque test reçoit des informations associées à l'initiateur ainsi qu'une règle indiquant les opérations à effectuer sur les cibles. Chaque test renvoie un opérateur booléen dont la valeur indique si la proposition «l'initiateur satisfait à la règle» est vraie ou fausse.

L'évaluation d'une opération demandée sur une cible spécifique peut nécessiter des informations au sujet d'une classe ou instance d'objets gérés de type supérieur, valeurs de référence ou valeurs initiales. Ces informations seront révélées à l'initiateur. Si celui-ci n'a pas accès aux informations requises (permission GET) au sujet de la classe ou instance d'objets gérés de type supérieur, valeurs de référence ou valeurs initiales, selon ce qui est déterminé conformément aux alinéas a) à e) ci-dessous, la règle doit renvoyer une évaluation de type FALSE (proposition fausse).

a) Lorsque cela est requis par un mode de type liste ACL, l'identité, le groupe ou le rôle de l'initiateur doit être comparé aux identités des initiateurs qui sont associés à la règle. Si une correspondance à l'identique est constatée et que l'opération ainsi que la cible associées à la demande soient compatibles avec les opérations et cibles spécifiées par la règle, celle-ci doit renvoyer une évaluation de type TRUE

(proposition vraie). Si une correspondance à l'identique n'est pas constatée ou que l'opération ou la cible associée à la demande soit incompatible avec les opérations et cibles spécifiées par la règle, celle-ci doit renvoyer une évaluation de type FALSE.

b) Lorsque cela est requis par un mode de type capacités, l'identité associée à l'initiateur doit être comparée aux identités des initiateurs qui sont associés à la règle. Si une correspondance à l'identique est constatée (permettant d'utiliser la capacité désignée) et que l'opération ainsi que la cible associées à la demande soient compatibles avec les opérations et cibles spécifiées par la règle associée à cette capacité, cette règle doit renvoyer une évaluation de type TRUE (proposition vraie). Si une correspondance à l'identique n'est pas constatée ou que l'opération ou la cible associée à la demande soit incompatible avec les opérations et cibles spécifiées par la règle associée à la capacité, cette règle doit renvoyer une évaluation de type FALSE.

c) Lorsque cela est requis par un mode de type contexte, les conditions contextuelles associées à la règle doivent être contrôlées. Si toutes les conditions contextuelles sont satisfaites, la règle doit renvoyer une évaluation de type TRUE et de type FALSE si ce n'est pas le cas.

d) Lorsque cela est requis par un mode de type étiquette, l'étiquette associée à l'initiateur doit être validée en fonction de l'étiquette associée à la cible. Si l'étiquette associée à l'initiateur est déterminée par l'algorithme de comparaison d'étiquettes comme étant compatible avec l'étiquette associée à la cible, la règle doit renvoyer une évaluation de type TRUE et de type FALSE si ce n'est pas le cas.

e) Lorsque cela est requis par la politique de sécurité, effectuer tout autre test associé à la règle.

Lors de l'évaluation des règles de contrôle d'accès dans un domaine de sécurité utilisant une combinaison de mécanismes de contrôle d'accès, une seule règle doit répondre aux exigences de tous les mécanismes associés à cette règle.

NOTE – Certaines politiques de sécurité peuvent appliquer à la règle des combinaisons des modes de contrôle d'accès. Dans ce cas, la politique de contrôle d'accès doit préciser l'ordre de préséance de ces modes.

7.4.4 Modification d'informations ADI

Si cela est spécifié par la politique de sécurité, les informations ADI retenues par la fonction ADF peuvent être modifiées au moyen des informations ADI d'initiateur qui ont été fournies avec la demande d'opération de gestion. Les informations en cause sont les suivantes:

- informations ACI associées en permanence à l'initiateur;
- informations ADI retenues lors de précédentes associations;
- informations ACI fournies par l'initiateur;
- informations ACI obtenues à l'issue de la procédure d'authentification;
- informations contextuelles.

NOTE – Les mécanismes de gestion, lecture, mémorisation et récupération d'informations ADI retenues sont hors du domaine d'application de la présente Recommandation | Norme internationale.

7.4.5 Modification d'informations ADI de cible

Si cela est spécifié par la politique de sécurité, les informations ADI associées à la cible peuvent être modifiées au moyen des informations ADI d'action qui ont été fournies avec la demande d'opération de gestion, conformément aux procédures suivantes.

7.4.5.1 Pour l'opération de création, les informations ADI d'action peuvent être utilisées afin de créer des informations ADI de cible correspondant expressément au nouvel objet géré à créer. Les informations ADI associées à d'autres cibles ne peuvent pas être modifiées.

7.4.5.2 Pour l'opération de suppression, les informations ADI d'action peuvent être utilisées afin de modifier ou supprimer des informations ADI de cible correspondant expressément à l'objet (ou aux objets) géré(s) à supprimer. Les informations ADI associées à d'autres cibles ne peuvent pas être modifiées.

7.4.5.3 Pour les opérations de remplacement de la valeur d'attribut, de remplacement par valeur par défaut, d'adjonction de membre et de retrait de membre, les informations ADI d'action peuvent être utilisées afin de modifier des informations ADI de cible correspondant expressément à l'attribut (ou aux attributs) de cible à modifier par l'opération. Les informations ADI associées à d'autres cibles ne peuvent pas être modifiées.

NOTE – Des informations ADI de cible peuvent être modifiées indépendamment des opérations de gestion décrites ci-dessus. Des informations ADI associées à des cibles peuvent être créées, supprimées ou modifiées par des moyens qui sont hors du domaine d'application de la présente Recommandation | Norme internationale. Par exemple, la création d'un objet géré qui représente une ressource peut aussi créer une information ADI de cible en conséquence directe de la création de cet objet géré. S'il est permis d'utiliser la gestion des systèmes OSI pour gérer les informations ACI, les informations ADI de cible peuvent être modifiées conformément à l'article 8.

7.4.6 Exécution de la décision

La fonction exécutive de contrôle d'accès (AEF) a la tâche d'exécuter la décision de politique qui est indiquée par la fonction décisionnelle de contrôle d'accès (ADF).

Les paragraphes suivants décriront:

a) les significations des éventuelles réponses de refus d'accès qui peuvent être renvoyées à l'initiateur à la suite d'une action due à la fonction AEF;

b) la procédure que la fonction AEF doit appliquer à la suite de la réception d'une demande d'opération de gestion assortie d'une invalidité d'information ACI liée à l'initiateur;

c) la procédure que la fonction AEF doit appliquer à la suite du rejet d'une demande d'opération de gestion en raison d'un refus d'accès par une règle globale;

d) la procédure que la fonction AEF doit appliquer à la suite du rejet d'une demande d'opération de gestion en raison d'un refus d'accès par une règle d'item ou par la règle par défaut;

e) les prescriptions applicables à la journalisation et à la signalisation des événements importants pour l'application d'un mode de contrôle d'accès.

7.4.6.1 Exécution d'un refus d'accès

L'exécution d'un refus d'accès nécessite la spécification de la réponse de refus appropriée qu'il convient de renvoyer à l'initiateur, ainsi que la spécification des cibles spécifiques auxquelles l'accès est refusé.

Une des actions suivantes de refus en réponse doit être spécifiée:

– une réponse de refus est donnée et la fonction AEF doit faire en sorte que la réponse d'erreur avec refus d'accès soit renvoyée à l'initiateur si le service d'opération de gestion a été demandé en mode confirmé;

– aucune réponse n'est donnée et la fonction AEF doit faire en sorte qu'aucune réponse ne soit renvoyée à l'initiateur;

– l'association est dissoute et la fonction AEF doit faire en sorte que la procédure A-ABORT de l'élément ACSE soit invoquée; ou

– une réponse de type false est donnée et la fonction AEF doit faire en sorte que le message information de gestion incorrecte soit renvoyé à l'initiateur si le service d'opération de gestion a été demandé en mode confirmé.

Si aucune action de refus en réponse n'est spécifiée, l'action par défaut de refus en réponse relève de la politique locale.

La réponse de refus doit être assortie d'un niveau de granularité. Les granularités de refus possibles sont les suivantes:

– refus unique au niveau de toute la demande de gestion. Le refus d'accès à un élément quelconque des informations de gestion se traduira par le refus de toute la demande. Aucune opération de gestion ne doit être effectuée sur des cibles associées à cette demande;

– refus au niveau de chaque objet géré indiqué dans la demande. Le refus d'accès à des opérations et attributs quelconques de cet objet géré se traduira par un refus d'accès à cet objet géré mais non à d'autres objets gérés associés à la demande. Aucune opération de gestion ne doit être effectuée sur des cibles associées à l'objet géré auquel l'accès est refusé;

– refus au niveau de chaque attribut relevant de chaque objet géré indiqué dans la demande. Un refus d'accès à un attribut spécifique d'un objet géré se traduira par un refus d'accès à cet attribut mais non à d'autres attributs relatifs à l'objet géré dont ils dépendent ni aux attributs contenus dans d'autres objets gérés. Aucune opération de gestion ne doit être effectuée sur les attributs spécifiques auxquels l'accès est refusé.

Si aucune granularité de refus n'est spécifiée, la valeur par défaut de cette caractéristique relève de la politique locale.

NOTES

1 Une valeur par défaut de la granularité de refus au niveau de la demande totale est recommandée pour satisfaire au principe du moindre privilège.

2 Une action exécutive par défaut de type non-réponse ou dissolution d'association est recommandée pour satisfaire au principe du moindre privilège.

7.4.6.2 Refus pour cause d'invalidité d'information ACI liée à l'initiateur

Si la décision consiste à rejeter la demande pour cause d'invalidité d'information ACI liée à l'initiateur,

- aucune opération ne doit être effectuée sur une quelconque des cibles spécifiées dans la demande;
- l'action exécutive de refus spécifiée ou par défaut doit être invoquée, sauf qu'une action exécutive de refus spécifiée de type Faux doit être remplacée par une réponse de type dissolution de l'association;
- la granularité spécifiée de la réponse éventuelle de refus doit être ignorée et la réponse doit être donnée au niveau de granularité de la demande totale.

7.4.6.3 Refus pour cause d'application d'une règle globale

Si la décision consiste à rejeter la demande pour cause d'application d'une règle globale,

- aucune opération ne doit être effectuée sur une quelconque des cibles spécifiées dans la demande;
- l'action exécutive de refus spécifiée ou par défaut doit être invoquée;
- la granularité spécifiée de la réponse éventuelle de refus doit être ignorée et la réponse doit être donnée au niveau de granularité de la demande totale.

7.4.6.4 Refus pour cause d'application d'une règle d'item ou d'une règle par défaut

Si la décision consiste à refuser l'exécution de l'opération de gestion sur la cible indiquée pour cause d'application d'une règle d'item ou par défaut,

- si la cible délimite un sous-arbre de l'arbre des informations de gestion, c'est-à-dire si la cible a été sélectionnée selon l'alinéa a) du 7.4.2, aucune opération ne doit être effectuée sur un quelconque des objets gérés contenus dans le sous-arbre délimité;
- si la cible est une sélection d'objets gérés [selon l'alinéa b) du 7.4.2], aucune opération ne doit être effectuée sur les objets gérés désignés par la cible;
- si la cible fait partie de celles qui ont été sélectionnées selon l'alinéa c) du 7.4.2, aucune opération ne doit être effectuée sur cette cible;
- l'action exécutive de refus spécifiée ou par défaut doit être invoquée;
- la granularité de la réponse doit être au niveau spécifié de granularité de refus, s'il n'est pas spécifié, au niveau de granularité par défaut.

NOTES

1 Dans le cas de l'objet sélection d'objets multiples, le service CMIS ne définit pas de réponse appropriée à la granularité de refus du niveau objet géré. C'est-à-dire qu'une réponse de refus doit alors être donnée pour chaque attribut sélectionné.

2 En cas de refus parce que l'accès est interdit à un attribut du filtre, le service CMIS ne prévoit aucun moyen d'indiquer que cet attribut de filtre est la cause du refus. Si la cible fait partie de celles qui ont été définies à l'alinéa b) du 7.4.2 et qu'une réponse de refus soit justifiée, la réponse peut ne pas fournir d'indication quant à l'attribut qui a provoqué le refus.

7.4.6.5 Prescription de journalisation des événements importants pour le contrôle d'accès

Si la politique de sécurité prescrit la journalisation des demandes d'opérations de gestion, la fonction AEF doit faire en sorte que la notification appropriée soit produite comme suit:

- si la demande d'accès est rejetée, la notification doit prendre la forme d'une signalisation d'alarme de sécurité; ou
- si la demande d'accès est acceptée, la notification doit prendre la forme d'un journal d'audit de sécurité.

7.5 Représentation des règles de contrôle d'accès

Une règle de contrôle d'accès consiste à appliquer les paires combinées (initiateur-cible) à la permission d'accès (autorisation ou refus).

L'espace des initiateurs comprend tous les utilisateurs possibles des applications de gestion. Cet espace peut être subdivisé selon la politique de sécurité (par exemple en modes de liste ACL, en modes de capacités, en modes d'étiquettes). Il peut y avoir autant de façons de subdiviser cet espace qu'il existe de politiques de sécurité.

Des groupes d'initiateurs peuvent être représentés par des objets gérés. L'identification des initiateurs dépend de la politique de sécurité.

L'espace des cibles comprend toutes les paires concevables de valeurs (opération de gestion-argument). Cet espace peut être subdivisé de nombreuses façons:

- par opération;
- par objet géré;
- par classe d'objets gérés;
- par attribut;
- par action;
- par valeur d'attribut/d'argument;
- par paramètre de visibilité; et
- par paramètre de synchronisation.

Le degré de granularité le plus fin qui puisse être obtenu est celui des paramètres de classe d'objets gérés, d'instance d'objet géré, d'opération, d'identificateur d'action, de valeur d'action, d'identificateur d'attribut et de valeur d'attribut. Lorsque la sélection d'objets multiples est autorisée, le contrôle d'accès peut être imposé sur des combinaisons des paramètres de classe d'objets gérés, d'instance d'objet géré, de visibilité et de synchronisation. Lorsque la détection de visibilité et le filtrage sont autorisés, le contrôle d'accès peut être imposé, pour chaque objet géré du domaine visible, sur la combinaison des paramètres de classe d'objets gérés, d'instance d'objet géré, d'identificateur d'attribut et de valeur d'attribut.

Des groupes de cibles peuvent être représentés par des objets gérés.

Le contrôle d'accès est exécuté dans le contexte d'un domaine de sécurité. Dans chaque domaine, les règles de contrôle d'accès se présentent sur cinq niveaux hiérarchiques:

- règles qui refusent à des initiateurs spécifiques l'accès à l'une quelconque des cibles du domaine;
- règles qui refusent à des initiateurs spécifiques l'accès à des cibles spécifiques du domaine;
- règles qui autorisent à des initiateurs spécifiques l'accès à toutes les cibles du domaine;
- règles qui autorisent à des initiateurs spécifiques l'accès à des cibles spécifiques du domaine;
- règles qui autorisent ou refusent l'accès en l'absence d'une quelconque autre règle autorisant ou refusant l'accès (règles par défaut).

8 Définitions génériques

Les informations et procédures de contrôle d'accès qui ont été décrites dans l'article 7 ci-dessus peuvent être représentées sous la forme d'objets gérés. La Figure 1 montre la hiérarchie d'héritage pour les classes d'objets gérés qui sont définies dans la présente Recommandation | Norme internationale.

La Figure 2 montre les relations entre certains des objets gérés indiqués sur la Figure 1.

NOTE – Tous les objets gérés représentés sur la Figure 1 ne sont pas reproduits sur la Figure 2. Les objets gérés qui n'y figurent pas (la classe des objets gérés de contrôle d'accès et les classes d'objets gérés d'étiquettes assignées et d'objets gérés dérivés) ont été omis pour des raisons de clarté.

8.1 Objets gérés

8.1.1 Objets de type contrôle d'accès

Cette classe d'objets gérés comprend les éléments et le comportement d'informations de gestion qui sont communs à tous les objets gérés représentant des informations de contrôle d'accès. Elle n'est spécifiée que pour constituer un point unique de spécialisation pour d'autres classes d'objets gérés représentant des informations de contrôle d'accès.

8.1.1.1 Attributs des objets de type contrôle d'accès

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type contrôle d'accès.

8.1.1.1.1 Nom des objets de type contrôle d'accès

Cet attribut est utilisé pour identifier des instanciations de spécialisations de la classe des objets gérés de type contrôle d'accès.

8.1.1.2 Notifications de contrôle d'accès

Les notifications suivantes sont définies pour la classe des objets gérés de type contrôle d'accès:

- modification de valeur d'attribut;
- création d'objet;
- suppression d'objet.

