

Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

X.519

(08/2005)

SÉRIE X: RÉSEAUX DE DONNÉES, COMMUNICATION
ENTRE SYSTÈMES OUVERTS ET SÉCURITÉ

Annuaire

**Technologies de l'information –
Interconnexion des systèmes ouverts –
L'annuaire: spécification des protocoles**

Recommandation UIT-T X.519



RECOMMANDATIONS UIT-T DE LA SÉRIE X
RÉSEAUX DE DONNÉES, COMMUNICATION ENTRE SYSTÈMES OUVERTS ET SÉCURITÉ

| | |
|--|--------------------|
| RÉSEAUX PUBLICS DE DONNÉES | |
| Services et fonctionnalités | 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 |
| Définitions 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 des couches | X.280–X.289 |
| Tests de conformité | X.290–X.299 |
| INTERFONCTIONNEMENT DES RÉSEAUX | |
| Généralités | X.300–X.349 |
| Systèmes de transmission de données par satellite | X.350–X.369 |
| Réseaux à protocole Internet | X.370–X.379 |
| SYSTÈMES DE MESSAGERIE | X.400–X.499 |
| ANNUAIRE | X.500–X.599 |
| RÉSEAUTAGE OSI ET ASPECTS SYSTÈMES | |
| Réseautage | X.600–X.629 |
| Efficacité | X.630–X.639 |
| Qualité de service | X.640–X.649 |
| Dénomination, adressage et enregistrement | X.650–X.679 |
| Notation de syntaxe abstraite numéro un (ASN.1) | X.680–X.699 |
| GESTION OSI | |
| Cadre général et architecture de la gestion-systèmes | X.700–X.709 |
| Service et protocole de communication de gestion | X.710–X.719 |
| Structure de l'information de gestion | X.720–X.729 |
| Fonctions de gestion et fonctions ODMA | X.730–X.799 |
| SÉCURITÉ | X.800–X.849 |
| APPLICATIONS OSI | |
| Engagement, concomitance et rétablissement | X.850–X.859 |
| Traitement transactionnel | X.860–X.879 |
| Opérations distantes | X.880–X.889 |
| Applications génériques de l'ASN.1 | X.890–X.899 |
| TRAITEMENT RÉPARTI OUVERT | X.900–X.999 |
| SÉCURITÉ DES TÉLÉCOMMUNICATIONS | X.1000– |

Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.

Technologies de l'information – Interconnexion des systèmes ouverts –
L'annuaire: spécification des protocoles

Résumé

La présente Recommandation | Norme internationale spécifie le protocole d'accès à l'annuaire, le protocole du système d'annuaire, le protocole de duplication miroir d'informations de l'annuaire et le protocole de gestion des liens opérationnels pour l'annuaire, pour l'accomplissement des services abstraits spécifiés dans les Recommandations UIT-T X.501 | ISO/CEI 9594-2, X.511 | ISO/CEI 9594-3, X.518 | ISO/CEI 9594-4 et X.525 | ISO/CEI 9594-9.

Source

La Recommandation UIT-T X.519 a été approuvée le 29 août 2005 par la Commission d'études 17 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8. Un texte identique est publié comme Norme Internationale ISO/CEI 9594-5.

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. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée mondiale de normalisation des télécommunications (AMNT), qui se réunit tous les quatre ans, détermine les thèmes d'étude à traiter par les Commissions d'études de l'UIT-T, lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution 1 de l'AMNT.

Dans certains secteurs des technologies 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.

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.

Le respect de cette Recommandation se fait à titre volontaire. Cependant, il se peut que la Recommandation contienne certaines dispositions obligatoires (pour assurer, par exemple, l'interopérabilité et l'applicabilité) et considère que la Recommandation est respectée lorsque toutes ces dispositions sont observées. Le futur d'obligation et les autres moyens d'expression de l'obligation comme le verbe "devoir" ainsi que leurs formes négatives servent à énoncer des prescriptions. L'utilisation de ces formes ne signifie pas qu'il est obligatoire de respecter la Recommandation.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou l'implémentation de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de l'implémentation de consulter la base de données des brevets du TSB.

© UIT 2006

Tous droits réservés. Aucune partie de cette publication ne peut être reproduite, par quelque procédé que ce soit, sans l'accord écrit préalable de l'UIT.

TABLE DES MATIÈRES

| | | <i>Page</i> |
|----|---|-------------|
| 1 | Domaine d'application | 1 |
| 2 | Références | 1 |
| | 2.1 Références normatives | 1 |
| | 2.2 Références non normatives | 2 |
| 3 | Définitions | 3 |
| | 3.1 Définitions de base relatives à l'annuaire..... | 3 |
| | 3.2 Définitions relatives aux opérations réparties..... | 3 |
| | 3.3 Définitions relatives à la spécification du protocole..... | 3 |
| 4 | Abréviations | 4 |
| 5 | Conventions | 4 |
| 6 | Spécifications communes des protocoles | 5 |
| | 6.1 Associations et opérations d'annuaire | 5 |
| | 6.2 Spécification des opérations d'annuaire..... | 5 |
| | 6.3 Aperçu général du protocole d'annuaire | 7 |
| | 6.4 Codes d'opération..... | 8 |
| | 6.5 Codes d'erreur | 8 |
| | 6.6 Syntaxes abstraites | 9 |
| 7 | Protocoles d'annuaire utilisant la pile OSI..... | 9 |
| | 7.1 Unités PDU OSI | 9 |
| | 7.2 Structure de l'unité PDU d'annuaire | 9 |
| | 7.3 Unités PDU de la couche Session | 10 |
| | 7.4 Adressage OSI..... | 10 |
| | 7.5 Procédure et séquençement | 11 |
| | 7.6 Spécifications des unités PDU d'annuaire..... | 11 |
| 8 | Mappage du protocole d'annuaire sur les services OSI | 25 |
| | 8.1 Syntaxes abstraites et syntaxes de transfert..... | 25 |
| | 8.2 Contextes d'application | 26 |
| | 8.3 Spécification de la couche Session..... | 27 |
| | 8.4 Utilisation du service de transport..... | 34 |
| 9 | Protocole IDM | 34 |
| | 9.1 Unités IDM-PDU | 34 |
| | 9.2 Exigences de séquençement | 37 |
| | 9.3 Protocoles..... | 37 |
| | 9.4 Causes de rejet..... | 38 |
| | 9.5 Causes d'abandon | 38 |
| | 9.6 Mappage sur protocole TCP/IP | 39 |
| | 9.7 Adressage | 39 |
| | 9.8 Utilisation de la connexion TLS | 40 |
| 10 | Mappage du protocole d'annuaire sur le protocole IDM..... | 40 |
| | 10.1 Protocole DAP-IP | 40 |
| | 10.2 Protocole DSP-IP | 41 |
| | 10.3 Protocole DISP-IP..... | 41 |
| | 10.4 Protocole DOP-IP | 41 |
| 11 | Coexistence entre piles protocolaires | 42 |
| | 11.1 Coexistence entre piles OSI et piles IDM..... | 42 |
| | 11.2 Coexistence en présence du protocole LDAP..... | 42 |
| | 11.3 Définition d'un format de point NSAP pour le protocole LDAP..... | 42 |
| 12 | Versions et règles d'extensibilité | 43 |
| | 12.1 DUA à DSA..... | 43 |
| | 12.2 DSA à DSA | 44 |
| | 12.3 Règles d'extensibilité des classes d'objets..... | 45 |

| | <i>Page</i> |
|---|-------------|
| 12.4 Règles d'extensibilité des types d'attribut d'utilisateur | 46 |
| 13 Conformité | 46 |
| 13.1 Conformité des DUA | 46 |
| 13.2 Conformité par les DSA | 46 |
| 13.3 Conformité du fournisseur de la duplication | 50 |
| 13.4 Conformité du consommateur de duplication | 51 |
| Annexe A – Spécifications communes des protocoles en notation ASN.1 | 53 |
| Annexe B – Protocole OSI en notation ASN.1 | 55 |
| Annexe C – Protocoles d'annuaire OSI en notation ASN.1 | 61 |
| Annexe D – Protocole IDM en notation ASN.1 | 64 |
| Annexe E – Protocoles IDM d'annuaire en notation ASN.1 | 67 |
| Annexe F – Types de liens opérationnels pour l'annuaire | 69 |
| Annexe G – Amendements et corrigenda | 70 |

Introduction

La présente Recommandation | Norme internationale a été élaborée, ainsi que les autres Recommandations | Normes internationales, pour faciliter l'interconnexion des systèmes de traitement de l'information et permettre ainsi d'assurer des services d'annuaire. L'ensemble de tous ces systèmes, avec les informations d'annuaire qu'ils contiennent, peut être considéré comme un tout intégré, appelé *annuaire*. Les informations de l'annuaire, appelées collectivement base d'informations d'annuaire (DIB, *directory information base*), sont généralement utilisées pour faciliter la communication entre, avec ou à propos d'objets tels que des entités d'application, des personnes, des terminaux et des listes de distribution.

L'annuaire joue un rôle important dans l'interconnexion des systèmes ouverts dont le but est de permettre, moyennant un minimum d'accords techniques en dehors des normes d'interconnexion proprement dites, l'interconnexion des systèmes de traitement de l'information:

- provenant de divers fabricants;
- gérés différemment;
- de niveaux de complexité différents;
- de générations différentes.

La présente Recommandation | Norme internationale spécifie les éléments des services d'application et les contextes d'application pour deux protocoles – le protocole d'accès à l'annuaire (DAP, *directory access protocol*) et le protocole du système d'annuaire (DSP, *directory system protocol*). Le DAP assure l'accès à l'annuaire pour rechercher ou modifier l'information qu'il contient. Le DSP assure le chaînage des demandes de recherche ou de modification d'information d'annuaire avec d'autres parties du système d'annuaire réparti où peut se trouver l'information.

En outre, la présente Recommandation | Norme internationale spécifie les éléments des services d'application et les contextes d'application pour le protocole de duplication miroir d'informations de l'annuaire (DISP, *directory information shadowing protocol*) et pour le protocole de gestion des liens opérationnels pour l'annuaire (DOP, *directory operational binding management protocol*). Le DISP permet la duplication miroir d'informations détenues par un DSA dans un autre DSA. Le DOP permet l'établissement, la modification et la terminaison de liens entre deux DSA pour l'administration des relations entre les DSA (telles que des relations hiérarchiques de duplication).

La présente Recommandation | Norme internationale indique les cadres de base à partir desquels des profils industriels peuvent être définis par d'autres groupes de normalisation et dans d'autres instances industrielles. L'utilisation d'un grand nombre de caractéristiques définies comme étant facultatives dans ces cadres peut être prescrite dans certains environnements au moyen de profils. Cette cinquième édition révisé techniquement et améliore, mais ne remplace pas, la quatrième édition de la présente Recommandation | Norme internationale. Les implémentations peuvent encore revendiquer la conformité à la quatrième édition mais celle-ci finira par ne plus être prise en compte (c'est-à-dire que les erreurs signalées ne seront plus corrigées). Il est recommandé que les implémentations se conforment, dès que possible, à la présente cinquième édition.

Cette cinquième édition spécifie les versions 1 et 2 des protocoles d'annuaire.

Les première et deuxième éditions ne spécifiaient que la version 1. La plupart des services et protocoles spécifiés dans la présente édition sont conçus pour fonctionner selon la version 1. Certains services et protocoles améliorés, par exemple les erreurs signées, ne fonctionneront cependant pas avant que toutes les entités d'annuaire mises en jeu dans l'exploitation aient négocié la version 2. Quelle que soit la version négociée, on a traité les différences entre les services et entre les protocoles définis dans les cinq éditions, à l'exception de ceux qui sont spécifiquement définis dans la version 2, en utilisant les règles d'extensibilité définies dans l'édition actuelle de la Rec. UIT-T X.519 | ISO/CEI 9594-5.

L'Annexe A, qui fait partie intégrante de la présente Recommandation | Norme internationale, spécifie le module ASN.1 associé aux spécifications communes des protocoles d'annuaire.

L'Annexe B, qui fait partie intégrante de la présente Recommandation | Norme internationale, spécifie le module ASN.1 pour la spécification du protocole OSI.

L'Annexe C, qui fait partie intégrante de la présente Recommandation | Norme internationale, spécifie le module ASN.1 pour les protocoles d'annuaire OSI.

L'Annexe D, qui fait partie intégrante de la présente Recommandation | Norme internationale, spécifie le module ASN.1 pour la spécification du protocole IDM.

L'Annexe E, qui fait partie intégrante de la présente Recommandation | Norme internationale, spécifie le module ASN.1 pour les protocoles d'annuaire IDM.

L'Annexe F, qui fait partie intégrante de la présente Recommandation | Norme internationale, spécifie le module ASN.1 qui contient tous les identificateurs d'objet ASN.1 assignés pour identifier les types de lien opérationnels dans cette série de Recommandations | Normes internationales.

L'Annexe G, qui fait partie intégrante de la Recommandation | Norme internationale, spécifie la liste des modifications et des signalements de défaut qui ont été incorporés pour constituer la présente édition de la présente Recommandation | Norme internationale.

**NORME INTERNATIONALE
RECOMMANDATION UIT**

**Technologies de l'information – Interconnexion des systèmes ouverts –
L'annuaire: spécification des protocoles**

1 Domaine d'application

La présente Recommandation | Norme internationale spécifie le protocole d'accès à l'annuaire, le protocole de système d'annuaire, le protocole de duplication miroir d'informations de l'annuaire et le protocole de gestion des liens opérationnels pour l'annuaire, pour l'accomplissement des services abstraits spécifiés dans la Rec. UIT-T X.511 | ISO/CEI 9594-3, la Rec. UIT-T X.518 | ISO/CEI 9594-4, la Rec. UIT-T X.525 | ISO/CEI 9594-9 et la Rec. UIT-T X.501 | ISO/CEI 9594-2.

2 Références

2.1 Références normatives

Les Recommandations et 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. Toutes Recommandations et Normes sont sujettes 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 de l'UIT-T en vigueur.

2.1.1 Recommandations | Normes internationales identiques

- Recommandation UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: le modèle de référence de base.*
- Recommandation UIT-T X.213 (2001) | ISO/CEI 8348:2002, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service réseau.*
- Recommandation UIT-T X.214 (1995) | ISO/CEI 8072:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service de transport.*
- Recommandation UIT-T X.500 (2005) | ISO/CEI 9594-1:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: aperçu général des concepts, modèles et services.*
- Recommandation UIT-T X.501 (2005) | ISO/CEI 9594-2:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: les modèles.*
- Recommandation UIT-T X.509 (2005) | ISO/CEI 9594-8:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: cadre général des certificats de clé publique et d'attribut.*
- Recommandation UIT-T X.511 (2005) | ISO/CEI 9594-3:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: définition du service abstrait.*
- Recommandation UIT-T X.518 (2005) | ISO/CEI 9594-4:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: procédures pour le fonctionnement réparti.*
- Recommandation UIT-T X.520 (2005) | ISO/CEI 9594-6:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: types d'attributs sélectionnés.*
- Recommandation UIT-T X.521 (2005) | ISO/CEI 9594-7:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: classes d'objets sélectionnées.*
- Recommandation UIT-T X.525 (2005) | ISO/CEI 9594-9:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: duplication.*

ISO/CEI 9594-5:2005 (F)

- Recommandation UIT-T X.530 (2005) | ISO/CEI 9594-10:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: utilisation de la gestion-systèmes pour l'administration de l'annuaire.*
- Recommandation UIT-T X.680 (2002) | ISO/CEI 8824-1:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- Recommandation UIT-T X.681 (2002) | ISO/CEI 8824-2:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des objets informationnels.*
- Recommandation UIT-T X.682 (2002) | ISO/CEI 8824-3:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des contraintes.*
- Recommandation UIT-T X.683 (2002) | ISO/CEI 8824-4:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: paramétrage des spécifications de la notation de syntaxe abstraite numéro un.*
- Recommandation UIT-T X.690 (2002) | ISO/CEI 8825-1:2002, *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives.*

2.1.2 Normes ISO/CEI

- ISO/CEI 10646:2003, *Technologies de l'information – Jeu universel de caractères codés sur plusieurs octets (JUC).*

2.1.3 Autres références

- Recommandation UIT-T E.164 (2005), *Plan de numérotage des télécommunications publiques internationales.*
- Recommandation UIT-T X.121 (2000), *Plan de numérotage international pour les réseaux publics de données.*
- IETF RFC 2025 (1996), *The Simple Public-Key GSS-API Mechanism (SPKM).*
- IETF RFC 793 (1981), *Transmission Control Protocol – DARPA Internet Program – Protocol Specification.*
- IETF RFC 1277 (1991), *Encoding Network Addresses to Support Operation over Non-OSI Lower Layers.*
- IETF RFC 1738 (1994), *Uniform Resource Locators (URL).*
- IETF RFC 2246 (1999), *The TLS Protocol Version 1.0.*
- IETF RFC 2251 (1997), *Lightweight Directory Access Protocol (v3).*
- IETF RFC 3546 (2003), *Transport Layer Security (TLS) Extensions.*

2.2 Références non normatives

- Recommandation UIT-T X.217 (1995) | ISO/CEI 8649:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition de service applicable à l'élément de service de contrôle d'association.*
- Recommandation UIT-T X.225 (1995) | ISO/CEI 8327-1:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Protocole de session en mode connexion: spécification du protocole.*
- Recommandation UIT-T X.226 (1994) | ISO/CEI 8823-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Protocole de présentation en mode connexion: spécification du protocole.*
- Recommandation UIT-T X.227 (1995) | ISO/CEI 8650-1:1996, *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.*
- Recommandation UIT-T X.881 (1994) | ISO/CEI 13712-2:1995, *Technologies de l'information – Opérations distantes: réalisations OSI – Définition du service de l'élément de service d'opérations distantes.*

3 Définitions

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

3.1 Définitions de base relatives à l'annuaire

Les termes suivants sont définis dans la Rec. UIT-T X.501 | ISO/CEI 9594-2:

- a) *l'annuaire*;
- b) *utilisateur (de l'annuaire)*;
- c) *agent de système d'annuaire (DSA)*;
- d) *agent d'utilisateur d'annuaire (DUA)*.

3.2 Définitions relatives aux opérations réparties

Les termes suivants sont définis dans la Rec. UIT-T X.518 | ISO/CEI 9594-4:

- a) *chaînage*;
- b) *renvoi de référence*.

3.3 Définitions relatives à la spécification du protocole

Les termes suivants sont définis dans la présente Recommandation | Norme internationale.

NOTE – Les définitions données dans ce paragraphe sont générales et visent aussi bien les cas OSI que les cas TCP/IP, sauf en ce qui concerne les exceptions indiquées.

3.3.1 syntaxe abstraite: spécification de types de données et/ou de valeurs de données à l'aide de règles de notation qui sont indépendantes de la technique de codage utilisée pour les représenter.

3.3.2 association d'applications: relation coopérative entre deux entités d'application établies par l'opération de rattachement.

3.3.3 contexte d'application: (définition OSI uniquement) ensemble de règles partagées par deux entités d'application pour prendre en charge une association d'applications.

3.3.4 nom de contexte d'application: identificateur d'objet ASN.1 qui identifie (désigne) un contexte d'application.

3.3.5 couche Application: couche supérieure du modèle OSI à sept couches représentant la sémantique de la communication.

3.3.6 entité d'application: représentation du comportement externe d'un processus d'application sous forme de capacités de communication.

3.3.7 titre d'entité d'application: nom distinctif d'une entité d'application d'annuaire et, en particulier, d'une entité d'application représentant un processus d'application d'annuaire.

3.3.8 processus d'application: processus à l'intérieur d'un système qui traite des renseignements dans un but particulier, notamment les opérations d'annuaire.

3.3.9 opération de rattachement: type d'opération utilisé pour établir une association d'applications.

3.3.10 opération d'annuaire: type d'opération pour un échange d'informations d'annuaire.

3.3.11 unité de données protocolaire d'annuaire: unité de données d'un protocole d'annuaire comprenant des informations de commande et également, en général, des données d'application spécifiées par les opérations d'annuaire.

NOTE 1 – Dans l'environnement OSI, une unité PDU d'annuaire comprend tous les éléments de protocole de la couche Présentation OSI et, le cas échéant, des éléments de protocole ACSE, outre les éléments de protocole propres à l'annuaire.

NOTE 2 – L'unité de données protocolaire d'Application (APDU, *application-protocol-data-unit*) est une unité de données définie par un protocole d'application OSI. Ce terme n'est pas utilisé dans l'édition 5 et les éditions ultérieures des présentes Spécifications d'annuaire. Cependant, l'abréviation peut figurer dans quelques éléments ASN.1.

3.3.12 initiateur: processus d'application qui initie une association d'applications en envoyant une demande de rattachement.

3.3.13 opération: échange entre deux processus d'application en vue de la réalisation d'une tâche particulière. Elle comprend une demande envoyée par un processus d'application à un autre processus ainsi que le renvoi de zéro ou de plusieurs réponses (résultat et/ou erreurs). Une opération implique qu'un certain processus doit être effectué par le processus d'application recevant la demande.

3.3.14 unité de données protocolaire: elle comprend les éléments de protocole de présentation ou les éléments de protocole ACSE d'une unité de données protocolaire d'annuaire.

3.3.15 couche Présentation: sixième couche du modèle de référence OSI.

3.3.16 erreur protocolaire: réception d'unité de données protocolaire non reconnue ou inattendue, ou d'une unité de données protocolaire ayant un paramètre inattendu ou non valable.

3.3.17 preneur: processus d'application qui reçoit une demande de rattachement et qui accepte ou refuse l'association d'applications.

3.3.18 couche Session: cinquième couche du modèle de référence OSI.

3.3.19 unité de données protocolaire de Session: (définition OSI uniquement) unité de données de la couche Session OSI comprenant des informations de commande et également, en général, une unité de données protocolaire d'annuaire.

4 Abréviations

Pour les besoins de la présente Recommandation | Norme internationale, les abréviations suivantes sont utilisées:

| | |
|--------|--|
| AC | Contexte d'application (<i>application context</i>) |
| ACSE | Élément de service de contrôle d'association (<i>association control service element</i>) |
| AE | Entité d'application (<i>application-entity</i>) |
| APDU | Unité de données protocolaire d'Application (<i>application-protocol-data-unit</i>) |
| DAP | Protocole d'accès à l'annuaire (<i>directory access protocol</i>) |
| DISP | Protocole de duplication miroir d'informations de l'annuaire (<i>directory information shadowing protocol</i>) |
| DOP | Protocole de gestion des liens opérationnels d'annuaire (<i>directory operational binding management protocol</i>) |
| DSA | Agent de système d'annuaire (<i>directory system agent</i>) |
| DSP | Protocole du système d'annuaire (<i>directory system protocol</i>) |
| DUA | Agent d'utilisateur d'annuaire (<i>directory user agent</i>) |
| IDM | Mappage Internet direct (<i>Internet directly mapped</i>) |
| LDAP | Protocole rapide d'accès à l'annuaire (<i>lightweight directory access protocol</i>) |
| PDU | Unité de données protocolaire (<i>protocol-data-unit</i>) |
| PPDU | Unité de données protocolaire de Présentation (<i>presentation-protocol-data-unit</i>) |
| SPDU | Unité de données protocolaire de Session (<i>session-protocol-data-unit</i>) |
| TCP/IP | Protocole de commande de transmission/protocole Internet (<i>transmission control protocol/Internet protocol</i>) |
| TSDU | Unité de données de service de transport (<i>transport-service-data-unit</i>) |

5 Conventions

A quelques exceptions mineures près, la présente Spécification d'annuaire a été élaborée conformément aux *Règles de présentation des textes communs UIT-T | ISO/CEI*, novembre 2001.

Le terme "Spécification d'annuaire" (comme dans "la présente Spécification d'annuaire") s'entend selon l'acceptation de la présente Recommandation | Norme internationale. Le terme "Spécifications d'annuaire" s'entend selon l'acceptation de toutes les Recommandations de la série X.500 et de toutes les parties de l'ISO/CEI 9594.

La présente Spécification d'annuaire utilise le terme "*systèmes de la première édition*" pour désigner les systèmes conformes à la première édition des Spécifications d'annuaire, c'est-à-dire à l'édition 1988 des Recommandations CCITT de la série X.500 et de l'ISO/CEI 9594:1990. La présente Spécification d'annuaire utilise le terme "*systèmes de la deuxième édition*" pour désigner les systèmes conformes à la deuxième édition (1993) des Spécifications d'annuaire, c'est-à-dire à l'édition 1993 des Recommandations UIT-T de la série X.500 et de l'ISO/CEI 9594:1995. La présente Spécification d'annuaire utilise le terme "*systèmes de la troisième édition*" pour désigner les systèmes conformes à la troisième édition des Spécifications d'annuaire, c'est-à-dire à l'édition 1997 des Recommandations UIT-T de la série X.500 et de l'ISO/CEI 9594:1998. La présente Spécification d'annuaire utilise le terme "*systèmes de la quatrième*

édition" pour désigner les systèmes conformes à la quatrième édition des Spécifications d'annuaire, c'est-à-dire aux éditions 2001 des Recommandations UIT-T X.500, X.501, X.511, X.518, X.519, X.520, X.521, X.525 et X.530 et à l'édition 2000 de la Rec. UIT-T X.509 et aux parties 1 à 10 de l'édition ISO-CEI 9594:2001.

La présente Spécification d'annuaire utilise le terme "*systèmes de la cinquième édition*" pour désigner les systèmes conformes à la cinquième édition des Spécifications d'annuaire, c'est-à-dire aux éditions 2005 des Recommandations UIT-T X.500, X.501, X.509, X.511, X.518, X.519, X.520, X.521, X.525 et X.530 et aux parties 1 à 10 de l'édition ISO-CEI 9594:2005.