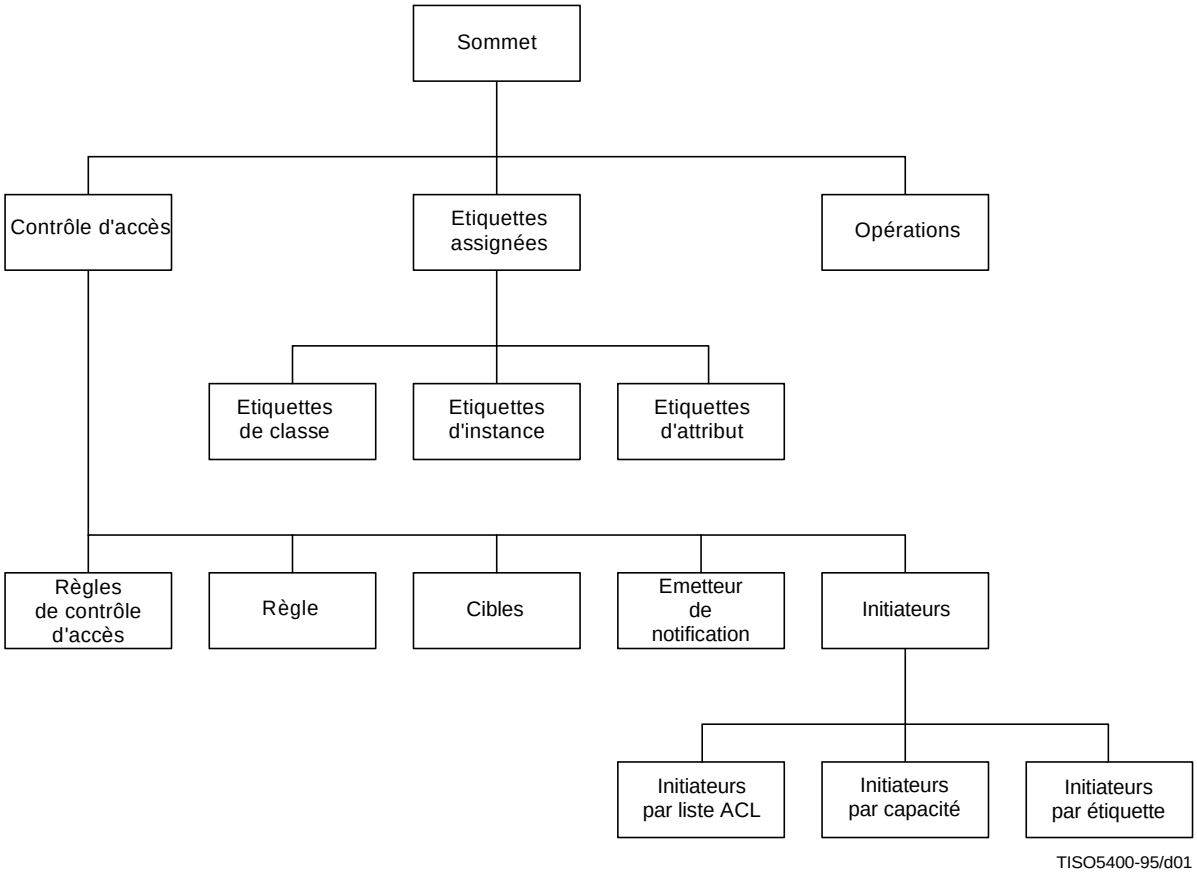

Figure 1 – Hiérarchie d'héritage des classes d'objets gérés

8.1.2 Objets de type règles de contrôle d'accès

Les objets gérés de type règles de contrôle d'accès représentent la fonction ADF pour un domaine de sécurité. Leurs attributs et leurs sous-classes d'objets gérés de type règle désignent les règles de contrôle d'accès applicables à leur domaine de sécurité. La classe des objets gérés de type règles de contrôle d'accès est une sous-classe de la classe des objets gérés de type contrôle d'accès.

L'objet géré règles de contrôle d'accès contient les autres objets gérés qui représentent des règles de contrôle d'accès pour la fonction ADF.

8.1.2.1 Attributs des objets de type règles de contrôle d'accès

Les attributs obligatoires suivants sont définis pour la classe des objets gérés de type règles de contrôle d'accès.

8.1.2.1.1 Accès par défaut

L'attribut accès par défaut identifie, conformément au 7.4.3.1.6, les droits d'accès par défaut pour chaque type d'opération.

8.1.2.1.2 Réponse de refus par défaut

Cet attribut identifie la réponse par défaut qui est renvoyée à l'initiateur en cas de refus, par la fonction ADF, de l'accès à la cible sur la base de la règle par défaut.

8.1.2.1.3 Identité du domaine

Cet attribut identifie le domaine de contrôle d'accès qui régit ces règles de contrôle d'accès.

8.1.2.1.4 Granularité de refus

Cet attribut identifie le niveau auquel le refus d'accès doit être signalé, s'il existe.

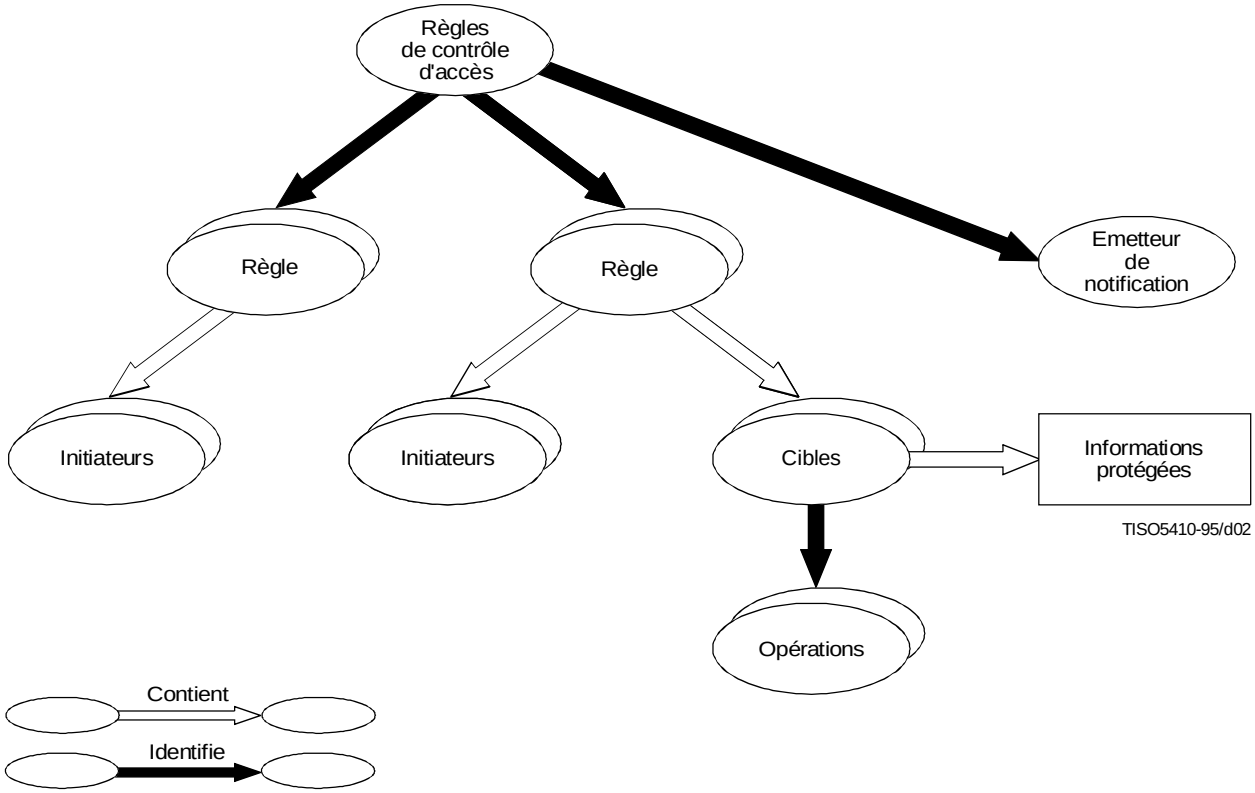

Figure 2 – Relation entre les objets gérés

8.1.3 Objets de type règle

La classe des objets gérés de type règle représente les règles globales et les règles d'item. La classe des objets gérés de type règle est une sous-classe de la classe des objets gérés de type contrôle d'accès.

8.1.3.1 Attributs des objets de type règle

Les attributs obligatoires suivants sont définis pour la classe des objets gérés de type règle.

8.1.3.1.1 Action exécutive

Cet attribut identifie l'action que la fonction AEF doit exécuter si la règle est observée.

8.1.3.1.2 Liste d'initiateurs

Cet attribut, valué sur un ensemble, identifie les sous-classes d'objets gérés de type initiateur qui spécifient les initiateurs auxquels la règle s'applique.

8.1.3.1.3 Liste de cibles

Cet attribut, valué sur un ensemble, identifie les objets gérés de type cible qui eux-mêmes spécifient les cibles auxquelles la règle s'applique.

8.1.3.2 Ensembles de planification

Pour tenir compte des divers niveaux de complexité pour planifier les périodes d'activité d'une règle, on définira pour celle-ci des ensembles conditionnels qui se rapportent à la planification.

Les ensembles de planification permettent aux objets gérés de type règle de passer automatiquement de l'état «actif» à l'état «inactif» et inversement, chacun de ces états étant signalé par l'attribut descripteur d'état de disponibilité, qui prendra, selon le cas, la valeur { } ou {offDuty}.

8.1.3.2.1 Ensemble descripteur d'état de disponibilité

Cet ensemble conditionnel doit être présent si un quelconque des autres ensembles de planification est présent. Si cet ensemble n'est pas présent, l'objet géré de type règle doit toujours être à l'état de disponibilité.

8.1.3.2.2 Ensemble durée

Cet ensemble conditionnel donne la possibilité de commander automatiquement les moments auxquels un objet géré commence à fonctionner et s'arrête. Cet ensemble doit être présent si une telle caractéristique fonctionnelle est requise. Si cet ensemble et un des autres ensembles de planification sont présents en même temps, l'ensemble descripteur d'état de disponibilité doit prendre la valeur {offDuty} à moins que les deux ensembles signalent l'état «actif», auquel cas le descripteur d'état prend la valeur { }.

8.1.3.2.3 Ensemble planification journalière

Cet ensemble conditionnel donne la possibilité de planifier une opération sur une période de 24 h. Il doit être présent si une telle caractéristique fonctionnelle est requise. Il ne doit pas être présent en même temps que l'ensemble planification hebdomadaire ou l'ensemble planification par ordonnanceur externe.

8.1.3.2.4 Ensemble planification hebdomadaire

Cet ensemble conditionnel donne la possibilité de planifier une opération sur une période d'une semaine. Il doit être présent si une telle caractéristique fonctionnelle est requise. Il ne doit pas être présent en même temps que l'ensemble planification journalière ou l'ensemble planification par ordonnanceur externe.

8.1.3.2.5 Ensemble planification par ordonnanceur externe

Cet ensemble conditionnel donne la possibilité de planifier une opération selon un ordonnancement défini dans un autre objet géré. Il doit être présent si une telle caractéristique fonctionnelle est requise. Il ne doit pas être présent en même temps que l'ensemble planification journalière ou l'ensemble planification hebdomadaire.

8.1.3.3 Ensemble contexte d'état

Cet ensemble conditionnel permet à une règle de fonctionner dans le contexte de l'état d'objets gérés spécifiques. Il doit être présent si une telle caractéristique fonctionnelle est requise.

8.1.3.3.1 Attributs de l'ensemble contexte d'état

L'attribut obligatoire suivant est défini pour l'ensemble contexte d'état.

8.1.3.3.1.1 Contextes d'état

Cet attribut, valué sur un ensemble, identifie les objets gérés et les filtres d'attribut qui leur sont associés.

8.1.3.4 Ensemble contexte d’authentification

Cet ensemble conditionnel, s'il est présent dans un objet géré de type règle, spécifie l'identité de la politique d'authentification et les prescriptions d'authentification auxquelles un initiateur est tenu de satisfaire.

8.1.3.4.1 Attributs de l’ensemble contexte d’authentification

L'attribut obligatoire suivant est défini pour l'ensemble contexte d'authentification.

8.1.3.4.1.1 Contexte d'authentification

Cet attribut se présente sous la forme d'une séquence comprenant l'identificateur de politique d'authentification et les prescriptions ainsi identifiées.

8.1.4 Objets de type émetteur de notification

Cette classe d'objets gérés sert à permettre l'émission de notifications applicables à la fourniture du service de contrôle d'accès pour la gestion OSI. Les types de notification qui sont applicables comprennent certaines alarmes de sécurité telles que définies dans la Rec. X.736 du CCITT | ISO/CEI 10164-7 et certaines notifications de journal d'audit de sécurité telles que définies dans la Rec. X.740 du CCITT | ISO/CEI 10164-8.

Un objet géré de type règles de contrôle d'accès ne peut contenir qu'un seul objet géré de type émetteur de notification.

8.1.4.1 Ensembles conditionnels de l'objet émetteur de notification

Cette classe d'objets gérés vise les ensembles conditionnels suivants, qui offrent une capacité flexible pour l'émission des notifications relatives au contrôle d'accès:

- ensemble alarme de violation relative à la sécurité;
- ensemble alarme de violation relative au temps;
- ensemble alarme de violation relative à l'opération;
- ensemble usage du contrôle d'accès;
- ensemble rapport sur le service de contrôle d'accès.

8.1.4.1.1 Ensemble alarme de violation relative à la sécurité

Cet ensemble conditionnel déclenche une notification d'alarme de sécurité de type «violation du service ou mécanisme de sécurité» et l'émission du message «tentative d'accès non autorisé» si les vérifications de contrôle d'accès donnent un résultat défavorable.

8.1.4.1.2 Ensemble alarme de violation relative au temps

Cet ensemble conditionnel déclenche une notification d'alarme de sécurité de type «violation dans le domaine du temps» et l'émission des messages «expiration de validité de clé» et «activité hors des heures ouvrables», si les vérifications de contrôle d'accès donnent un résultat défavorable. La cause «expiration de validité de clé» doit être utilisée lorsque la clé identifiée par le cachet du certificat ACC est périmée. La cause «activité hors des heures ouvrables» doit être utilisée lorsque les vérifications de contexte temporel donnent un résultat défavorable.

8.1.4.1.3 Ensemble alarme de violation relative à l'opération

Cet ensemble conditionnel déclenche une notification d'alarme de sécurité de type «violation opérationnelle» et l'émission des messages «hors service» et «raison non spécifiée» si les vérifications de contrôle d'accès donnent un résultat défavorable. La cause «hors service» doit être utilisée lorsque le mécanisme de contrôle d'accès identifié n'est pas disponible. La cause «raison non spécifiée» doit être utilisée dans les autres cas.

8.1.4.1.4 Ensemble usage du contrôle d'accès

Cet ensemble conditionnel sert à compter le nombre de tentatives d'accès valides et invalides et à permettre d'envoyer, à un journal enregistré d'audit de sécurité, des rapports d'usage contenant ces informations. Ces rapports sont envoyés à des intervalles de temps définis par la politique de sécurité. Le champ informations additionnelles sert à acheminer les valeurs relevées dans les compteurs.

8.1.4.1.4.1 Attributs de l'ensemble usage du contrôle d'accès

Les attributs suivants sont définis pour l'ensemble usage du contrôle d'accès.

8.1.4.1.4.1.1 Tentatives d'accès valides

Cet attribut sert à compter le nombre de fois où une fonction ADF a autorisé un accès.

8.1.4.1.4.1.2 Tentatives d'accès invalides

Cet attribut sert à compter le nombre de fois où une fonction ADF n'a pas autorisé un accès.

8.1.4.1.5 Ensemble rapport sur le service de contrôle d'accès

Cet ensemble conditionnel permet d'émettre des notifications pour journalisation d'audit de sécurité de type «rapport sur le service» en vue de leur éventuel enregistrement dans un journal d'audit de sécurité.

8.1.5 Objets de type cibles

L'objet géré cibles identifie un ensemble d'informations de gestion à soumettre au contrôle d'accès. La classe des objets gérés de type cibles est une sous-classe de la classe des objets gérés de type contrôle d'accès.

8.1.5.1 Attributs des objets de type cibles

Les attributs obligatoires suivants sont définis pour les objets gérés de type cibles.

8.1.5.1.1 Classes d'objets gérés

Cet attribut, valué sur un ensemble, identifie les classes d'objets gérés protégés ainsi que les corrélations de noms qui leur sont facultativement associées.

8.1.5.1.2 Instances d'objet géré

Cet attribut, valué sur un ensemble, identifie les objets gérés protégés.

8.1.5.1.3 Visibilité

Cet attribut identifie le champ de visibilité pour la sélection d'objets gérés protégés.

8.1.5.1.4 Filtre

Cet attribut identifie un filtre à appliquer aux objets gérés identifiés par les autres attributs de l'objet géré cibles afin de déterminer s'il convient de les inclure dans une classe d'objets gérés protégés.

8.1.5.2 Ensemble liste d'opérations

Cet ensemble conditionnel permet le fonctionnement de l'attribut type d'opération, en variante à l'objet géré opérations. Il ne peut être inclus dans l'objet géré cibles que s'il ne contient aucune instanciation de l'objet géré opérations.

8.1.5.2.1 Attributs de l'ensemble liste d'opérations

L'attribut obligatoire suivant est défini pour l'ensemble liste d'opérations.

8.1.5.2.1.1 Liste d'opérations

Cet attribut, valué sur un ensemble, identifie les types d'opération qui sont soumis aux règles applicables à l'objet géré cibles contenant ces opérations.

8.1.6 Objets de type opérations

L'objet géré opérations identifie les contraintes portant sur les types d'opération pour les objets gérés identifiés par l'objet géré cibles dont ils dépendent.

Les types d'opération, définis dans la Rec. X.720 du CCITT | ISO/CEI 10165-1, sont les suivants:

- action;
- création;
- suppression;
- lecture;
- remplacement;
- adjonction de membre;
- retrait de membre;
- remplacement par défaut;
- filtre; et
- sélection d'objets multiples.

Il ne doit y avoir qu'un seul objet géré de type opérations pour un type d'opération donné et contenu dans un objet géré de type cibles.

Le type d'opération est utilisé comme valeur de l'attribut dénomination pour la classe d'objets gérés de type opérations.

Des ensembles conditionnels contiennent des attributs spécifiant des contraintes applicables aux attributs et aux actions associés aux objets de type opérations. En outre, certains attributs contenus dans des ensembles conditionnels ont pour fonction d'imposer des contraintes sur les valeurs des paramètres visibilité et synchronisation qui sont autorisées dans une demande d'accès mettant en jeu le paramètre sélection d'objets multiples.