La présente Spécification d'annuaire présente la notation ASN.1 en caractères gras de la police Helvetica. Lorsque des types et des valeurs ASN.1 sont cités dans le texte normal, ils en sont différenciés par leur présentation en caractères gras Helvetica. Les noms des procédures, normalement cités lors de la spécification des sémantèmes de traitement, sont différenciés du texte normal par une présentation en caractères gras de la police Times. Les autorisations de commande d'accès sont présentées en caractères italiques de la police Times.

Si, dans une liste, les points sont numérotés (au lieu d'utiliser des tirets ou des lettres), ils sont considérés comme des étapes d'une procédure.

6 Spécifications communes des protocoles

6.1 Associations et opérations d'annuaire

Les protocoles relatifs aux présentes Spécifications d'annuaire sont décrits en tant qu'ensemble d'*opérations*. Une opération est définie en termes de demande envoyée par un système à un autre système dans l'attente que ce dernier traite la demande et, le cas échéant, renvoie une ou plusieurs réponses qui constituent le résultat. Une opération peut être une *opération de rattachement* ou une opération invoquée pour accéder à des informations d'annuaire (une *opération d'annuaire*).

Si des conditions d'exception sont rencontrées, une ou plusieurs erreurs peuvent être renvoyées au lieu ou en plus des résultats éventuels.

NOTE 1 – Les opérations définies renverront un ou plusieurs résultats ou encore une seule erreur.

Les protocoles d'annuaire définis par les présentes Spécifications d'annuaire peuvent utiliser une pile de protocoles OSI, une pile de protocoles TCP/IP, ou les deux. La spécification indiquée par le présent paragraphe est indépendante d'une pile de protocoles en particulier. La spécification OSI pertinente est indiquée aux § 7 et 8, la spécification TCP/IP étant indiquée aux § 9 et 10.

Un processus d'un système qui traite des opérations d'annuaire est appelé *processus d'application*. Une *entité d'application* est le reflet du comportement externe d'un processus d'application.

Pour que des opérations d'annuaire puissent être invoquées entre deux processus d'applications d'annuaire, une *association d'applications* doit être établie entre les entités d'application correspondantes. Une association d'applications est une relation coopérative entre deux entités d'application découlant de l'échange d'informations de commande dans la demande et le résultat d'une opération de rattachement et de l'utilisation d'un service sous-jacent commun.

NOTE 2 – Il s'agit de la définition modifiée d'une association d'applications indiquée dans la Rec. UIT-T X.217 | ISO/CEI 8649, définition qui est censée viser à la fois l'utilisation d'une pile de protocoles OSI sous-jacente et l'utilisation d'une pile TCP/IP sous-jacente.

Une association d'applications prend fin au moyen d'un échange d'informations de détachement. Le détachement d'une association d'applications n'est pas défini en tant qu'opération.

6.2 Spécification des opérations d'annuaire

Les présentes Spécifications d'annuaire définissent plusieurs types d'opérations. Un type d'opération est spécifié par la classe d'objets informationnels ASN.1 **OPERATION**. Les erreurs pouvant être associées à un type d'opération sont définies par la classe d'objets informationnels ASN.1 **ERRORS**.

```
OPERATION ::= CLASS {
    &ArgumentType,
    &ResultType OPTIONAL,
    &Errors ERROR OPTIONAL,
    &operationCode Code UNIQUE OPTIONAL }
WITH SYNTAX {
    ARGUMENT &ArgumentType
    [RESULT &ResultType]
    [ERRORS &Errors]
```

```
[CODE      &operationCode] }
```

```
ERROR ::= CLASS {
    &ParameterType,
    &errorCode Code UNIQUE OPTIONAL }
WITH SYNTAX {
    PARAMETER &ParameterType
    [CODE      &errorCode] }
```

```
Code ::= CHOICE {
    local      INTEGER,
    global    OBJECT IDENTIFIER }
```

La classe d'objets informationnels **OPERATION** est un moyen commode d'exprimer la syntaxe des demandes d'annuaire, des résultats et des erreurs pour un type d'opération particulier.

Cette classe d'objets informationnels ASN.1 comprend les champs ci-après:

- le champ **&ArgumentType** spécifie un type de données ouvert pour la partie demande d'une opération;
- le champ **&ResultType** définit un type de données ouvert pour une ou plusieurs réponses constituant le résultat de la demande. S'il est absent, aucun résultat n'est associé à l'opération;
- le champ **&Errors** spécifie une ou plusieurs erreurs susceptibles de se produire par suite du traitement de la demande. S'il est absent, aucune erreur n'est associée à l'opération;
- le champ **&operationCode** indique le type d'opération d'annuaire à effectuer. Il est absent pour l'opération de rattachement. Voir le § 6.4 pour les codes d'opération actuellement définis.

Les opérations d'annuaire peuvent en principe être effectuées dans deux modes différents:

- si une opération d'annuaire doit être achevée avant qu'une nouvelle opération d'annuaire puisse être invoquée, le mode opératoire est *synchrone*;
- si plusieurs opérations peuvent se dérouler en même temps, le mode opératoire est *asynchrone*.

Si toutes les opérations d'annuaire définies pour un type particulier d'association d'applications

- comportent à la fois une demande et un ou plusieurs résultats et/ou erreurs;
- ne peuvent être invoquées que par un système désigné;

elles peuvent être exécutées en mode synchrone ou asynchrone. Dans le cas contraire, le mode opératoire est toujours asynchrone.

La classe d'objets informationnels **OPERATION** n'implique pas en soi un séquençement. Une demande d'annuaire peut ne pas se solder par un résultat et/ou une erreur, ou peut se solder par plusieurs résultats et/ou erreurs. Cela étant, elle rattache bel et bien une demande à des réponses éventuelles (résultats et erreurs) en contenant le même code d'opération et le même identificateur d'invocation (voir ci-après). La spécification d'un type d'opération particulier peut cependant imposer des restrictions en matière de séquençement.

Une erreur indique l'échec d'une opération. Elle est représentée par la classe d'objets informationnels ASN.1 **ERROR**. Les différents champs sont décrits ci-après:

- le champ **&ParameterType** définit le type de données du paramètre de l'erreur indiquant la nature de l'erreur;
- le champ **&errorCode** spécifie le code qui identifie l'erreur (voir le § 6.5 pour la définition des codes d'erreur).

Bien que cela ne soit pas indiqué par les classes d'objets informationnels **OPERATION** ou **ERRORS**, chaque invocation d'une opération d'annuaire reçoit un identificateur **Invokeld**, qui est acheminé dans le protocole. Il est donc possible d'indiquer à quelle opération d'annuaire une demande, un résultat ou une erreur appartient. Le paramètre **Invokeld** est ainsi défini:

```
Invokeld ::= CHOICE {
    present INTEGER,
    absent  NULL }
```

Si un type d'opération ne spécifie pas un code **&operationCode**, les opérations de ce type ne peuvent pas se voir affecter un identificateur **Invokeld**.

6.3 Aperçu général du protocole d'annuaire

6.3.1 Utilisation des services sous-jacents

Lorsque deux processus d'application provenant de systèmes ouverts différents interagissent, l'association d'applications est effectuée sous la forme d'un protocole de couche Application à l'aide d'un service sous-jacent OSI ou TCP/IP.

On trouvera des précisions sur l'utilisation du service OSI à l'article 8 et des précisions sur l'utilisation du service TCP/IP au § 10.

6.3.2 Protocole d'accès à l'annuaire (DAP)

Pour qu'une interaction soit possible entre un agent DUA et un agent DSA provenant de systèmes ouverts différents, une opération de rattachement doit être invoquée entre eux afin d'établir une association d'applications prenant en charge un protocole d'annuaire appelé protocole d'accès à l'annuaire (DAP, *directory access protocol*).

L'opération de rattachement (**directoryBind**) permettant d'établir une association d'applications DAP est définie au § 8 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.

La présente édition et toutes les éditions précédentes des présentes Spécifications d'annuaire permettent seulement à un agent DUA d'invoquer une opération de rattachement et d'initier ultérieurement des opérations d'annuaire. Si la pile OSI sous-jacente est utilisée, les opérations d'annuaire peuvent être invoquées en mode synchrone ou asynchrone. En cas d'utilisation de la pile TCP/IP sous-jacente, les opérations d'annuaire sont toujours invoquées en mode asynchrone.

Toutes les opérations d'annuaire exigent le renvoi d'une réponse ou d'une erreur.

6.3.3 Protocole du système d'annuaire (DSP)

Pour que deux agents DSA de deux systèmes ouverts différents puissent interagir, une opération de rattachement doit être invoquée entre eux pour établir une association d'applications qui prend en charge un protocole d'annuaire appelé protocole du système d'annuaire (DSP, *directory system protocol*).

L'opération de rattachement (**dSABind**) servant à établir une association d'applications DSP est définie au § 11 de la Rec. UIT-T X.518 | ISO/CEI 9594-4.

L'un ou l'autre des agents DSA peut invoquer une opération de rattachement. Tant le DSA initiateur que le DSA preneur peuvent invoquer des opérations d'annuaire ultérieures. Les opérations d'annuaire sont toujours invoquées en mode asynchrone dans le protocole DSP.

Toutes les opérations d'annuaire exigent le renvoi d'une réponse ou d'une erreur.

6.3.4 Protocole de duplication miroir d'informations de l'annuaire (DISP)

Pour qu'une interaction entre deux agents DSA provenant de systèmes ouverts différents soit possible en vue d'un échange d'informations miroirs, une opération de rattachement doit être invoquée entre eux pour établir une association d'applications prenant en charge un protocole d'annuaire appelé protocole de duplication miroir d'informations de l'annuaire (DISP, *directory information shadowing protocol*).

L'opération de rattachement (**dSAShadowBind**) servant à établir une association d'applications DISP est définie au § 7.4.1 de la Rec. UIT-T X.525 | ISO/CEI 9594-9.

En cas d'utilisation de la pile sous-jacente OSI, le mode opératoire est synchrone ou asynchrone selon le contexte d'application choisi pour l'opération de rattachement. Si la pile sous-jacente TCP/IP est utilisée, les opérations d'annuaire sont toujours invoquées en mode asynchrone.

Toutes les opérations d'annuaire exigent le renvoi d'une réponse ou d'une erreur.

6.3.5 Protocole de gestion des liens opérationnels pour l'annuaire (DOP)

Pour que deux agents DSA relevant de deux systèmes ouverts différents puissent interagir aux fins du maintien des liens opérationnels, une opération de rattachement doit être invoquée pour établir une association d'application prenant en charge un protocole d'annuaire appelé protocole de gestion des liens opérationnels pour l'annuaire (DOP, *directory operational binding management protocol*).

Le DSA qui peut jouer le rôle d'initiateur de l'opération de rattachement dépend des rôles DSA assignés pour les liens opérationnels à gérer au moyen des opérations d'annuaire dans l'association d'applications. Seul l'initiateur peut invoquer des opérations d'annuaire. Plusieurs types de liens opérationnels ne peuvent être gérés dans le cadre de cette association d'applications que si les rôles DSA associés aux différents types sont compatibles (par exemple, si un agent DSA joue le rôle A pour chaque type de lien).

Toutes les opérations d'annuaire exigent le renvoi d'une réponse ou d'une erreur.

6.4 Codes d'opération

6.4.1 Codes d'opération pour les protocoles DAP et DSP

Les codes d'opération ci-après sont utilisés dans les protocoles DAP et DSP:

| | | |
|------------------------------|-------------|----------------------|
| id-opcode-read | Code | ::= local : 1 |
| id-opcode-compare | Code | ::= local : 2 |
| id-opcode-abandon | Code | ::= local : 3 |
| id-opcode-list | Code | ::= local : 4 |
| id-opcode-search | Code | ::= local : 5 |
| id-opcode-addEntry | Code | ::= local : 6 |
| id-opcode-removeEntry | Code | ::= local : 7 |
| id-opcode-modifyEntry | Code | ::= local : 8 |
| id-opcode-modifyDN | Code | ::= local : 9 |

L'utilisation de ces codes d'opération est spécifiée dans la Rec. UIT-T X.511 | ISO/CEI 9594-3.

6.4.2 Codes d'opération pour le protocole DISP

Les codes d'opération ci-après sont utilisés dans le protocole DISP.

| | | |
|---|-------------|----------------------|
| id-opcode-requestShadowUpdate | Code | ::= local : 1 |
| id-opcode-updateShadow | Code | ::= local : 2 |
| id-opcode-coordinateShadowUpdate | Code | ::= local : 3 |

L'utilisation de ces codes d'opération est spécifiée dans la Rec. X.525 | ISO/CEI 9594-9.

6.4.3 Codes d'opération pour le protocole DOP

Les codes d'opération ci-après sont utilisés dans le protocole DOP.

| | | |
|--|-------------|------------------------|
| id-op-establishOperationalBinding | Code | ::= local : 100 |
| id-op-modifyOperationalBinding | Code | ::= local : 102 |
| id-op-terminateOperationalBinding | Code | ::= local : 101 |

L'utilisation de ces codes d'opération est spécifiée dans la Rec. UIT-T X.501 | ISO/CEI 9594-2.

6.5 Codes d'erreur

6.5.1 Codes d'erreur pour les protocoles DAP et DSP

Les codes d'erreur suivants sont utilisés dans les protocoles DAP et DSP. Le code **id-errcode-referral** est uniquement utilisé dans le protocole DAP. Le code **id-opcode-dsaReferral** est uniquement utilisé dans le protocole DSP:

| | | |
|----------------------------------|-------------|----------------------|
| id-errcode-attributeError | Code | ::= local : 1 |
| id-errcode-nameError | Code | ::= local : 2 |
| id-errcode-serviceError | Code | ::= local : 3 |
| id-errcode-referral | Code | ::= local : 4 |
| id-errcode-abandoned | Code | ::= local : 5 |
| id-errcode-securityError | Code | ::= local : 6 |
| id-errcode-abandonFailed | Code | ::= local : 7 |
| id-errcode-updateError | Code | ::= local : 8 |
| id-errcode-dsaReferral | Code | ::= local : 9 |

6.5.2 Codes d'erreur pour le protocole DISP

Le code d'erreur suivant est utilisé dans le protocole DISP:

| | | |
|-------------------------------|-------------|----------------------|
| id-errcode-shadowError | Code | ::= local : 1 |
|-------------------------------|-------------|----------------------|

6.5.3 Codes d'erreur pour le protocole DOP

Le code d'erreur suivant est utilisé dans le protocole DOP:

| | | |
|---------------------------------------|-------------|------------------------|
| id-err-operationalBindingError | Code | ::= local : 100 |
|---------------------------------------|-------------|------------------------|

6.6 Syntaxes abstraites

Une spécification de protocole comprend une spécification des types de données qui peuvent être transférés dans le cadre des échanges protocolaires. Les types de données sont définis à l'aide d'une notation abstraite comme la notation ASN.1 et constituent la syntaxe abstraite pour le protocole. Les syntaxes abstraites sont très similaires pour la communication OSI et la communication TCP/IP, malgré quelques différences. Quatre syntaxes abstraites sont définies pour chacun de ces types de communication correspondant aux quatre protocoles d'annuaire différents. C'est seulement pour la communication OSI que des identificateurs d'objet sont affectés aux syntaxes abstraites. Lors de l'établissement d'une association d'applications OSI, l'identificateur d'objet pertinent pour la syntaxe abstraite est signalé dans l'opération de rattachement (voir le § 7.6.1).

7 Protocoles d'annuaire utilisant la pile OSI

Le présent paragraphe définit les protocoles d'annuaire et leur mappage avec le protocole de session OSI. Il intègre les éléments pertinents du protocole de présentation OSI définis par la Rec. UIT-T X.226 | ISO/CEI 8823-1 et l'élément de service de contrôle d'association (ACSE) défini par la Rec. UIT-T X.227 | ISO/CEI 8650-1. Ces éléments ont été incorporés de manière à assurer un codage compatible avec les systèmes antérieurs à l'édition 5.

La partie pertinente du protocole de session OSI est définie au § 8.3.

7.1 Unités PDU OSI

Les messages des protocoles de type OSI sont acheminés dans une association d'applications OSI en tant qu'unités de données protocolaires d'annuaire représentées par le type de données **OSI-PDU** comme suit:

```
OSI-PDU {APPLICATION-CONTEXT:protocol} ::= TYPE-IDENTIFIER.&Type (
    OsiBind { {protocol} } |
    OsiBindResult { {protocol} } |
    OsiBindError { {protocol} } |
    OsiOperation { {protocol.&Operations} } |
    PresentationAbort )
```

7.2 Structure de l'unité PDU d'annuaire

Une unité PDU d'annuaire de l'environnement OSI comprend des éléments de protocole de la couche Présentation OSI tels que les définit la Rec. UIT-T X.226 | ISO/CEI 8823-1, des éléments de protocole ACSE tels que les définit la Rec. UIT-T X.227 | ISO/CEI 8650-1, le cas échéant, et les éléments de protocole d'annuaire propres au protocole en question.

Outre les éléments de protocole propres à l'annuaire, **OsiBind**, **OsiBindResult** et **OsiBindError** ont des éléments de protocole de présentation et des éléments de protocole ACSE, tandis que **OsiOperation** a uniquement des éléments de protocole de présentation en plus des éléments de protocole d'annuaire spécifiques. L'abandon **PresentationAbort** a uniquement des éléments de protocole de présentation.

Les éléments de protocole de la couche Présentation inclus dans une unité PDU d'annuaire spécifique comprennent une unité PPDU.

NOTE 1 – Le terme PPDU (unité de données protocolaire de Présentation, *presentation-protocol-data-unit*) est introduit ici car il y est fait référence lors de la description des erreurs de protocole de Présentation et par le type de données **Abort-reason**. Autrement, ce terme n'est pas pertinent pour les présentes Spécifications d'annuaire.

Les éléments de protocole ACSE inclus dans une unité PDU d'annuaire spécifique comprennent une unité PDU ACSE.

NOTE 2 – La Rec. UIT-T X.227 | ISO/CEI 8650-1 emploie le terme APDU (unité de données protocolaire d'Application, *application-protocol-data-unit*) pour une unité PDU ACSE. Comme les éléments de protocole d'annuaire d'une unité PDU d'annuaire spécifique comprennent aussi en principe une unité APDU, les termes PDU ACSE sont utilisés ici pour éviter une confusion.

Les unités PPDU ci-après sont utilisées par les présentes Spécifications d'annuaire:

- a) unité PPDU CP, qui correspond au type de données **CP-type** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Elle fait partie du type de données **OsiBind**;
- b) unité PPDU CPA, qui correspond au type de données **CPA-PPDU** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Elle fait partie du type de données **OsiBindResult**;
- c) unité PPDU CPR, qui correspond au type de données **CPR-PPDU** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Elle fait partie des types de données **OsiBindError**;

- d) unité PDU TD, qui correspond au type de données **User-data** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Elle fait partie des types de données **OsiOperation**;
- e) unité PDU ARU, qui correspond au type de données **ARU-PPDU** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Elle fait partie du type de données **ARU-PPDU** tel que le définissent les présentes Spécifications d'annuaire;
- f) unité PDU ARP, qui correspond au type de données **ARP-PPDU** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Elle constitue le type de données **ARP-PPDU** tel que le définissent les présentes Spécifications d'annuaire.

Aucune unité PDU n'est définie pour la libération d'une association d'application (**OsiUnbind** et **OsiUnbindResult**). Toutefois, le type de données **User-data** défini par la Rec. UIT-T X.226 | ISO/CEI 8823-1 sert à acheminer **OsiUnbind** et **OsiUnbindResult**.

Les unités PDU ACSE ci-après sont utilisées par la présente Spécification d'annuaire:

- a) **AARQ-apdu** fait partie du type de données **OsiBind**;
- b) **AARE-apdu** fait partie des types de données **OsiBindResult** et **OsiBindError**;
- c) **RLRQ-apdu** fait partie du type de données **OsiUnbind**;
- d) **RLRE-apdu** fait partie du type de données **OsiUnbindresult**;
- e) **ABRT-apdu** fait partie du type de données **ARU-PPDU**.

7.3 Unités PDU de la couche Session

Outre les unités PDU d'annuaire, la présente Spécification d'annuaire définit aussi des unités de données protocolaires de Session (SPDU, *session-protocol-data-unit*). Toutes les unités PDU d'annuaire sont acheminées dans une unité SPDU.

Les unités SPDU ci-après sont utilisées par la présente Spécification d'annuaire:

- a) unité SPDU CONNECT, utilisée pour acheminer **OsiBind**;
- b) unité SPDU ACCEPT, utilisée pour acheminer **OsiBindResult**;
NOTE – L'unité PDU ACSE AARE (telle qu'elle est représentée par **AARE-apdu** et **AAREerr-apdu** est, conformément au § 8.1.3 de la Rec. UIT-T X.227 | ISO/CEI 8650-1, mappée avec la réponse/confirmation P-CONNECT, "result" étant défini par la valeur "user rejection". Conformément au § 6.2.5.6 de la Rec. UIT-T X.226 | ISO/CEI 8823-1, une unité PDU CPR sera utilisée dans la couche Présentation. De même, conformément au § 7.1.3 de la Rec. UIT-T X.226 | ISO/CEI 8823-1, l'unité PDU CPR est acheminée dans les primitives de session de réponse et de confirmation S-CONNECT.
- c) unité SPDU REFUSE, utilisée pour acheminer **OsiBindError** et pour rejeter une association d'application à cause des conditions de la couche Session;
- d) unité SPDU FINISH, utilisée pour acheminer **OsiUnbind** afin de lancer la terminaison d'une association d'application;
- e) unité SPDU DISCONNECT, utilisée pour acheminer **OsiUnbindResult** afin d'achever la terminaison d'une association d'application;
- f) unité SPDU ABORT, utilisée pour acheminer **ARU-PPDU** et **ARP-PPDU** et utilisée toute seule en cas d'abandon à cause d'un problème de la couche Session;
- g) unité SPDU ABORT ACCEPT, qui ne transporte aucune information de couche supérieure, mais qui indique qu'un message d'abandon a été reçu par le système homologue;
- h) unité SPDU DATA TRANSFER, utilisée pour acheminer **OsiOperation**.

On trouvera au § 8.3 des précisions sur les unités SPDU.

7.4 Adressage OSI

Le système OSI définit des adresses pour la couche Réseau, jusqu'à et y compris la couche Présentation. L'adresse en haut de la couche Réseau est appelée adresse de point d'accès de service de réseau (NSAP, *network-service-access-point*). La structure d'une adresse NSAP est définie dans la Rec. UIT-T X.213 | ISO/CEI 8348. Une adresse de transport en haut de la couche Transport est définie en tant qu'adresse NSAP avec en plus un sélecteur de transport optionnel. Une adresse de session en haut de la couche Session est définie en tant qu'adresse de transport avec en plus un sélecteur de session optionnel. Une adresse de présentation est définie en tant qu'adresse de session avec en plus un sélecteur de présentation optionnel. La présente Spécification d'annuaire fait uniquement référence au sélecteur de session et au sélecteur de présentation.

7.5 Procédure et séquençement

Une association d'applications entre deux processus d'application est initiée par l'un des processus d'application émettant une demande **OsiBind** telle que la définit le § 7.6.1. Le processus d'application initiateur attendra ensuite un résultat **OsiBindResult** pour confirmer l'établissement de l'argument avant d'envoyer une unité PDU d'annuaire à cette association d'applications.

Indépendamment de toute règle de séquençement, le processus d'application initiateur peut à tout moment émettre une unité **ARU-PPDU** ou **ARP-PPDU** (voir ci-après le § 7.6.7) après avoir émis une demande **OsiBind**. De même, l'application preneuse peut à tout moment émettre une unité **ARU-PPDU** ou **ARP-PPDU** après avoir reçu une demande **OsiBind**.

Si un résultat **OsiBindResult** est reçu, le processus d'application initiateur peut envoyer des unités PDU **OsiOperation** contenant **OsiReq**, **OsiRes**, **OsiErr** et **OsiRej** comme le prescrit le protocole en question.

Une association d'applications n'est pas établie si un élément **OsiBindError** (voir le § 7.6.3) est reçu en réponse à la demande **OsiBind**, ou si l'association d'applications est refusée au niveau de la session (voir le § 8.3.5).

Deux processus d'application peuvent presque simultanément s'envoyer mutuellement un résultat **OsiBind**. Cela sera considéré comme deux tentatives indépendantes d'établissement d'une association d'applications.

Des erreurs de protocole peuvent se produire dans le protocole de session, dans les éléments de protocole de présentation, dans les éléments de protocole ACSE et dans les éléments de protocole propres à l'annuaire.

Une erreur de protocole peut se produire:

- a) à la réception d'une unité PDU non reconnue ou inattendue;
- b) lorsqu'un ou plusieurs paramètres d'une unité PDU reçue ne sont pas valables ou inattendus.

NOTE 1 – Selon les règles d'extensibilité énoncées au § 12, les paramètres inconnus seront écartés. Le paragraphe 8.5 de la Rec. UIT-T X.226 | ISO/CEI 8823-1 et le § 7.4 de la Rec. UIT-T X.227 | ISO/CEI 8650-1 définissent des règles analogues.

NOTE 2 – Les paragraphes 6.4.4.2 et 6.4.4.3 de la Rec. UIT-T X.226 | ISO/CEI 8823-1 établissent une distinction entre une erreur de protocole et une unité PDU non valable. Comme les deux cas provoquent le même type d'abandon, la présente Spécification d'annuaire ne fait pas une telle distinction, ni le § 7.3.3.4 de la Rec. UIT-T X.227 | ISO/CEI 8650-1.

Dans les deux cas, l'association d'applications considérée ou une association d'applications en cours d'établissement/de terminaison sera interrompue.