NOTE – Il est parfois nécessaire d'appliquer différentes contraintes au même type d'opération, ou à des sous-opérations du même type d'opération, pour le(s) même(s) objet(s) géré(s). Par exemple, cette règle peut s'appliquer lorsque différentes contraintes doivent être imposées à différentes actions spécifiques pour le(s) même(s) objet(s) géré(s). Dans de tels cas, un nouvel objet géré de type cibles doit être créé de manière à contenir l'objet géré auquel la contrainte s'applique et à contenir l'objet géré de type opérations qui spécifie les nouvelles contraintes à appliquer quant au type d'opération.

8.1.6.1 Attributs des objets de type opérations

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type opérations.

8.1.6.1.1 Type d'opération

Cet attribut identifie le type d'opération auquel les contraintes sont applicables. Il est utilisé pour la dénomination d'objets gérés de type opérations.

8.1.6.2 Notifications

Les notifications obligatoires ci-après sont définies pour la classe des objets gérés de type opérations:

a) création d'objet;

b) suppression d'objet;

c) modification de valeur d'attribut.

8.1.6.3 Ensembles conditionnels des objets de type opérations

Cette classe d'objets gérés met en jeu les ensembles conditionnels suivants:

– ensemble identificateurs d'attribut;

– ensemble modification d'attribut;

– ensemble actions;

– ensemble visibilité.

8.1.6.3.1 Ensemble identificateurs d'attribut

Cet ensemble conditionnel identifie les attributs auxquels l'accès doit être contrôlé. Il doit être présent si l'opération est de type lecture, remplacement, adjonction de valeur, retrait de valeur, remplacement par défaut ou filtre. Il ne doit pas être présent si l'opération est d'un autre type.

8.1.6.3.1.1 Attributs de l'ensemble identificateurs d'attribut

Cet ensemble conditionnel comprend l'attribut liste d'identificateurs d'attribut qui est spécifié dans la Rec. X.721 du CCITT | ISO/CEI 10165-2.

8.1.6.3.2 Ensemble modification d'attribut

Cet ensemble conditionnel identifie les contraintes portant sur la modification de valeurs d'attribut. Il doit être présent si l'opération est du type remplacement, adjonction de valeur, retrait de valeur ou création. Il ne doit pas être présent si l'opération est d'un autre type.

8.1.6.3.2.1 Attributs de l'ensemble modification d'attribut

Cet ensemble conditionnel comprend l'attribut liste de filtres d'attribut.

8.1.6.3.2.1.1 Liste de filtres d'attribut

Cet attribut, valué sur un ensemble, identifie des contraintes relatives à la valeur des attributs contenus dans une demande d'opération, au moyen d'un ensemble de filtres du service CMIS (un filtre CMIS par attribut soumis à contrainte spécifiée).

8.1.6.3.3 Ensemble actions

Cet ensemble conditionnel identifie les contraintes portant sur les valeurs des arguments de type information d'action. Il doit être présent si l'opération est du type action. Il ne doit pas être présent si l'opération est d'un autre type.

8.1.6.3.3.1 Attributs de l'ensemble actions

Cet ensemble conditionnel comprend l'attribut liste de filtres d'action.

8.1.6.3.3.1.1 Liste de filtres d'action

Au moyen d'un filtre du service CMIS, cet attribut, valué sur un ensemble, identifie des actions et, en option, des contraintes sur les valeurs des arguments de type information d'action.

8.1.6.3.4 Ensemble visibilité

Cet ensemble conditionnel identifie des contraintes portant sur les paramètres visibilité et synchronisation d'opérations de gestion mettant en jeu une sélection d'objets multiples. Cet ensemble doit être présent si l'opération est du type sélection d'objets multiples. Il ne doit pas être présent si l'opération est d'un autre type.

NOTE – L'ensemble visibilité peut par exemple être utilisé:

– pour empêcher la découverte d'objets gérés, ou l'accès à des objets gérés au cours d'une opération de lecture d'un sous-arbre entier au moyen d'un analyseur – comme cela peut être effectué pour explorer illégalement un système;

– pour protéger certains objets gérés contre leur suppression dans le cadre d'un sous-arbre, de telle manière que ces objets gérés ne puissent être supprimés par un certain initiateur que s'ils font l'objet d'une désignation directe et jamais dans le cadre d'une suppression par analyseur.

8.1.6.3.4.1 Attributs de l'ensemble visibilité

Les attributs suivants sont définis pour l'ensemble visibilité.

8.1.6.3.4.1.1 Filtre de visibilité

Pour les demandes qui sélectionnent de multiples objets gérés, l'attribut filtre de visibilité spécifie des contraintes sur le paramètre visibilité de la demande. L'identificateur de l'attribut visibilité (voir 8.1.5.1.3) est utilisé pour tous les items de filtrage contenus dans le filtre.

Si l'attribut filtre de visibilité ne contient aucun item de filtrage, toutes les valeurs possibles du paramètre visibilité doivent être considérées comme des cibles.

8.1.6.3.4.1.2 Filtre de synchronisation

Pour les demandes qui sélectionnent de multiples objets gérés, l'attribut filtre de synchronisation spécifie des contraintes sur le paramètre synchronisation de la demande. L'identificateur de l'attribut synchronisation (voir 8.4.2) est utilisé pour tous les items de filtrage contenus dans le filtre. Si l'attribut filtre de synchronisation ne contient aucun item de filtrage, toutes les valeurs possibles du paramètre synchronisation doivent être considérées comme des cibles.

8.1.7 Objets de type initiateurs

La classe des objets gérés de type initiateurs identifie les initiateurs admis à effectuer des opérations de gestion.

8.1.7.1 Attributs des objets de type initiateurs

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type initiateurs.

8.1.7.1.1 Information ACI d'initiateur requise

Cet attribut sert à indiquer si, pour satisfaire au mode de contrôle d'accès en cours, une information ACI d'initiateur est requise avec chaque demande d'opération de gestion.

8.1.8 Objets de type initiateurs par liste ACL

La classe des objets gérés de type initiateurs par liste ACL contient une liste de noms ou d'autres identités qui forment ensemble une liste de contrôle d'accès (liste ACL).

Plusieurs objets gérés de type initiateurs par liste ACL peuvent être instanciés dans le cadre d'un même objet géré de type règle.

8.1.8.1 Attributs des objets de type initiateurs par liste ACL

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type initiateurs par liste ACL.

8.1.8.1.1 Liste de contrôle d'accès

Cet attribut sert à contenir les identités d'initiateurs qui reçoivent soit une autorisation spécifique d'accès à des informations de gestion ou un refus spécifique d'accès à des informations de gestion.

8.1.9 Objets de type initiateurs par capacité

La classe des objets gérés de type initiateurs par capacité contient une liste d'identités.

Plusieurs objets gérés de type initiateurs par capacité peuvent être instanciés dans le cadre d'un même objet géré de type règle.

8.1.9.1 Attributs des objets de type initiateurs par capacité

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type initiateurs par capacité.

8.1.9.1.1 Liste d'identités de capacité

Cet attribut contient un ensemble d'identités.

Les identités peuvent prendre la forme d'un nom individuel, d'un nom de groupe, d'un nom de rôle ou d'un nom d'application, chaque nom étant assorti d'un ensemble facultatif de paires (nom d'autorité de domaine de sécurité – type d'opération); l'identité peut aussi prendre une forme non spécifiée dans le cadre de la présente Recommandation | Norme internationale.

8.1.10 Objets de type initiateurs par étiquette

Les objets gérés de type initiateurs par étiquette peuvent être utilisés pour spécifier des contraintes sur des opérations de gestion qui s'ajoutent aux contraintes relatives à un contrôle de compatibilité entre l'étiquette de sécurité associée à l'initiateur et l'étiquette de sécurité associée à la cible.

Si un objet géré de type initiateurs par étiquette est présent dans un objet géré de type règle, les contraintes définies par l'objet géré de type règle et par les objets gérés de type cibles contenus dans cet objet règle doivent s'ajouter aux contraintes relatives au contrôle de compatibilité d'étiquettes de sécurité dues au mode de contrôle d'accès par étiquettes.

Si un objet géré de type initiateurs par étiquette n'est pas présent dans un objet géré de type règle, aucune contrainte additionnelle n'est imposée quant à l'accès, autre que les contraintes de contrôle de compatibilité d'étiquettes de sécurité dues au mode de contrôle d'accès par étiquettes.

Plusieurs objets gérés de type initiateurs par étiquette peuvent être instanciés dans un même objet géré de type règle.

8.1.10.1 Attributs des objets de type initiateurs par étiquette

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type initiateurs par étiquette.

8.1.10.1.1 Etiquette de sécurité

Cet attribut contient une étiquette de sécurité.

8.1.11 Objets de type étiquettes assignées

Ce type d'objet géré est la racine du sous-arbre qui contient les objets gérés de type étiquette. Ces objets attribuent à des cibles, en fonction de relations de préséance, une unique étiquette de sécurité. Ce type d'objet géré peut contenir les différents objets gérés de type étiquette et fournit une étiquette de sécurité par défaut à attribuer aux éléments de gestion auxquels une étiquette de sécurité n'a pas été spécifiquement assignée.

8.1.11.1 Attributs des objets de type étiquettes assignées

Les attributs obligatoires suivants sont définis pour la classe des objets gérés de type étiquettes assignées.

8.1.11.1.1 Nom d'étiquette

Cet attribut sert à identifier des instanciations et des spécialisations de la classe des objets gérés de type étiquettes assignées.

8.1.11.1.2 Etiquette de sécurité

Cet attribut contient l'étiquette de sécurité à attribuer à la cible.

8.1.12 Objets de type étiquette d'attribut

Ce type d'objet géré sert à associer une unique étiquette de sécurité à des cibles qui sont des attributs spécifiques dans un objet géré.

8.1.12.1 Attributs des objets de type étiquette d'attribut

Les attributs obligatoires suivants sont définis pour la classe des objets gérés de type étiquette d'attribut.

8.1.12.1.1 Instance d'objet géré

Cet attribut sert à identifier un objet géré spécifique.

8.1.12.1.2 Liste d'identificateurs d'attribut

Cet attribut sert à identifier des attributs spécifiques de l'objet géré qui est identifié par l'attribut instance d'objet géré.

8.1.13 Objets de type étiquette d'instance

Ce type d'objet géré sert à associer une unique étiquette de sécurité à des cibles qui sont des objets gérés individuels.

8.1.13.1 Attributs des objets de type étiquette d'instance

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type étiquette d'instance.

8.1.13.1.1 Instances d'objet géré

Cet attribut sert à identifier une liste de cibles spécifiques au moyen de leurs identificateurs d'objet géré.

8.1.14 Objets de type étiquette de classe

Ce type d'objet géré sert à associer une unique étiquette de sécurité à des cibles qui sont des classes d'objets gérés.

8.1.14.1 Attributs des objets de type étiquette de classe

L'attribut obligatoire suivant est défini pour la classe des objets gérés de type étiquette de classe.

8.1.14.1.1 Classes d'objets gérés

Cet attribut sert à identifier une liste de classes d'objets gérés spécifiques.

8.2 Paramètres

8.2.1 Paramètre filtre de contrôle d'accès invalide

Ce paramètre d'erreur spécifique signale qu'une erreur est contenue dans un élément proposé de filtre de contrôle d'accès. Sa valeur se présente sous la forme de la séquence suivante: identificateur d'erreur (prenant une de ces valeurs: *duplicateId, heterogeneousId*, ou *invalidId*) et filtre facultatif du service CMIS (contenant le filtre où se trouve l'erreur).

8.3 Corrélation de noms

8.3.1 Corrélation règle – Règle de contrôle d'accès

Cette corrélation de noms indique que des objets de type règle – et leurs spécialisations – seront contenus dans des objets de type règles de contrôle d'accès – et leurs spécialisations. L'attribut nom des objets de type contrôle d'accès doit être utilisé pour dénommer les règles. Des objets gérés de type règle peuvent être créés par une opération de gestion, avec dénomination automatique d'instance et objet de référence. Les objets gérés de type règle peuvent être supprimés par une opération de gestion.

8.3.2 Corrélation opérations – Cibles

Cette corrélation de noms indique que des objets gérés de type opérations seront contenus dans des objets gérés de type cibles. L'attribut type d'opération doit être utilisé pour dénommer les opérations. Des objets gérés de type opérations peuvent être créés par une opération de gestion, avec objet de référence. Les objets gérés de type opérations peuvent être supprimés par une opération de gestion.

8.3.3 Corrélation émetteur de notification – Règles de contrôle d'accès

Cette corrélation de noms indique qu'un unique objet géré de type émetteur de notification sera contenu dans un objet géré de type règles de contrôle d'accès. L'émetteur de notification peut être – par opération de gestion – créé avec objet de référence et supprimé. Il peut être nommé automatiquement.

8.3.4 Corrélation étiquette d'attribut – Etiquettes assignées

Cette corrélation de noms indique que des objets gérés de type étiquette d'attribut seront contenus dans des objets gérés de type étiquettes assignées. Les objets gérés de type étiquette d'attribut peuvent être créés et supprimés par une opération de gestion.

8.3.5 Corrélation étiquette d'instance – Etiquettes assignées

Cette corrélation de noms indique que des objets gérés de type étiquette d'instance seront contenus dans des objets gérés de type étiquettes assignées. Les objets gérés de type étiquette d'instance peuvent être créés et supprimés par une opération de gestion.

8.3.6 Corrélation étiquette de classe – Etiquettes assignées

Cette corrélation de noms indique que des objets gérés de type étiquette de classe seront contenus dans des objets gérés de type étiquettes assignées. Les objets gérés de type étiquette de classe peuvent être créés et supprimés par une opération de gestion.

8.4 Attributs

Les attributs suivants ne sont pas définis pour un quelconque ensemble conditionnel ou pour une quelconque classe d'objets gérés; ils sont utilisés pour spécifier d'autres attributs ou pour effectuer un filtrage.

8.4.1 Filtre de contrôle d'accès

Cet attribut, valué sur un ensemble, identifie des contraintes portant sur la valeur de certains paramètres des opérations de gestion. Chaque élément de cet ensemble est constitué d'un filtre du service CMIS, visant un élément unique d'une information de gestion. Chaque élément d'une information de gestion est adressé par un élément et un seul de cet attribut. Un ensemble vide indique que toutes les valeurs possibles sont visées.

8.4.2 Synchronisation

Cette valeur d'attribut représente le paramètre synchronisation des opérations de gestion. Cet attribut sert à représenter des filtres appliqués à ce paramètre.

8.5 Définitions génériques importées

La présente Recommandation | Norme internationale fait appel aux définitions génériques suivantes, qui figurent dans la Rec. X.730 du CCITT | ISO/CEI 10164-1, dans la Rec. X.731 du CCITT | ISO/CEI 10164-2, dans la Rec. X.732 du CCITT | ISO/CEI 10164-3, dans la Rec. X.734 du CCITT | ISO/CEI 10164-5, dans la Rec. X.736 du CCITT | ISO/CEI 10164-7 et dans la Rec. X.740 du CCITT | ISO/CEI 10164-8:

- liste d'identificateurs d'attribut;
- notification de modification de valeur d'attribut;
- ensemble descripteur d'état de disponibilité;
- compteur;
- ensemble planification quotidienne;
- structure de discriminateur;
- ensemble durée;
- ensemble planification par ordonnanceur externe;

- membre;
- instance d'objet géré;
- notification de création d'objet;
- notification de suppression d'objet;
- violation opérationnelle;
- violation de service ou de mécanisme de sécurité;
- rapport sur le service;
- violation relative au temps;
- rapport sur l'usage;
- ensemble planification hebdomadaire.

8.6 Compatibilité

Les définitions des classes d'objets gérés sont compatibles avec les fonctions définies dans la présente Recommandation | Norme internationale du fait qu'elles comprennent la spécification des objets gérés, des attributs et des notifications définis dans la présente Recommandation | Norme internationale, dans la Rec. X.721 du CCITT | ISO/CEI 10165-2, dans la Rec. X.736 du CCITT | ISO/CEI 10164-7 et dans la Rec. X.740 du CCITT | ISO/CEI 10164-8. Le mécanisme de référence est défini dans la Rec. X.722 du CCITT | ISO/CEI 10165-4.

9 Définitions de service

9.1 Introduction

Le contrôle d'accès peut être appliqué aux informations de gestion. Il peut varier en fonction du temps ou de modifications de la politique de sécurité de contrôle d'accès. Il est donc nécessaire de prévoir un mécanisme permettant d'administrer le service de contrôle d'accès.

9.2 Service de gestion du contrôle d'accès

Ce service assure l'administration des règles de contrôle d'accès utilisées par un système.

Il permet d'administrer des objets gérés appartenant aux classes suivantes:

- règles de contrôle d'accès;
- règles.

9.2.1 Lancement de règles de contrôle d'accès

Le service PT-CREATE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système crée des instances des classes d'objets gérés énumérées au 9.2.

9.2.2 Modification de règles de contrôle d'accès

Le service PT-SET défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système modifie les valeurs d'attributs des classes d'objets gérés énumérées au 9.2.

9.2.3 Suppression de règles de contrôle d'accès

Le service PT-DELETE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système supprime des instances des classes d'objets gérés énumérées au 9.2.

9.3 Service d’administration des cibles

Ce service est utilisé pour administrer les objets gérés de type cibles qui identifient les informations de gestion protégées par contrôle d'accès.

9.3.1 Lancement de cibles de contrôle d'accès

Le service PT-CREATE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système crée des instances des classes d'objets gérés de type cibles.

9.3.2 Modification de cibles de contrôle d'accès

Le service PT-SET défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système modifie les valeurs d'attributs des classes d'objets gérés de type cibles.

9.3.3 Suppression de cibles de contrôle d'accès

Le service PT-DELETE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système supprime des instances des classes d'objets gérés de type cibles.