Si le problème est détecté dans le protocole de session, une unité SPDU ABORT sera émise (voir le § 8.3.8) sans données d'utilisateur.

Si le problème est détecté dans le protocole de présentation, une unité **ARP-PPDU** (voir le § 7.6.7.2) sera émise.

Si le problème est détecté dans le protocole ACSE, une unité **ARU-PPDU** ayant le paramètre **abort-source** mis à la valeur **acse-service-provider** (voir le § 7.6.7.1) sera émise.

Si le problème est détecté dans le protocole d'annuaire, une unité **ARU-PPDU** ayant le paramètre **abort-source** mis à la valeur **acse-service-user** sera émise.

7.6 Spécifications des unités PDU d'annuaire

7.6.1 Demande de rattachement OSI

```
OsiBind {APPLICATION-CONTEXT:Protocols} ::= SET {
  mode-selector                [0] IMPLICIT SET {mode-value [0] IMPLICIT INTEGER (1) },
  normal-mode-parameters      [2] IMPLICIT SEQUENCE {
    protocol-version           [0] IMPLICIT BIT STRING {version-1(0)}
                                DEFAULT {version-1},
    calling-presentation-selector [1] IMPLICIT Presentation-selector OPTIONAL,
    called-presentation-selector [2] IMPLICIT Presentation-selector OPTIONAL,
    presentation-context-definition-list [4] IMPLICIT Context-list,
    user-data                   CHOICE {
      fully-encoded-data
        transfer-syntax-name      Transfer-syntax-name OPTIONAL,
        presentation-context-identifier Presentation-context-identifiant,
        presentation-data-values  CHOICE {
          single-ASN1-type       [0] AARQ-apdu {{Protocols}} } } } }
```

```
Presentation-selector ::= OCTET STRING(SIZE (1..4, ..., 5..MAX))
```

Context-list ::= SEQUENCE SIZE (2) OF
SEQUENCE {
presentation-context-identif **Presentation-context-identif**,
abstract-syntax-name **Abstract-syntax-name**,
transfer-syntax-name-list **SEQUENCE OF Transfer-syntax-name }**

Presentation-context-identif ::= INTEGER(1..127, ..., 128..MAX)

Abstract-syntax-name ::= OBJECT IDENTIFIER

Transfer-syntax-name ::= OBJECT IDENTIFIER

AARQ-apdu {APPLICATION-CONTEXT:Protocols} ::= [APPLICATION 0] IMPLICIT SEQUENCE {
protocol-version [0] **IMPLICIT BIT STRING {version1(0)} DEFAULT {version1},**
application-context-name [1] **Application-context-name,**
called-AP-title [2] **Name** **OPTIONAL,**
called-AE-qualifier [3] **RelativeDistinguishedName** **OPTIONAL,**
called-AP-invocation-identif [4] **AP-invocation-identif** **OPTIONAL,**
called-AE-invocation-identif [5] **AE-invocation-identif** **OPTIONAL,**
calling-AP-title [6] **Name** **OPTIONAL,**
calling-AE-qualifier [7] **RelativeDistinguishedName** **OPTIONAL,**
calling-AP-invocation-identif [8] **AP-invocation-identif** **OPTIONAL,**
calling-AE-invocation-identif [9] **AE-invocation-identif** **OPTIONAL,**
implementation-information [29] **IMPLICIT Implementation-data** **OPTIONAL,**
user-information [30]
IMPLICIT SEQUENCE SIZE(1) OF [UNIVERSAL 8] IMPLICIT SEQUENCE {
direct-reference **OBJECT IDENTIFIER OPTIONAL,**
indirect-reference **Presentation-context-identif,**
encoding **CHOICE {**
single-ASN1-type **[0] TheOsiBind {{Protocols}} }**

NOTE – Le composant **user-information** est défini dans la Rec. UIT-T Rec. X.226 | ISO/CEI 8823-1 comme élément EXTERNAL. Etant donné que le contenu de l'élément externe est connu, il pourrait constituer une aide pour les responsables de la mise en œuvre si le codage exact de l'élément EXTERNAL est fourni. L'élément externe est présenté ici conformément au codage défini dans les Rec. UIT-T X.690 | ISO/CEI 8825-1, UIT-T X.691 | ISO/CEI 8825-2 et UIT-T. X.693 | ISO/CEI 8825-4. Il ne s'agit pas d'une présentation ASN.1 totalement légale. La spécification ASN.1 formelle et légale utilisant la notation EXTERNAL est indiquée dans l'Annexe B.

Application-context-name ::= OBJECT IDENTIFIER

AP-invocation-identif ::= INTEGER

AE-invocation-identif ::= INTEGER

Implementation-data ::= GraphicString

TheOsiBind {APPLICATION-CONTEXT:Protocols} ::=
[16] APPLICATION-CONTEXT.&bind-operation.&ArgumentType ({Protocols})

La demande **OsiBind** permet d'initier une association d'applications. Elle comprend des éléments de protocole de présentation (voir le § 7.6.1.1), des éléments de protocole ACSE (voir le § 7.6.1.2) et des éléments de protocole de rattachement d'annuaire (voir le § 7.6.1.3). La demande de rattachement sera formatée comme spécifié dans les paragraphes mentionnés.

La demande **OsiBind** est acheminée par le paramètre Données d'utilisateur ou le paramètre Données d'utilisateur en mode étendu de l'unité SPDU CONNECT de la couche Session (voir le § 8.3.3).

Le preneur de l'association d'applications vérifiera les éléments de protocole dans l'ordre suivant:

- 1) Les éléments de protocole de session seront vérifiés. Si l'un ou plusieurs de ces éléments de protocole sont inacceptables, une unité SPDU REFUSE (voir le § 8.3.5) sera renvoyée. Dans le cas contraire, la procédure se poursuit.
- 2) Les éléments de protocole de présentation seront vérifiés. Si l'un ou plusieurs de ces éléments de protocole sont inacceptables, une erreur **OsiBindError** avec un composant **provider-reason** et sans composant **user-data** (voir le § 7.6.3.1) sera renvoyée. Dans le cas contraire, la procédure se poursuit.
- 3) Les éléments de protocole ACSE seront vérifiés. Si l'un ou plusieurs de ces éléments de protocole sont inacceptables, une erreur **OsiBindError** sera renvoyée avec les composants **result** et **result-source-**

diagnostic de l'unité **AAREerr-apdu** présents et le composant **user-information** absent comme cela est spécifié au § 7.6.3.2. Dans le cas contraire, la procédure se poursuit.

- 4) Le rattachement de l'annuaire sera vérifié selon les règles du protocole d'annuaire en question. Un résultat **OsiBindResult** (voir le § 7.6.2) sera renvoyé si le preneur peut accepter le rattachement de l'annuaire; dans le cas contraire, une erreur **OsiBindError** sera renvoyée avec le composant **user-information** de l'unité **AAREerr-apdu** présent.

Si une erreur de protocole est détectée à tout moment pendant la séquence, l'opération d'abandon appropriée sera initiée comme indiqué au § 7.5.

7.6.1.1 Éléments de protocole de présentation

Les éléments de protocole de présentation constituant une unité PPDU CP sont ceux qui sont définis par le type de données **OsiBind** susmentionné à l'exception de l'unité **AARQ-apdu** intégrée.

Le composant **mode-selector** sera toujours mis à 1.

NOTE 1 – La Rec. UIT-T X.226 | ISO/CEI 8823-1 définit deux modes de connexion de présentation. Les présentes Spécifications d'annuaire utilisent toujours le mode **normal-mode**.

Le composant **normal-mode-parameters** comprend les sous-composants suivants:

- a) Le sous-composant **protocol-version** sera omis ou recevra la valeur **version-1**. S'il est défini différemment, le preneur renverra une erreur **OsiBindError** dont l'élément **provider-reason** est mis à la valeur **protocol-version-not-supported**.
- b) Si elle est fournie, la valeur du sous-composant **calling-presentation-selector** sera obtenue à partir d'informations locales.

Pour une définition du sélecteur de présentation, voir le § 7.4.

- c) Si elle est fournie, la valeur du sous-composant **called-presentation-selector** sera obtenue à partir:
- d'informations provenant de la valeur **AccessPoint** de l'élément **ContinuationReference** à la suite d'une opération d'annuaire précédente (voir la Rec. UIT-T X.518 | ISO/CEI 9594-4);
 - d'informations locales.

Si le preneur n'utilise pas l'adresse du sélecteur de présentation ou si le sélecteur de présentation fourni n'est pas destiné à un processus d'application d'annuaire, le preneur renverra une erreur **OsiBindError** dont l'élément **provider-reason** est mis à la valeur **called-presentation-address-unknown**.

- d) Le sous-composant **presentation-context-definition-list** comprendra deux éléments dont chacun est un type de séquence doté:
- d'un élément **presentation-context-identifiant** qui est choisi par l'initiateur, sera un entier impair et sera différent pour les deux éléments;
 - d'un élément **abstract-syntax-name**, qui
 - i) pour l'un des éléments, sera un identificateur d'objet qui identifie la syntaxe abstraite ACSE (**id-acseAS**);
 - ii) pour l'autre élément, sera un identificateur d'objet pour une syntaxe abstraite d'annuaire correspondant au type d'association d'applications à établir (**id-as-directoryAccessAS**, **id-as-directorySystemAS**, **id-as-directoryShadowAS** ou **id-as-directoryOperationalBindingManagementAS**, selon qu'il sera approprié);
 - d'un élément **transfer-syntax-name-list**, qui comprendra un seul élément, à savoir l'identificateur d'objet pour les règles de codage de base (BER, *basic encoding rules*).

NOTE 2 – La Rec. UIT-T X.226 | ISO/CEI 8823-1 permet de proposer plusieurs syntaxes de transfert, l'une d'entre elles étant ensuite choisie par le preneur. Les règles d'extensibilité définies au § 12 imposent l'application des règles de codage de base.

Voir le § 8.1 pour des précisions sur les syntaxes abstraites et les syntaxes de transfert.

- e) Le sous-composant **user-data** comprend les éléments suivants:

NOTE 3 – Le sous-composant **user-data** reflète la valeur **fully-encoded-data** choisie pour les données **user-data** de l'unité PPDU CP définie par la Rec. UIT-T X.226 | ISO/CEI 8823-1. Les données **fully-encoded-data** comprennent une séquence de listes **PVD-list**. La présente Spécification d'annuaire prescrit exactement une liste **PVD-list**. Par conséquent, le type *sequence-of* définit exactement une valeur.

- le sous-composant **transfer-syntax-name** s'il est présent, sera l'identificateur d'objet pour les règles de codage de base;

NOTE 4 – Conformément au § 8.4.2.7 de la Rec. UIT-T X.226 | ISO/CEI 8823-1: "Le nom de la syntaxe de transfert doit être présent quand plusieurs noms de syntaxe de transfert ont été proposés pour le contexte de présentation des valeurs de données de présentation".

- le sous-composant **presentation-context-identifiant** recevra la même valeur que l'identificateur **presentation-context-identifiant** de l'élément de la liste **presentation-context-definition-list** qui spécifie la syntaxe abstraite ACSE;
- le sous-composant **presentation-data-values** inclura les éléments de protocole ACSE définis au § 7.6.1.2.

7.6.1.2 Eléments de protocole ACSE

Les éléments de protocole ACSE sont ceux qui sont définis par le type de données **AARQ-apdu** ci-dessus sauf pour l'élément **TheOsBind** intégré.

NOTE 1 – Les éléments de protocole ACSE sont les composants pertinents de l'unité **AARQ-apdu** définis par la Rec. UIT-T X.227 | ISO/CEI 8650-1. Seule l'unité fonctionnelle "noyau" de l'élément ACSE est utilisée dans les présentes Spécifications d'annuaire. Conformément au § 9.1 de la Rec. UIT-T X.227 | ISO/CEI 8650-1, les composants **sender-acse-requirements**, **mechanism-name**, **calling-authentication-value** et **application-context-name-list** ne sont pas pertinents.

Le composant **protocol-version** sera omis ou recevra la valeur **version1**, c'est-à-dire que le bit 0 est défini. Si le composant est présent, l'initiateur n'inclura aucun bit après le bit 0. Si le preneur reçoit une demande de rattachement dans laquelle ce composant est présent, dans laquelle le bit 0 est défini et dans laquelle un ou plusieurs bits sont définis, il ne sera pas tenu compte de ces bits. Si le bit 0 n'est pas défini, mais qu'un autre bit est défini, le processus d'application preneur répondra par un élément **OsBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **no-common-acse-version**.

Le composant **application-context-name** sera mis:

- a) pour le protocole DAP, à la valeur **id-ac-directoryAccessAC**;
- b) pour le protocole DSP, à la valeur **id-as-directorySystemAC**;
- c) pour le protocole DISP, à l'une des valeurs suivantes:
 - **id-ac-shadowConsumerInitiatedAC**;
 - **id-ac-shadowSupplierInitiatedAC**;
 - **id-ac-shadowSupplierInitiatedAsynchronousAC**; ou
 - **id-ac-shadowConsumerInitiatedAsynchronousAC**;
- d) pour le protocole DOP, à la valeur **id-ac-directoryOperationalBindingManagementAC**.

Si le preneur ne prend pas en charge le composant **application-context-name** indiqué, il répondra par un élément **OsBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **application-context-name-not-supported**.

S'il est présent, le composant **called-AP-title** sera obtenu à partir:

- d'informations renvoyées dans une référence **ContinuationReference** à la suite d'une opération d'annuaire précédente;
- d'informations locales.

Si le preneur ne reconnaît pas le composant **called-AP-title**, il répondra par un élément **OsBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **called-AP-title-not-recognized**.

S'il est présent, le composant **called-AE-qualifier** sera obtenu à partir:

- d'informations renvoyées dans une référence **ContinuationReference** à la suite d'une opération d'annuaire précédente;
- d'informations locales.

Si le preneur ne reconnaît pas le composant **called-AE-qualifier**, il répondra par un élément **OsBindError** (voir le § 7.6.3) dans lequel **Associate-source-diagnostic** a la valeur **called-AE-qualifier-not-recognized**.

Le composant **called-AP-invocation-identifiant** peut optionnellement être fourni si les informations ayant trait à sa valeur provenant d'une association d'applications précédente sont conservées. Si le preneur ne reconnaît pas le composant **called-AP-invocation-identifiant**, il répondra par un élément **OsBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **called-AP-invocation-identifiant-not-recognized**.

Le composant **called-AE-invocation-identifiant** peut optionnellement être fourni si les informations ayant trait à sa valeur provenant d'une association d'applications précédente sont conservées. Si le preneur ne reconnaît pas le

composant **called-AE-invocation-identifiant**, il répondra par un élément **OsiBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **called-AE-invocation-identifiant-not-recognized**.

S'il est fourni, le composant **calling-AP-title** sera obtenu à partir des informations locales. Si le preneur veut s'assurer de l'identité de l'initiateur, mais qu'il ne reconnaît pas le composant **calling-AP-title**, il peut rejeter l'association d'applications au moyen d'un élément **OsiBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **calling-AP-title-not-recognized**.

S'il est fourni, le composant **calling-AE-qualifier** sera obtenu à partir des informations locales. Si le preneur veut s'assurer de l'identité de l'initiateur, mais qu'il ne reconnaît pas le composant **calling-AE-qualifier**, il peut rejeter l'association d'applications au moyen d'un élément **OsiBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **calling-AE-qualifier-not-recognized**.

Le composant **calling-AP-invocation-identifiant** peut optionnellement être fourni. Un système récepteur peut faire abstraction de cette valeur, si elle est présente. Si le preneur veut s'assurer de l'identité de l'initiateur, mais qu'il ne reconnaît pas le composant **calling-AP-invocation-identifiant**, il peut rejeter l'association d'applications au moyen d'un élément **OsiBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **calling-AP-invocation-identifiant-not-recognized**.

Le composant **calling-AE-invocation-identifiant** peut optionnellement être fourni. Un système récepteur peut faire abstraction de cette valeur, si elle est présente. Si le preneur veut s'assurer de l'identité de l'initiateur, mais qu'il ne reconnaît pas le composant **calling-AE-invocation-identifiant**, il peut rejeter l'association d'applications au moyen d'un élément **OsiBindError** (voir le § 7.6.3), dans lequel **Associate-source-diagnostic** a la valeur **calling-AE-invocation-identifiant-not-recognized**.

Le composant **implementation-information** peut détenir des informations propres à l'implémentation. Ces informations n'affectent pas la procédure d'établissement de l'association d'applications.

Le composant **user-information** comprend les sous-composants suivants:

- a) **direct-reference**, s'il est présent, qui contiendra l'identificateur d'objet pour les règles de codage de base;
- b) **indirect-reference**, qui identifiera la syntaxe abstraite d'annuaire dans l'élément **presentation-context-definition-list** conformément au § 7.6.1.1 d);
- c) **single-ASN1-type**, qui contiendra les éléments de protocole de rattachement comme indiqué au § 7.6.1.3.

NOTE 2 – Le composant **user-information** correspond au composant **user-information** de l'élément **AARQ-apdu** comme le définit la Rec. UIT-T X.227 | ISO/CEI 8650-1. Ce composant est une séquence de l'élément **EXTERNAL**. Les présentes Spécifications d'annuaire exigent exactement une occurrence de l'élément **EXTERNAL** (voir la Note dans le 7.6.1).

7.6.1.3 Eléments de protocole de rattachement

L'élément **TheOsiBind** sera l'argument de la demande de rattachement tel que le définit le protocole d'annuaire en question.

NOTE – L'argument de rattachement commence par l'étiquette [16] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

7.6.2 Résultat du rattachement OSI

Un résultat **OsiBindResult** est renvoyé par le preneur si la demande **OsiBind** est acceptable et que le preneur décide d'initier une association d'applications.

```
OsiBindResult {APPLICATION-CONTEXT:Protocols} ::= SET {
  mode-selector          [0] IMPLICIT SET {mode-value [0] IMPLICIT INTEGER (1) },
  normal-modeparameters [2] IMPLICIT SEQUENCE {
    protocol-version     [0] IMPLICIT BIT STRING {version-1(0)} DEFAULT {version-1},
    responding-presentation-selector
                          [3] IMPLICIT Presentation-selector OPTIONAL,
    presentation-context-definition-result-list
                          [5] IMPLICIT SEQUENCE SIZE (2) OF SEQUENCE {
      result              [0] IMPLICIT Result (acceptance),
      transfer-syntax-name [1] IMPLICIT Transfer-syntax-name },
    user-data             CHOICE {
      fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        transfer-syntax-name Transfer-syntax-name OPTIONAL,
        presentation-context-identifier Presentation-context-identifier,
        presentation-data-values CHOICE {
          single-ASN1-type [0] AARE-apdu {{Protocols}} } } } }
```

Result ::= INTEGER {
acceptance (0),
user-rejection (1),
provider-rejection (2) }

AARE-apdu {APPLICATION-CONTEXT:Protocols} ::= [APPLICATION 1] IMPLICIT SEQUENCE {
protocol-version [0]
IMPLICIT BIT STRING {version1(0)} DEFAULT {version1},
application-context-name [1] Application-context-name,
result [2] Associate-result (accepted),
result-source-diagnostic [3] Associate-source-diagnostic,
responding-AP-title [4] Name OPTIONAL,
responding-AE-qualifier [5] RelativeDistinguishedName OPTIONAL,
responding-AP-invocation-identifier [6] AP-invocation-identifier OPTIONAL,
responding-AE-invocation-identifier [7] AE-invocation-identifier OPTIONAL,
implementation-information [29] IMPLICIT Implementation-data OPTIONAL,
user-information [30]
IMPLICIT SEQUENCE SIZE(1) OF [UNIVERSAL 8] IMPLICIT SEQUENCE {
direct-reference OBJECT IDENTIFIER OPTIONAL,
indirect-reference Presentation-context-identifier,
encoding CHOICE {
single-ASN1-type [0] TheOsiBindRes {{Protocols}} } }

NOTE – Voir Note dans § 7.6.1.

Associate-result ::= INTEGER {
accepted (0),
rejected-permanent (1),
rejected-transient (2) }{0..2, ...}

Associate-source-diagnostic ::= CHOICE {
acse-service-user [1] INTEGER {
null (0),
no-reason-given (1),
application-context-name-not-supported (2),
calling-AP-title-not-recognized (3),
calling-AP-invocation-identifier-not-recognized (4),
calling-AE-qualifier-not-recognized (5),
calling-AE-invocation-identifier-not-recognized (6),
called-AP-title-not-recognized (7),
called-AP-invocation-identifier-not-recognized (8),
called-AE-qualifier-not-recognized (9),
called-AE-invocation-identifier-not-recognized (10) } (0..10, ...),
acse-service-provider [2] INTEGER {
null (0),
no-reason-given (1),
no-common-acse-version (2) } (0..2, ...) }

TheOsiBindRes {APPLICATION-CONTEXT:Protocols} ::=
[17] APPLICATION-CONTEXT.&bind-operation.&ResultType ({Protocols})

L'élément **OsiBindResult** est acheminé dans le paramètre Données d'utilisateur de l'unité SPDU ACCEPT de la couche Session (voir le § 8.3.4).

7.6.2.1 Eléments de protocole de présentation

Les éléments de protocole de présentation constituant une unité PPDU CPA sont ceux qui sont définis par le type de données **OsiBindResult** ci-dessus sauf pour l'unité **AARE-apdu** intégrée.

Le composant **mode-selector** sera toujours mis à 1.

Le composant **normal-mode-parameters** comprend les sous-composants suivants:

- le sous-composant **protocol-version** sera omis ou aura la valeur **version-1**;
- s'il est fourni, le sous-composant **responding-presentation-selector** sera obtenu à partir des informations locales;
- le sous-composant **presentation-context-definition-result-list** comprendra deux éléments dans une séquence correspondant à la séquence d'éléments fournie dans l'élément **presentation-context-definition-list** de la demande de rattachement, chacun fournissant le résultat de la négociation du contexte pour l'élément correspondant comme suit:

- l'élément **result** sera présent et aura la valeur **acceptance**;
 - l'élément **transfer-syntax-name** sera présent et spécifiera l'identificateur d'objet pour les règles de codage de base (BER);
- d) le sous-composant **user-data** comprend les éléments suivants:
- Le sous-composant **transfer-syntax-name**, s'il est présent, sera l'identificateur d'objet pour les règles de codage de base;
 - le sous-composant **presentation-context-identifiant** recevra la même valeur que le sous-composant **presentation-context-identifiant** de l'élément de la liste **presentation-context-definition-list** de la demande de rattachement qui a spécifié le nom de syntaxe abstraite ACSE;
 - le sous-composant **presentation-data-values** contiendra les éléments de protocole ACSE comme indiqué au § 7.6.2.2.

7.6.2.2 Eléments de protocole ACSE

Le composant **protocol-version** sera omis ou recevra la valeur **version1**, c'est-à-dire que le bit 0 est défini. Si le composant est présent, le preneur n'inclura aucun bit après le bit 0.

Le composant **result** sera mis à la valeur **accepted** par le preneur.

Le composant **result-source-diagnostic** acceptera le choix **acse-service-user** et prendra la valeur **null** ou **no-reason-given**.

Le composant **application-context-name** sera présent et mis à la valeur du composant correspondant de la demande de rattachement.

S'il est fourni, le composant **responding-AP-title** sera obtenu à partir des informations locales.

S'il est fourni, le **responding-AE-qualifier** sera obtenu à partir des informations locales.

Le composant **responding-AP-invocation-identifiant** peut optionnellement être fourni. L'initiateur peut ne pas tenir compte de ce composant s'il est présent.

Le composant **responding-AE-invocation-identifiant** peut optionnellement être fourni. L'initiateur peut ne pas tenir compte de ce composant s'il est présent.

Le composant **implementation-information** peut contenir des informations propres à l'implémentation. Ces informations n'affectent pas la procédure d'établissement de l'association d'applications.

Le composant **user-information** comprend les sous-composants suivants:

- a) **direct-reference**, s'il est présent, qui contiendra l'identificateur d'objet pour les règles de codage de base ASN.1;
- b) **indirect-reference**, qui identifiera la syntaxe abstraite d'annuaire au moyen de l'élément **presentation-context-definition-list** comme défini au § 7.6.1.1 d);
- c) **single-ASN1-type**, qui contiendra les éléments de protocole de résultat comme indiqué au § 7.6.2.3.

7.6.2.3 Eléments de protocole de résultat de rattachement

TheOsiBindRes sera du type de résultat de rattachement défini pour le protocole d'annuaire en question.