9.4 Service d'administration des initiateurs

Ce service sert à administrer des sous-classes des objets gérés de type initiateurs.

9.4.1 Lancement d'initiateurs de contrôle d'accès

Le service PT-CREATE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système crée des instances des classes d'objets gérés de type initiateurs.

9.4.2 Modification d'initiateurs de contrôle d'accès

Le service PT-SET défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système modifie les valeurs d'attributs des classes d'objets gérés de type initiateurs.

9.4.3 Suppression d'initiateurs de contrôle d'accès

Le service PT-DELETE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système supprime des instances des classes d'objets gérés de type initiateurs.

9.5 Service d'administration des opérations

Ce service sert à administrer des sous-classes des objets gérés de type opérations.

9.5.1 Lancement d'opérations de contrôle d'accès

Le service PT-CREATE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système crée des instances des classes d'objets gérés de type opérations.

9.5.2 Modification d'opérations de contrôle d'accès

Le service PT-SET défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système modifie les valeurs d'attributs des classes d'objets gérés de type opérations.

9.5.3 Suppression d'opérations de contrôle d'accès

Le service PT-DELETE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système supprime des instances des classes d'objets gérés de type opérations.

9.6 Service d'administration des étiquettes

Ce service sert à administrer des sous-classes des objets gérés de types étiquettes assignées, étiquette d'attribut, étiquette d'instance et étiquette de classe.

9.6.1 Lancement d'étiquettes de contrôle d'accès

Le service PT-CREATE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système crée des instances des classes d'objets gérés de types étiquettes assignées, étiquette d'attribut, étiquette d'instance et étiquette de classe.

9.6.2 Modification d'étiquettes de contrôle d'accès

Le service PT-SET défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système modifie les valeurs d'attributs des classes d'objets gérés de types étiquettes assignées, étiquette d'attribut, étiquette d'instance et étiquette de classe.

9.6.3 Suppression d'étiquettes de contrôle d'accès

Le service PT-DELETE défini dans la Rec. X.730 du CCITT | ISO/CEI 10164-1 est utilisé pour permettre à un certain système ouvert de demander qu'un autre système supprime des instances des classes d'objets gérés de types étiquettes assignées, étiquette d'attribut, étiquette d'instance et étiquette d'objet.

9.7 Service de notification de contrôle d'accès

Ce service permet de transmettre, d'un système ouvert à un autre, des rapports d'événement concernant la surveillance et l'administration d'un service de contrôle d'accès. La fonction de signalisation des alarmes de sécurité, spécifiée dans la Rec. X.736 du CCITT | ISO/CEI 10164-7, est utilisée pour retransmettre des enregistrements d'alarme et des journaux d'audit de sécurité concernant des tentatives d'accès illégal à des informations de gestion; par ailleurs, la fonction de journal d'audit de sécurité, spécifiée dans la Rec. X.740 du CCITT | ISO/CEI 10164-8, est utilisée pour retransmettre des rapports concernant l'usage général du service de contrôle d'accès.

10 Unités fonctionnelles

Les objets et attributs du service de contrôle d'accès constituent une unique unité fonctionnelle de gestion.

11 Protocole

11.1 Eléments de procédure

La présente Recommandation | Norme internationale fait appel aux éléments de procédure qui sont définis dans la Rec. X.730 du CCITT | ISO/CEI 10164-1, dans la Rec. X.736 du CCITT | ISO/CEI 10164-7 et dans la Rec. X.740 du CCITT | ISO/CEI 10164-8 pour les services décrits à l'article 9 ci-dessus. Il n'existe aucun élément de procédure qui soit spécifique de la présente Recommandation | Norme internationale.

11.2 Syntaxe abstraite

11.2.1 Objets gérés

Le Tableau 1 décrit la relation entre les objets gérés de contrôle d'accès et les objets gérés dont la syntaxe abstraite est spécifiée dans l'Annexe A.

11.2.2 Attributs

Le Tableau 2 décrit la relation entre les attributs de contrôle d'accès et les attributs de gestion dont la syntaxe abstraite est spécifiée dans l'Annexe A.

11.2.3 Groupes d’attributs

La présente fonction de gestion des systèmes ne définit aucun groupe d'attributs.

11.2.4 Actions

La présente fonction de gestion des systèmes ne définit aucune action.

11.2.5 Notifications

La présente fonction de gestion des systèmes ne définit aucune notification.

Tableau 1 – Objets gérés

Objet géré	Classe d'objets gérés
Contrôle d'accès	accessControl
Règles de contrôle d'accès	accessControlRules
Initiateurs par liste ACL	aclInitiators
Etiquettes assignées	assignedLabels
Etiquette d'attribut	attributeLabel
Initiateurs par capacité	capabilityInitiators
Etiquette de classe	classLabel
Etiquette d'instance	instanceLabel
Initiateurs par étiquette	labelInitiators
Emetteur de notification	notificationEmitter
Opérations	operations
Règle	rule
Cibles	targets

11.2.6 Paramètres

Le Tableau 3 décrit la relation entre les paramètres de contrôle d'accès et les paramètres de gestion dont la syntaxe abstraite est spécifiée dans l'Annexe A.

11.3 Négociation de l'unité fonctionnelle de contrôle d'accès

La présente Recommandation | Norme internationale assigne l'identificateur d'objet suivant:

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

en tant que valeur de l'identificateur de type ASN.1 **FunctionalUnitPackageId**, qui est défini dans la Rec. X.701 du CCITT | ISO/CEI 10040 pour usage lors de la négociation de l'unité fonctionnelle suivante:

0 unité fonctionnelle de contrôle d'accès

où le numéro indique la position binaire attribuée à l'unité fonctionnelle et où le nom renvoie à l'unité fonctionnelle définie à l'article 10.

Dans le contexte de la gestion des systèmes OSI, le mécanisme permettant de négocier l'unité fonctionnelle de contrôle d'accès est décrit par la Rec. X.701 du CCITT | ISO/CEI 10040.

NOTE – La prescription de négocier les unités fonctionnelles est spécifiée par le contexte d'application.

12 Relation avec d'autres fonctions

Les notifications de création d'objet et de suppression d'objet, définies dans la Rec. X.730 du CCITT | ISO/CEI 10164-1, servent à signaler, selon le cas, la création ou la suppression d'instances d'objets gérés appartenant aux classes définies dans la présente Recommandation | Norme internationale.

Tableau 2 – Attributs de gestion

Attribut de contrôle d'accès	Nom de l'attribut
Filtre de contrôle d'accès	accessControlFilter
Nom d'objet de contrôle d'accès	accessControlObjectName
Liste des filtres d'action	actionFilterList
Liste des filtres d'attribut	attributeFilterList
Contexte d'authentification	authenticationContext
Liste des identités de capacité	capabilityIdentitiesList
Accès par défaut	defaultAccess
Réponse de refus par défaut	defaultDenialResponse
Granularité de refus	denialGranularity
Identité de domaine	domainIdentity
Action exécutive	enforcementAction
Filtre	filter
Information ACI d'initiateur requise	initiatorACImandated
Liste d'initiateurs	initiatorsList
Tentatives d'accès invalides	invalidAccessAttempts
Nom d'étiquette	labelName
Classes d'objets gérés	managedObjectClasses
Instances d'objet géré	managedObjectInstances
Type d'opération	operationType
Liste d'opérations	operationsList
Visibilité	scope
Filtre de visibilité	scopeFilter
Etiquette de sécurité	securityLabel
Contextes d'état	stateConditions
Synchronisation	synchronization
Filtre de synchronisation	synchronizationFilter
Liste de cibles	targetsList
Tentatives d'accès valides	validAccessAttempts

Tableau 3 – Paramètres de gestion

Paramètre de contrôle d'accès	Nom du paramètre
Filtre de contrôle d'accès invalide	invalidAccessControlFilter

La notification de modification d'une valeur d'attribut, définie dans la Rec. X.730 du CCITT | ISO/CEI 10164-1, sert à signaler des modifications de valeur d'attribut dans des instances des objets gérés définis dans la présente Recommandation | Norme internationale.

Les notifications de violation opérationnelle, de violation du service ou mécanisme de sécurité, et de violation relative au domaine temporel, définies dans la Rec. X.736 du CCITT | ISO/CEI 10164-7, sont utilisées pour signaler des alarmes de sécurité associées au fonctionnement des mécanismes de contrôle d'accès.

Les notifications de rapport sur le service et d'usage du service, définies dans la Rec. X.740 du CCITT | ISO/CEI 10164-8, servent à acheminer des rapports de journalisation d'audit de sécurité associés à l'utilisation des services et mécanismes de contrôle d'accès.

La gestion des informations de contrôle d'accès peut faire appel aux services suivants de gestion des systèmes OSI, définis dans la Rec. X.730 du CCITT | ISO/CEI 10164-1:

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

La politique de sécurité peut stipuler que ces services soient utilisés dans le cadre d'une association fiable, de manière que les informations ACI soient protégées d'une découverte ou modification intempestive.

13 Conformité

Les mises en œuvre réputées conformes à la présente Recommandation | Norme internationale doivent respecter les prescriptions de conformité qui sont définies dans les paragraphes suivants.

13.1 Conformité statique

La mise en œuvre doit êre conforme aux prescriptions de la présente Recommandation | Norme internationale dans le rôle de gestionnaire, dans le rôle d'agent ou dans ces deux rôles. Une revendication de conformité à au moins un de ces rôles doit être déclarée dans le Tableau B.1.

Si une revendication de conformité est faite pour remplir le rôle de gestionnaire, la mise en œuvre doit supporter au moins une opération ou notification de gestion d'au moins un des objets gérés décrits dans la présente Recommandation | Norme internationale. Une revendication de conformité dans le rôle de gestionnaire pour ces opérations de gestion est spécifiée dans le Tableau B.3 et les tableaux suivants de l'Annexe B.

Si une revendication de conformité est faite à l'appui du rôle d'agent, la mise en œuvre doit supporter une ou plusieurs des classes d'objets gérés des règles de contrôle d'accès spécifiées dans le Tableau B.4.

Le système doit supporter la syntaxe de transfert déduite des règles de codage spécifiées dans la Rec. X.209 du CCITT | ISO/CEI 8825 pour les types abstraits de données indiqués dans les définitions avec lesquelles la compatibilité est revendiquée; ces règles de codage sont dénommées comme suit: {joint-iso-ccitt asn1(1) basic encoding(1)}.

13.2 Conformité dynamique

Les mises en œuvre revendiquant la conformité à la présente Recommandation | Norme internationale doivent supporter les éléments de procédure et les définitions de sémantèmes qui correspondent aux définitions dont le support est revendiqué.

13.3 Prescriptions relatives à la conformité des informations de gestion

Tout formulaire MCS, MICS, MOCS, MRCS ou MIDS qui est conforme à la présente Recommandation | Norme internationale doit être techniquement identique aux formulaires spécifiés dans les Annexes B, C, D, E et F sans modification de la numérotation des tableaux et du numéro d'index des items, à la seule exception de la pagination et des en-têtes, qui peuvent différer.

Le fournisseur d'une mise en œuvre réputée conforme à la présente Recommandation | Norme internationale doit remplir un exemplaire de la déclaration de conformité de gestion (MCS) qui est décrite dans l'Annexe B au titre des prescriptions de conformité, ainsi que tout autre formulaire de déclaration de conformité d'instance (ICS) signalé comme applicable à partir de cette déclaration MCS. Une déclaration ICS réputée conforme à la présente Recommandation | Norme internationale doit:

- décrire une mise en œuvre conforme à la présente Recommandation | Norme internationale;
- avoir été remplie conformément aux instructions de remplissage indiquées dans la Rec. UIT-T X.724 | ISO/CEI 10165-6;
- comporter les renseignements nécessaires pour identifier de manière univoque aussi bien le fournisseur que la mise en œuvre.

Les revendications de conformité à des informations de gestion définies dans la présente Recommandation | Norme internationale comme appartenant à des classes d'objets gérés définies ailleurs doivent comporter, dans leurs formulaires MOCS, les prescriptions relatives aux formulaires MIDS pour ces classes d'objets gérés.

Annexe A

Définition des informations de gestion

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

A.1 Attribution des identificateurs d'objet

La présente Recommandation | Norme internationale attribue les identificateurs d'objet suivants:

```
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 Définition des classes d'objets gérés pour le contrôle d'accès

A.2.1 Classe des objets gérés de classe contrôle d'accès

La classe des objets gérés de classe contrôle d'accès sert à offrir un attribut commun de dénomination et une signalisation commune des changements de valeur d'attribut pour les objets gérés représentant des informations de contrôle d'accès. Elle n'est pas destinée à être instanciée.

```
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
                    ! La classe d'objets gérés contrôle d'accès doit émettre les notifications de création d'objet et de
                    suppression d'objet. Les spécialisations de la classe d'objets gérés contrôle d'accès doivent définir les
                    conditions dans lesquelles les notifications de changement de valeur d'attribut doivent être
                    émises. ! ;;
            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 Règles de contrôle d'accès

La classe d'objets gérés de classe règles de contrôle d'accès sert à définir une représentation des règles de contrôle d'accès. Chaque fonction décisionnelle d'accès nécessite, à l'intérieur d'un domaine de sécurité, un seul objet géré de classe règles de contrôle d'accès.

```
accessControlRules MANAGED OBJECT CLASS
        DERIVED FROM accessControl;

        CHARACTERIZED BY accessControlRulesPackage PACKAGE
            BEHAVIOUR accessControlRulesBehaviour BEHAVIOUR
                DEFINED AS
                    ! Un objet géré de classe règles de contrôle d'accès peut contenir des objets gérés de classe règle,
                    dont chacun représente une règle globale ou une règle d'item. Cette classe doit utiliser ces règles
                    conformément aux procédures du 7.4 et conformément à la politique du domaine de contrôle
                    d'accès.
```

Une notification de changement de valeur d'attribut doit être émise lorsqu'un attribut quelconque de cette classe d'objets est modifié.

NOTE – Un objet géré de classe règles de contrôle d'accès peut contenir des objets gérés de classe règle qui sont en conflit pour une paire donnée initiateur/cible. Les procédures du 7.4.3.1 font en sorte que le principe du moindre privilège soit appliqué.

! ;;

ATTRIBUTES

defaultAccess	**REPLACE-WITH-DEFAULT DEFAULT VALUE AccessControl-ASN1Module.denyAll GET-REPLACE,**
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

! Chaque règle identifie sa nature: octroi ou refus d'accès. Si l'attribut d'action exécutive a la valeur d'autorisation, l'accès est autorisé; sinon, l'attribut d'action exécutive définit le type de réponse négative qui est communiquée à l'initiateur de l'opération de gestion.

Un objet géré règle peut comporter des propriétés représentant un contexte pour la règle.

La possibilité de planification constitue un tel contexte. Lorsqu'ils sont présents, les ensembles de planification commandent la valeur de l'attribut descripteur d'état de disponibilité. Cet attribut doit afficher la valeur {off duty} lorsque la planification exige que la règle ne soit pas disponible et, dans le cas contraire, la valeur {}.

Un autre contexte est l'état d'autres objets gérés. Lorsqu'il est présent, l'ensemble contexte d'état identifie les objets gérés et les filtres d'après leurs attributs. Cette règle ne doit s'appliquer que si les objets gérés existent et que les filtres donnent le résultat TRUE.

L'attribut liste d'initiateurs identifie les objets gérés de classe initiateur qui désignent des initiateurs dans le contexte d'un ou de plusieurs systèmes de contrôle d'accès. Si la liste est vide, la règle doit s'appliquer à tous les initiateurs.

L'attribut liste de cibles identifie les objets gérés de classe cible qui désignent les cibles auxquelles la règle se rapporte. Si la liste est vide, la règle est globale; sinon, c'est une règle d'item.

La création et la suppression de règles doivent être signalées, respectivement, par des notifications de création d'objet et de suppression d'objet.

Une notification de changement de valeur d'attribut doit être émise lorsqu'un quelconque attribut de cette classe d'objets est modifié. ! ;;

ATTRIBUTES

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

CONDITIONAL PACKAGES

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": availabilityStatusPackage
PRESENT IF ! Un quelconque des ensembles de planification (durée, planification journalière, hebdomadaire, externe) est présent. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": duration
PRESENT IF ! L'objet doit être disponible à partir d'un moment de départ spécifié, en permanence ou jusqu'à un moment d'arrêt spécifié. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": dailyScheduling
PRESENT IF ! Les deux ensembles de planification, hebdomadaire et externe, ne sont pas présents et que la planification journalière soit supportée. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": weeklyScheduling
PRESENT IF ! Les deux ensembles de planification, journalière et externe, ne sont pas présents et que la planification hebdomadaire soit supportée. !,

"CCITT Rec. X.721 | ISO/IEC 10165-2:1992": externalScheduler
PRESENT IF ! Les deux ensembles de planification, journalière et hebdomadaire, ne sont pas présents et que la planification externe soit supportée. !,

stateConditionsPackage PACKAGE
BEHAVIOUR stateConditionsBehaviour BEHAVIOUR
DEFINED AS
! Lorsque cet ensemble est présent dans un objet de classe règle, les filtres désignés par l'attribut de contexte d'état doivent être appliqués aux objets gérés désignés par cet attribut. Si les objets gérés ne sont pas disponibles ou si les filtres donnent la valeur FALSE, la règle doit donner la valeur FALSE. Si les filtres donnent la valeur TRUE, la règle doit donner la valeur 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 ! L'état d'un autre objet géré fournit un contexte pour cette règle. !,

authenticationContextPackage PACKAGE
BEHAVIOUR authenticationContextBehaviour BEHAVIOUR
DEFINED AS
! Lorsque cet ensemble est présent dans un objet géré de classe règle, les prescriptions d'authentification spécifiées par l'attribut de contexte d'authentification doivent être satisfaites avant d'effectuer une autre évaluation des droits d'accès d'un initiateur.

Si les prescriptions d'authentification ne sont pas satisfaites, la règle doit donner la valeur FALSE. ! ;;
ATTRIBUTES authenticationContext GET-REPLACE;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) (2) };
PRESENT IF ! Le contexte d'authentification est requis. !;

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

A.2.4 Emetteur de notification

notificationEmitter MANAGED OBJECT CLASS
DERIVED FROM accessControl;
CHARACTERIZED BY accessControlNotificationEmitterPkg PACKAGE
BEHAVIOUR accessControlNotificationEmitterDefinition BEHAVIOUR
DEFINED AS
! Cette classe d'objets gérés permet au système de contrôle d'accès de signaler des attaques potentielles ou réelles contre la sûreté d'applications de gestion et d'informations de gestion. Une instance de cette classe d'objets gérés doit supporter au moins un des ensembles conditionnels définis ci-dessous. ! ;;;
CONDITIONAL PACKAGES

securityViolationAlarmPkg PACKAGE
BEHAVIOUR securityViolationAlarmBehaviour BEHAVIOUR
DEFINED AS
! Cet ensemble permet une notification d'alarme de sécurité de type "violation de service ou de mécanisme de sécurité" avec la cause "tentative d'accès non autorisée", en cas d'échec des vérifications du contrôle d'accès. ! ;;
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 ! la politique de sécurité exige que ce type d'alarme de sécurité soit émis lors d'un échec des vérifications du contrôle de sécurité. !,

timeViolationAlarmPkg PACKAGE
BEHAVIOUR timeViolationAlarmBehaviour BEHAVIOUR
DEFINED AS
! Cet ensemble permet une notification d'alarme de sécurité de type "violation relative au temps" et provoque l'envoi d'une cause de type "expiration de validité de clé" et "activité hors des heures ouvrables" en cas d'échec des vérifications de contrôle d'accès. La cause "expiration de validité de clé" doit être utilisée lorsque la clé identifiée par le cachet du certificat ACC est périmée. La cause "activité hors des heures ouvrables" doit être utilisée si les vérifications de contexte temporel donnent un résultat défavorable. ! ;;

```
        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 ! la politique de sécurité exige que ce type d'alarme de sécurité soit émis si un des deux
                    événements suivants se produit: détection de dépassement des heures ouvrables ou utili-
                    sation de clé périmée. !,

    operationalViolationAlarmPkg  PACKAGE
        BEHAVIOUR  operationalViolationAlarmBehaviour BEHAVIOUR
            DEFINED AS
                ! Cet ensemble permet, en cas d'échec des vérifications de contrôle d'accès, une notification
                d'alarme de sécurité du type "violation opérationnelle" et provoque l'émission d'une cause
                "hors service" ou "raison non spécifiée". La cause "hors service" doit être utilisée lorsque le
                mécanisme de contrôle d'accès identifié n'est pas disponible. La cause "raison non spécifiée"
                doit être utilisée dans les autres cas. ! ;;
        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 ! la politique de sécurité prescrit que ce type d'alarme de sécurité soit émis si un des deux
                    événements suivants se produit: indisponibilité du mécanisme de contrôle d'accès ou
                    identification d'une autre cause par la politique de sécurité. !,

    accessControlUsagePkg  PACKAGE
        BEHAVIOUR  accessControlUsagePkgBehaviour BEHAVIOUR
            DEFINED AS
                ! Cet ensemble est utilisé pour compter le nombre de tentatives d'accès valides ou invalides
                ainsi que pour permettre l'envoi à un  journal d'audit de sécurité de rapports d'usage
                contenant ces informations. Le rapport sur l'usage est envoyé à un intervalle de temps défini
                par la politique de sécurité. Le champ d'informations supplémentaires est utilisé pour
                acheminer les valeurs de compteur. ! ;;
        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 ! la politique de sécurité prescrit que le nombre de tentatives d'accès valides et invalides soit
                    consigné dans un journal. !,

    accessControlServiceReportPkg  PACKAGE
        BEHAVIOUR accessControlServiceReportPkgBehaviour BEHAVIOUR
            DEFINED AS
                !  Cet ensemble permet l'émission de notifications de type "rapport sur le service" pour
                éventuelle inclusion dans un journal d'audit de sécurité.  ! ;;
        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 ! la politique de sécurité prescrit que les rapports sur le service soient consignés dans
                        un journal. !;

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

A.2.5 Cibles

La classe d'objets gérés de classe cibles est utilisée pour identifier des objets gérés auxquels on contrôle l'accès.

```
targets MANAGED OBJECT CLASS
        DERIVED FROM accessControl;
        CHARACTERIZED BY targetsPackage PACKAGE
            BEHAVIOUR targetsBehaviour BEHAVIOUR
                DEFINED AS
                    ! Les cibles identifient des objets gérés dans le cadre du domaine de sécurité. Ces objets gérés sont
                    identifiés conformément aux règles suivantes:

                    a) tous les objets gérés contenus dans le domaine de sécurité et appartenant aux classes d'objets
                    gérés identifiées par l'attribut classes d'objets gérés sont identifiés avec les corrélations de noms
                    spécifiées;
```

b) tous les objets gérés contenus dans le domaine de sécurité sont identifiés dans le domaine de sécurité explicitement désigné par les instances de ces objets gérés;

c) chaque objet géré sélectionné conformément à a) et à b) doit être considéré comme étant un objet géré de base pour la sélection d'objets gérés conformément aux attributs de visibilité et de filtre;

d) tous les objets gérés sélectionnés conformément à c) doivent être considérés comme des objets gérés de classe cible.

A moins que l'objet géré cibles ne contienne que des objets gérés de classe opérations, l'objet géré cibles désigne toutes les opérations effectuées sur les objets gérés sélectionnés.

Une notification de changement de valeur d'attribut doit être émise lorsqu'un attribut de cet objet géré est modifié. ! ;;

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
! Cet ensemble permet la prise en charge de l'attribut liste d'opérations en variante de l'objet géré opérations. Il ne peut être inclus dans l'objet géré cibles que si celui-ci ne contient aucune instanciation de l'objet géré opérations.
! ;;
ATTRIBUTES
operationsList GET-REPLACE ADD-REMOVE;;
REGISTERED AS
{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) operationsListPackage(15) }
PRESENT IF ! Aucun objet de classe opérations n'est contenu !

REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) managedObjectClass(3) targets(5) };

A.2.6 Opérations

Les instanciations de l'objet géré opérations désignent les opérations qui peuvent être effectuées sur les informations de gestion spécifiées (désignées par l'objet géré cibles).

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
! L'objet géré opérations désigne les contraintes sur les types d'opération pour les objets gérés identifiés par l'objet géré de la classe supérieure cibles.

Le type d'opération est spécifié par l'attribut type d'opération. Cet attribut est également l'attribut de dénomination pour la classe d'objets gérés opérations.

Les contraintes sur le type d'opération, dont certaines sont spécifiques de chaque type d'opération, sont spécifiées par d'autres attributs contenus dans des ensembles conditionnels.

Lorsqu'un objet géré de classe cibles désigne l'objet géré spécifié dans la demande d'accès et contient un ou plusieurs objets gérés opérations, cette demande d'accès doit répondre aux conditions suivantes pour satisfaire à la règle de confinement:

a) la demande d'accès correspond au type d'opération pour un des objets gérés opérations contenus dans la classe cibles; et

b) les contraintes spécifiées par l'attribut type d'opération sont satisfaites.

L'objet géré opérations doit émettre la notification de création d'objet au moment de sa création et la notification de suppression d'objet au moment de sa suppression. Une notification de changement de valeur d'attribut doit être émise lors d'un changement d'un attribut de cette classe d'objets gérés. ! ;;
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
! Les attributs désignés par l'attribut liste d'identificateurs d'attribut doivent faire partie de la cible. Si l'attribut liste d'identificateurs d'attribut est vide, tous les attributs doivent faire partie de la cible pour l'opération désignée et pour les objets gérés indiqués par l'objet géré de la classe supérieure cibles. ! ;;
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 ! l'opération est de type lecture, remplacement par défaut ou filtrage !,

attributeModificationPackage PACKAGE
BEHAVIOUR attributeModificationBehaviour BEHAVIOUR
DEFINED AS
! Les valeurs d'attribut identifiées par l'attribut liste de filtres d'attribut doivent faire partie de la cible. Si l'attribut liste de filtres d'attribut est vide, tous les attributs et leurs valeurs doivent faire partie de la cible pour l'opération à effectuer sur les objets gérés désignés par l'objet géré de la classe supérieure cibles. Si l'attribut liste de filtres d'attribut désigne un attribut sans contrainte sur sa valeur, toutes les valeurs de cet attribut doivent faire partie de la cible pour l'opération à effectuer sur les objets gérés désignés par l'objet géré de la classe supérieure cibles. ! ;;
ATTRIBUTES
attributeFilterList GET-REPLACE ADD-REMOVE;
REGISTERED AS
{ joint-iso-ccitt(2) ms(9) function(2) part9(9) package(4) attributeModificationPackage(9) };
PRESENT IF ! l'opération est du type remplacement, adjonction, suppression ou création !,

actionsPackage PACKAGE
BEHAVIOUR actionsBehaviour BEHAVIOUR
DEFINED AS
! Les valeurs d'attribut identifiées par l'attribut liste de filtres d'action doivent faire partie de la cible. Si l'attribut liste de filtres d'action est vide, toutes les actions et leurs valeurs d'information doivent faire partie de la cible pour l'opération à effectuer sur les objets gérés désignés par l'objet géré de la classe supérieure cibles. Si l'attribut liste de filtres d'action désigne une action sans contrainte sur sa valeur, toutes les valeurs de cette action doivent faire partie de la cible pour l'opération à effectuer sur les objets gérés désignés par l'objet géré de la classe supérieure cibles.

NOTE – Aux fins du filtrage, les paramètres d'action peuvent être désignés sous la forme d'attributs utilisant le modèle paramétrique défini dans la Rec. X.722 du CCITT | ISO/CEI 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 ! l'opération est du type action !,

scopePackage PACKAGE
BEHAVIOUR scopeBehaviour BEHAVIOUR
DEFINED AS
! Les valeurs de visibilité et de synchronisation indiquées par les attributs visibilité et synchronisation doivent faire partie de la cible. ! ;;
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 ! l'opération est du type sélection d'objets multiples !;

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

A.2.7 Initiateurs

La classe d'objets gérés initiateurs sert à identifier un ensemble d'éventuels demandeurs d'opération. Le moyen précis d'identifier les initiateurs dépend de la politique de sécurité. La classe d'objets gérés initiateurs n'est pas destinée à être instanciée mais vise à être spécialisée pour que les demandeurs d'opération puissent être identifiés conformément à une politique de sécurité donnée. Il est recommandé de faire enregistrer tous les ensembles contenant des attributs d'identification des demandeurs, de manière que l'attribut ensemble puisse être utilisé afin d'identifier la politique de sécurité.

```
initiators MANAGED OBJECT CLASS
        DERIVED FROM accessControl;
        CHARACTERIZED BY initiatorsPackage PACKAGE
            BEHAVIOUR initiatorsBehaviour BEHAVIOUR
                DEFINED AS
                    ! La classe initiateurs désigne des demandeurs individuels d'opérations de gestion conformément
                    aux règles applicables dans les systèmes de contrôle d'accès. La diversité des mécanismes possibles
                    empêche une représentation unique des initiateurs. Des spécialisations de la classe d'objets gérés
                    initiateurs fourniront des attributs permettant d'identifier les demandeurs conformément aux
                    mécanismes de contrôle d'accès indiqués.

                    Lorsqu'une spécialisation identifie plusieurs mécanismes de contrôle d'accès, elle doit également
                    contenir un attribut de comportements afin de résoudre les conflits entre droits d'accès associés aux
                    différents mécanismes. ! ;;
            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 Initiateurs par liste ACL

```
aclInitiators MANAGED OBJECT CLASS
        DERIVED FROM initiators;
        CHARACTERIZED BY aclPackage PACKAGE
            BEHAVIOUR aclInitiatorsBehaviour BEHAVIOUR
                DEFINED AS
                    ! Cette classe d'objets gérés est utilisée pour prendre en charge un système de contrôle d'accès
                    utilisant une liste de contrôle d'accès (liste ACL).

                    La classe d'objets gérés initiateurs par liste ACL contient une liste de noms ou d'autres
                    identificateurs formant ensemble une liste de contrôle d'accès. L'identité d'un demandeur
                    d'opération de gestion doit être mise en correspondance avec les entrées d'une liste de commande
                    d'accès afin de déterminer si le demandeur est un initiateur autorisé.

                    Plusieurs objets gérés de classe initiateurs par liste ACL peuvent être instanciés dans le cadre d'un
                    objet géré de classe règle.

                    Une notification de changement de valeur d'attribut doit être émise lorsqu'un attribut de cette classe
                    d'objets est modifié. ! ;;
            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 Initiateurs par capacité

```
capabilityInitiators MANAGED OBJECT CLASS
        DERIVED FROM initiators;
        CHARACTERIZED BY capabilityPackage PACKAGE
            BEHAVIOUR capabilityInitiatorsBehaviour BEHAVIOUR
                DEFINED AS
                    ! La classe d'objets gérés initiateurs par capacité contient une liste d'identités qui sont utilisées pour
                    déterminer si l'initiateur de la demande est autorisé à utiliser la capacité de sécurité associée à la
                    demande d'accès.

                    L'identité associée à la demande d'accès est mise en correspondance avec le contenu de l'attribut
                    Liste d'identités de capacité afin de déterminer si la capacité de sécurité associée à la demande
                    d'accès peut être utilisée par l'initiateur de cette demande.
```

Ces identificateurs peuvent être des noms d'individus, des noms de groupe, des noms de rôle ou des noms d'application qui peuvent être associés à un ensemble facultatif de paires (nom d'autorité du domaine de sécurité – type d'opération); ces identités peuvent aussi prendre une forme non spécifiée dans le cadre de la présente Recommandation | Norme internationale.

NOTE – Lorsqu'un mécanisme de capacité est utilisé, les objets gérés de classe règle spécifiant le refus d'autorisation ne sont pas requis. L'absence d'identité dans l'attribut liste d'identités de capacité invalide celle-ci. En outre, les objets gérés de classe cibles et les objets gérés opérations associés ne sont pas requis, sauf si d'autres contraintes d'accès sont requises afin de mettre en œuvre des raffinements de politique locale de sécurité dans la politique générale du domaine de sécurité.

Une notification de changement de valeur d'attribut doit être émise lorsqu'un attribut de cette classe d'objets est modifié. ! ;;

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 Initiateurs par étiquette

labelInitiators MANAGED OBJECT CLASS

DERIVED FROM initiators;

CHARACTERIZED BY labelPackage PACKAGE

BEHAVIOUR labelInitiatorsBehaviour BEHAVIOUR

DEFINED AS

! L'objet géré initiateurs par étiquette peut être utilisé pour spécifier des contraintes sur des opérations de gestion qui s'ajoutent à la contrainte prescrivant une correspondance de compatibilité entre l'étiquette de sécurité associée à l'initiateur et l'étiquette de sécurité associée à la cible.

L'accès ne doit être accordé ou refusé à un initiateur conformément à la règle supérieure que si l'étiquette de sécurité de cet initiateur fait partie de l'ensemble d'étiquettes de sécurité désigné par l'attribut étiquettes de sécurité, que si l'opération effectuée sur la cible est conforme aux conditions spécifiées par l'objet géré cibles applicable et par les objets gérés opérations associés à la règle, et si l'étiquette de sécurité de l'initiateur est compatible avec l'étiquette de sécurité assignée à la cible.

NOTE – L'association d'une étiquette de sécurité à une cible doit avoir été effectuée avant l'utilisation de cette étiquette dans la procédure ci-dessus. Les étiquettes de sécurité sont associées aux cibles au moyen des objets gérés étiquettes assignées, étiquette d'attribut, étiquette d'instance et étiquette de classe et conformément aux procédures associées qui sont décrites en 7.4.

Une notification de changement de valeur d'attribut doit être émise lorsqu'un attribut de cette classe d'objets est modifié. ! ;;

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 Etiquettes assignées

assignedLabels MANAGED OBJECT CLASS

DERIVED FROM top;

CHARACTERIZED BY assignedLabelsPackage PACKAGE

BEHAVIOUR assignedLabelsPkgBehav BEHAVIOUR

DEFINED AS

! Cet objet géré contient les objets gérés étiquette d'attribut, étiquette d'instance et étiquette de classe qui, en combinaison avec les relations de préséance, attribuent une unique étiquette de sécurité aux cibles.

Il doit y avoir un seul objet géré de cette classe par fonction décisionnelle de contrôle d'accès.

Pour assurer l'association d'une unique étiquette de sécurité avec une cible, une relation de préséance est spécifiée comme suit entre les classes d'objets gérés étiquette d'attribut, étiquette d'instance et étiquette de classe, ainsi qu'à l'intérieur de ces classes:

– **Relations de préséance entre classes:**

objet géré étiquette d'attribut > objet géré étiquette d'instance > objet géré étiquette de classe d'objets

– **Relation de préséance à l'intérieur d'une classe:**

Tous les objets gérés de classe étiquette d'attribut, étiquette d'instance et étiquette de classe doivent être considérés comme ordonnés à l'intérieur de leurs classes respectives d'objet géré en fonction de la valeur de l'attribut dénomination de chacun de ces objets.

La valeur de l'attribut étiquette de sécurité à l'intérieur de l'objet géré étiquette d'attribut, étiquette d'instance ou étiquette de classe qui fait référence à la cible, directement ou indirectement, qui possède la plus grande préséance de classe et qui est la première dans l'ordre lexicographique de cette classe, doit être associée à la cible.

Si une étiquette de sécurité n'est pas associée à une cible par un objet géré étiquette d'attribut, étiquette d'instance ou étiquette de classe, l'étiquette de sécurité contenue par défaut dans l'attribut étiquette de sécurité de cet objet géré doit être associée à la cible.

La classe d'objets gérés étiquettes assignées doit émettre la notification de création d'objet lorsqu'un objet géré de cette classe est créé; et elle doit émettre la notification de suppression d'objet lorsqu'un objet géré de cette classe est supprimé. Une notification de changement de valeur d'attribut doit être émise lorsqu'un attribut de cette classe d'objets gérés est modifié. ! ;;

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 Etiquette d'attribut

attributeLabel MANAGED OBJECT CLASS
DERIVED FROM assignedLabels;
CHARACTERIZED BY attributeLabelPackage PACKAGE
BEHAVIOUR attributeLabelPkgBehav BEHAVIOUR
DEFINED AS

! Cet objet géré associe une étiquette de sécurité à des attributs spécifiques à l'intérieur d'un objet géré.

L'étiquette de sécurité est la valeur contenue dans l'attribut étiquette de sécurité.

Les attributs sont les valeurs contenues dans l'attribut liste d'identificateurs d'attribut.

L'objet géré est la valeur contenue dans l'attribut instance d'objet géré.

Un objet géré étiquettes assignées peut contenir plusieurs objets gérés de cette classe.

Le comportement des objets gérés de classe étiquette d'attribut par rapport aux autres objets de cette classe et le comportement des objets gérés à l'intérieur des classes d'objets gérés étiquette d'instance et étiquette de classe doivent être tels que définis dans le comportement de l'objet géré étiquettes assignées. ! ;;

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 Etiquette d'instance

instanceLabel MANAGED OBJECT CLASS
DERIVED FROM assignedLabels;
CHARACTERIZED BY instanceLabelPackage PACKAGE
BEHAVIOUR instanceLabelPkgBehav BEHAVIOUR
DEFINED AS

! Cet objet géré associe une étiquette de sécurité à des objets gérés spécifiques.

L'étiquette de sécurité est la valeur contenue dans l'attribut étiquette de sécurité.

Les identificateurs d'objet géré sont contenus dans l'attribut instances d'objet géré.

Un objet géré étiquettes assignées peut contenir plusieurs objets gérés de cette classe.

Le comportement des objets gérés de classe étiquette d'instance par rapport aux autres objets de cette classe, ainsi que le comportement des objets gérés contenus dans les classes d'objets gérés de type étiquette d'attribut et étiquette de classe doivent être tels que définis dans le comportement de l'objet géré étiquettes assignées. ! ;;

ATTRIBUTES
managedObjectInstances GET;;;

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

A.2.14 Etiquette de classe