NOTE – Le résultat du rattachement commence par l'étiquette [17] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

7.6.3 Erreur de rattachement OSI

```
OsiBindError {APPLICATION-CONTEXT:Protocols} ::= CHOICE {
  normal-mode-parameters SEQUENCE {
    protocol-version [0] IMPLICIT BIT STRING {version-1(0)} DEFAULT {version-1},
    responding-presentation-selector
      [3] IMPLICIT Presentation-selector OPTIONAL,
    presentation-context-definition-result-list
      [5] IMPLICIT Result-list OPTIONAL,
    provider-reason [10] IMPLICIT Provider-reason OPTIONAL,
    user-data CHOICE {
      fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        transfer-syntax-name Transfer-syntax-name OPTIONAL,
        presentation-context-identifiant Presentation-context-identifiant,
        presentation-data-values CHOICE {
```

single-ASN1-type [0] AAREerr-apdu {{Protocols}} } } } OPTIONAL } }

Result-list ::= SEQUENCE SIZE (2) OF SEQUENCE {
 result [0] IMPLICIT Result,
 transfer-syntax-name [1] IMPLICIT Transfer-syntax-name OPTIONAL,
 provider-reason [2] IMPLICIT INTEGER {
 reason-not-specified (0),
 abstract-syntax-not-supported (1),
 proposed-transfer-syntaxes-not-supported (2) } OPTIONAL }

Provider-reason ::= INTEGER {
 reason-not-specified (0),
 temporary-congestion (1),
 local-limit-exceeded (2),
 called-presentation-address-unknown (3),
 protocol-version-not-supported (4),
 default-context-not-supported (5),
 user-data-not-readable (6),
 no-PSAP-available (7) }

AAREerr-apdu {APPLICATION-CONTEXT:Protocols} ::= [APPLICATION 1] IMPLICIT SEQUENCE {
 protocol-version [0] IMPLICIT BIT STRING {version1(0)}
 DEFAULT {version1},
 application-context-name [1] Application-context-name,
 result [2] Associate-result (rejected-permanent..rejected-transient),
 result-source-diagnostic [3] Associate-source-diagnostic,
 responding-AP-title [4] Name OPTIONAL,
 responding-AE-qualifier [5] RelativeDistinguishedName OPTIONAL,
 responding-AP-invocation-identifier [6] AP-invocation-identifier OPTIONAL,
 responding-AE-invocation-identifier [7] AE-invocation-identifier OPTIONAL,
 implementation-information [29] IMPLICIT Implementation-data OPTIONAL,
 user-information [30]
 IMPLICIT SEQUENCE SIZE(1) OF [UNIVERSAL 8] IMPLICIT SEQUENCE {
 direct-reference OBJECT IDENTIFIER OPTIONAL,
 indirect-reference Presentation-context-identifier,
 encoding CHOICE {
 single-ASN1-type [0] TheOsiBindErr {{Protocols}} } OPTIONAL } }

NOTE – Voir Note au § 7.6.1.

TheOsiBindErr {APPLICATION-CONTEXT:Protocols} ::=
 [18] APPLICATION-CONTEXT.&bind-operation.&Errors.&ParameterType ({{Protocols}})

OsiBindError est acheminé dans le champ Code raison de l'unité SPDU REFUSE de la couche Session (voir le § 8.3.5).

7.6.3.1 Eléments de protocole de présentation

Les éléments de protocole de présentation constituant une unité PPDU CPR sont ceux qui sont définis par le type de données **OsiBindError** ci-dessus sauf pour l'unité **AAREerr-apdu** intégrée.

Le composant **normal-mode-parameters** comprend les sous-composants suivants:

NOTE 1 – L'unité PPDU CPR correspond à un choix entre le mode X.410 et le mode normal. Les présentes Spécifications d'annuaire utilisent uniquement le mode normal. La déclaration CHOICE est conservée pour assurer une compatibilité binaire en amont lorsque les règles de codage de base ou un codage analogue sont utilisés.

- a) Le sous-composant **protocol-version** aura les caractéristiques indiquées au § 7.6.2.1
- b) S'il est fourni, le sous-composant **responding-presentation-selector** aura les caractéristiques indiquées au § 7.6.2.1.
- c) Le sous-composant **presentation-context-definition-result-list** aura les caractéristiques indiquées ci-après:
 - si le refus n'est pas associé à la négociation du contexte de présentation, l'élément **result** aura la valeur **acceptance**, le sous-composant **transfer-syntax-name** sera présent et spécifiera l'identificateur d'objets pour les règles de codage de base et l'élément **provider-reason** sera absent;
 - si la syntaxe abstraite en question n'est prise en charge par aucune des syntaxes de transfert proposées, l'élément **result** aura la valeur **provider-rejection** et l'élément **provider-reason** sera présent avec la valeur appropriée; ou

- si la syntaxe abstraite en question n'est pas du tout prise en charge et que la puce précédente ne s'applique pas, l'élément **result** aura la valeur **user-rejection** et l'élément **provider-reason** sera présent avec la valeur appropriée.
- d) Le sous-composant **provider-reason** sera présent si l'association d'application est refusée en raison de problèmes détectés dans les éléments du protocole de présentation de la demande de rattachement. Dans le cas contraire, ce sous-composant sera absent.

NOTE 2 – Au § 6.2.4.9 de la Rec. UIT-T X.226 | ISO/CEI 8823-1, il est dit en ce qui concerne la raison **provider-reason**: "Si ce paramètre figure, il indique que le refus émane du fournisseur de service de présentation répondant; s'il ne figure pas, cette absence indique que le refus émane de l'utilisateur du service de présentation répondant."

- e) Le sous-composant **user-data** sera absent si le sous-composant **provider-reason** est présent. Dans le cas contraire, il sera présent avec les éléments suivants:
 - Le sous-composant **transfer-syntax-name** s'il est présent, sera l'identificateur d'objets pour les règles de codage de base de l'ASN.1.
 - le sous-composant **presentation-context-identifiant** recevra la même valeur que le sous-composant **presentation-context-identifiant** de l'élément de la liste **presentation-context-definition-list** de la demande de rattachement qui a spécifié le nom de syntaxe abstraite ACSE;
 - le sous-composant **presentation-data-values** contiendra les éléments de protocole ACSE comme indiqué au § 7.6.3.2.

7.6.3.2 Eléments de protocole ACSE

Le composant **protocol-version** aura les caractéristiques indiquées au § 7.6.2.2.

Le composant **application-context-name** sera présent et mis à la valeur du composant correspondant de la demande de rattachement.

Le composant **result** recevra la valeur **rejected-permanent** ou **rejected-transient** selon les facteurs locaux à considérer.

NOTE – Conformément au § 11.1.1 de la Rec. UIT-T X.881 | ISO/CEI 13712-2, une erreur de rattachement est acheminée dans la réponse/confirmation A-ASSOCIATE, la valeur du paramètre Result des primitives de service A-ASSOCIATE étant mise à la valeur "rejected (permanent)" ou "rejected (transient)" et la valeur d'erreur de l'opération de rattachement étant mappée avec le paramètre Information d'utilisateur de ces primitives de service. Au niveau du protocole, cela se traduit par le fait que le composant **result** est mis à la valeur **rejected-permanent** ou **rejected-transient**. La plupart des erreurs de rattachement témoignent d'une condition permanente. Cela étant, l'erreur **serviceError** caractérisée par le problème **unavailable** peut être considérée comme transitoire.

Le composant **result-source-diagnostic** prendra les valeurs suivantes selon l'état considéré:

- a) si le refus se situe dans un protocole d'annuaire, le choix **acse-service-user** sera accepté avec la valeur **null** ou **no-reason-given**; ou
- b) si le refus est associé au protocole ACSE ou est dû à des erreurs dans le titre du processus d'application spécifié, le titre de l'entité d'application ou le contexte d'application, le choix **acse-service-user** sera accepté avec la valeur appropriée.

Si elle est présente, la valeur du composant **responding-AP-title** sera obtenue à partir des informations locales.

S'il est présent, le composant **responding-AE-qualifier** sera obtenu à partir des informations locales.

S'il est présent, le composant **responding-AP-invocation-identifiant** peut être écarté ou conservé pour une future association avec l'agent DSA considéré.

S'il est présent, le composant **responding-AE-invocation-identifiant** peut être écarté ou conservé pour une future association avec l'agent DSA considéré.

Le composant **implementation-information** peut contenir des informations propres à l'implémentation.

Le composant **user-information** comprend les sous-composants suivants:

- a) **direct-reference**, lorsqu'il est présent contiendra l'identificateur d'objet pour les règles de codage de base (BER) de l'ASN1.
- b) **indirect-reference**, qui identifiera la syntaxe abstraite d'annuaire dans l'élément **presentation-context-definition-list** ainsi qu'il est défini au § 7.6.1.1 d);
- c) **single-ASN1-type**, qui contiendra les éléments de protocole d'erreur de rattachement comme indiqué au § 7.6.3.3.

7.6.3.3 Eléments de protocole d'erreur de rattachement

TheOsiBindErr sera le type d'erreur de rattachement pertinent.

NOTE – L'erreur de rattachement commence par l'étiquette [18] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

7.6.4 Demande de détachement OSI

```
OsiUnbind ::= CHOICE {
    fully-encoded-data    [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        presentation-context-identifiant Presentation-context-identifiant,
        presentation-data-values CHOICE {
            single-ASN1-type    [0] TheOsiUnbind } } }
```

```
TheOsiUnbind ::= [APPLICATION 2] IMPLICIT SEQUENCE {
    reason [0] IMPLICIT Release-request-reason OPTIONAL }
```

```
Release-request-reason ::= INTEGER {
    normal (0) }
```

OsiUnbind est acheminé dans les données d'utilisateur de l'unité SPDU FINISH de la couche Session (voir le § 8.3.6).

Seul l'initiateur d'une association d'applications peut invoquer une demande de détachement.

NOTE 1 – Le paragraphe 8.5 de la Rec. UIT-T X.880 | ISO/CEI 13712-1 définit une classe d'objets informationnels **CONNECTION-PACKAGE**, dans laquelle le champ **&responderCanUnbind** indique si le preneur peut émettre une demande de détachement ou non. Sa valeur par défaut est **FALSE**. La quatrième édition de la présente spécification d'annuaire n'a ajouté l'élément **&responderCanUnbind** pour aucun des protocoles. Le protocole IDM permet au preneur d'émettre une demande de détachement, sauf pour le protocole DAP (voir le § 9.2.2).

NOTE 2 – Le paragraphe 8.5 de la Rec. UIT-T X.880 | ISO/CEI 13712-1 définit aussi un champ **&unbindCanFail** dans la classe d'objets informationnels **CONNECTION-PACKAGE** dont la valeur par défaut est **FALSE**. La quatrième édition de la présente spécification d'annuaire n'a ajouté l'élément **&unbindCanFail** pour aucun des protocoles.

7.6.4.1 Eléments de protocole de présentation

Les éléments de protocole de présentation sont uniquement ceux qui sont définis par le type d'erreur **User-data** spécifié par la Rec. UIT-T X.226 | ISO/CEI 8823-1.

Le composant **presentation-context-identifiant** recevra la même valeur que le composant **presentation-context-identifiant** de l'élément de la liste **presentation-context-definition-list** de la demande de rattachement qui a spécifié le nom de syntaxe abstraite ACSE.

Le composant **presentation-data-values** contiendra les éléments de protocole ACSE comme indiqué au § 7.6.4.2.

7.6.4.2 Eléments de protocole ACSE

Le composant **reason** sera mis à la valeur **normal** ou sera absent. L'absence du composant **reason** indique une libération normale.

NOTE 1 – Conformément au § 11.1.2 de la Rec. UIT-T X.881 | ISO/CEI 13712-2, le composant **reason** sera toujours mis à la valeur **normal**.

NOTE 2 – Conformément à la Rec. UIT-T X.226 | ISO/CEI 8823-1, il n'y a pas d'éléments de protocole de présentation pour la libération normale d'une connexion. Une libération normale est consécutive à la libération normale de la connexion de session sous-jacente.

7.6.5 Résultat de l'opération de détachement OSI

```
OsiUnbindResult ::= CHOICE {
    fully-encoded-data    [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        presentation-context-identifiant Presentation-context-identifiant,
        presentation-data-values CHOICE {
            single-ASN1-type    [0] TheOsiUnbindRes } } }
```

```
TheOsiUnbindRes ::= [APPLICATION 3] IMPLICIT SEQUENCE {
    reason [0] IMPLICIT Release-response-reason OPTIONAL }
```

```
Release-response-reason ::= INTEGER {
    normal (0) }
```

NOTE – Les spécifications antérieures à l'édition 5 stipulent que le paramètre Result du service A-RELEASE tel que le définit la Rec. UIT-T X.217 | ISO/CEI 8649 sera mis à la valeur "affirmative".

OsiUnbindResult est acheminé dans les données d'utilisateur de l'unité SPDU DISCONNECT de la couche Session (voir le § 8.3.7).

7.6.5.1 Eléments de protocole de présentation

Les éléments de protocole de présentation sont uniquement ceux qui sont définis par le type de données **User-data** spécifié par la Rec. UIT-T X.226 | ISO/CEI 8823-1.

Le composant **presentation-context-identifiant** recevra la même valeur que le composant **presentation-context-identifiant** de l'élément de la liste **presentation-context-definition-list** de la demande de rattachement qui a spécifié la syntaxe abstraite ACSE.

Le composant **presentation-data-values** contiendra la demande de libération ACSE.

7.6.5.2 Eléments du protocole ACSE

L'absence du composant reason indique une libération normale.

7.6.6 Opérations OSI

```
OsiOperation {OPERATION:Operations} ::= CHOICE {
    fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        presentation-context-identifiant Presentation-context-identifiant,
        presentation-data-values CHOICE {
            single-ASN1-type [0] CHOICE {
                request OsiReq {{Operations}},
                result OsiRes {{Operations}},
                error OsiErr {{Operations}},
                reject OsiRej } } }
    }
```

L'élément **OsiOperation** est acheminé dans le champ Informations d'utilisateur de l'unité SPDU DATA TRANSFER de la couche Session (voir le § 8.3.10).

7.6.6.1 Eléments du protocole de présentation

Le composant **presentation-context-identifiant** recevra la même valeur que le composant **presentation-context-identifiant** de l'élément de la liste **presentation-context-definition-list** de la demande de rattachement qui a spécifié le nom de syntaxe abstraite d'annuaire pour le protocole d'annuaire en question.

Le composant **presentation-data-values** contiendra la demande, le résultat, l'erreur ou le refus d'annuaire.

7.6.6.2 Demande OSI

```
OsiReq {OPERATION:Operations} ::= [1] IMPLICIT SEQUENCE {
    invokeld Invokeld,
    opcode OPERATION.&operationCode {{Operations}},
    argument OPERATION.&ArgumentType {{Operations}} {@opcode} }
```

NOTE 1 – La demande commence par l'étiquette [1] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

Le composant **invokeld** identifie une invocation déterminée. Ce ne sera pas une valeur qui a été utilisée pour une demande précédente qui exige une réponse (résultat et/ou erreur) et qui est encore en cours. Si cette règle est enfreinte, le récepteur émettra un élément **OsiReject** dont l'élément **InvokeProblem** aura la valeur **duplicateInvocation**. Si le résultat de la demande n'est pas forcément une réponse, le délai à respecter pour pouvoir utiliser à nouveau un composant **invokeld** est décidé localement.

NOTE 2 – Toutes les opérations d'annuaire actuellement définies exigent une réponse.

Le composant **opcode** contiendra le code d'opération pour le type particulier d'opération. Si un code d'opération inconnu est défini, le récepteur émettra un élément **OsiReject** dont l'élément **InvokeProblem** aura la valeur **unrecognizedOperation**.

Le composant **argument** contiendra l'argument constitué selon le champ **&ArgumentType** du type d'opération identifié par le composant **opcode** pour le protocole en question.

7.6.6.3 Résultat OSI

```
OsiRes { OPERATION:Operations } ::= [2] IMPLICIT SEQUENCE {
    invokeld Invokeld,
    result SEQUENCE {
        opcode OPERATION.&operationCode {{Operations}},
        result OPERATION.&ResultType {{Operations}} {@opcode} } }
```

ISO/CEI 9594-5:2005 (F)

NOTE – Le résultat commence par l'étiquette [2] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

Le composant **invokeID** sera identique à celui qui est spécifié dans la demande correspondante.

Le composant **opcode** sera identique à celui qui est spécifié dans la demande correspondante.

Le composant **result** contiendra le résultat constitué selon le champ **&ResultType** du type d'opération identifié par le composant **opcode** pour le protocole en question.

7.6.6.4 Erreur OSI

```
OsiErr {OPERATION:Operations} ::= [3] IMPLICIT SEQUENCE {  
  invokeID      Invokeld,  
  errcode       OPERATION.&Errors.&errorCode ({Operations}),  
  error         OPERATION.&Errors.&ParameterType ({Operations} {@.errcode}) }
```

NOTE – L'erreur commence par l'étiquette [3] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

Le composant **invokeID** sera identique à celui qui est spécifié dans la demande **OsiRequest** correspondante.

Le composant **errcode** recevra le code de l'une des erreurs identifiées par le champ **ERRORS** de l'objet informationnel **OPERATION** identifié par l'élément **opcode** de la demande **OsiRequest** correspondante.

Le composant **error** contiendra les paramètres identifiés par le composant **errcode**.

7.6.6.5 Rejet OSI

Le type **OsiRej** permet de signaler une utilisation erronée des autres unités PDU d'annuaire. Il est défini comme suit:

```
OsiRej ::= [4] IMPLICIT SEQUENCE {  
  invokeld      Invokeld,  
  problem      CHOICE {  
    general      [0] GeneralProblem,  
    invoke       [1] InvokeProblem,  
    returnResult [2] ReturnResultProblem,  
    returnError  [3] ReturnErrorProblem } }
```

NOTE – Le rejet commence par l'étiquette [4] ainsi qu'il est défini dans la Rec. UIT-T X.880 | ISO/CEI 13712-1.

Le composant **invokeld** sera identique à celui qui est spécifié dans l'unité PDU qui doit être rejeté, sauf si le composant **invokeld** ne peut pas être déterminé; il prendra à la place la valeur **absent** (voir le § 6.2).

Le composant **problem** contiendra l'élément **Reject** ainsi qu'il est défini au § 7.6.6.6.

7.6.6.6 Problèmes de rejet

```
GeneralProblem ::= INTEGER {  
  unrecognizedPDU      (0),  
  mistypedPDU          (1),  
  badlyStructuredPDU  (2) }
```

L'élément **GeneralProblem** dénote un problème fondamental concernant la forme ou la structure d'une unité PDU d'annuaire. Les possibilités sont les suivantes:

- unrecognizedPDU** – L'étiquette de début de l'unité PDU indique qu'il ne s'agit pas d'un élément **OsiRequest**, **OsiResult**, **OsiError** ou **OsiReject**;
- mistypedPDU** – La structure de l'unité PDU n'est pas conforme à la définition appropriée;
- badlyStructuredPDU** – La structure de l'unité PDU ne peut pas être déterminée à partir de la syntaxe abstraite prévue.

```
InvokeProblem ::= INTEGER {  
  duplicateInvocation      (0),  
  unrecognizedOperation    (1),  
  mistypedArgument        (2),  
  resourceLimitation       (3),  
  releaseInProgress        (4) }
```

Un élément **InvokeProblem** indique qu'un certain composant d'une demande **OsiRequest** était erroné. Les possibilités sont les suivantes:

- duplicateInvocation** – Voir le § 7.6.6.2;
- unrecognizedOperation** – Le code d'opération n'est pas l'un de ceux qui sont définis pour le protocole d'annuaire en question;

- c) **mistypedArgument** – L'argument n'est pas constitué selon le champ **&ArgumentType** de l'opération identifiée par le composant **opcode**;
- d) **resourceLimitation** – L'opérateur prévu n'est pas disposé à effectuer l'opération à cause d'une limitation des ressources;
- e) **releaseInProgress** – L'opérateur prévu n'est pas disposé à effectuer l'opération car il est sur le point de libérer l'association d'applications.

```
ReturnResultProblem ::= INTEGER {
    unrecognizedInvocation      (0),
    resultResponseUnexpected    (1),
    mistypedResult              (2) }
```

Un élément **ReturnResultProblem** indique qu'un certain composant d'un élément **OsiResult** était erroné. Les possibilités sont les suivantes:

- a) **unrecognizedInvocation** – L'élément **Invokeld** n'était pas celui qui identifie une demande en cours;
- b) **resultResponseUnexpected** – Un résultat a été reçu pour une opération pour laquelle aucun résultat n'est défini;
NOTE 1 – Tous les types d'opération d'annuaire actuellement définis spécifient un résultat.
- c) **mistypedResult** – Le résultat n'est pas constitué selon le champ **&ResultType** de l'opération identifiée par le composant **opcode**.

```
ReturnErrorProblem ::= INTEGER {
    unrecognizedInvocation      (0),
    errorResponseUnexpected     (1),
    unrecognizedError           (2),
    unexpectedError             (3),
    mistypedParameter           (4) }
```

Un élément **ReturnErrorProblem** indique qu'un certain composant d'un élément **OsiError** était erroné. Les possibilités sont les suivantes:

- a) **unrecognizedInvocation** – L'élément **Invokeld** n'était pas celui qui identifie une demande en cours;
- b) **errorResponseUnexpected** – Une erreur a été reçue pour une opération pour laquelle aucune erreur n'est définie;
NOTE 2 – Tous les types d'opération d'annuaire actuellement définis spécifient une ou plusieurs erreurs.
- c) **unrecognizedError** – Une erreur a été reçue, qui n'était pas l'une de celles spécifiées par les présentes Spécifications d'annuaire;
- d) **unexpectedError** – Une erreur a été reçue, qui n'était pas l'une de celles identifiées par le champ **&Errors** de l'opération identifiée par le composant **opcode**;
- e) **mistypedParameter** – Le paramètre du résultat erroné n'est pas constitué selon le champ **&ParameterType** de l'erreur identifiée par le composant **errcode**.

7.6.7 Abandon de présentation

L'abandon peut être causé par un problème d'application (**ARU-PPDU**) ou un problème de couche Présentation (élément **ARP-PPDU**).

```
PresentationAbort ::= CHOICE {
    aru-ppdu ARU-PPDU,
    arp-ppdu ARP-PPDU }
```

7.6.7.1 Abandon d'application OSI

```
ARU-PPDU ::= CHOICE {
    normal-mode-parameters [0] IMPLICIT SEQUENCE {
        presentation-context-identifier-list [0] IMPLICIT Presentation-context-identifier-list,
        user-data CHOICE {
            fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
                presentation-context-identifier Presentation-context-identifier,
                presentation-data-values CHOICE {
                    single-ASN1-type [0] ABRT-apdu } } } }
```

Presentation-context-identif-ier-list ::=
SEQUENCE SIZE (1) OF SEQUENCE {
presentation-context-identif-ier Presentation-context-identif-ier,
transfer-syntax-name Transfer-syntax-name}

ABRT-apdu ::= [APPLICATION 4] IMPLICIT SEQUENCE {
abort-source ABRT-source }

ABRT-source ::= INTEGER {
acse-service-user (0),
acse-service-provider (1) }

L'élément **ABRT-PPDU** est utilisé si l'abandon est causé par des problèmes au niveau du protocole d'annuaire ou dans l'élément ACSE et non dans les éléments du protocole de présentation.

L'élément **ABRT-PPDU** est acheminé dans les données d'utilisateur de l'unité SPDU ABORT de la couche Session et le bit 2 du champ Déconnexion de transport sera défini tandis que le bit 3 sera réinitialisé (voir le § 8.3.8).

L'élément **ABRT-PPDU** peut entraîner une perte d'informations pendant le transfert.

La réception d'une unité **ABRT-PPDU** au sujet d'une association prenant en charge le DAP met fin à toutes les opérations de traitement d'une demande, sauf dans certaines conditions décrites dans la Rec. UIT-T X.518 | ISO/CEI 9594-4. Il en va de même pour le DSP. Il incombe à l'utilisateur de l'annuaire de confirmer que les modifications de la DIB qu'il a demandées ont bien été apportées.

La réception d'une unité **ABRT-PPDU** au sujet d'une association prenant en charge le DISP est décrite dans la Rec. UIT-T X.525 | ISO/CEI 9594-9.

La réception d'une unité **ABRT-PPDU** au sujet d'une association prenant en charge le DOP est décrite dans la Rec. UIT-T X.518 | ISO/CEI 9594-4.

7.6.7.1.1 Eléments du protocole de présentation

Le composant **normal-mode-parameters** comprend les sous-composants suivants:

- a) le sous-composant **presentation-context-identif-ier-list**, qui indique quelle syntaxe de transfert est utilisée pour les données d'utilisateur. Seules les informations ACSE sont incluses dans les données d'utilisateur. Elles comprendront un élément, qui est un type de séquence dont:
 - le sous-composant **presentation-context-identif-ier** recevra la même valeur que le sous-composant **presentation-context-identif-ier** de l'élément de la liste **presentation-context-definition-list** de la demande de rattachement qui a spécifié le nom de syntaxe abstraite ACSE;
 - le sous-composant **transfer-syntax-name** sera l'identificateur d'objet pour les règles de codage de base;
- e) le sous-composant **user-data**, qui comprend les éléments suivants:
 - le sous-composant **presentation-context-identif-ier**, qui recevra la même valeur que le sous-composant **presentation-context-identif-ier** de l'élément de la liste **presentation-context-definition-list** qui spécifie la syntaxe abstraite ACSE;
 - le sous-composant **presentation-data-values**, qui contiendra les éléments du protocole ACSE définis au § 7.6.7.1.2.

7.6.7.1.2 Eléments du protocole ACSE

Si l'abandon est causé au niveau du protocole d'annuaire, l'élément **ABRT-source** sera mis à la valeur **acse-service-user**. S'il est causé au niveau ACSE, l'élément **ABRT-source** sera mis à la valeur **acse-service-provider**.

NOTE – L'élément **ABRT-apdu** définie par la Rec. UIT-T X.227 | ISO/CEI 8650-1 comprend deux paramètres supplémentaires. Le paramètre **abort-diagnostics** ne sera pas présent si seul le noyau est utilisé, ce qui signifie que l'abandon est utilisé pour signaler une erreur de protocole. Le paramètre **user-information** n'est pas utilisé car les présentes Spécifications d'annuaire ne donnent pas d'informations sur l'abandon.

7.6.7.2 Abandon au niveau du protocole de présentation OSI

ARP-PPDU ::= SEQUENCE {
provider-reason [0] IMPLICIT Abort-reason OPTIONAL,
event-identif-ier [1] IMPLICIT Event-identif-ier OPTIONAL }

```

Abort-reason ::= INTEGER {
    reason-not-specified           (0),
    unrecognized-ppdu              (1),
    unexpected-ppdu                 (2),
    unexpected-session-service-primitive (3),
    unrecognized-ppdu-parameter    (4),
    unexpected-ppdu-parameter      (5),
    invalid-ppdu-parameter-value   (6) }

```

```

Event-identifiant ::= INTEGER {
    cp-PPDU           (0),
    cpa-PPDU          (1),
    cpr-PPDU          (2),
    aru-PPDU          (3),
    arp-PPDU          (4),
    td-PPDU           (7),
    s-release-indication (14),
    s-release-confirm  (15) }

```

L'élément **ARP-PDU** est utilisé si l'abandon est causé par des problèmes au niveau du protocole de présentation.