```
classLabel MANAGED OBJECT CLASS
        DERIVED FROM assignedLabels;
        CHARACTERIZED BY  classLabelPackage  PACKAGE
            BEHAVIOUR  classLabelPkgBehav BEHAVIOUR
                DEFINED AS
                    ! Cet objet géré associe une étiquette de sécurité à des objets gérés spécifiques.

                    L'étiquette de sécurité est la valeur contenue dans l'attribut étiquette de sécurité.

                    Les identificateurs d'objet géré sont contenus dans l'attribut classes d'objets gérés.

                    Un objet géré étiquettes assignées peut contenir plusieurs objets gérés de cette classe.

                    Le comportement des objets gérés de classe étiquette de classe par rapport aux autres objets de cette
                    classe, ainsi que le comportement des objets gérés contenus dans les classes d'objets gérés de type
                    étiquette d'attribut et étiquette d'instance doivent être tels que définis dans le comportement de
                    l'objet géré étiquettes assignées. ! ;;
        ATTRIBUTES
            managedObjectClasses        GET;;;

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

A.3 Définitions des corrélations de noms

A.3.1 Règle – Règle de contrôle d'accès

```
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 Opérations – Cibles

```
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 Emetteur de notification – Règles de contrôle d'accès

```
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 Etiquette d'attribut – Etiquettes assignées

```
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 Etiquette d'instance – Etiquettes assignées

```
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 Etiquette de classe – Etiquettes assignées

```
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 Définition de paramètres

A.4.1 Filtre de contrôle d'accès invalide

```
invalidAccessControlFilter PARAMETER
        CONTEXT SPECIFIC-ERROR;
            WITH SYNTAX AccessControlDefinitions.InvalidAccessControlFilter;
                BEHAVIOUR invalidAccessControlFilterBehaviour BEHAVIOUR
                    DEFINED AS
                        ! Cette erreur spécifique du traitement par service CMIS signale une erreur survenue dans un
                        élément de filtrage de contrôle d'accès qui est proposé. Sa valeur doit être une séquence
                        d'identificateurs d'erreur prenant une des valeurs suivantes: "duplicateId", "heterogeneousId"
                        ou "invalidId", assortie d'un filtre CMIS facultatif contenant le filtre signalé en erreur. ! ;;

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

A.5 Définition des attributs

A.5.1 Liste de contrôle d'accès

```
accessControlList ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.AccessControlList;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
        BEHAVIOUR aclBehaviour BEHAVIOUR
            DEFINED AS
                ! Cet attribut sert à spécifier une liste d'initiateurs à utiliser dans un système fondé sur une liste de
                contrôle d'accès. Les initiateurs sont identifiés par leur nom personnel, par une référence anonyme ou par
                un nom de groupe, par des rôles ou par des titres d'entité d'application. Les initiateurs peuvent être
                associés à des applications spécifiées. Les noms de groupe individuels peuvent être utilisés dans le cadre
                de l'Annuaire OSI.

                Cet attribut permet d'utiliser un nom d'initiateur ou un nom par procuration (alias). Le nom d'initiateur
                peut prendre la forme syntaxique d'un nom distinctif ou d'un titre d'entité d'application, alors que le nom
                par procuration (alias) prend la forme d'un identificateur d'objet et d'une valeur.

                La forme de nom distinctif peut être utilisée pour identifier un initiateur spécifique, un groupe
                d'initiateurs ou un rôle particulier.

                La forme de nom par titre d'entité d'application désigne le titre d'entité d'application et, par référence, le
                système qui a émis la demande.

                La forme de nom par procuration (alias) est utilisée lorsque la forme du nom n'est pas spécifiquement
                celle d'un initiateur, d'un groupe d'initiateurs, d'un rôle ou d'un titre d'entité d'application. La
                procuration permet donc à l'initiateur d'être anonyme. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) accessControlList(1) };
```

A.5.2 Filtre de contrôle d'accès

L'attribut suivant est défini aux fins des héritages.

accessControlFilter ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.FilterList;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR accessControlFilterBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut de type sorte (évalué sur un ensemble) fournit un ensemble de filtres CMIS pour contraindre les paramètres des opérations de gestion. Si cet ensemble est vide, le filtre CMIS doit être considéré comme désignant toutes les cibles pouvant être identifiées par l'attribut dérivé.

Pour tout filtre compris dans l'ensemble, chaque item de filtre CMIS doit désigner le même attribut. Les tentatives de violation de cette contrainte doivent donner lieu à une erreur spécifique d'invalidité du filtre de contrôle d'accès avec l'identificateur d'erreur heterogeneousIds.

Aucun attribut ne doit être associé à plus d'un filtre CMIS. Les tentatives de violation de cette contrainte doivent donner lieu à une erreur spécifique d'invalidité du filtre de contrôle d'accès avec l'identificateur d'erreur duplicateIds.

Toutes les valeurs des champs d'identification d'attribut dans les items de filtre CMIS doivent désigner des informations de gestion qui sont valides pour la spécification donnée de cet attribut. Toute violation de cette contrainte doit donner lieu à une erreur spécifique d'invalidité du filtre de contrôle d'accès avec l'identificateur d'erreur invalidIdentifier. ! ;;
PARAMETERS invalidAccessControlFilter;

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

A.5.3 Nom des objets de classe contrôle d'accès

accessControlObjectName ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.AccessControlObjectName;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR accessControlObjectNameBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut est utilisé pour identifier des instanciations de spécialisations de la classe des objets gérés contrôle d'accès. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) accessControlObjectName(3) };

A.5.4 Liste de filtres d'action

actionFilterList ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.ActionFilterList;
MATCHES FOR EQUALITY, SET-INTERSECTION, SET-COMPARISON;
BEHAVIOUR actionFilterlistBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut évalué sur un ensemble identifie les actions et, en option, les contraintes sur leurs valeurs d'argument au moyen d'un filtre CMIS.

Pour tout filtre CMIS de l'ensemble, chaque item de filtre CMIS doit identifier le même attribut. Les tentatives de violation de cette contrainte doivent donner lieu à l'erreur spécifique d'invalidité du filtre de contrôle d'accès avec l'identificateur d'erreur heterogeneousIds.

Aucun attribut ne doit être associé à plus d'un seul filtre CMIS. Les tentatives de violation de cette contrainte doivent donner lieu à une erreur spécifique d'invalidité du filtre de contrôle d'accès avec l'identificateur d'erreur duplicateIds.

Toutes les valeurs des champs de l'identificateur d'attribut dans les items de filtre CMIS doivent désigner des informations de gestion qui sont valides pour la spécialisation donnée de cet attribut. Toute violation de cette contrainte doit donner lieu à l'erreur spécifique d'invalidité du filtre de contrôle d'accès avec l'identificateur d'erreur invalidIdentifier. ! ;;
PARAMETERS invalidAccesscontrolFilter;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) actionFilterList(4) };

A.5.5 Liste de filtres d'attribut

attributeFilterList ATTRIBUTE
DERIVED FROM accessControlFilter;
BEHAVIOUR attributeFilterListBehaviour BEHAVIOUR
DEFINED AS

! Cet attribut identifie les contraintes portant sur les valeurs des attributs.

Si un attribut est identifié sans contraintes sur sa valeur, par exemple:

{ item : present : globalForm : accessControlList }

toutes les valeurs de cet attribut sont identifiées.

Si l'ensemble est vide, il n'y a pas de contraintes. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) attributeFilterList(5) };

A.5.6 Contexte d'authentification

authenticationContext ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.AuthenticationContext;
BEHAVIOUR authenticationContextPackageBehaviour BEHAVIOUR
DEFINED AS
! L'attribut contexte d'authentification est une séquence constituée de l'identificateur de politique d'authentification et des prescriptions ainsi désignées. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) authenticationContext(6) };

A.5.7 Liste d'identités de capacité

capabilityIdentitiesList ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.CapabilityIdentitiesList;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR capabilityBehaviour BEHAVIOUR
DEFINED AS
! L'attribut liste d'identités de capacité contient un ensemble d'identités.

Ces identités peuvent être un nom d'individu, un nom de groupe, un nom de rôle, ou un nom d'application, chacun de ces noms pouvant être associé à un ensemble facultatif constitué de paires {nom d'autorité du domaine de sécurité/type d'opération}; ou bien l'identité peut avoir une forme non spécifiée dans la présente Recommandation | Norme internationale. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) capabilityIdentitiesList(7) };

A.5.8 Accès par défaut

defaultAccess ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DefaultAccess;
MATCHES FOR EQUALITY;
BEHAVIOUR defaultAccessBehaviour BEHAVIOUR
DEFINED AS
! L'attribut accès par défaut identifie, conformément au 7.4.3.1.6, les droits d'accès par défaut pour chaque type d'opération. Sa valeur est une séquence énumérant les actions exécutives pour chaque type d'opération. La valeur par défaut de cet attribut doit être le refus de toutes les opérations avec la réponse de refus d'accès. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) defaultAccess(8) };

A.5.9 Réponse de refus par défaut

defaultDenialResponse ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DenialResponse;
MATCHES FOR EQUALITY;
BEHAVIOUR denialResponseBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut définit la réponse de refus à renvoyer si le refus a été choisi à la suite de l'application de la règle par défaut. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) defaultDenialResponse(9) };

A.5.10 Granularité du refus

denialGranularity ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DenialGranularity;
MATCHES FOR EQUALITY;
BEHAVIOUR denialGranularityBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut identifie le niveau auquel le refus d'accès doit être manifesté, s'il doit l'être. Il doit prendre une des valeurs suivantes: demande, objet ou attribut. Si la valeur est demande, celle-ci doit être entièrement rejetée si une quelconque cible contenue dans cette demande est refusée. Si la valeur de

l'attribut est objet, la demande visant cet objet géré doit être rejetée si une quelconque cible contenue dans la demande de cet objet est refusée. Si la valeur de l'attribut est attribut, la demande doit être rejetée au niveau de l'attribut. ! ;;

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

A.5.11 Identité de domaine

domainIdentity ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.DomainIdentity;
MATCHES FOR EQUALITY;
BEHAVIOUR domainNameBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut identifie le domaine de contrôle d'accès régissant les présentes règles de contrôle d'acccès. ! ;;

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

A.5.12 Action exécutive

enforcementAction ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.EnforcementAction;
MATCHES FOR EQUALITY;
BEHAVIOUR enforcementActionBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut identifie l'action à exécuter si la règle est appliquée. Il prend une des valeurs suivantes: refus, refus avec réponse (valeur par défaut), refus sans réponse, abandon d'association, refus avec réponse d'erreur, et autorisation. ! ;;

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

A.5.13 Filtre

filter ATTRIBUTE
DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": discriminatorConstruct;
BEHAVIOUR filterBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut identifie un filtre à appliquer aux objets gérés identifiés par les autres attributs de l'objet géré cibles afin de déterminer leur inclusion sous forme d'objet géré protégé. ! ;;

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

A.5.14 Information ACI d'initiateur requise

initiatorACImandated ATTRIBUTE
WITH ATTRIBUTE SYNTAX AccessControlDefinitions.Boolean;
MATCHES FOR EQUALITY;
BEHAVIOUR initiatorACImandatedBehaviour BEHAVIOUR
DEFINED AS
! L'attribut information ACI d'initiateur requise est du type booléen. Cet attribut sert à indiquer si, pour appliquer le système de contrôle d'accès choisi, des informations ACI d'initiateur sont requises avec chaque demande d'opération de gestion individuelle. Lorsque cet attribut prend la valeur TRUE, cela indique que des informations ACI d'initiateur sont requises dans chaque demande d'opération de gestion, tandis que la valeur FALSE indique qu'aucune information ACI d'initiateur n'est requise. Si l'attribut a la valeur TRUE et que la demande d'opération de gestion ne contienne pas d'information ACI d'initiateur, l'accès est refusé. ! ;;

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

A.5.15 Liste d'initiateurs

initiatorsList ATTRIBUTE
DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": member;
BEHAVIOUR initiatorsListBehaviour BEHAVIOUR
DEFINED AS
! Cet attribut, évalué sur un ensemble, identifie les sous-classes de la classe des objets gérés de classe initiateurs qui spécifient les initiateurs auxquels la règle doit s'appliquer. Une erreur doit découler de toute tentative d'inclure dans l'attribut liste d'initiateurs une valeur qui n'est pas le nom d'un objet géré de la classe initiateurs. ! ;;

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

A.5.16 Tentatives d'accès invalides

```
invalidAccessAttempts  ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992": counter;
        BEHAVIOUR invalidAccessAttemptBehaviourPkg  BEHAVIOUR
            DEFINED AS
                !  Cet attribut est utilisé pour compter le nombre d'occasions où une fonction décisionnelle de contrôle
                d'accès n'a pas autorisé l'accès. Cet attribut prend la forme d'un compteur non réarmable, comme défini
                par la Rec. X.721 du CCITT | ISO/CEI 10165-2.  ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) invalidAccessAttempts(16) };
```

A.5.17 Nom d'étiquette

```
labelName  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  AccessControlDefinitions.LabelName;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR  labelNameBehaviourPkg  BEHAVIOUR
            DEFINED AS
                !  Cet attribut attribue aux étiquettes de sécurité un entier de type name of qui permet de lancer une
                vérification d'ordonnancement. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) labelName(17) };
```

A.5.18 Classes d'objets gérés

```
managedObjectClasses ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.ObjectClassList;
        MATCHES FOR EQUALITY SET-COMPARISON, SET-INTERSECTION;
        BEHAVIOUR managedObjectClassesBehaviour BEHAVIOUR
            DEFINED AS
                !  Cet attribut, évalué sur un ensemble, identifie les classes d'objets gérés protégés ainsi que les
                corrélations de noms facultativement associées.