L'élément **ARP-PDU** est acheminé dans les données d'utilisateur de l'unité SPDU ABORT de la couche Session et le bit 2 du champ de déconnexion de transport sera défini tandis que le bit 3 sera réinitialisé (voir le § 8.3.8).

L'élément **ARP-PDU** peut entraîner une perte d'informations pendant le transfert.

La réception d'un élément **ARP-PDU** sera traitée comme indiqué pour l'élément **ARU-PDU** au § 7.6.7.1.

Le composant provider-reason peut prendre l'une des valeurs suivantes:

- a) **reason-not-specified**;
- b) **unrecognized-ppdu**, indiquant la réception d'une unité PPDU inconnue;

NOTE – Il peut s'agir d'une unité PPDU définie par la Rec. UIT-T X.226 | ISO/CEI 8823-1, mais qui n'est pas utilisée dans la présente Spécification d'annuaire. Certaines implémentations peuvent la signaler comme une unité **unexpected-ppdu**. Toutefois, il n'est pas obligatoire qu'une implémentation reconnaisse les unités PPDU non définies par la présente Spécification d'annuaire.
- c) **unexpected-ppdu**, indiquant la réception hors séquence d'une unité PPDU identifiée par l'élément **Event-identifiant**;
- d) **unexpected-session-service-primitive**, comme indiqué par l'élément **Event-identifiant**;
- e) **unrecognized-ppdu-parameter**, qui ne devrait pas être utilisé selon les règles d'extensibilité (voir la Note 1 au § 7.5);
- f) **unexpected-ppdu-parameter**, indiquant que même si un paramètre a été reconnu, il n'était pas attendu à ce moment ou à cet endroit en particulier, dans une unité PPDU identifiée par l'élément **Event-identifiant**;
- g) **invalid-ppdu-parameter-value**, indiquant qu'un paramètre avait une valeur non valable dans une unité PPDU identifiée par l'élément **Event-identifiant**.

L'élément **Event-identifiant** sera présent dans les conditions indiquées ci-dessus, sinon il sera absent.

- a) **s-release-indication** indique que la fonction du système homologue dans la couche Session a mis fin de manière inattendue à l'association d'applications;
- b) **s-release-confirm** indique que la fonction locale dans la couche Session a mis fin de manière inattendue à l'association d'applications.

8 Mappage du protocole d'annuaire sur les services OSI

8.1 Syntaxes abstraites et syntaxes de transfert

Faisant partie d'une association d'applications, les éléments du protocole utilisé doivent être convenus entre les deux parties, au moyen de la signalisation des syntaxes abstraites pertinentes dans le cadre de l'opération de rattachement. Des identificateurs d'objet, qui sont acheminés lors du rattachement, sont attribués aux syntaxes abstraites.

Les protocoles d'annuaire nécessitent chacun deux syntaxes abstraites, l'une correspondant à l'élément de protocole du protocole ACSE et l'autre au protocole d'annuaire lui-même (syntaxe abstraite d'annuaire).

NOTE – Les éléments de protocole ACSE font partie des spécifications d'annuaire des éditions 5 et suivantes. Cela étant, pour la compatibilité en amont, il est encore nécessaire de signaler deux syntaxes abstraites dans l'opération de rattachement.

Les identificateurs d'objet pour les syntaxes abstraites d'annuaire sont les suivants:

```
id-as-directoryAccessAS          OBJECT IDENTIFIER ::= {id-as 1}
id-as-directorySystemAS         OBJECT IDENTIFIER ::= {id-as 2}
id-as-directoryShadowAS        OBJECT IDENTIFIER ::= {id-as 3}
id-as-directoryOperationalBindingManagementAS OBJECT IDENTIFIER ::= {id-as 4}
```

La syntaxe abstraite ACSE est identifiée par:

```
id-acseAS                        OBJECT IDENTIFIER ::=
    { joint-iso-itu-t association-control(2) abstract-syntax(1) apdus(0) version(1) }
```

Les règles de codage ASN.1 pour une syntaxe abstraite sont signalées par un identificateur d'objet.

Les identificateurs d'objet pour les règles de codage ASN.1 sont définis dans la Rec. UIT-T X.690 | ISO/CEI 8825-1. Par commodité, l'identificateur d'objet pour les règles de codage de base est indiqué ici:

```
{ joint-iso-itu-t asn1(1) basic-encoding(1) }
```

8.2 Contextes d'application

Un *contexte d'application* est un ensemble de règles communes partagées par deux entités d'application afin de prendre en charge une association d'applications. Un contexte d'application est identifié par un nom *application-context-name* sous la forme d'un identificateur d'objet. Le nom de contexte d'application est signalé par l'opération de rattachement.

Un contexte d'application est défini à l'aide de la classe d'objets informationnels ASN.1 ci-après:

```
APPLICATION-CONTEXT ::= CLASS {
    &bind-operation          OPERATION,
    &Operations              OPERATION,
    &applicationContextName OBJECT IDENTIFIER UNIQUE }
WITH SYNTAX {
    BIND-OPERATION          &bind-operation
    OPERATIONS              &Operations
    APPLICATION CONTEXT NAME &applicationContextName }
```

Le champ **&bind-operation** permet d'indiquer le type d'opération de rattachement par lequel le contexte d'application est signalé.

Le champ **&Operations** sert à énumérer toutes les opérations d'annuaire pertinentes pour le contexte d'application.

Le champ **&applicationContextName** est utilisé pour indiquer l'identificateur d'objet pour le contexte d'application.

NOTE – Cette classe d'objets informationnels ASN.1 est une version abrégée de celle qui est définie par la Rec. UIT-T X.881 | ISO/CEI 13712-2 et elle est indiquée ici car certaines spécifications utilisent la référence d'objet informationnel ASN.1 au lieu de l'identificateur d'objet attribué.

8.2.1 Contexte d'application pour le protocole DAP

```
directoryAccessAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          directoryBind
    OPERATIONS              { read | compare | abandon | list | search | addEntry
                            | removeEntry | modifyEntry | modifyDN }
    APPLICATION CONTEXT NAME id-ac-directoryAccessAC }
```

Le contexte d'application **directoryAccessAC** est celui qui définit le protocole DAP. La prise en charge de ce contexte d'application passe obligatoirement par celle des syntaxes abstraites **id-acseAS** et **id-as-directoryAccessAS**.

Pour un agent DUA, cela implique la prise en charge d'au moins un type d'opération DAP, au-delà éventuellement du type d'opération Abandon. Pour un agent DSA, cela implique la prise en charge de toutes les opérations DAP.

8.2.2 Contexte d'application pour le protocole DSP

```
directorySystemAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSABind
    OPERATIONS              { chainedRead | chainedCompare | chainedAbandon
                            | chainedList | chainedSearch
                            | chainedAddEntry | chainedRemoveEntry
                            | chainedModifyEntry | chainedModifyDN }
    APPLICATION CONTEXT NAME id-ac-directorySystemAC }
```

Le contexte d'application **directorySystemAC** est celui qui définit le protocole DSP. La prise en charge de ce contexte d'application passe obligatoirement par celle des syntaxes abstraites **id-acseAS** et **id-as-directorySystemAS**.

Cela implique la prise en charge de toutes les opérations DSP indiquées ci-dessus.

8.2.3 Contextes d'application pour le protocole DISP

```
shadowSupplierInitiatedAC APPLICATION-CONTEXT ::= {
  BIND-OPERATION      dSAShadowBind
  OPERATIONS          { updateShadow
                       | coordinateShadowUpdate }
  APPLICATION CONTEXT NAME id-ac-shadowSupplierInitiatedAC }
```

Le contexte d'application **shadowSupplierInitiatedAC** est un contexte d'application DISP pour une association d'applications dans laquelle la mise à jour des informations miroir est initiée par le fournisseur, le mode opératoire étant synchrone.

NOTE – Les termes consommateur et fournisseur sont utilisés pour désigner deux rôles. Ces rôles correspondent respectivement aux termes consommateur d'informations miroir et fournisseur d'informations miroir utilisés dans la Rec. UIT-T X.525 | ISO/CEI 9594-9.

```
shadowConsumerInitiatedAC APPLICATION-CONTEXT ::= {
  BIND-OPERATION      dSAShadowBind
  OPERATIONS          { requestShadowUpdate
                       | updateShadow }
  APPLICATION CONTEXT NAME id-ac-shadowConsumerInitiatedAC }
```

Le contexte d'application **shadowConsumerInitiatedAC** est un contexte d'application DISP pour une association d'applications dans laquelle la mise à jour des informations miroir est initiée par le consommateur, le mode opératoire étant synchrone.

```
shadowSupplierInitiatedAsynchronousAC APPLICATION-CONTEXT ::= {
  BIND-OPERATION      dSAShadowBind
  OPERATIONS          { updateShadow
                       | coordinateShadowUpdate }
  APPLICATION CONTEXT NAME id-ac-shadowSupplierInitiatedAsynchronousAC }
```

Le contexte d'application **shadowSupplierInitiatedAsynchronousAC** est un contexte d'application DISP pour une association d'applications dans laquelle la mise à jour des informations miroir est initiée par le fournisseur, le mode opératoire étant asynchrone.

```
shadowConsumerInitiatedAsynchronousAC APPLICATION-CONTEXT ::= {
  BIND-OPERATION      dSAShadowBind
  OPERATIONS          { requestShadowUpdate
                       | updateShadow }
  APPLICATION CONTEXT NAME id-ac-shadowConsumerInitiatedAsynchronousAC }
```

Le contexte d'application **shadowConsumerInitiatedAsynchronousAC** est un contexte d'application DISP pour une association d'applications dans laquelle la mise à jour des informations miroir est initiée par le consommateur, le mode opératoire étant asynchrone.

8.2.4 Contexte d'application pour le protocole DOP

```
directoryOperationalBindingManagementAC APPLICATION-CONTEXT ::= {
  BIND-OPERATION      dSAOperationalBindingManagementBind
  OPERATIONS          { establishOperationalBinding
                       | modifyOperationalBinding
                       | terminateOperationalBinding}
  APPLICATION CONTEXT NAME id-ac-directoryOperationalBindingManagementAC }
```

Le contexte d'application **directoryOperationalBindingManagementAC** est celui qui définit le protocole DOP.

8.3 Spécification de la couche Session

8.3.1 Structure de l'unité de données protocolaire de Session (SPDU)

Une *unité de données protocolaire de Session* (SPDU, *session-protocol-data-unit*) comprend un *identificateur SPDU* (SI, *SPDU identifier*), aucun ou plusieurs paramètres identifiés chacun par un *identificateur de paramètre* (PI, *parameter identifier*) et éventuellement une valeur de paramètre (PV, *parameter value*). Les paramètres associés peuvent être groupés puis identifiés par un *identificateur de groupe de paramètres* (PGI, *parameter group identifier*).

La première partie d'une unité SPDU est le champ identificateur SPDU (SI), qui comprend un seul octet et dont la valeur est un chiffre binaire.

Un indicateur de longueur (LI, *length indicator*) est utilisé pour indiquer la longueur d'une unité SPDU, d'un paramètre ou d'un groupe de paramètres. Les champs LI indiquant des longueurs comprises entre 0 et 254 seront constitués d'un octet. Les champs LI indiquant des longueurs comprises entre 255 et 65 535 seront constitués de trois octets. Le premier octet sera ensuite codé 1111 1111, et les deuxième et troisième octets contiennent la longueur du champ de paramètre associé, les bits de poids fort se trouvant dans le premier de ces deux octets.

La valeur du champ LI ne comprend pas la valeur du champ lui-même ni d'aucun champ Informations d'utilisateur ultérieur.

NOTE – Parmi les unités SPDU utilisées par la présente Spécification d'annuaire, seule l'unité SPDU DATA TRANSFER a un champ Informations d'utilisateur.

Les bits d'un octet sont des nombres compris entre 1 à 8, le bit 1 étant le bit de poids faible.

La Figure 1 représente le cas où une unité SPDU n'a pas de paramètres. L'unité SPDU ABORT ACCEPT en est un exemple. Le champ LI a alors la valeur 0.

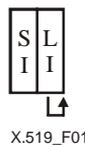


Figure 1 – Unité SPDU sans paramètres

La Figure 2 illustre le cas où une unité SPDU a deux paramètres distincts, chacun étant identifié par un identificateur de paramètre PI. Le premier champ LI indique la longueur de l'unité SPDU, à l'exclusion du champ SI et du champ LI lui-même. Les deux autres champs LI indiquent la longueur des paramètres.

Exemple: si la première valeur de paramètre PV est de 3 octets et que la deuxième valeur PV est de 4 octets, le premier champ LI a la valeur 11, le deuxième champ LI la valeur 3 et le troisième champ LI la valeur 4.

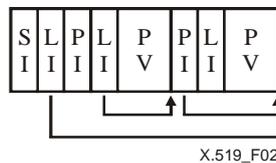


Figure 2 – Unité SPDU ayant des paramètres – Non groupés

La Figure 3 représente le cas où une unité SPDU a deux paramètres groupés ensemble, chacun étant identifié par un identificateur PI. Le groupe est identifié par un champ PGI. Le premier champ LI indique la longueur de l'unité SPDU, à l'exclusion du champ SI et du champ LI lui-même. Le champ LI suivant indique la longueur du groupe, à l'exclusion du champ PGI et du champ LI lui-même. Les deux autres champs LI indiquent la longueur des paramètres.

Exemple: si la première valeur de paramètre PV est de 5 octets et que la deuxième valeur PV est de 3 octets, le premier champ LI a la valeur 14, le deuxième champ LI la valeur 12, le troisième champ LI la valeur 5 et le quatrième champ la valeur 3.

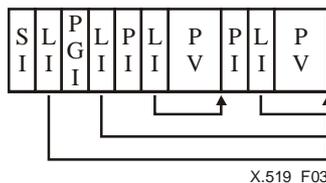


Figure 3 – Unité SPDU ayant des paramètres – Groupés

8.3.2 Taille et segmentation de l'unité TSDU

La taille maximale de l'unité de données de service de transport (TSDU, *transport-service-data-unit*) représente le nombre maximal d'octets à présenter à la couche Transport pour la transmission. Elle est négociée pendant l'établissement de l'association d'applications pour chaque sens de transmission (voir le § 8.3.3 et le § 8.3.4). Si une

unité PDU d'annuaire comprenant un préfixe de protocole de session dépasse cette valeur maximale, il est nécessaire de segmenter l'unité PDU d'annuaire en plusieurs unités SPDU.

Chaque processus d'application propose une taille maximale d'unité TSDU que l'initiateur est autorisé à envoyer. Le plus petit des deux nombres est utilisé. Une valeur égale à zéro est interprétée comme signifiant une taille illimitée. Si l'un ou l'autre des processus d'application propose la valeur zéro, l'initiateur n'enverra pas de données segmentées dans l'association d'applications.

Chaque processus d'application propose une taille maximale d'unité TSDU que le preneur est autorisé à envoyer. Le plus petit des deux nombres est utilisé. Une valeur égale à zéro est interprétée comme signifiant une taille illimitée. Si l'un ou l'autre des processus d'application propose la valeur zéro, le preneur n'enverra pas de données segmentées dans l'association d'applications.

8.3.3 Unité SPDU CONNECT de la couche Session

Tableau 1 – Paramètres de l'unité SPDU CONNECT

| Identificateur PGI | O/F | Code | Identificateur PI | O/F | Code | Longueur |
|--------------------------------------|-----|------|---|-----|------|----------------------|
| Identificateur de connexion | F | 1 | Référence de l'utilisateur appelant du service de session | F | 10 | 64 octets maximum |
| | | | Référence commune | F | 11 | 64 octets maximum |
| | | | Informations de référence additionnelles | F | 12 | 4 octets maximum |
| | | | Options de protocole | O | 19 | 1 octet |
| Item de connexion ou d'acceptation | O | 5 | Taille maximale d'unité TSDU | F | 21 | 4 octets |
| | | | Numéro de version | O | 22 | 1 octet |
| | | | Caractéristiques de session proposées par l'utilisateur | O | 20 | 2 octets |
| | | | Sélecteur de session appelant | F | 51 | 16 octets maximum |
| | | | Sélecteur de session appelé | F | 52 | 16 octets maximum |
| Données d'utilisateur | O | 193 | | | | 512 octets maximum |
| Données d'utilisateur en mode étendu | O | 194 | | | | 10240 octets maximum |

O: obligatoire

F: facultatif

Le champ SI field recevra la valeur 13 ('0D'H).

L'identificateur de connexion est un groupe de paramètres facultatifs rempli au moyen de données créées localement qui permettent d'identifier cette connexion de session. Il peut avoir les paramètres facultatifs suivants:

- a) référence de l'utilisateur appelant du service de session, c'est-à-dire une référence choisie par l'initiateur;

NOTE 1 – Un utilisateur du service de session est, conformément à la Rec. UIT-T X.200 | ISO/CEI 7498-1, une fonction de la couche Présentation utilisant le service de session.
- b) référence commune;
- c) informations de référence additionnelles.

Le paramètre Item de connexion ou d'acceptation est un groupe de paramètres obligatoires comprenant les paramètres suivants:

- a) options de protocole – La concaténation étendue, telle qu'elle est définie par la Rec. UIT-T X.225 | ISO/CEI 8327-1, n'est pas pertinente pour la présente Spécification d'annuaire. Ce champ sera absent ou sa valeur sera mise à '00'H (valeur par défaut). Pour autant, une implémentation devrait accepter la valeur '01'H;
- b) le champ de paramètre Taille maximale d'unité TSDU devrait être présent si une taille maximale d'unité TSDU est proposée. Si le champ de paramètre Taille maximale d'unité TSDU est présent:
 - i) les deux premiers octets du champ de paramètre contiendront la taille maximale d'unité TSDU proposée, exprimée en octets, dans le sens initiateur vers preneur, codée sous la forme d'un nombre binaire, dont le premier des deux octets est la partie de poids fort du nombre;

ISO/CEI 9594-5:2005 (F)

- ii) les deux octets suivants du champ de paramètre contiendront la taille maximale d'unité TSDU proposée, exprimée en octets, dans le sens preneur vers initiateur, codée sous la forme d'un nombre binaire, dont le premier des deux octets est la partie de poids fort du nombre.

Si ce paramètre est absent, la taille maximale d'unité TSDU n'est pas limitée. Si l'une ou l'autre paire d'octets a la valeur zéro, la taille de l'unité TSDU n'est pas limitée dans le sens de transfert associé à cette paire d'octets;

- c) numéro de version – Ce champ recevra la valeur '02'H.

Le champ Caractéristiques de session proposées par l'utilisateur sera mis à '0002'H.

NOTE 2 – Seule l'unité fonctionnelle de transfert duplex de session est utilisée par l'annuaire.

Le champ Sélecteur de session appelant contiendra la valeur du sélecteur de session de l'initiateur, si un sélecteur de session est attribué, et aura une valeur obtenue à partir des informations locales. Si l'initiateur n'a pas de sélecteur de session dans son adresse de présentation, ce champ sera absent.

Le champ Sélecteur de session appelé sera présent si on sait qu'il fait partie de l'adresse du système récepteur. Dans le cas contraire, il sera absent. S'il est présent, la valeur sera obtenue à partir:

- d'informations renvoyées dans un élément **ContinuationReference** à la suite d'une opération d'annuaire précédente;
- d'informations locales.

Tant le paramètre Données d'utilisateur que le paramètre Données d'utilisateur en mode étendu seront pris en charge, mais seuls ces deux paramètres peuvent être utilisés pour une instance de communications. Si les données d'utilisateur à inclure comprennent 512 octets ou moins, le paramètre Données d'utilisateur sera utilisé. Si les données d'utilisateur comprennent plus de 512 octets, le paramètre Données d'utilisateur en mode étendu sera utilisé et le paramètre Données d'utilisateur ne sera pas utilisé.

La demande de rattachement OSI est acheminée en tant que données d'utilisateur de l'unité SPDU CONNECT de la couche Session (voir le § 7.6.1). Elle ne dépassera pas 10 240 octets.

8.3.4 Unité SPDU ACCEPT de la couche Session

Tableau 2 – Paramètres de l'unité SPDU ACCEPT

| Identificateur PGI | O/F | Code | Identificateur PI | O/F | Code | Longueur |
|------------------------------------|-----|------|---|-----|------|-------------------|
| Identificateur de connexion | F | 1 | Référence de l'utilisateur appelé du service de session | F | 9 | 64 octets maximum |
| | | | Référence commune | F | 11 | 64 octets maximum |
| | | | Informations de référence additionnelles | F | 12 | 4 octets maximum |
| Item de connexion ou d'acceptation | F | 5 | Options de protocole | O | 19 | 1 octet |
| | | | Taille maximale d'unité TSDU | F | 21 | 4 octets |
| | | | Numéro de version | O | 22 | 1 octet |
| | | | Caractéristiques de session proposées par l'utilisateur | O | 20 | 2 octets |
| | | | Sélecteur de session appelant | F | 51 | 16 octets maximum |
| | | | Sélecteur de session appelé | F | 52 | 16 octets maximum |
| Données d'utilisateur | O | 193 | | | | |

O: obligatoire

F: facultatif

Le champ SI recevra la valeur 14 ('0E'H).

Le paramètre Identificateur de connexion est un groupe de paramètres facultatifs rempli au moyen de données créées localement qui permettent d'identifier cette connexion de session. Il peut avoir les paramètres facultatifs suivants:

- a) référence de l'utilisateur appelé du service de session;
- b) référence commune;
- c) informations de référence additionnelles.

Le paramètre Item de connexion ou d'acceptation est un groupe de paramètres obligatoires comprenant les paramètres suivants:

- Options de protocole – Ce champ sera absent ou la valeur sera mise à '00'H (valeur par défaut). Pour autant, une implémentation devrait accepter la valeur '01'H.
- Taille maximale d'unité TSDU – Ce champ sera présent si une taille maximale d'unité TSDU est proposée par le preneur. Le codage et la valeur par défaut de ce champ sont identiques à ceux de l'unité SPDU CONNECT (voir le § 8.3.3).
- Numéro de version – Ce champ recevra la valeur '02'H.

Le champ Caractéristiques de session proposées par l'utilisateur sera mis à '0002'H.

Le champ Sélecteur de session appelant sera présent si le champ correspondant était présent dans l'unité SPDU CONNECT et aura alors la même valeur que ce champ, sinon il sera absent.

S'il est fourni, le champ Sélecteur de session appelé aura une valeur obtenue à partir d'informations locales.

Le paramètre Données d'utilisateur sera pris en charge. Il sera utilisé pour acheminer l'élément **OsiBindResult** (voir le § 7.6.2).

La longueur de l'unité SPDU ACCEPT ne dépassera pas 65 539 octets.

8.3.5 Unité SPDU REFUSE de la couche Session

L'unité SPDU REFUSE de la couche Session est utilisée par le preneur pour refuser une association d'applications.

Tableau 3 – Paramètres de l'unité SPDU REFUSE

| PGI | O/F | Code | PI | O/F | Code | Longueur |
|-----------------------------|-----|------|---|-----|------|-------------------|
| Identificateur de connexion | F | 1 | Référence de l'utilisateur appelé du service de session | F | 9 | 64 octets maximum |
| | | | Référence commune | F | 11 | 64 octets maximum |
| | | | Informations de référence additionnelles | F | 12 | 4 octets maximum |
| | | | Déconnexion de transport | F | 17 | 1 octet |
| | | | Caractéristiques de session proposées par l'utilisateur | F | 20 | 2 octets |
| | | | Numéro de version | F | 22 | 1 octet |
| | | | Code raison | O | 50 | Voir ci-après |

O: obligatoire

F: facultatif

Le champ SI recevra la valeur 12 ('0C'H).

Le paramètre Identificateur de connexion est un groupe de paramètres facultatifs rempli au moyen de données créées localement qui permettent d'identifier cette connexion de session. Il peut avoir les paramètres facultatifs suivants:

- référence de l'utilisateur appelé du service de session;
- référence commune;
- informations de référence additionnelles.

Le champ Déconnexion de transport indique si la connexion de transport sous-jacente sera libérée ou conservée. Son codage sera le suivant:

- bit 1 = 0: la connexion de transport sera conservée;
- bit 1 = 1: la connexion de transport sera libérée.

Les bits 2 à 8 sont réservés.

Si ce champ est absent, la connexion de transport sera libérée.

Le champ Caractéristiques de session proposées par l'utilisateur ne sera pas présent si le champ Code raison n'est pas mis à 2. Si le champ Code raison est mis à 2, ce champ sera présent et mis à '0002'H.

Le champ Code raison contiendra un code raison dans le premier octet. Selon la valeur de ce premier octet, des octets supplémentaires peuvent être utilisés. Les valeurs suivantes sont définies pour le premier octet:

ISO/CEI 9594-5:2005 (F)

- a) 0: rejet par l'utilisateur appelé du service de session; raison non spécifiée;
- b) 1: rejet par l'utilisateur appelé du service de session en raison d'un encombrement temporaire;
- c) 2: rejet par l'utilisateur appelé du service de session. Les octets suivants peuvent être utilisés pour des données d'utilisateur jusqu'à une longueur de 512 octets si la version 1 du protocole a été sélectionnée et jusqu'à une longueur telle que la longueur totale (y compris l'identificateur SI et l'indicateur LI) de l'unité SPDU ne dépasse pas 65 539 octets si la version 2 du protocole a été sélectionnée;
- d) * 128 + 1: sélecteur de session inconnu;
- e) * 128 + 2: utilisateur du service de session non relié au point SSAP;
- f) 128 + 3: engorgement de la machine de protocole de session au moment de la connexion;
- g) * 128 + 4: versions de protocole proposées non prises en charge;
- h) * 128 + 5: rejet par la machine de protocole de session; raison non spécifiée;
- i) * 128 + 6: rejet par la machine de protocole de session; restriction imposée à l'implémentation indiquée dans la déclaration PICS.

NOTE – Les raisons signalées par un astérisque (*) peuvent être considérées comme persistantes et les autres comme transitoires.

Toutes les autres valeurs sont réservées.

8.3.6 Unité SPDU FINISH de la couche Session

Tableau 4 – Paramètres de l'unité SPDU FINISH

| Identificateur PGI | O/F | Code | Identificateur PI | O/F | Code | Longueur |
|-----------------------|-----|------|--------------------------|-----|------|----------|
| | | | Déconnexion de transport | F | 17 | |
| Données d'utilisateur | O | 193 | | | | |

O: obligatoire

F: facultatif

Le champ SI recevra la valeur 9.

Le champ Déconnexion de transport indique si la connexion de transport sous-jacente sera libérée ou conservée. Son codage sera le suivant:

- a) bit 1 = 0: la connexion de transport est conservée;
- b) bit 1 = 1: la connexion de transport est libérée.

Si ce champ est absent, la connexion de transport sera libérée.

Le champ Données d'utilisateur contiendra l'élément **OsiUnbind** (voir le § 7.6.4). Sa longueur est limitée de sorte que la longueur totale (y compris les champs SI et LI) de l'unité SPDU ne dépasse pas 65 539 octets.

NOTE – Le paramètre Item de délimitation défini pour l'unité SPDU FINISH dans la Rec. UIT-T X.225 | ISO/CEI 8327-1 n'est pas pertinent car seule une quantité limitée de données d'utilisateur sera transférée.

8.3.7 Unité SPDU DISCONNECT de la couche Session

Tableau 5 – Paramètres de l'unité SPDU DISCONNECT

| Identificateur PGI | O/F | Code | Identificateur PI | O/F | Code | Longueur |
|-----------------------|-----|------|-------------------|-----|------|----------|
| Données d'utilisateur | O | 193 | | | | |

O: obligatoire

F: facultatif

Le champ SI recevra la valeur 10.

Le champ Données d'utilisateur contiendra l'élément **OsiUnbindResult** (voir le § 7.6.5). Sa longueur est limitée de sorte que la longueur totale (y compris les champs SI et LI) de l'unité SPDU ne dépasse pas 65 539 octets.

NOTE – Le paramètre Item de délimitation défini pour l'unité SPDU FINISH dans la Rec. UIT-T X.225 | ISO/CEI 8327-1 n'est pas pertinent car seule une quantité limitée de données d'utilisateur sera transférée.

8.3.8 Unité SPDU ABORT de la couche Session

Tableau 6 – Paramètres de l'unité SPDU ABORT

| Identificateur PGI | O/F | Code | Identificateur PI | O/F | Code | Longueur |
|-----------------------|-----|------|---------------------------------|-----|------|------------------|
| | | | Déconnexion de transport | O | 17 | |
| | | | Copie des valeurs paramétriques | F | 49 | 9 octets maximum |
| Données d'utilisateur | F | 193 | | | | |

O: obligatoire

F: facultatif

Le champ SI recevra la valeur 25.

Le champ Déconnexion de transport indique si la connexion de transport doit être conservée ou non, en mentionnant éventuellement un code raison facultatif. Le codage de ce champ est le suivant:

- a) bit 1 = 0: la connexion de transport est conservée;
- b) bit 1 = 1: la connexion de transport est libérée;
- c) bit 2 = 1: coupure par l'utilisateur;
- d) bit 3 = 1: erreur de protocole;
- e) bit 4 = 1: raison non spécifiée;
- f) bit 5 = 1: la restriction imposée à l'implémentation est indiquée dans la déclaration de conformité d'une implémentation de protocole.

Les bits 6 à 8 sont réservés.

Le champ Copie des valeurs paramétriques n'est présent que si le champ Déconnexion de transport indique une erreur de protocole; la valeur et la signification du contenu de ce champ dépendent de l'implémentation.

Le champ Données d'utilisateur ne doit être présent que si le champ Déconnexion de transport indique une coupure par l'utilisateur et contiendra l'unité **ARU-PPDU** (voir le § 7.6.7.1) ou **ARP-PPDU** (voir le § 7.6.7.2). Sa longueur est limitée de sorte que la longueur totale (y compris les champs SI et LI) de l'unité SPDU ne dépasse pas 65 539 octets.

NOTE – Le paramètre Item de délimitation défini pour l'unité SPDU FINISH dans la Rec. UIT-T X.225 | ISO/CEI 8327-1 n'est pas pertinent car seule une quantité limitée de données d'utilisateur sera transférée.

8.3.9 Unité SPDU ABORT ACCEPT de la couche Session

Le champ SI recevra la valeur 26.

Aucun champ de paramètre n'est associé à cette unité SPDU.

8.3.10 Unité SPDU DATA TRANSFER de la couche Session

L'unité SPDU DATA Transfer de la couche Session comprend en principe deux unités SPDU concaténées, dont la première est appelée unité SPDU GIVE TOKEN. Sous la forme utilisée dans la présente Spécification d'annuaire, elle comprend uniquement le champ SI de valeur 1, le champ de longueur ayant la valeur zéro.

NOTE – La Rec. UIT-T X.225 | ISO/CEI 8327-1 définit la concaténation de base et la concaténation étendue. Cette dernière n'est pas utilisée dans la présente Spécification d'annuaire. La concaténation de base est pertinente uniquement pour l'unité SPDU DATA et le Tableau 7 de la Rec. UIT-T X.225 | ISO/CEI 8327-1 indique que l'unité SPDU DATA TRANSFER sera concaténée avec l'unité SPDU GIVE TOKEN. Comme nous utilisons uniquement l'unité fonctionnelle duplex intégral, le paramètre Item de jetons n'est pas nécessaire, ni les données d'utilisateur.

Tableau 7 – Paramètres de l'unité SPDU DATA TRANSFER

| Identificateur PGI | O/F | Code | Identificateur PI | O/F | Code | Longueur |
|----------------------------------|-----|------|----------------------|-----|------|----------|
| | | | Item de délimitation | F | 25 | 1 octet |
| Champ Informations d'utilisateur | | | | | | |

O: obligatoire

F: facultatif

Le champ SI de l'unité SPDU DATA TRANSFER recevra aussi la valeur 1.

ISO/CEI 9594-5:2005 (F)

Le champ Informations d'utilisateur contient l'unité PDU d'annuaire complète ou une partie de cette unité. Le champ SI ne comprend pas le champ Informations d'utilisateur.

Le champ Item de délimitation, s'il est présent, indiquera si cette unité SPDU est ou non le début ou la fin de l'unité PDU d'annuaire. Il sera présent si la segmentation peut être utilisée. Il ne sera pas présent si la segmentation ne doit pas être utilisée. Son codage sera le suivant:

- a) bit 1 = 1: début de l'unité PDU d'annuaire;
bit 1 = 0: pas le début de l'unité PDU d'annuaire;
- b) bit 2 = 1: fin de l'unité PDU d'annuaire;
bit 2 = 0: pas la fin de l'unité PDU d'annuaire.

Les bits 3 à 8 sont réservés.

Si ce champ n'est pas présent, la segmentation n'est pas autorisée et cette unité SPDU contient une unité PDU d'annuaire complète.

Exemple de codage:

Si le champ Item de délimitation n'est pas inclus, le codage des unités SPDU concaténées serait '01 00 01 00'H.

Si le champ susmentionné est inclus et que l'unité SPDU contient une unité PDU d'annuaire complète, le codage des unités SPDU concaténées serait '01 00 01 03 19 01 03'H.

8.4 Utilisation du service de transport

Avant de pouvoir établir une association d'applications, il faut établir une connexion de transport ainsi qu'il est défini dans la Rec. UIT-T X.214 | ISO/CEI 8072.

Seul l'initiateur d'une connexion de transport est autorisé à initier une association d'applications.

NOTE – Cette restriction est spécifiée au § 6.1.4 de la Rec. UIT-T X.225 | ISO/CEI 8327-1.

Toutes les unités SPDU de la couche Session sont mappées avec la demande T-DATA et l'indication T-DATA.

Lorsqu'une association d'applications est refusée ou a été connectée avec succès, puis déconnectée, au moyen d'un abandon ou d'une libération sans problème, la connexion de transport sous-jacente peut être déconnectée ou réutilisée.

La connexion de transport peut être conservée pour être réutilisée à condition que:

- a) le processus d'application qui a établi la connexion de transport demande la conservation de cette connexion au moyen d'un paramètre de l'unité SPDU ABORT ou FINISH;
- b) le processus d'application qui a établi la connexion de transport reçoit une unité SPDU REFUSE ou ABORT qui indique au moyen d'un paramètre que la connexion de transport doit être conservée.

Pour éviter un conflit au sujet d'une connexion de transport conservée, seul l'initiateur de cette connexion peut la réutiliser en envoyant une demande de rattachement pour établir une nouvelle association d'applications.

Le flux de transport exprès n'est pas utilisé.

9 Protocole IDM

Le présent paragraphe définit le protocole Internet directement mappé (IDM, *Internet directly mapped protocol*), qui est un mappage direct des éléments du service de demande-réponse sur le protocole Internet TCP/IP, omettant les couches ACSE, Présentation, Session et Transport du modèle OSI. Ce protocole est délibérément minimal et est conçu pour simplifier l'implémentation. Il est en mode connexion et entièrement asynchrone.

Ce protocole fait appel à un certain nombre d'unités de données protocolaires effectuant le transfert des messages d'association, de demande, de réponse et d'erreur.

9.1 Unités IDM-PDU

Les messages du protocole Internet directement mappés sont acheminés sur une connexion TCP/IP sous la forme d'unités de données protocolaires appelées IDM-PDU et sont mappés sur une connexion TCP/IP comme spécifié au § 9.6. La connexion TCP/IP peut facultativement être protégée au moyen d'une connexion TLS, comme spécifié au § 9.8. La connexion TLS est définie dans les documents RFC 2246 et RFC 3546. La définition ASN.1 de ces unités est donnée ci-dessous.

```

IDM-PDU {IDM-PROTOCOL:protocol} ::= CHOICE {
    bind                [0] IdmBind{ {protocol} },
    bindResult          [1] IdmBindResult{ {protocol} },
    bindError           [2] IdmBindError{ {protocol} },
    request             [3] Request{ {protocol.&Operations} },
    result              [4] IdmResult{ {protocol.&Operations} },
    error               [5] Error{ {protocol.&Operations} },
    reject              [6] IdmReject,
    unbind              [7] Unbind,
    abort               [8] Abort,
    startTLS            [9] StartTLS,
    tLSResponse         [10] TLSResponse }

IdmBind {IDM-PROTOCOL:Protocols} ::= SEQUENCE {
    protocolID          IDM-PROTOCOL.&id ({Protocols}),
    callingAETitle      [0] GeneralName OPTIONAL,
    calledAETitle       [1] GeneralName OPTIONAL,
    argument            [2] IDM-PROTOCOL.&bind-operation.&ArgumentType
                        ({Protocols} {@protocolID}) }

IdmBindResult {IDM-PROTOCOL:Protocols} ::= SEQUENCE {
    protocolID          IDM-PROTOCOL.&id ({Protocols}),
    respondingAETitle   [0] GeneralName OPTIONAL,
    result              [1] IDM-PROTOCOL.&bind-operation.&ResultType
                        ({Protocols} {@protocolID}) }

IdmBindError {IDM-PROTOCOL:Protocols} ::= SEQUENCE {
    protocolID          IDM-PROTOCOL.&id ({Protocols}),
    errcode             IDM-PROTOCOL.&bind-operation.&Errors.&errorCode
                        ({Protocols} {@protocolID}),
    respondingAETitle   [0] GeneralName OPTIONAL,
    aETitleError        ENUMERATED {
                        callingAETitleNotAccepted (0),
                        calledAETitleNotRecognized (1) } OPTIONAL,
    error               [1] IDM-PROTOCOL.&bind-operation.&Errors.&ParameterType
                        ({Protocols} {@protocolID, @errcode}) }

Request {OPERATION:Operations} ::= SEQUENCE {
    invokeID           INTEGER,
    opcode             OPERATION.&operationCode ({Operations}),
    argument            OPERATION.&ArgumentType ({Operations} {@opcode}) }

IdmResult {OPERATION:Operations} ::= SEQUENCE {
    invokeID           INTEGER,
    opcode             OPERATION.&operationCode ({Operations}),
    result             OPERATION.&ResultType ({Operations} {@opcode}) }

Error {OPERATION:Operations} ::= SEQUENCE {
    invokeID           INTEGER,
    errcode            OPERATION.&Errors.&errorCode ({Operations}),
    error              OPERATION.&Errors.&ParameterType
                        ({Operations} {@errcode}) }

IdmReject ::= SEQUENCE {
    invokeID           INTEGER,
    reason             ENUMERATED {
        mistypedPDU (0),
        duplicateInvokeIDRequest (1),
        unsupportedOperationRequest (2),
        unknownOperationRequest (3),
        mistypedArgumentRequest (4),
        resourceLimitationRequest (5),
        unknownInvokeIDResult (6),
        mistypedResultRequest (7),
        unknownInvokeIDError (8),
        unknownError (9),
        mistypedParameterError (10) } }

Unbind ::= NULL

```

```

Abort ::= ENUMERATED {
    mistypedPDU          (0),
    unboundRequest      (1),
    invalidPDU           (2),
    resourceLimitation  (3),
    connectionFailed    (4),
    invalidProtocol      (5),
    reasonNotSpecified  (6) }

```

StartTLS ::= NULL

```

TLSResponse ::= ENUMERATED {
    success              (0),
    operationsError     (1),
    protocolError       (2),
    unavailable         (3) }

```

Une unité PDU de type **bind** est envoyée pour demander une association entre l'expéditeur et le répondeur. L'unité **protocolID** désigne le protocole **IDM-PROTOCOL** à utiliser (voir § 9.4). L'unité **argument** est une valeur pour le champ **ARGUMENT** de l'opération **BIND-OPERATION** du protocole identifié. L'unité **callingAETitle** est le nom de l'entité applicative locale qui envoie l'unité PDU **bind**. L'unité **calledAETitle** est le nom de l'entité applicative distante à laquelle l'unité PDU **bind** est envoyée.

Une unité **bindResult** est renvoyée en réponse à une demande d'association effective. L'unité **protocolID** est la valeur déjà envoyée dans l'unité PDU **bind** correspondante. L'unité **result** est une valeur du champ **RESULT** de l'opération **BIND-OPERATION** du protocole identifié. L'unité **respondingAETitle** est le nom de l'entité applicative distante qui a envoyé l'unité **bindResult**.

Une unité PDU **bindError** est renvoyée en réponse à une demande d'association ineffective. L'unité **protocolID** est la valeur déjà envoyée dans l'unité PDU **bind** correspondante. L'unité **errcode** est le code de l'une des erreurs énumérées dans le champ **ERRORS** de l'opération **BIND-OPERATION** du protocole identifié. L'unité **error** est une valeur du champ **PARAMETER** de l'**ERROR** identifiée par l'unité **errcode**. L'unité **respondingAETitle** est le nom de l'entité applicative distante qui a envoyé l'unité **bindError**. L'unité **aETitleError** est mise à la valeur **callingAETitleNotAccepted** si une unité PDU **bind** est reçue et si l'unité **callingAETitle** fournie n'est pas acceptable par le système appelé. L'unité **aETitleError** est mise à la valeur **calledAETitleNotRecognized** si une unité PDU **bind** est reçue et si l'entité applicative distante connaît l'entité applicative qui effectue l'association, mais n'accepte pas le titre **calledAETitle** envoyé dans l'unité PDU **bind** pour désigner son propre nom.

Une unité PDU **request** est envoyée pour demander l'exécution d'une opération. L'unité **invokeID** désigne une demande particulière et les réponses qui lui sont associées. C'est un entier positif qui est choisi de façon à être différent de la valeur envoyée dans toute demande antérieure sur cette connexion TCP/IP. L'unité **opcode** est le code d'une des opérations énumérées dans le champ **OPERATIONS** du protocole choisi. L'unité **argument** est une valeur du champ **ARGUMENT** de l'opération désignée par l'unité **opcode**.

NOTE – L'élément **InvokeID** des systèmes X.500 est, du point de vue sémantique, équivalent à l'élément **messageID** des systèmes LDAP, comme défini au § 4.1.1.1 du document RFC 2251.

Une unité PDU **result** est renvoyée en réponse à une demande d'opération effective. Les unités **invokeID** et **opcode** sont les valeurs déjà envoyées dans l'unité PDU de la demande à laquelle cette unité PDU est une réponse. L'unité **result** est une valeur du champ **RESULT** de l'**OPERATION** identifiée par l'unité **opcode**.

Une unité PDU **error** est renvoyée en réponse à une demande d'opération ineffective. L'unité **invokeID** a la valeur déjà envoyée dans l'unité PDU de demande à laquelle cette unité PDU est une réponse. L'unité **errcode** est le code de l'une des erreurs énumérées dans le champ **ERRORS** de l'opération indiquée dans l'unité PDU de demande. L'unité **error** est une valeur du champ **PARAMETER** de l'**ERROR** identifiée par l'unité **errcode**.

Une unité PDU **reject** est renvoyée en réponse à une erreur de protocole détectée dans une unité PDU **request**, **result** ou **error** qui a été reçue et à partir de laquelle un identificateur d'invocation peut être récupéré. L'unité **invokeID** est l'identificateur d'invocation de l'unité PDU reçue en erreur. L'unité **reason** est un code d'entier pour l'erreur, comme décrit en § 9.4.

Une unité PDU **unbind** est envoyée afin de clore une association de façon ordonnée, comme décrit au § 9.2. Elle n'a pas de paramètres.

Une unité PDU **startTLS** est envoyée à l'initiateur de la connexion TCP/IP pour demander l'établissement d'une connexion TLS.

Une unité PDU **tlsResponse** PDU est envoyée par le preneur de la connexion TCP/IP après la réception d'une unité PDU **startTLS**. Une réponse **tlsResponse** de type **success** indique que le preneur est disposé et apte à négocier une

connexion TLS. Une réponse **tLSResponse** de type autre que **success** indique que le preneur n'est pas disposé ou apte à négocier une connexion TLS. Le preneur renverra la réponse **operationsError** s'il détecte tout séquençement incorrect des opérations, par exemple la réception d'une unité PDU **startTLS** après l'établissement de la connexion TLS. Il renverra la réponse **protocolError** s'il ne prend pas en charge la connexion TLS, soit de par sa conception soit de par sa configuration. Le preneur renverra la réponse **unavailable** s'il prend en charge la connexion TLS mais ne peut pas l'établir au moment de la demande **startTLS**.

9.2 Exigences de séquençement

9.2.1 Rattachement

L'initiateur de la connexion TCP/IP doit envoyer l'unité PDU **bind** au répondeur. Celui-ci doit répondre par l'envoi de l'unité PDU **bindResponse** ou **bindError**. Une fois que l'unité PDU **bindResponse** a été reçue, une *association* est considérée comme établie entre l'initiateur et le répondeur.

L'initiateur doit envoyer une unité PDU **bind** avant d'envoyer des unités PDU **request**. Il peut envoyer des unités PDU **request** après avoir envoyé l'unité PDU **bind** mais avant de recevoir une unité **bindResponse** ou **bindError**. Le répondeur doit traiter une unité PDU **bind** reçue et y répondre avant de traiter des unités PDU **request** reçues et d'y répondre.

Si le protocole permet au répondeur d'émettre des demandes, celui-ci peut émettre de telles demandes dès qu'il a envoyé une unité PDU **bindResponse**. L'initiateur doit traiter l'unité **bindResponse** avant de répondre aux unités PDU **request** reçues.

Si une unité **bindError** est reçue, l'initiateur peut choisir soit d'essayer un autre rattachement par l'envoi d'une nouvelle unité PDU de rattachement soit de fermer la connexion TCP/IP.

Si les deux entités applicatives utilisent les informations de titre **AETitle** de l'unité PDU **bind**, une unité PDU **bindError**, dont le paramètre **aETitleError** est mis à **callingAETitleNotAccepted** ou à **calledAETitleNotRecognized**, peut être reçue en tant que réponse à une unité PDU **bind**.

9.2.2 Détachement

Lorsque le protocole DAP est utilisé, seul l'initiateur du rattachement doit envoyer une unité PDU **unbind**. Avec tout autre protocole, l'initiateur ou le répondeur peut envoyer une unité PDU **unbind**. Celle-ci est destructive dans la mesure où les résultats de toutes les opérations en cours sont perdus (indéfinis). Pour éviter la perte de données, l'initiateur ne devrait lancer le détachement que lorsque toutes les demandes ont été satisfaites.

Soit l'initiateur soit le répondeur peut clore la connexion TCP/IP sous-jacente quand il le souhaite. Toutes les demandes en suspens sont perdues.

9.2.3 Demandes et réponses

Une unité PDU **request** peut être envoyée à tout moment après l'envoi d'une unité PDU **bind** ou **bindResult**, afin de demander au destinataire de cette unité PDU d'effectuer l'opération indiquée. Le destinataire de l'unité PDU **request** doit répondre par une unité PDU **result**, **error** ou **reject**.

Les demandes sont asynchrones et rien ne garantit que l'ordre des réponses sera le même que celui des demandes.

Le récepteur d'une réponse doit utiliser l'identificateur d'invocation comme indicateur primaire de la demande à laquelle la réponse se rapporte. Il doit rejeter la réponse si l'identificateur d'invocation est erroné.

9.2.4 Rejets

L'unité PDU **reject** doit être utilisée pour indiquer qu'un problème a été rencontré lors du traitement d'une unité PDU **request**, **result** ou **error**.

Si une autre erreur de protocole se produit ou si l'identificateur d'invocation ne peut pas être déterminé, la connexion doit être close.

9.3 Protocoles

Les protocoles à utiliser à l'intérieur du protocole IDM sont définis par l'utilisation de la classe d'objets informationnels **IDM-PROTOCOL**, qui est définie comme suit:

```
IDM-PROTOCOL ::= CLASS {
    &bind-operation OPERATION,
    &Operations OPERATION,
    &id OBJECT IDENTIFIER UNIQUE }
```

```

WITH SYNTAX {
    BIND-OPERATION      &bind-operation
    OPERATIONS          &Operations
    ID                  &id }

```

Chaque instance d'une classe **IDM-PROTOCOL** définit l'opération de rattachement et les opérations de demande/réponse à utiliser dans le protocole IDM. Le champ **bindOperation** définit l'opération à utiliser pour le rattachement; le champ **ARGUMENT** de cette opération est utilisé avec l'unité PDU **bind** qui signale le protocole; le champ **RESULT** est utilisé avec l'unité PDU **bindResult** et l'une des erreurs indiquées dans le champ **ERRORS** de cette opération est utilisée avec l'unité PDU **bindError**. Le champ **Operations** définit les opérations qui peuvent être utilisées à l'intérieur des unités PDU **request**, **result** et **error** du protocole IDM. Le champ **id** est l'identificateur de protocole. Implicitement, il désigne également le contexte d'application pour une opération de rattachement. En conséquence, une classe **IDM-PROTOCOL** distincte est définie pour chaque contexte d'application requis.

9.4 Causes de rejet

Une unité PDU **reject** est renvoyée en réponse à diverses conditions d'erreur. Ces conditions d'erreur et le code raison par lequel elles sont signalées sont décrits ci-dessous:

une cause **mistypedPDU** est renvoyée si la construction de l'unité PDU est non valide.

Une cause **duplicateInvokeIDRequest** est renvoyée si une unité PDU **request** est reçu et si l'unité **invokeID** a déjà été utilisée depuis l'établissement de la connexion.

Une cause **unsupportedOperationRequest** est renvoyée si une unité PDU **request** est reçue et que l'opération demandée ne soit pas prise en charge.

Une cause **unknownOperationRequest** est renvoyée si une unité PDU **request** de demande est reçue et que l'opération demandée soit inconnue.

Une cause **mistypedArgumentRequest** est renvoyée si une unité PDU **request** est reçue et que l'unité **argument** soit construite de façon non valide.

Une cause **resourceLimitationRequest** est envoyée si une unité PDU **request** est reçue et qu'aucune opération ne puisse être effectuée en raison de limitations de ressource.

Une cause **unknownInvokeIDResult** est renvoyée si une unité PDU **result** est reçue et que l'unité **invokeID** ne corresponde pas à celle d'une opération à laquelle une réponse est attendue.

Une cause **mistypedResultRequest** est renvoyée si une unité PDU **result** est reçue et que l'unité **result** soit construite de façon non valide, ou que l'unité **opcode** ne corresponde pas à l'unité PDU **request** correspondante.

Une cause **unknownInvokeIDError** est renvoyée si une unité PDU **error** est reçue et que l'unité **invokeID** ne corresponde pas à celle d'une opération à laquelle une réponse est attendue.

Une cause **unknownError** est renvoyée si une unité PDU **error** est reçue et que l'unité **error** indiquée n'appartienne pas au protocole indiqué ou ne soit pas autorisée en tant que réponse à l'opération.

Une cause **mistypedParameterError** est renvoyée si une unité PDU **error** est reçue et que l'unité **parameter** soit construite de façon non valide ou que l'unité **opcode** ne corresponde pas à celle de l'unité PDU **request** correspondante.