                Toute tentative d'inclusion d'une valeur non reconnue comme étant de la classe des objets gérés dans le
                domaine doit donner lieu à l'erreur spécifique d'invalidité d'attribut de service CMIS. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) managedObjectClasses(18) };
```

A.5.19 Instances d'objets gérés

```
managedObjectInstances ATTRIBUTE
        DERIVED FROM "CCITT Rec. X.721 | ISO/IEC 10165-2:1992": member;
        BEHAVIOUR managedObjectInstancesBehaviourPkg  BEHAVIOUR
            DEFINED AS
                !  Cet attribut, évalué sur un ensemble, identifie des objets gérés protégés.  ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) managedObjectInstances(19) };
```

A.5.20 Type d'opération

```
operationType ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.OperationType;
        MATCHES FOR EQUALITY;
            BEHAVIOUR operationTypeBehaviourPkg  BEHAVIOUR
                DEFINED AS
                    !  Cet attribut en lecture seulement est utilisé pour dénommer des objets gérés de la classe des
                    opérations. Il peut prendre une des valeurs suivantes: lecture, remplacement, adjonction de
                    membre, suppression de membre, remplacement par défaut, sélection d'objets multiples, filtre,
                    création, suppression, et action. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) operationType(20) };
```

A.5.21 Liste d'opérations

```
operationsList  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  AccessControlDefinitions.OperationsList;
        MATCHES FOR EQUALITY,  SET-COMPARISON,  SET-INTERSECTION;
        BEHAVIOUR  operationsListBehaviourPkg  BEHAVIOUR
            DEFINED AS
                ! Cet attribut, évalué sur un ensemble, identifie les opérations qui sont à autoriser ou à refuser, selon les
                permissions contenues dans l'objet géré de la classe supérieure règle, ces opérations étant à appliquer à
                des cibles désignées par l'objet géré de classe cibles. Les opérations sont désignées par l'attribut type
                d'opération. L'attribut liste d'opérations peut être utilisé lorsque aucune contrainte conditionnelle n'est
                imposée aux paramètres d'une opération.  ! ;;

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

A.5.22 Visibilité

```
scope ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.Scope;
        MATCHES FOR EQUALITY;
        BEHAVIOUR scopeBehaviourPkg  BEHAVIOUR
            DEFINED AS
                ! L'attribut visibilité identifie la portée d'une sélection d'objets gérés protégés.  ! ;;

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

A.5.23 Filtre de visibilité

```
scopeFilter ATTRIBUTE
        DERIVED FROM accessControlFilter;
        BEHAVIOUR scopeFilterBehaviour BEHAVIOUR
            DEFINED AS
                !  Pour les demandes qui sélectionnent des objets gérés multiples, le filtre de visibilité spécifie des
                contraintes imposées au paramètre visibilité de la demande. L'identificateur de l'attribut visibilité est
                utilisé pour tous les items de type filtre contenus dans le filtre.

                Cet attribut identifie un filtre applicable au paramètre "visibilité" des opérations de gestion. Il ne doit
                avoir qu'un seul élément ou aucun.  ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) scopeFilter(23) };
```

A.5.24 Etiquette de sécurité

```
securityLabel ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.SecurityLabel;
        MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
        BEHAVIOUR securityLabelBehaviour BEHAVIOUR
            DEFINED AS
                ! L'attribut étiquette de sécurité contient une étiquette de sécurité.  ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) securityLabel(24) };
```

A.5.25 Contextes d'état

```
stateConditions ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.StateConditions;
        MATCHES FOR EQUALITY;
        BEHAVIOUR stateConditionsPackageBehaviour BEHAVIOUR
            DEFINED AS
                ! Cet attribut identifie un objet géré et un filtre applicables aux attributs de cet objet géré. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) stateConditions(25) };
```

A.5.26 Synchronisation

Cet attribut fournit un identificateur d'attribut et une syntaxe de filtrage pour le paramètre de synchronisation des opérations de gestion.

```
synchronization ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AccessControlDefinitions.CMISSync;
        BEHAVIOUR synchronizationBehaviour BEHAVIOUR
            DEFINED AS
                !  Cette valeur d'attribut représente le paramètre de synchronisation des opérations de gestion. Cet
                attribut est utilisé pour représenter des filtres applicables à ce paramètre.  ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) synchronization(26) };
```

A.5.27 Filtre de synchronisation

```
synchronizationFilter ATTRIBUTE
        DERIVED FROM accessControlFilter;
        BEHAVIOUR synchronizationFilterBehaviour BEHAVIOUR
            DEFINED AS
                ! Pour les demandes qui sélectionnent des objets gérés multiples, le filtre de synchronisation spécifie des
                contraintes sur le paramètre de synchronisation de la demande et l'identificateur de l'attribut
                synchronisation est utilisé pour tous les items de type filtre contenus dans le filtre.

                Cet attribut identifie un filtre applicable au paramètre de synchronisation des opérations de gestion. Il ne
                doit avoir qu'un seul élément ou aucun. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) synchronizationFilter(27) };
```

A.5.28 Liste de cibles

```
targetsList ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992": member;
        BEHAVIOUR targetsListBehaviour BEHAVIOUR
            DEFINED AS
                ! Cet attribut, évalué sur un ensemble, identifie les objets gérés de classe cibles qui eux-mêmes spécifient
                les cibles auxquelles la règle de refus ou d'autorisation d'item est applicable. Une erreur doit découler
                d'une tentative d'inclure une valeur qui n'est pas reconnue comme étant le nom d'un objet géré de classe
                cibles ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) targetsList(28) };
```

A.5.29 Tentatives d'accès valides

```
validAccessAttempts  ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992": counter;
        BEHAVIOUR validAccessAttemptBehaviourPkg  BEHAVIOUR
            DEFINED AS
                ! Cet attribut est utilisé pour compter le nombre de fois où une fonction décisionnelle de contrôle d'accès
                a autorisé l'accès. Cet attribut prend la forme d'un compteur non réarmable, tel que défini par la
                Rec. X.721 du CCITT | ISO/CEI 10165-2. ! ;;
REGISTERED AS { joint-iso-ccitt(2) ms(9) function(2) part9(9) attribute(7) validAccessAttempts(29) };
```

A.6 Définitions de syntaxe abstraite