9.5 Causes d'abandon

Une unité **Abort** PDU est renvoyée en réponse à diverses conditions d'erreur qui ne sont pas assurées par les unités PDU **Reject** et **BindError**. Les conditions d'erreur et le code raison par lequel elles sont signalées sont décrits ci-dessous.

Une cause **mistypedPDU** est renvoyée si l'unité PDU reçue a une construction non valide.

Une cause **unboundRequest** est renvoyée si une unité PDU **request** est reçue avant qu'une association ait été établie.

Une cause **invalidPDU** est renvoyée si un agent DSA reçoit une unité PDU qui n'est pas une unité IDM-PDU.

Une cause **resourceLimitation** est renvoyée si une unité PDU **Bind** est reçue et qu'aucune opération ne puisse être effectuée en raison de limitations de ressources, par exemple le dépassement du nombre maximal de connexions.

Une cause **connectionFailed** est renvoyée si l'agent DSA n'a pas été en mesure de créer la connexion TCP/IP afin d'envoyer une unité PDU **Bind**.

Une cause **invalidProtocol** est renvoyée si une unité PDU **resultBind**, **BindResult** ou **BindError** est reçue et que l'unité **protocolID** soit inconnue ou non prise en charge.

Une cause **reasonNotSpecified** est renvoyée si l'initiateur ou le répondeur souhaite clore l'association pour toute autre raison.

NOTE – Un abandon peut être produit par le service sous-jacent de l'initiateur, ce qui empêchera le protocole de passer par la connexion. Par exemple, un abandon avec demande **unboundRequest** sera lancé par le service sous-jacent et non par le système cible, qui ne peut pas être atteint.

9.6 Mappage sur protocole TCP/IP

Chaque unité PDU du protocole IDM est codée sans restriction au moyen des règles de codage de base de la notation ASN.1. Les données binaires résultant du codage sont ensuite partitionnées et classées dans un ou plusieurs segments à envoyer sur la connexion TCP/IP. Chaque segment possède un *en-tête* et contient le *fragment* ou la portion de données codées faisant suite à celles du paquet précédent. La subdivision en fragments d'une unité PDU du protocole IDM et la longueur d'un fragment quelconque sont au choix de l'expéditeur et n'ont pas de portée particulière. Tous les fragments d'une unité PDU-IDM doivent être envoyés avant qu'une autre unité PDU-IDM soit envoyée.

Le format d'un segment (en-tête plus fragment d'unité PDU-IDM) est le suivant:

| | | | |
|----------------------|--------------------------------|------------------------|---|
| version (1 octet) | fragment final (1 octet) | longueur (4 octets) | données (octets de <i>longueur</i>) |
|----------------------|--------------------------------|------------------------|---|

Le champ de *version* indique la version de l'unité PDU-IDM et son mappage sur le protocole TCP/IP. La version décrite dans la présente Spécification d'annuaire doit être indiquée avec la valeur 1. Tous les paquets d'une connexion doivent avoir la même valeur du champ *version*.

NOTE – La façon dont les parties communicantes négocient le numéro de version fera l'objet d'un complément d'étude.

Le champ de *fragment final* indique si le champ de données *data* contient un fragment non final d'unité PDU-IDM (valeur 0) ou s'il contient la valeur entière ou le fragment final (valeur 1).

Le champ de *longueur* *length* indique la longueur en octets du champ de données *data*. Il est envoyé dans "l'ordre des octets du réseau", les octets de poids fort précédant les octets de poids faible. La valeur minimale de longueur est 1. Pour des motifs de performance, il est recommandé que toute l'unité PDU-IDM soit contenue dans un même segment si la longueur peut être exprimée dans les 4 octets du champ de longueur. La fragmentation IDM ne doit être utilisée que si la longueur de l'unité PDU-IDM ne peut pas être exprimée sur 4 octets.

Le champ données *data* contient le fragment suivant de l'unité PDU-IDM qui est acheminé ou l'unité PDU-IDM entière si la valeur entière est acheminée dans le même fragment.

9.7 Adressage

Une extrémité de communication en mode IDM est définie par son adresse IP et par son numéro d'accès, ce qui peut s'écrire comme suit en notation IETF RFC 1738:

```
idm://host:port
```

Le présent paragraphe définit un format équivalent d'adresse de couche Réseau OSI pour une telle extrémité, afin de permettre l'utilisation du protocole IDM avec des définitions de service faisant référence à des adresses de couche Présentation OSI (comme les définitions du service d'annuaire). Une adresse de présentation dans un système prenant en charge l'accès IDM est structurée exactement comme pour l'accès OSI sauf que les éventuels sélecteurs P, S et T ne sont pas pris en compte et que l'adresse réseau a la forme spécifiée ci-dessous. Les systèmes prenant en charge aussi bien les piles OSI que les piles IDM peuvent avoir une même adresse de couche Présentation OSI contenant des adresses réseau OSI et IDM.

Le format d'adresse réseau OSI pour une extrémité IDM est conforme au format IETF RFC 1277. Exprimé sous forme de chaîne d'octets, il se compose de 29 chiffres décimaux codés binaires et d'un seul chiffre de remplissage, comme suit:

- l'identificateur AFI (les 2 premiers chiffres) est "54" (format F.69, décimal, zéro initial significatif);
- l'identificateur IDI (les 8 chiffres suivants) est "00728722";
- le champ de protocole DSP (les 20 chiffres suivants) est construit comme suit:
 - les 2 premiers chiffres forment le préfixe DSP et ont la valeur '10' pour l'IDM
 - NOTE 1 – Les valeurs 01, 02, 03 et 06 sont déjà attribuées par l'IETF RFC 1277. La valeur 03 désigne la pile RFC 1006.
 - les 12 chiffres suivants forment l'adresse IP décimale à 4 composants séparés par des points, chaque composant comptant donc 3 chiffres;
 - les 5 chiffres suivants forment le numéro d'accès.

ISO/CEI 9594-5:2005 (F)

NOTE 2 – Le numéro d'accès est facultatif selon l'IETF RFC 1277 mais obligatoire pour le protocole IDM.

- le dernier chiffre est le caractère hexadécimal 'F' final qui remplit le champ DSP de façon qu'il se termine sur une limite d'octet.

Un agent DSA capable de communiquer sur deux piles différentes (par exemple IDM sur TCP/IP ou OSI sur TCP/IP selon RFC 1006) aura dans son adresse de couche présentation deux adresses de couche Réseau. Par exemple, si l'agent DSA utilise l'accès 1200 pour la pile IDM et l'accès par défaut 102 pour la pile OSI, son extrémité **myAccessPoint** contiendra une adresse de Présentation composée des éléments suivants:

- une adresse réseau n°1 pour le protocole IDM avec le codage suivant (contenant comme adresse IP individuelle l'adresse de bouclage 127.0.0.1 et l'accès 1200):

'54007287221012700000000101200F'H

- une adresse réseau n°2 pour le protocole OSI selon RFC 1006 avec le codage suivant (contenant également comme adresse IP individuelle l'adresse de bouclage 127.0.0.1; l'accès 102 (qui est la valeur par défaut selon RFC 1006) ne figure plus explicitement dans le codage, ce qui est autorisé car le préfixe DSP est dans ce cas 03 selon RFC 1006 et non plus 10 comme pour le protocole IDM):

'540072872203127000000001'H

9.8 Utilisation de la connexion TLS

9.8.1 Etablissement de la connexion TLS

A tout moment, l'initiateur de la connexion TCP/IP peut demander l'établissement d'une connexion TLS en envoyant une unité PDU **StartTLS**. Après sa demande, il n'enverra aucune unité PDU tant qu'il n'a pas reçu une unité PDU **TLSResponse**.

9.8.2 Fermeture de la connexion TLS

Deux types de fermeture de la connexion TLS sont pris en charge, la fermeture progressive et la fermeture brusque.

9.8.2.1 Fermeture progressive

L'initiateur ou le preneur de la connexion TCP/IP peut mettre fin à la connexion TLS en envoyant une alerte de fermeture de connexion TLS. Il cessera alors d'envoyer toute autre unité PDU de protocole d'enregistrement TLS et ignorera toute unité PDU de la couche d'enregistrement TLS reçue tant qu'il n'aura pas reçu une alerte de fermeture de connexion TLS de l'autre partie. Lorsqu'il aura reçu cette alerte, il pourra continuer à envoyer et à recevoir des unités IDM.

Lorsqu'une partie reçoit une alerte de fermeture de connexion TLS non sollicitée, elle peut décider de laisser inchangée la connexion TCP/IP. Dans ce cas, elle répondra immédiatement par une alerte de fermeture de connexion TLS, après quoi elle peut envoyer et recevoir des unités PDU IDM. Après la fermeture d'une connexion TLS, un agent DSA ne répondra à aucune demande reçue avant cette fermeture.

Chaque partie peut décider d'abandonner la connexion TCP/IP sous-jacente après l'envoi ou la réception d'une alerte de fermeture de connexion TLS.

9.8.2.2 Fermeture brusque

L'initiateur ou le preneur de la connexion TCP/IP peut brusquement mettre fin à la connexion TLS en fermant la connexion TCP/IP sous-jacente.

10 Mappage du protocole d'annuaire sur le protocole IDM

Le présent paragraphe contient des définitions pour le mappage des protocoles de l'annuaire sur le protocole IDM. Le module **DirectoryIDMProtocols** complet est reproduit dans l'Annexe E. Ses composants sont répétés ci-dessous pour plus de clarté.

10.1 Protocole DAP-IP

L'unité **dap-ip** du protocole DAP-IP (protocole d'accès à l'annuaire sur protocole TCP/IP) est utilisée pour invoquer des opérations du service abstrait **DirectoryAbstractService**. Elle est définie comme suit:

DAP-IDM-PDUs ::= IDM-PDU (dap-ip)

```

dap-ip IDM-PROTOCOL ::= {
    BIND-OPERATION    directoryBind
    OPERATIONS        { read | compare | abandon | list | search
                       | addEntry | removeEntry | modifyEntry | modifyDN }
    ID                 id-idm-dap }

```

Les codes d'opération et d'erreur pour ce protocole sont indiqués aux § 6.4.1 et 6.5.1.

Seuls les agents DUA peuvent établir des connexions utilisant ce protocole. Seul l'initiateur d'une connexion peut demander des opérations de ce protocole.

10.2 Protocole DSP-IP

L'unité **dsp-ip** du protocole DSP-IP (protocole de système d'annuaire sur protocole TCP/IP) est utilisée pour invoquer des opérations du service abstrait **DistributedOperations**. Elle est définie comme suit:

DSP-IDM-PDUs ::= IDM-PDU (dsp-ip)

```

dsp-ip IDM-PROTOCOL ::= {
    BIND-OPERATION    directoryBind
    OPERATIONS        { chainedRead | chainedCompare | chainedAbandon
                       | chainedList | chainedSearch
                       | chainedAddEntry | chainedRemoveEntry
                       | chainedModifyEntry | chainedModifyDN }
    ID                 id-idm-dsp }

```

Les codes d'opération et d'erreur pour ce protocole sont indiqués aux § 6.4.1 et 6.5.1.

Les agents DSA peuvent utiliser ce protocole et aussi bien l'initiateur que l'accepteur d'une connexion peuvent demander des opérations de ce protocole.

10.3 Protocole DISP-IP

L'unité **disp-ip** du protocole DISP-IP (protocole de duplication miroir d'informations de l'annuaire sur protocole TCP/IP) est utilisée pour invoquer des opérations du service abstrait **DirectoryShadowAbstractService**. Elle est définie comme suit:

DISP-IDM-PDUs ::= IDM-PDU (disp-ip)

```

disp-ip IDM-PROTOCOL ::= {
    BIND-OPERATION    directoryBind
    OPERATIONS        { requestShadowUpdate
                       | updateShadow
                       | coordinateShadowUpdate }
    ID                 id-idm-disp }

```

Les codes d'opération et d'erreur pour ce protocole sont indiqués aux § 6.4.2 et 6.5.2.

Les agents DSA peuvent utiliser ce protocole et aussi bien l'initiateur que l'accepteur d'une connexion peuvent demander des opérations de ce protocole.

10.4 Protocole DOP-IP

L'unité **dop-ip** du protocole DOP-IP (protocole de gestion des liens opérationnels de l'annuaire sur protocole TCP/IP) est utilisée pour invoquer des opérations du service abstrait **OperationalBindingManagement**. Elle est définie comme suit:

DOP-IDM-PDUs ::= IDM-PDU (dop-ip)

```

dop-ip IDM-PROTOCOL ::= {
    BIND-OPERATION    directoryBind
    OPERATIONS        { establishOperationalBinding
                       | modifyOperationalBinding
                       | terminateOperationalBinding}
    ID                 id-idm-dop }

```

Les codes d'opération et d'erreur pour ce protocole sont indiqués aux § 6.4.3 et 6.5.3.

Les agents DSA peuvent utiliser ce protocole et aussi bien l'initiateur que l'accepteur d'une connexion peuvent demander des opérations de ce protocole.

11 Coexistence entre piles protocolaires

Le paragraphe 9.7 a défini un format d'adresse de couche Réseau OSI pour une extrémité de communications en protocole IDM. Le présent paragraphe recommande une méthode permettant la coexistence d'agents DSA prenant en charge des piles protocolaires différentes, telles que OSI, IDM et LDAP. Afin que les renvois de référence puissent contenir des points d'accès en protocole LDAP, le présent paragraphe spécifie également un format d'adresse réseau OSI pour une extrémité de communications en protocole LDAP.

11.1 Coexistence entre piles OSI et piles IDM

Une implémentation conforme implémentera la pile OSI définie aux § 7 et 8, la pile IDM définie aux § 9 et 10, ou les deux.

Si un agent DSA de chaînage a besoin de renvoyer une demande vers un agent DSA cible et si ces deux agents ne prennent en charge aucune pile protocolaire commune, cet agent DSA de chaînage doit renvoyer, à la place de la demande, un renvoi de référence. Celui-ci sera renvoyé par chaque agent DSA qui a concaténé la demande. Si l'un quelconque de ces agents DSA prend en charge la pile protocolaire de l'agent DSA cible, cet agent peut choisir d'envoyer la demande directement à l'agent cible qui est indiqué dans le renvoi de référence.

Si aucun des agents DSA de chaînage ne prend en charge la pile protocolaire de l'agent DSA cible, le renvoi de référence doit être renvoyé à l'agent DUA, qui sera peut-être en mesure d'envoyer la demande directement à l'agent DSA cible.

Si l'on déploie dans un domaine un assortiment de produits d'agent DSA dont certains ne prennent en charge qu'une seule pile protocolaire, il est recommandé que:

- a) que cette pile protocolaire soit prise en charge par les agents DSA détenant la connaissance d'homologues ne prenant en charge qu'une seule pile protocolaire;
- b) ou bien que les deux piles protocolaires soient prises en charge par l'agent DSA auquel l'agent DUA se rattache.

11.2 Coexistence en présence du protocole LDAP

Les agents DSA prenant en charge soit la pile protocolaire de couches supérieures OSI soit la pile protocolaire IDM peuvent également décider de prendre en charge le protocole LDAP. L'interopérabilité entre de tels agents DSA peut être réalisée par l'emploi de concaténations ou de renvois de référence. L'interopérabilité entre de tels agents DSA et les agents DUA peut être réalisée par l'emploi du protocole LDAP ou DAP.

Afin qu'un agent DSA soit en mesure de fournir des renvois de référence utiles à des agents DUA ne prenant en charge que le protocole LDAP, il est nécessaire de représenter, dans une adresse de présentation OSI, le point d'accès LDAP d'un éventuel agent DSA cible. Le paragraphe 11.3 définit un format de point NSAP pour le protocole LDAP. Un agent DSA recevant un renvoi de référence contenant un point NSAP de ce type peut le convertir en un renvoi de référence LDAP et le renvoyer au client LDAP connecté.

11.3 Définition d'un format de point NSAP pour le protocole LDAP

Le présent paragraphe définit un format d'adresse réseau OSI pour une extrémité de communication en protocole LDAP, afin de permettre l'utilisation de ce point NSAP avec des définitions de service faisant référence à des adresses de couche Présentation OSI (telles que les définitions de service d'annuaire). Une adresse de présentation pour un système prenant en charge le protocole LDAP est structurée exactement comme pour l'accès OSI, sauf que les éventuels sélecteurs P, S et T ne sont pas pris en compte et que l'adresse réseau a la forme spécifiée ci-dessous. Les systèmes prenant en charge simultanément les piles OSI, IDM et LDAP peuvent avoir une même adresse de couche Présentation OSI contenant des adresses réseau OSI, IDM et LDAP.

Le format d'adresse réseau OSI pour une extrémité LDAP est conforme au format IETF RFC 1277. Exprimé sous forme de chaîne d'octets, il se compose de 29 chiffres décimaux codés binaires et d'un seul chiffre de remplissage, comme suit:

- l'identificateur AFI (les 2 premiers chiffres) est '54' (format F.69, décimal, zéro initial significatif);
- l'identificateur IDI (les 8 chiffres suivants) est '00728722';
- le champ de protocole DSP (les 20 chiffres suivants) est construit comme suit:
 - les 2 premiers chiffres forment le préfixe DSP et ont la valeur '11' pour le protocole LDAP;

NOTE 1 – Les valeurs 01, 02, 03 et 06 sont déjà attribuées par l'IETF RFC 1277. La valeur 03 désigne la pile RFC 1006. La valeur '10' désigne la pile IDM.

- les 12 chiffres suivants forment l'adresse IP décimale à 4 composants séparés par des points, chaque composant comptant donc 3 chiffres;
- les 5 chiffres suivants forment le numéro d'accès.

NOTE 2 – Le numéro d'accès est facultatif selon l'IETF RFC 1277 mais obligatoire pour le protocole LDAP.

- le dernier chiffre est le caractère hexadécimal 'F' final qui remplit le champ DSP de façon qu'il se termine sur une limite d'octet.

Un agent DSA capable de communiquer sur trois piles différentes (par exemple IDM sur TCP/IP ou OSI sur TCP/IP selon RFC 1006 ou LDAP) aura dans son adresse de couche présentation trois adresses de couche Réseau. Par exemple, si l'agent DSA utilise l'accès 1200 pour la pile IDM et l'accès par défaut 102 pour la pile OSI et l'accès 389 pour le protocole LDAP, son extrémité **myAccessPoint** contiendra une adresse de présentation composée des éléments suivants:

- une adresse réseau n°1 pour le protocole IDM avec le codage suivant (contenant comme adresse IP individuelle l'adresse de bouclage 127.0.0.1 et l'accès 1200):

'54007287221012700000000101200F'H

- une adresse réseau n°2 pour le protocole OSI selon RFC 1006 avec le codage suivant (contenant également comme adresse IP individuelle l'adresse de bouclage 127.0.0.1; l'accès 102 (qui est la valeur par défaut selon RFC 1006) ne figure plus explicitement dans le codage, ce qui est autorisé car le préfixe DSP est dans ce cas 03 selon RFC 1006 et non plus 10 comme pour le protocole IDM):

'540072872203127000000001'H

- une adresse réseau n°3 pour le protocole LDAP avec le codage suivant (contenant comme adresse IP individuelle l'adresse de bouclage 127.0.0.1 et l'accès 389):

'54007287221112700000000100389F'H

12 Versions et règles d'extensibilité

Le présent paragraphe décrit les règles de négociation de la version ainsi que les règles d'extensibilité pour les protocoles mappés sur l'OSI qui sont définis dans le § 7 et pour les protocoles mappés sur le protocole IDM qui sont définis dans le § 10.

L'annuaire peut être réparti et deux entités d'application d'annuaire ou plus peuvent fonctionner conjointement pour satisfaire une demande. Les AE de l'annuaire peuvent être implémentées en conformité avec différentes éditions de la Spécification d'annuaire du service d'annuaire, qui peuvent être représentées ou non par des numéros de version de protocole différents. Le numéro de version est négocié en fonction de la plus récente version commune à deux AE d'annuaire directement rattachées.

NOTE 1 – Il existe actuellement deux versions de chaque protocole d'annuaire. L'édition de 1988 et celle de 1993 correspondent à la version 1. La plupart des fonctionnalités introduites dans les éditions 4 et ultérieures sont également disponibles dans la version 1. Toutefois, certains services et protocoles améliorés (les erreurs signées par exemple) imposent la négociation préalable de la version 2 par les parties en communication.

Un DUA peut émettre une demande conformément à l'édition la plus récente de la Spécification d'annuaire à partir de laquelle cet agent a été implémenté, c'est-à-dire conformément à son niveau. Au moyen des règles d'extensibilité définies ci-dessous, cette demande devra être transmise au DSA approprié qui répondra à la demande, quelle que soit l'édition des DSA en cause. Le DSA preneur fonctionnera selon les dispositions décrites ci-dessous.

NOTE 2 – Un DSA intermédiaire qui ne fait que chaîner la demande peut choisir d'examiner des éléments choisis de l'unité PDU d'annuaire qui est nécessaire à l'opération, par exemple la résolution du nom.

12.1 DUA à DSA

12.1.1 Négociation de la version

Lorsqu'elle accepte une association (c'est-à-dire un lien) qui utilise le DAP, la version négociée ne concerne que les aspects point à point du protocole d'échange entre le DUA et le DSA auquel il est connecté. Les demandes ou réponses ultérieures au sujet de cette association ne devront pas subir de contraintes dues à la version négociée.

NOTE – Il n'existe pas actuellement d'aspects point à point du DAP qui soient indiqués par des versions de protocole différentes.

12.1.2 Traitement des demandes et des réponses

Le DUA peut formuler des demandes en utilisant l'édition des spécifications le plus élevé de la demande qu'il prend en charge. Si un ou plusieurs éléments de la demande sont critiques, il devra indiquer les numéros d'extension dans le paramètre **criticalExtensions**.

NOTE 1 – Si l'information que l'extension a remplacée dans un type **CHOICE**, **ENUMERATED**, ou **INTEGER** (utilisé comme **ENUMERATED**) est indispensable pour l'exécution correcte d'une opération dans un DSA implémenté conformément à une édition antérieure de la Spécification, il est recommandé de qualifier cette extension de critique.

Lorsqu'il traite une demande émanant d'un DUA, un DSA doit se conformer aux règles définies au § 12.2.2.

Lorsqu'il traite une réponse, un DUA doit:

- a) ne prendre en compte aucune assignation de nom binaire inconnu dans une chaîne de bits;
- b) ne prendre en compte aucun numéro indiqué comme étant inconnu dans un type **ENUMERATED** ou dans un type **INTEGER** utilisé dans le style énuméré, à condition que le numéro apparaisse comme élément facultatif du **SET** ou d'une **SEQUENCE**;
- c) ignorer tous les éléments inconnus dans des **SET**, à la fin des **SEQUENCE**, ou dans des **CHOICE** lorsque le **CHOICE** est lui-même un élément facultatif d'un **SET** ou d'une **SEQUENCE**;
NOTE 2 – Les implémentations peuvent à titre d'option locale ignorer certains autres éléments dans une PDU d'annuaire. En particulier, certains numéros nommés inconnus et certains **CHOICE** inconnus contenus dans les éléments obligatoires de **SET** et **SEQUENCE**, peuvent être ignorés sans invalider l'opération. L'identification de ces éléments appelle un complément d'étude.
- d) ne pas considérer la réception de types d'attribut et de valeurs d'attribut inconnus comme une violation de protocole;
- e) à titre facultatif, signaler à l'utilisateur les types d'attribut et les valeurs d'attribut inconnus.

12.1.3 Règles d'extensibilité pour le traitement des erreurs

Lorsqu'il traite un type d'erreur connu avec des problèmes et des paramètres indiqués comme étant inconnus, un DUA doit:

- a) ne pas considérer comme une violation de protocole la réception de problèmes et de paramètres indiqués comme étant inconnus (c'est-à-dire qu'il ne doit pas émettre de **OSI Reject** ou de **Reject**, selon le cas, ou abandonner l'association d'application);
- b) à titre facultatif, signaler à l'utilisateur l'information supplémentaire d'erreur.

Quand il traite un type d'erreur de type inconnu, un DUA doit:

- a) ne pas considérer la réception d'un type d'erreur inconnu comme une violation de protocole (c'est-à-dire qu'il ne doit pas émettre de **OSI Reject** ou de **Reject**, selon le cas, ou abandonner l'association d'application);
- b) à titre facultatif, signaler l'erreur à l'utilisateur.

12.2 DSA à DSA

12.2.1 Négociation de la version

Lorsqu'elle établit ou qu'elle accepte une association (c'est-à-dire un lien) utilisant le DSP, la version négociée ne concerne que les aspects point à point du protocole d'échange entre les DSA. Les demandes ou les réponses suivantes, relatives à l'association, ne doivent pas subir de contraintes dues à la version négociée.

NOTE 1 – Il n'existe pas actuellement d'aspects point à point du DSP qui soient indiqués par des versions de protocole différentes.

Lorsqu'elle établit ou accepte une association (c'est-à-dire un lien) utilisant le DISP, la version négociée doit définir tous les aspects du protocole d'échange entre les DSA. Les demandes ou réponses suivantes, relatives à l'association, doivent être contraintes par la version négociée.

NOTE 2 – Il n'existe actuellement qu'une seule version du DISP.

Lorsqu'elle établit ou accepte une association (c'est-à-dire un lien) utilisant le DOP, la version négociée doit définir tous les aspects du protocole d'échange entre les DSA. Les demandes ou réponses suivantes, relatives à l'association, doivent être contraintes par la version négociée.

NOTE 3 – Il n'y a actuellement qu'une seule version du DOP.

12.2.2 Règles d'extensibilité pour le traitement des opérations

Si au cours d'une opération (une fois la résolution du nom achevée), un DSA détecte un élément de **criticalExtensions** dont la sémantique est inconnue, cet agent devra renvoyer une indication **unavailableCriticalExtension** comme **serviceError** ou dans un **PartialOutcomeQualifier**.