```
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
```

Annexe B

Formulaire MCS[5)]

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

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.

5) Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MCS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété. Les instructions pour compléter le formulaire MCS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 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

		Manager		Agent		
Index	Systems management functional unit name	Status	Support	Status	Support	Additional information
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	accessControl-Rules	–	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	notification-Emitter	–	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	capability-Initiators	–	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	securityAlarm-record	–	c9			
15	CCITT Rec. X.740 | ISO/IEC 10164-8	Annex D, all tables	securityAudit-Trailrecord	–	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 –.								

Annexe C

Formulaire MICS[6)]

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

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.

---

6) Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MICS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété.

Table C.1 – Attribute support

				Set by create		Get	
Index	Attribute template label	Value of object identifier for the attribute	Constraints and values	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**

	Replace		Add		Remove		Set to default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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		

Annexe D

Formulaire MOCS[7)]

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

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}		

7) Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MOCS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété. Les instructions pour compléter le formulaire MOCS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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

					Support		
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information
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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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	3.4	correlatedNotifications	{dmi-att 12}	–	o		
(cont.)	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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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

					Support		
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information
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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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

					Support		
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information
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	3.4.1	correlatedNotification	–	–	c:m		
(cont.)	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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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

					Support		
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information
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

					Support		
Index	Parameter template label	Value of parameter identifier	Constraints and values	Status	Confirmed	Non-confirmed	Additional information
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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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		
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	3.4.1	correlatedNotification	–	–	c:m		
(cont.)	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

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	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**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
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

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

Table D.36 *(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}	–	c91		
	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.27		
	1.5.2.2	nonSpecificForm	–	–	c:o.27		
	1.5.2.3	localDistinguishedName	–	–	c:o.27		
	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}	–	c92		
	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.28		
	2.4.2.2	nonSpecificForm	–	–	c:o.28		
	2.4.2.3	localDistinguishedName	–	–	c:o.28		
	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}	–	c93		
	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.29		
	3.4.2.2	nonSpecificForm	–	–	c:o.29		
	3.4.2.3	localDistinguishedName	–	–	c:o.29		
	3.5	additionalText	{dmi-att 7}	–	o		
	3.6	additionalInformation	{dmi-att 6}	–	o		

c91: if D.36/1.5 then m else –.

c92: if D.36/2.4 then m else –.

c93: if D.36/3.4 then m else –.

D.11 Label initiators managed object class

D.11.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the label 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.37.

Table D.37 – 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	labelInitiators	{ac-obj 10}		

If the answer to the actual class question in the managed object class support Table D.37 is "N", the supplier of the implementation shall supply the actual class support details, in Table D.38.

Table D.38 – Actual class support

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

D.11.2 Packages

See Table D.39.

Table D.39 – 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}	–	c94		
3	allomorphicPackage	{dmi-pkg 17}	–	c95		
4	accessControlPackage	–	–	m		
5	initiatorsPackage	–	–	m		
6	labelPackage	{ac-pkg 14}	–	m		
c94: if D.39/3 then m else –. c95: if D.37/1b then – else m.						

D.11.3 Attributes

See Table D.40.

Table D.40 – Attribute support

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c96		c96	
4	allomorphs	{dmi-att 50}	–	c97		c97	
5	accessControlObjectName	{ac-att 3}	–	m		m	
6	initiatorACImandated	{ac-att 14}	–	m		m	
7	securityLabel	{ac-att 24}	–	m		m	

Table D.40 *(concluded)* **– Attribute support**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	m		–		–		m		
7	m		–		–		–		
c96: if D.39/2 then m else –. c97: if D.39/3 then m else –.									

D.11.4 Notifications

See Table D.41.

Table D.41 – Notification support

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

Table D.41 *(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}	–	c98		
	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.30		
	1.5.2.2	nonSpecificForm	–	–	c:o.30		
	1.5.2.3	localDistinguishedName	–	–	c:o.30		
	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}	–	c99		
	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.31		
	2.4.2.2	nonSpecificForm	–	–	c:o.31		
	2.4.2.3	localDistinguishedName	–	–	c:o.31		
	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}	–	c100		
	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.32		
	3.4.2.2	nonSpecificForm	–	–	c:o.32		
	3.4.2.3	localDistinguishedName	–	–	c:o.32		
	3.5	additionalText	{dmi-att 7}	–	o		
	3.6	additionalInformation	{dmi-att 6}	–	o		

c98: if D.41/1.5 then m else –.

c99: if D.41/2.4 then m else –.

c100: if D.41/3.4 then m else –.

D.12 Assigned labels managed object class

D.12.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the assigned labels 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.42.

Table D.42 – 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	labelInitiators	{ac-obj 10}		

If the answer to the actual class question in the managed object class support Table D.42 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.43.

Table D.43 – Actual class support

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

D.12.2 Packages

See Table D.44.

Table D.44 – 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}	–	c101		
3	allomorphicPackage	{dmi-pkg 17}	–	c102		
4	assignedLabelsPackage	–	–	m		
c101: if D.44/3 then m else –. c102: if D.42/1b then – else m.						

D.12.3 Attributes

See Table D.45.

Table D.45 – Attribute support

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c103		c103	
4	allomorphs	{dmi-att 50}	–	c104		c104	
5	labelName	{ac-att 17}	–	m		m	
6	securityLabel	{ac-att 24}	–	m		m	

Table D.45 *(concluded)* **– Attribute support**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	–		–		–		m		
c103: if D.44/2 then m else –. c104: if D.44/3 then m else –.									

D.12.4 Notifications

See Table D.46.

Table D.46 – Notification support

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

Table D.46 *(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	attributeList	{dmi-att 9}	–	o		
	1.3	notificationIdentifier	{dmi-att 16}	–	c105		
	1.4	correlatedNotifications	{dmi-att 12}	–	o		
	1.4.1	correlatedNotification	–	–	c:m		
	1.4.2	sourceObjectInst	–	–	c:o		
	1.4.2.1	distinguishedName	–	–	c:o.33		
	1.4.2.2	nonSpecificForm	–	–	c:o.33		
	1.4.2.3	localDistinguishedName	–	–	c:o.33		
	1.5	additionalText	{dmi-att 7}	–	o		
	1.6	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}	–	c106		
	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.34		
	2.4.2.2	nonSpecificForm	–	–	c:o.34		
	2.4.2.3	localDistinguishedName	–	–	c:o.34		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	o		
c105: if D.46/1.4 then m else –. c106: if D.46/1.4 then m else –.							

D.13 Attribute labels managed object class

D.13.1 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.47.

Table D.47 – 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	attributeLabel	{ac-obj 12}		

If the answer to the actual class question in the managed object class support Table D.47 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.48.

Table D.48 – Actual class support

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

D.13.2 Packages

See Table D.49.

Table D.49 – 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}	–	c107		
3	allomorphicPackage	{dmi-pkg 17}	–	c108		
4	assignedLabelsPackage	–	–	m		
5	attributeLabelPackage	–	–	m		
c107: if D.49/3 then m else –. c108: if D.47/1b then – else m.						

D.13.3 Attributes

See Table D.50.

Table D.50 – Attribute support

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c109		c109	
4	allomorphs	{dmi-att 50}	–	c110		c110	
5	labelName	{ac-att 17}	–	m		m	
6	securityLabel	{ac-att 24}	–	m		m	
7	managedObjectInstance	{dmi-att 61}	–	m		m	
8	attributeIdentifierList	{dmi-att 8}	–	m		m	

Table D.50 *(concluded)* **– Attribute support**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	–		–		–		m		
7	m		–		–		–		
8	m		–		–		–		
c109: if D.48/2 then m else –. c110: if D.49/3 then m else –.									

D.13.4 Notifications

See Table D.51.

Table D.51 – Notification support

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

Table D.51 *(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	attributeList	{dmi-att 9}	–	o		
	1.3	notificationIdentifier	{dmi-att 16}	–	c111		
	1.4	correlatedNotifications	{dmi-att 12}	–	o		
	1.4.1	correlatedNotification	–	–	c:m		
	1.4.2	sourceObjectInst	–	–	c:o		
	1.4.2.1	distinguishedName	–	–	c:o.35		
	1.4.2.2	nonSpecificForm	–	–	c:o.35		
	1.4.2.3	localDistinguishedName	–	–	c:o.35		
	1.5	additionalText	{dmi-att 7}	–	o		
	1.6	additionalInformation	{dmi-att 6}	–	o		

Table D.51 *(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}	–	c112		
	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.36		
	2.4.2.2	nonSpecificForm	–	–	c:o.36		
	2.4.2.3	localDistinguishedName	–	–	c:o.36		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	o		
c111: if D.51/1.4 then m else –. c112: if D.51/2.4 then m else –.							

D.14 Instance label managed object class

D.14.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the instance label 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.52.

Table D.52 – 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	instanceLabel	{ac-obj 13}		

If the answer to the actual class question in the managed object class support Table D.52 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.53.

Table D.53 – Actual class support

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

D.14.2 Packages

See Table D.54.

Table D.54 – 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}	–	c113		
3	allomorphicPackage	{dmi-pkg 17}	–	c114		
4	assignedLabelPackage	–	–	m		
5	instanceLabelPackage	–	–	m		
c113: if D.53/3 then m else –. c114: if D.51/1b then – else m.						

D.14.3 Attributes

See Table D.55.

Table D.55 – Attribute support

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c115		c115	
4	allomorphs	{dmi-att 50}	–	c116		c116	
5	labelName	{ac-att 17}	–	m		m	
6	securityLabel	{ac-att 24}	–	m		m	
7	managedObjectInstances	{ac-att 19}	–	m		m	

Table D.55 *(concluded)* **– Attribute support**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	–		–		–		m		
7	m		–		–		–		
c115: if D.53/2 then m else –. c116: if D.53/3 then m else –.									

D.14.4 Notifications

See Table D.56.

Table D.56 – 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	objectCreation	{dmi-not 6}	–	m			
2	objectDeletion	{dmi-not 7}	–	m			

Table D.56 *(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	attributeList	{dmi-att 9}	–	o		
	1.3	notificationIdentifier	{dmi-att 16}	–	c117		
	1.4	correlatedNotifications	{dmi-att 12}	–	o		
	1.4.1	correlatedNotification	–	–	c:m		
	1.4.2	sourceObjectInst	–	–	c:o		
	1.4.2.1	distinguishedName	–	–	c:o.37		
	1.4.2.2	nonSpecificForm	–	–	c:o.37		
	1.4.2.3	localDistinguishedName	–	–	c:o.37		
	1.5	additionalText	{dmi-att 7}	–	o		
	1.6	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}	–	c118		
	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.38		
	2.4.2.2	nonSpecificForm	–	–	c:o.38		
	2.4.2.3	localDistinguishedName	–	–	c:o.38		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	o		

c117: if D.56/1.4 then m else –.

c118: if D.56/2.4 then m else –.

D.15 Class label managed object class

D.15.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the class label 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.57.

Table D.57 – 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	classLabel	{ac-obj 14}		

If the answer to the actual class question in the managed object class support Table D.57 is "N", the supplier of the implementation shall supply the actual class support details, in Table D.58.

Table D.58 – Actual class support

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

D.15.2 Packages

See Table D.59.

Table D.59 – 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}	–	c119		
3	allomorphicPackage	{dmi-pkg 17}	–	c120		
4	assignedLabelsPackage	–	–	m		
5	classLabelPackage	–	–	m		
c119: if D.57/3 then m else –. c120: if D.55/1b then – else m.						

D.15.3 Attributes

See Table D.60.

Table D.60 – Attribute support

				Set by Create		Get	
Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Status	Support	Status	Support
1	objectClass	{dmi-att 65}	–	m		m	
2	nameBinding	{dmi-att 63}	–	m		m	
3	packages	{dmi-att 66}	–	c121		c121	
4	allomorphs	{dmi-att 50}	–	c122		c122	
5	labelName	{ac-att 17}	–	m		–	
6	securityLabel	{ac-att 24}	–	m		m	
7	managedObjectClasses	{ac-att 18}	–	m		m	

Table D.60 *(concluded)* **– Attribute support**

	Replace		Add		Remove		Set to Default		
Index	Status	Support	Status	Support	Status	Support	Status	Support	Additional information
1	–		–		–		–		
2	–		–		–		–		
3	–		–		–		–		
4	–		–		–		–		
5	–		–		–		–		
6	–		–		–		m		
7	–		–		–		–		
c121: if D.59/2 then m else –. c122: if D.59/3 then m else –.									

D.15.4 Notifications

See Table D.61.

Table D.61 – Notification support

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

Table D.61 *(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	attributeList	{dmi-att 9}	–	o		
	1.3	notificationIdentifier	{dmi-att 16}	–	c123		
	1.4	correlatedNotifications	{dmi-att 12}	–	o		
	1.4.1	correlatedNotification	–	–	c:m		
	1.4.2	sourceObjectInst	–	–	c:o		
	1.4.2.1	distinguishedName	–	–	c:o.39		
	1.4.2.2	nonSpecificForm	–	–	c:o.39		
	1.4.2.3	localDistinguishedName	–	–	c:o.39		
	1.5	additionalText	{dmi-att 7}	–	o		
	1.6	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}	–	c124		
	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.40		
	2.4.2.2	nonSpecificForm	–	–	c:o.40		
	2.4.2.3	localDistinguishedName	–	–	c:o.40		
	2.5	additionalText	{dmi-att 7}	–	o		
	2.6	additionalInformation	{dmi-att 6}	–	o		

c123: if D.61/2.4 then m else –.

c124: if D.61/3.4 then m else –.

Annexe E

Formulaire MRCS de corrélation de noms[8)]

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

E.1 Introduction

The purpose of this MRCS proforma for name bindings is to provide a mechanism for a supplier which claims conformance to a name binding to provide conformance information in a standard form.

E.2 Instructions for completing the MRCS proforma for name binding to produce a MRCS

The MRCS 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 Table E.1 and if necessary provide additional information.

Notations used in the Status and Support columns are specified in B.1.3.

E.3 Symbols, abbreviations and terms

The following abbreviation is used in this proforma:

ac-nb **joint-iso-ccitt ms(9) function(2) part9(9) nameBinding(6)**

E.4 Statement of conformance to the name binding

See Table E.1.

Table E.1 – Name binding support

Index	Name binding template label	Value of object identifier for name binding	Constraints and values	Status	Support	Additional information
1	rule-accessControlRules	{ac-nb 1}	–	o		
2	operations-targets	{ac-nb 2}	–	o		
3	notificationEmitter-accessControlRules	{ac-nb 3}	–	o		
4	attributeLabel-assignedLabels	{ac-nb 4}	–	o		
5	instanceLabel-assignedLabels	{ac-nb 5}	–	o		
6	classLabel-assignedLabels	{ac-nb 6}	–	o		

8) Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MRCS pour la corrélation de noms de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire MRCS une fois celui-ci complété. Les instructions pour compléter le formulaire MRCS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 10165-6.

Table E.1 *(concluded)* **– Name binding support**

Index	Sub-index	Operation	Constraints and values	Status	Support	Additional information
1	1.1	Create support	–	m		
	1.1.1	Create with automatic instance naming	–	m		
	1.1.2	Create with reference object	–	m		
	1.2	Delete support	–	m		
	1.2.1	Delete only if no contained objects	–	m		
	1.2.2	Delete contained objects	–	–		
2	2.1	Create support	–	m		
	2.1.1	Create with automatic instance naming	–	–		
	2.1.2	Create with reference object	–	m		
	2.2	Delete support	–	m		
	2.2.1	Delete only if no contained objects	–	m		
	2.2.2	Delete contained objects	–	–		
3	3.1	Create support	–	m		
	3.1.1	Create with automatic instance naming	–	m		
	3.1.2	Create with reference object	–	m		
	3.2	Delete support	–	m		
	3.2.1	Delete only if no contained objects	–	m		
	3.2.2	Delete contained objects	–	–		
4	4.1	Create support	–	m		
	4.1.1	Create with automatic instance naming	–	–		
	4.1.2	Create with reference object	–	–		
	4.2	Delete support	–	m		
	4.2.1	Delete only if no contained objects	–	–		
	4.2.2	Delete contained objects	–	–		
5	5.1	Create support	–	m		
	5.1.1	Create with automatic instance naming	–	–		
	5.1.2	Create with reference object	–	–		
	5.2	Delete support	–	m		
	5.2.1	Delete only if no contained objects	–	–		
	5.2.2	Delete contained objects	–	–		
6	6.1	Create support	–	m		
	6.1.1	Create with automatic instance naming	–	–		
	6.1.2	Create with reference object	–	–		
	6.2	Delete support	–	m		
	6.2.1	Delete only if no contained objects	–	–		
	6.2.2	Delete contained objects	–	–		

Annexe F

Formulaire MIDS (paramètres)[9)]

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

F.1 Introduction

The purpose of this MIDS proforma for parameters is to provide a mechanism for a supplier which claims conformance to the parameter to provide conformance information in a standard form.

F.2 Instructions for completing the MIDS proforma for parameters to produce a MIDS

The MIDS 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 Table F.1 and if necessary provide additional information.

Notations used in the Status and Support columns are specified in B.1.3.

F.3 Symbols, abbreviations and terms

The following abbreviation is used in this proforma:

ac-par **joint-iso-ccitt ms(9) function(2) part9(9) parameter(5)**

F.4 Instructions for completing the MIDS proforma

The specifier of a managed object class that claims to support the notifications specified by ITU-T Rec. X.741 | ISO/IEC 10164-9 shall import a copy of this annex and complete it according to the instructions specified in ITU-T Rec. X.724 | ISO/IEC 10165-6.

Table F.1 – Parameter support

					Support		
Index	Parameter template label	Value of parameter identifier	Constraints and values	Status	Confirmed	Non-confirmed	Additional information
1	invalidAccessControlFilter	{ac-par 1}	–	o			

9) Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MIDS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété. Les instructions pour la construction du formulaire MIDS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 10165-6.

Annexe G

Paramètre du service CMIP pour le contrôle d'accès

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

G.1 Certificat de contrôle d'accès

Cette information peut servir à spécifier le paramètre de contrôle d'accès qui peut être utilisé avec le protocole CMIP. La spécification d'une politique de contrôle d'accès peut comporter sa propre définition de cette information.

Un certificat de contrôle d'accès (ACC), également appelé certificat d'accès privilégié (PAC), peut contenir les types d'information ci-après:

- identité du domaine de sécurité et de l'autorité correspondante;
- informations de contrôle d'accès requises par la politique de contrôle d'accès. Ces informations peuvent prendre la forme d'une ou de plusieurs des capacités de l'initiateur, du nom de celui-ci ou d'étiquettes de sécurité;
- date de début de validité de l'information de contrôle d'accès;
- date de fin de validité de l'information de contrôle d'accès;
- date de création du certificat ACC;
- informations pouvant servir aux contrôles d'intégrité.

NOTE – Un certain nombre d'organisations normalisent actuellement des certificats ACC appropriés: le Sous-Comité 27 de l'ISO/CEI, l'Association des fabricants européens de matériels informatiques (ECMA), etc. Les réalisateurs sont invités à étudier la possibilité d'utiliser les certificats ACC de ces organisations.

Annexe H

Relation avec la Recommandation UIT-T X.812 | ISO/CEI 10181-3: Cadres de sécurité dans les systèmes ouverts – Cadre de contrôle d'accès

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

H.1 Introduction

Les procédures et informations de gestion définies dans la présente Recommandation | Norme internationale sont destinées à être utilisées de concert avec les modes de contrôle d'accès décrits dans la Rec. UIT-T X.812 | ISO/CEI 10181-3. La présente annexe informative se rapporte à la terminologie, aux procédures et aux informations de gestion définies dans cette Recommandation | Norme internationale, selon les prescriptions décrites dans la Rec. UIT-T X.812 | ISO/CEI 10181-3 concernant la terminologie, les procédures et les informations de gestion applicables.

H.2 Vue d'ensemble de la terminologie applicable de la Rec. UIT-T X.812 | ISO/CEI 10181-3

Il est nécessaire de connaître les termes suivants, qui sont définis dans la Rec. UIT-T X.812 | ISO/CEI 10181-3, pour que les prescriptions qui y sont également décrites concernant la terminologie, les procédures et les informations de gestion puissent être mises en relation avec la terminologie, les procédures et les informations de gestion définies dans la présente Recommandation | Norme internationale.

- *Information ACI liée à l'initiateur:* information de contrôle d'accès liée à un initiateur, c'est-à-dire soit transmise avec une demande d'accès ou associée à l'initiateur d'une demande via des mécanismes locaux. Ces informations peuvent inclure des informations ACI d'initiateur, certaines informations ACI de cible et des informations contextuelles sélectionnées. Il s'agira par exemple d'étiquettes de sécurité associées à des initiateurs, de capacités, de certificats de contrôle d'accès et d'informations contextuelles telles que la position d'un initiateur.

- *Information ACI liée à la cible:* information de contrôle d'accès liée à une cible via des informations soit directement mémorisées dans la base d'informations de gestion de sécurité ou indiquées dans cette base comme étant des informations fournies par des mécanismes locaux. Il s'agira par exemple d'étiquettes de sécurité associées à des cibles (informations de gestion protégées), d'identités figurant sur des listes de contrôle d'accès, d'opérations dont l'exécution sur les cibles est admise ou autorisée, d'autorités de domaine de sécurité et de l'accès qui leur est autorisé, et d'informations contextuelles comme l'heure et la date.

Les éléments d'information de gestion définis dans la présente Recommandation | Norme internationale sont des informations liées à la cible qui doivent être utilisées avec les modes de contrôle d'accès définis dans la Rec. UIT-T X.812 | ISO/CEI 10181-3.

H.3 Mode de contrôle d'accès par liste de contrôle d'accès (ACL)

Comme indiqué dans la Rec. UIT-T X.812 | ISO/CEI 10181-3, dans le cas du mode ACL, le «contrôle d'accès est géré au moyen d'une liste de couples (qualificateur d'initiateur – qualificateur de demande d'accès) formant des informations ACI liées à la cible et au moyen d'une liste d'identités d'individus, de groupes ou de rôles formant des informations ACI liées à l'initiateur». En option, un qualificateur de contexte peut être ajouté pour obtenir quelques variantes d'un même mode de contrôle d'accès.

Le qualificateur d'initiateur détermine l'identité unique, le groupe ou le rôle d'un initiateur auquel est appliqué le qualificateur de demande d'accès.

Le qualificateur de demande d'accès décrit les contraintes portant sur les demandes d'accès (opérations et cibles associées) pour lesquelles l'accès doit être autorisé ou refusé selon l'identité indiquée dans le qualificateur d'initiateur associé.

Le qualificateur de contexte décrit les contraintes contextuelles à ajouter aux contraintes sur les qualificateurs de demande d'accès, pour obtenir quelques variantes d'un même mode de contrôle d'accès.

Le couplage d'une paire (qualificateur d'initiateur – qualificateur de demande) et d'un qualificateur facultatif de contexte est représenté dans la présente Recommandation | Norme internationale sous la forme d'informations implantées dans un objet géré de classe règle qui contient un ou plusieurs objets gérés de classe initiateurs par liste ACL ainsi qu'un ou plusieurs objets gérés de classe cibles. Les informations additionnelles de demande d'accès seront représentées dans l'unique objet géré de classe règles de contrôle d'accès qui est à l'état actif pour la politique de sécurité en cours d'exécution.

Les informations relatives au qualificateur d'initiateur, au qualificateur de demande d'accès et au qualificateur de contexte sont mémorisées dans différents objets gérés comme suit:

qualificateur d'initiateur:

– identificateur de nom d'initiateur ou d'alias, implanté dans un élément de l'attribut liste de contrôle d'accès d'un objet géré de classe initiateurs par liste ACL.

qualificateur de demande d'accès:

– une ou plusieurs cibles, identifiées par des attributs dans un objet géré de classe cibles;

– contraintes relatives aux opérations et aux attributs, situées dans l'objet géré opérations contenu dans l'objet cibles et spécifique du type d'opération demandé;

– permission d'accès, implantée dans l'objet géré de classe règle;

– permissions d'accès par défaut pour chaque type d'opération, pour chaque nom de domaine de sécurité, pour chaque granularité des réponses en refus, implantées dans des attributs de l'objet géré de classe règles de contrôle d'accès.

qualificateur de contexte:

– contraintes contextuelles, implantées dans l'objet géré de classe règle associé, sous la forme de contraintes sur la disponibilité dans le temps de la règle, des contextes d'état au sujet des informations contenues dans d'autres objets gérés représentant des ressources, et sur l'authentification.

Les informations de gestion définies dans la présente Recommandation | Norme internationale peuvent être utilisées dans plusieurs variantes du même mode de contrôle d'accès, ou dans des combinaisons de telles variantes, comme indiqué dans la Rec. UIT-T X.812 | ISO/CEI 10181-3. Pour plus de détails sur les variantes suivantes, prière de consulter la Rec. UIT-T X.812 | ISO/CEI 10181-3.

– Les *listes ACL sans qualificateur de demande d'accès* sont représentées par des «règles globales», définies dans la présente Recommandation | Norme internationale, qui ne contiennent aucun objet géré de classe cibles ou de classe opérations.

– Les *listes ACL avec qualificateur de contexte* sont représentées par des objets gérés de classe règles qui contiennent des informations contextuelles. La règle peut être de type «règles globales» comme décrit ci-dessus ou «règles d'item» comme défini dans la présente Recommandation | Norme internationale, contenant des objets gérés de classes initiateurs par liste ACL, cibles et, le cas échéant, opérations.

– Les *listes ACL avec cibles groupées* sont représentées par des règles d'item avec plusieurs cibles protégées, spécifiées par les objets gérés de classe cibles qu'elles contiennent.

– Les *listes ACL avec qualificateur de cible* sont représentées par des règles d'item.

– Les *listes ACL avec initiateurs groupés* sont partiellement représentées par une combinaison de règles globales et de règles d'item. Des informations locales supplémentaires doivent toujours être fournies afin de contrôler l'ordre de traitement des règles et ainsi de tenir compte de la granularité du contrôle d'accès à des sous-groupes.

– Les *listes ACL ordonnancées* sont partiellement représentées par une combinaison de règles globales et de règles d'item, avec fourniture d'informations locales supplémentaires pour contrôler l'ordre de recherche lors du traitement des règles et de leurs listes de contrôle d'accès associées.

H.4 Mode de contrôle d'accès par capacité

Comme indiqué dans la Rec. UIT-T X.812 | ISO/CEI 10181-3, dans le cas d'un mode par capacité, le «contrôle d'accès est géré en termes d'informations ACI liées à l'initiateur (qui est une capacité) qui définissent un ensemble d'opérations pouvant être effectuées sur un ensemble de cibles déterminé».

Conformément à la Rec. UIT-T X.812 | ISO/CEI 10181-3, les informations ACI liées à la cible comprennent des identificateurs d'individus, de groupe et de rôle, ainsi que, le cas échéant, une liste de noms d'autorité de domaine de sécurité avec les opérations associées.

Les informations ACI liées à une cible dans le mode de contrôle par capacité sont toutes implantées dans l'attribut capacité des objets gérés de classe initiateurs par capacité.

Deux variantes seront décrites pour le mode de contrôle d'accès par capacité:

- *capacités sans opérations spécifiques:* seul un identificateur d'individu, de groupe ou de rôle doit être inclus dans l'information ACI liée à la cible;
- *capacités avec contraintes liées à l'autorité:* la liste additionnelle des noms d'autorité de domaine de sécurité et de leurs opérations associées doit toujours être fournie dans les informations ACI liées à la cible.

H.5 Mode de contrôle d'accès par étiquettes

Comme cela est indiqué dans la Rec. UIT-T X.812 | ISO/CEI 10181-3, dans le cas d'un mode par étiquettes, on fait appel à des «étiquettes de sécurité qui peuvent être assignées à des initiateurs et à des cibles pour transférer des données entre des systèmes».

Conformément à la Rec. UIT-T X.812 | ISO/CEI 10181-3, les informations ACI liées à la cible sont formées d'une (unique) étiquette de sécurité, associée à chaque cible.

La présente Recommandation | Norme internationale propose un mécanisme permettant d'associer à une cible une étiquette de sécurité unique. Cette étiquette est implantée dans un objet géré de classe étiquette, assortie d'une cible identifiée ou d'un ensemble de cibles identifiées. L'ordre d'évaluation des objets gérés de classe étiquette est défini de telle manière qu'on ne puisse associer à une cible qu'une seule étiquette.

H.6 Mode de contrôle d'accès par contexte

Comme indiqué dans la Rec. UIT-T X.812 | ISO/CEI 10181-3, dans le cas d'un mode par contexte, le «contrôle d'accès est géré en termes d'informations ACI liées à l'initiateur ou liées à la cible, ou encore indépendamment sous la forme d'informations obtenues par la fonction ADF».

Conformément à la Rec. UIT-T X.812 | ISO/CEI 10181-3, les listes de contrôle d'accès par contexte contiennent des entrées qui possèdent deux champs:

- le champ *qualificateur de contexte*, qui est une séquence de conditions contextuelles (comme l'heure, le conduit, le point) auxquelles le qualificateur d'opération est appliqué. Chaque condition contextuelle est associée individuellement à une déclaration de type vrai ou faux;
- le champ *qualificateur d'opération*, qui indique l'opération autorisée pour le qualificateur de contexte associé.

La présente Recommandation | Norme internationale ne spécifie pas directement de contraintes contextuelles sous la forme d'une liste de contrôle d'accès par contexte. En revanche, elle implante:

- des informations relatives au qualificateur de contexte dans les objets gérés de classe règle sous la forme d'informations de planification, de contextes d'état relatifs à des valeurs d'attribut insérées dans d'autres objets gérés, et sous la forme de contraintes contextuelles d'authentification;
- des informations relatives au qualificateur d'opération dans les objets gérés de classe cibles (attribut liste d'opérations) ou dans l'objet géré opérations (attribut type d'opération), ou encore des informations impliquant une opération quelconque dans le cas de règles globales ne contenant pas d'objets gérés de classes cibles ou opérations.

Le couplage des objets gérés de classe règle avec leurs objets gérés cibles et opérations associés ou, dans le cas de règles globales, l'implication d'une opération quelconque peut être considérée comme constituant une liste de contrôle d'accès par contexte.

La seule variante, dans le mode de contrôle d'accès par contexte décrit dans la Rec. UIT-T X.812 | ISO/CEI 10181-3, nécessite une recherche ordonnée dans la liste de contrôle d'accès par contexte. Cette variante ne peut être mise en œuvre que si des informations locales supplémentaires sont spécifiées afin de contrôler la recherche des objets gérés de classes règle, cibles ou opérations.