NOTE – Si on reçoit une chaîne **criticalExtensions** comportant une ou plusieurs valeurs nulles, cela indique que les extensions correspondant à ces valeurs ne sont pas présentes dans l'opération ou qu'elles ne sont pas critiques. Il ne faut pas déduire, de la présence d'une valeur nulle dans une chaîne **criticalExtensions**, la présence ou l'absence de l'extension correspondante dans l'unité PDU d'annuaire.

Autrement, lorsqu'il traite une PDU d'annuaire, un DSA doit:

- a) ne prendre en compte aucune assignation de nom binaire inconnu dans une chaîne de bits;
- b) ne prendre en compte aucun numéro indiqué comme étant inconnu dans un type **ENUMERATED** ou dans un type **INTEGER** utilisé dans le style énuméré, à condition que le numéro apparaisse comme élément facultatif du **SET** ou d'une **SEQUENCE**;
- c) ignorer tous les éléments inconnus dans des **SET**, à la fin des **SEQUENCE**, ou dans des **CHOICE** lorsque le **CHOICE** est lui-même un élément facultatif d'un **SET** ou d'une **SEQUENCE**.

12.2.3 Règles d'extensibilité pour le chaînage

Si la PDU est une demande, le DSA doit transmettre la demande contenant les types et les valeurs inconnus à tous les DSA supplémentaires déterminés par l'opération de résolution du nom.

Si la PDU est une réponse, le DSA devra traiter les types et les valeurs inconnus de la même façon qu'il traiterait les types et les valeurs connus (voir l'article sur le fusionnement des résultats dans la Spécification d'annuaire sur le fonctionnement réparti) et les envoyer au DSA ou au DUA initiateur.

Un agent DSA implémentant la cinquième édition ou les éditions ultérieures et agissant en tant qu'agent DSA intermédiaire qui ne fait que chaîner une demande transmettra une demande contenant une opération inconnue. Un agent DSA antérieur à la cinquième édition peut facultativement transmettre une demande contenant une opération inconnue.

12.2.4 Règles d'extensibilité pour le traitement des erreurs

Lorsqu'il traite un type d'erreur connu avec des problèmes et des paramètres indiqués comme étant inconnus, un DSA:

- a) ne doit pas considérer comme une violation de protocole la réception de problèmes et de paramètres indiqués comme étant inconnus (c'est-à-dire qu'il ne doit pas émettre de **OSI Reject** ou de **Reject**, selon le cas, ou abandonner l'association d'application);
- b) peut essayer d'effectuer une reprise selon son analyse du seul type d'erreur ou peut simplement transmettre l'erreur (et ses problèmes et paramètres indiqués comme étant inconnus) au DSA ou au DUA approprié suivant.

Lorsqu'il traite un type d'erreur inconnu, un DSA qui ne fait que chaîner la demande doit:

- a) ne pas considérer comme une violation de protocole le type d'erreur inconnu (c'est-à-dire qu'il ne doit pas émettre de **OSI Reject** ou de **Reject**, selon le cas, ou abandonner l'association d'application);
- b) ne pas essayer de corriger l'erreur et ses paramètres et problèmes indiqués ou d'effectuer une reprise à leur suite;
- c) transmettre le type d'erreur inconnu au DSA ou au DUA approprié suivant.

Lorsqu'il traite une erreur inconnue, un DSA qui établit la corrélation entre des réponses multiples doit:

- a) ne pas considérer comme une violation de protocole le type d'erreur inconnu (c'est-à-dire qu'il ne doit pas émettre de **OSI Reject** ou de **Reject**, selon le cas, ou abandonner l'association d'application);
- b) ne pas essayer de corriger l'erreur et ses paramètres et problèmes indiqués ou d'effectuer une reprise à leur suite;
- c) mettre l'erreur inconnue dans **PartialOutcomeQualifier**;
- d) continuer à mettre les résultats en corrélation comme en conditions normales.

12.3 Règles d'extensibilité des classes d'objets

Des attributs d'utilisateurs optionnels peuvent être ajoutés à une classe d'objets existante sans assigner un nouvel identificateur d'objet.

Un DSA ne prenant pas en charge une extension de classe d'objets peut rejeter toute opération qui tente de créer ou de modifier une entrée qui se traduit par la présence d'un attribut d'extension dans cette entrée.

12.4 Règles d'extensibilité des types d'attribut d'utilisateur

Une définition de type d'attribut d'utilisateur peut être étendue de manière à ce que ces caractéristiques de concordance ne soient pas modifiées. Cela peut inclure:

- l'adjonction de valeurs aux types **ENUMERATED** et **INTEGER** qui est utilisée dans le style énuméré;
- l'adjonction de bits à une chaîne binaire.

Il n'est pas exigé qu'un DSA traite une valeur d'attribut qui inclut de telles extensions.

Un DUA ne doit pas considérer la réception d'une valeur d'attribut étendue comme une erreur.

13 Conformité

Le présent paragraphe définit les conditions de conformité à la présente Spécification d'annuaire.

13.1 Conformité des DUA

Pour être conforme à la présente Spécification d'annuaire, un DUA mis en service doit satisfaire aux conditions spécifiées aux § 13.1.1 à 13.1.3.

13.1.1 Conditions de déclaration

Il sera déclaré ce qui suit:

- a) les opérations du contexte d'application **directoryAccessAC** et/ou du protocole **dap-ip** que l'agent DUA est capable d'invoquer et auxquelles est alléguée la conformité;
- b) le ou les niveaux de sécurité de rattachement pour lesquels la conformité est revendiquée (aucune, simple, poussée – et, s'il s'agit de la conformité simple, si le mot de passe est absent, présent ou protégé); ainsi que l'indication de la capacité de l'agent DUA à produire des arguments signés ou à valider des résultats signés;
- c) les extensions dont la liste est donnée dans le Tableau 1 de la Rec. UIT-T X.511 | ISO/CEI 9594-3 que le DUA est capable de déclencher et auxquelles est alléguée la conformité;
- d) si la conformité au contrôle d'accès par règles est alléguée;
- e) si la conformité est alléguée pour une authentification forte ou des opérations signées, une identification du certificat et des extensions CRL pour lesquelles la conformité est alléguée.

13.1.2 Conditions statiques

Un DUA doit:

- a) être à même de prendre en charge le contexte d'application **directoryAccessAC** respectant la syntaxe abstraite définie au § 7, et/ou le protocole **dap-ip** défini au § 10;
- b) se conformer aux extensions auxquelles la conformité a été alléguée au § 13.1.1 c);
- c) si la conformité au contrôle d'accès par règles est alléguée, être capable de prendre en charge les étiquettes de sécurité présentées au § 19.4 de la Rec. UIT-T X.501 | ISO/CEI 9594-2;
- d) être conforme aux § 8 et 15 de la Rec. UIT-T X.509 | ISO/CEI 9594-8 pour le certificat et les extensions CRL pour lesquels la conformité a été alléguée au § 13.1.1 e).

13.1.3 Conditions dynamiques

Un DUA doit:

- a) être conforme au mappage sur le service utilisé, défini au § 8 ou au § 10 ou aux deux;
- b) être conforme aux procédures des règles d'extensibilité définies au § 12.1.

13.2 Conformité par les DSA

Pour être conforme à la présente Spécification d'annuaire, un DSA mis en service doit satisfaire aux conditions spécifiées aux § 13.2.1 à 13.2.3.

13.2.1 Conditions de déclaration

Il sera déclaré ce qui suit:

- a) les contextes d'application et les protocoles IDM auxquels est alléguée la conformité: **directoryAccessAC**, **directorySystemAC**, **directoryOperationalBindingManagementAC**, **dap-ip**, **dsp-ip**, **dop-ip**, ou toute combinaison de ces contextes. Un DSA qui prétend être conforme au contexte **directoryOperationalBindingManagementAC** ou au contexte **dop-ip** pour la prise en charge des liens opérationnels hiérarchiques doit aussi prendre en charge le contexte **directorySystemAC** ou **dsp-ip**. Si un DSA est tel que la connaissance de son existence a été disséminée de sorte que des DSA n'appartenant pas à son DMD contiennent des informations de connaissance à son sujet, alors il devra prétendre à la conformité au contexte d'application **directorySystemAC** ou **dsp-ip**;
 NOTE 1 – Un contexte d'application ne doit pas être tronqué sauf comme indiqué ici; en particulier, il ne peut y avoir de déclaration de conformité à des opérations particulières.
- b) les types de liens opérationnels auxquels est alléguée la conformité: **shadowOperationalBindingID**, **specificHierarchicalBindingID**, **non-specificHierarchicalBindingID** ou toute combinaison de ceux-ci. Un DSA qui allègue la conformité avec le lien **shadowOperationalBindingID** doit prendre en charge un ou plusieurs contextes d'application pour les fournisseurs d'informations miroirs ou les consommateurs d'informations miroirs indiqués aux § 13.3 et 13.4;
- c) si le DSA est capable ou non d'agir en tant que DSA de premier niveau, comme défini dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- d) au cas où la conformité avec le contexte d'application spécifié par **directorySystemAC** et/ou associé au protocole **dap-ip** est alléguée, si le mode d'opération chaîné défini dans la Rec. UIT-T X.518 | ISO/CEI 9594-4 est pris en charge ou non;
- e) si la conformité au contexte d'application spécifié par **directoryAccessAC** et/ou associé au protocole **dap-ip** est revendiquée, le ou les niveaux de sécurité de rattachement pour lesquels la conformité est revendiquée (aucune, simple, poussée – et, s'il s'agit de la conformité simple, si le mot de passe est absent, présent ou protégé); si l'agent DSA peut effectuer l'authentification d'expéditeur comme défini au § 22.1 de la Rec. UIT-T X.518 | ISO/CEI 9594-4 et, si c'est le cas, si cette opération est fondée sur l'identité ou sur la signature; et si l'agent DSA peut effectuer l'authentification de résultat comme défini au § 22.2 de la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- f) si la conformité au contexte d'application spécifié par **directorySystemAC** et/ou associé au protocole **dsp-ip** est revendiquée, le ou les niveaux de sécurité de rattachement pour lesquels la conformité est revendiquée (aucune, simple, poussée – et, s'il s'agit de la conformité simple, si le mot de passe est absent, présent ou protégé); si l'agent DSA peut effectuer l'authentification d'expéditeur comme défini au § 22.1 de la Rec. UIT-T X.518 | ISO/CEI 9594-4 et, si c'est le cas, si cette opération est fondée sur l'identité ou sur la signature; et si l'agent DSA peut effectuer l'authentification de résultat comme défini au § 22.2 de la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- g) les types d'attributs sélectionnés et définis dans la Rec. UIT-T X.520 | ISO/CEI 9594-6 et tous autres types d'attributs auxquels est alléguée la conformité et si, pour les attributs fondés sur la syntaxe **DirectoryString**, est revendiquée la conformité pour le choix entre **UniversalString**, **BMPString**, ou **UTF8String**;
- h) les classes d'objets sélectionnées, définies dans la Rec. UIT-T X.521 | ISO/CEI 9594-7 et toutes autres classes d'objets auxquelles est alléguée la conformité;
- i) les extensions, dont la liste est reproduite au Tableau 1 de la Rec. UIT-X.511 | ISO/CEI 9594-3, auxquelles le DSA est capable de répondre et auxquelles est alléguée la conformité;
- j) s'il est allégué conformité aux attributs collectifs comme défini au § 8.9 de la Rec. UIT-T X.501 | ISO/CEI 9594-2 et aux § 7.6, 7.8.2 et 9.2.2 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
- k) s'il est allégué conformité aux attributs hiérarchiques comme défini aux § 7.6, 7.8.2 et 9.2.2 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
- l) les types d'attribut opérationnels définis dans la Rec. UIT-T X.501 | ISO/CEI 9594-2 et les autres types d'attribut opérationnels auxquels est alléguée la conformité;
- m) s'il est allégué conformité au renvoi de noms pseudonymes comme décrit au § 7.7.1 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
- n) s'il est allégué conformité indiquant que l'information d'entrée renvoyée est complète, comme décrit au § 7.7.1 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
- o) s'il est allégué conformité à la modification de l'attribut de la classe d'objets pour ajouter ou supprimer des valeurs identifiant des classes d'objets auxiliaires comme décrit au § 11.3.2 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;

- p) s'il est allégué conformité à la commande d'accès de base;
- q) s'il est allégué conformité à la commande d'accès simplifié;
- r) si le DSA est capable d'administrer le sous-schéma pour sa portion du DIT, comme défini dans la Rec. UIT-T X.501 | ISO/CEI 9594-2;
NOTE 2 – La capacité d'administrer un sous-schéma ne doit pas être amputée; en particulier, la capacité d'administrer les définitions d'un sous-schéma particulier ne doit pas être alléguée.
- s) les liens de noms sélectionnés définis dans la Rec. UIT-T X.521 | ISO/CEI 9594-7 et tout autre lien de nom, auxquels est alléguée la conformité;
- t) si le DSA est capable d'administrer les attributs collectifs définis dans la Rec. UIT-T X.501 | ISO/CEI 9594-2;
- u) les types de contextes sélectionnés, définis dans la Rec. UIT-T X.520 | ISO/CEI 9594-6, ainsi que tous les autres types de contextes auxquels est alléguée la conformité;
- v) s'il est allégué conformité aux contextes tels qu'ils sont définis aux § 8.8, 8.9 et 12.8 de la Rec. UIT-T X.501 | ISO/CEI 9594-2 et aux § 7.3 et 7.6 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
- w) s'il est allégué conformité à l'usage des contextes dans les RDN, selon les § 8.5 et 9.3 de la Rec. UIT-T X.501 | ISO/CEI 9594-2, § 7.7 de la Rec. UIT-T X.511 | ISO/CEI 9594-3 et la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- x) s'il est allégué conformité à la gestion de l'arbre d'informations du DSA, telle qu'elle est définie au § 7.13 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
- y) s'il est allégué conformité à l'usage de la gestion-systèmes aux fins de gestion de l'annuaire, tel qu'il est décrit dans la Rec. UIT-T X.530 | ISO/CEI 9594-10;
- z) les objets gérés sélectionnés et les types d'attributs de gestion définis dans la Rec. UIT-T X.530 | ISO/CEI 9594-10, ainsi que tous autres objets gérés ou types d'attributs de gestion, auxquels est alléguée la conformité;
- aa) s'il est allégué conformité au contrôle d'accès par règles.
NOTE 3 – La prise en charge des étiquettes de sécurité exige au minimum la prise en charge dans les contextes de Context lists selon le § 8.8 de la Rec. UIT-T X.501 | ISO/CEI 9594-2 et de **returnContexts** selon le § 7.6 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.
- bb) s'il est allégué conformité à l'intégrité des opérations de l'annuaire;
- cc) s'il est allégué conformité à la capacité de l'annuaire à détenir des informations chiffrées et signées et à offrir un accès à ces informations;
- dd) si la conformité est revendiquée pour une authentification forte, des opérations signées ou des opérations protégées, l'identification du certificat et des extensions CRL pour lesquelles la conformité est revendiquée.

13.2.2 Conditions statiques

Un DSA doit:

- a) être à même de prendre en charge les contextes d'application dont les syntaxes abstraites sont définies au § 7, ainsi que les protocoles IDM définis au § 10, auxquels la conformité est revendiquée;
- b) être à même de prendre en charge le cadre d'information défini par sa syntaxe abstraite dans la Rec. UIT-T X.501 | ISO/CEI 9594-2;
- c) se conformer aux conditions minimales de connaissance définies dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- d) s'il est déclaré conforme en tant que DSA de premier niveau, être conforme aux conditions de prise en charge du contexte de la racine, définies dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- e) être à même de prendre en charge les types d'attributs auxquels est alléguée la conformité, tels que définis par leur syntaxe abstraite;
- f) être à même de prendre en charge les classes d'objets auxquelles est alléguée la conformité, telles que définies par leur syntaxe abstraite;
- g) se conformer aux extensions auxquelles la conformité a été déclarée au § 13.2.1 i);
- h) si la capacité à gérer le sous-schéma tel que défini dans la Rec. UIT-T X.501 | ISO/CEI 9594-2 est alléguée, le DSA doit être capable de le gérer;
- i) s'il est allégué conformité aux attributs collectifs, avoir la capacité d'exécution des procédures pertinentes définies aux § 7.6, 7.8.2 et 9.2.2 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;

- j) s'il est allégué conformité aux attributs hiérarchiques, avoir la capacité d'exécution des procédures pertinents définies aux § 7.6, 7.8.2 et 9.2.2 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
 - k) avoir la capacité de prise en charge des types d'attributs auxquels est alléguée la conformité;
 - l) s'il est allégué conformité à la commande d'accès de base, être à même de détenir des éléments d'information ACI qui soient conformes aux définitions de la commande d'accès de base;
 - m) s'il est allégué conformité à la commande d'accès simplifié, être à même de détenir des éléments d'information ACI qui soient conformes aux définitions de la commande d'accès simplifié;
 - n) être à même de prendre en charge les types de contexte auxquels est alléguée la conformité tels que les définissent leurs syntaxes abstraites;
 - o) s'il est allégué conformité aux contextes, être en mesure d'exécuter les procédures pertinentes qui sont définies dans la Rec. UIT-T X.511 | ISO/CEI 9594-3;
 - p) s'il est allégué conformité à l'usage des contextes dans les RDN, être en mesure d'exécuter les procédures pertinentes telles qu'elles sont définies aux § 9.3 de la Rec. UIT-T X.501 | ISO/CEI 9594-2, § 7.7 de la Rec. UIT-T X.511 | ISO/CEI 9594-3 et dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
 - q) s'il est allégué conformité à la gestion de l'arbre d'informations du DSA, être en mesure d'exécuter les procédures pertinentes telles qu'elles sont définies aux § 7.5 et 7.13 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
 - r) s'il est allégué conformité pour la prise en charge des familles de caractéristiques dans des entrées, disposer des capacités définies aux § 7.3.2, 7.6.4 et 7.8.3 de la Rec. UIT-T X.511 | ISO/CEI 9594-3;
 - s) s'il est allégué conformité pour la caractéristique d'assouplissement de la recherche, disposer des capacités définies au § 13.6.2 de la Rec. UIT-T X.501 | ISO/CEI 9594-2 et au § 10.2.2 de la Rec. UIT-T X.511 | ISO/CEI 9594-3. En particulier, une implémentation doit indiquer:
 - si elle prend en charge l'inclusion de la construction **RelaxationPolicy** dans une demande de recherche;
 - si elle prend en charge l'adaptation par mappage, la substitution de règles d'adaptation, ou les deux;
 - les mappages qui sont pris en charge lorsqu'elle prend en charge l'adaptation par mappage.
 - t) s'il est allégué conformité avec la fonction groupe hiérarchique, disposer des capacités définies au § 7.5 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.
En outre, l'implémentation doit déclarer:
 - quelles options de hiérarchie sont prises en charge;
 - u) s'il est allégué conformité à l'administration de base des services, disposer des capacités définies au § 16 de la Rec. UIT-T X.501 | ISO/CEI 9594-2, et les procédures de vérification de base définies au § 13 de la Rec. UIT-T X.511 | ISO/CEI 9594-3. Cette prise en charge inclut:
 - la prise en charge du comptage des entrées;
 - la prise en charge des options de commande de service **entryCount** et **performExactly**;
 - la prise en charge de l'extension de **notification** définie au § 7.4 de la Rec. UIT-T X.511 | ISO/CEI 9594-3.
- En outre, l'implémentation doit déclarer si elle prend en charge:
- des points administratifs propres au service différents des points administratifs autonomes;
 - la fonctionnalité de contexte à l'intérieur des règles de recherche;
 - les familles de facilité d'entrée à l'intérieur des règles de recherche, qui nécessitent également une conformité générale avec cette fonctionnalité;
 - la fonctionnalité de l'assouplissement de recherche à l'intérieur des règles de recherche détaillées ci-dessus au s), qui exige également que l'implémentation allègue une conformité générale avec la fonctionnalité d'assouplissement de recherche;
 - les groupes hiérarchiques à l'intérieur des règles de recherche.
- v) s'il est allégué conformité à l'usage de la gestion-systèmes aux fins de gestion de l'annuaire, être en mesure d'exécuter, à l'égard des objets auxquels est alléguée la conformité, les procédures pertinentes telles qu'elles sont définies dans la Rec. UIT-T X.530 | ISO/CEI 9594-10;
 - w) s'il est allégué conformité au contrôle d'accès par règles, être en mesure de détenir des articles d'ACI conformes à la définition du contrôle d'accès par règles;

ISO/CEI 9594-5:2005 (F)

- x) s'il est allégué conformité à l'intégrité des opérations de l'annuaire, être en mesure de sceller toutes les opérations d'annuaire prises en charge;
- y) s'il est allégué conformité à l'intégrité des informations d'annuaire conservées en mémoire, être en mesure de prendre en charge le contexte **attributValueIntegrityInfoContext** pour protéger l'information d'annuaire;
- z) conforme au § 8 de la Rec. UIT-T X.509 | ISO/CEI 9594-8 pour les extensions Certificate et CRL pour lesquelles la conformité a été alléguée au 13.2.1 dd).

13.2.3 Conditions dynamiques

Un DSA doit:

- a) si la conformité avec d'éventuels contextes d'application définis aux § 8.2.2, 8.2.3 et 8.2.4 est revendiquée, être conforme au mappage sur les services utilisés, défini au § 8;
- b) être conforme aux procédures applicables au fonctionnement réparti de l'annuaire vis-à-vis des renvois de référence, définies dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- c) s'il est allégué conformité au contexte d'application spécifié par **directoryAccessAC** et/ou associé au protocole **dap-ip** être conforme aux procédures de la Rec. UIT-T X.518 | ISO/CEI 9594-4 en ce qui concerne le mode de renvoi de référence du DAP;
- d) s'il est allégué conformité au contexte d'application spécifié par **directorySystemAC** et/ou associé au protocole **dsp-ip**, être conforme au mode de renvoi de référence d'interaction défini dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- e) s'il est allégué conformité au mode chaîné d'interaction, être conforme au mode chaîné d'interaction défini dans la Rec. UIT-T X.518 | ISO/CEI 9594-4;
NOTE – C'est le seul cas où un DSA doit être capable d'invoquer des opérations de l'élément **directorySystemAC** et/ou **dsp-ip**.
- f) être conforme aux procédures des règles d'extensibilité définies au § 12.2;
- g) s'il est allégué conformité à la commande d'accès de base, être à même de protéger l'information dans le DSA en suivant les procédures de la commande d'accès de base;
- h) s'il est allégué conformité à la commande d'accès simplifié, être à même de protéger l'information dans le DSA selon les procédures de la commande d'accès simplifié;
- i) s'il est allégué conformité à **shadowOperationalBindingID**, être conforme aux procédures de la Rec. UIT-T X.525 | ISO/CEI 9594-9 et à celles de la Rec. UIT-T X.501 | ISO/CEI 9594-2 qui concernent le DOP;
- j) s'il est allégué conformité à **specificHierarchicalBindingID**, être conforme aux procédures de la Rec. UIT-T X.518 | ISO/CEI 9594-4 et de la Rec. UIT-T X.501 | ISO/CEI 9594-2 dans la mesure où elles se rapportent à des liens opérationnels hiérarchiques spécifiques;
- k) s'il est allégué conformité à **non-specificHierarchicalBindingID**, être conforme aux procédures de la Rec. UIT-T X.518 | ISO/CEI 9594-4 et de la Rec. UIT-T X.501 | ISO/CEI 9594-2 dans la mesure où elles se rapportent à des liens opérationnels hiérarchiques non spécifiques;
- l) s'il est allégué conformité à l'usage de contextes dans les RDN, être conforme à la résolution des noms impliquant les contextes telle que définie aux § 9.4 de la Rec. UIT-T X.501 | ISO/CEI 9594-2 et aux § 10.3, 10.4, 10.6, 10.10, 10.11 et 15.5.4 de la Rec. UIT-T X.518 | ISO/CEI 9594-4;
- m) s'il est allégué conformité au contrôle d'accès par règles, être capable de protéger les informations contenues dans un DSA dans le respect des procédures de contrôle d'accès par règles;
- n) s'il est allégué conformité à l'administration de base des services, avoir la capacité de traiter les règles de recherche spécifiées au § 19.3.2 de la Rec. UIT T X.518 | ISO/CEI 9594-4.

13.3 Conformité du fournisseur de la duplication

Pour être conforme à la présente Spécification d'annuaire, un DSA remplissant la fonction de fournisseur de la duplication doit satisfaire aux conditions spécifiées aux § 13.3.1 à 13.3.3.

13.3.1 Conditions de déclaration

Il sera déclaré ce qui suit:

- a) le ou les contextes d'application auxquels la conformité est revendiquée en tant que fournisseur de duplication:

shadowSupplierInitiatedAC, **shadowConsumerInitiatedAC**,
shadowSupplierInitiatedAsynchronousAC, **shadowConsumerInitiatedAsynchronousAC**, et **disp-ip**.

L'implémentation d'un agent DSA revendiquant la conformité en tant que fournisseur de duplication et ne prenant pas en charge le protocole **disp-ip** doit, au minimum, prendre en charge soit le contexte **shadowSupplierInitiatedAC** soit le contexte **shadowConsumerInitiatedAC**. Si l'agent DSA prend en charge le contexte **shadowSupplierInitiatedAC**, il peut, facultativement, prendre en charge le contexte **shadowSupplierInitiatedAsynchronousAC**. Si l'agent DSA prend en charge le contexte **shadowConsumerInitiatedAC**, il peut, facultativement, prendre en charge le contexte **shadowConsumerInitiatedAsynchronousAC**. Si cette implémentation revendique la conformité au protocole **disp-ip**, il faut déclarer si l'implémentation a la capacité d'invoquer l'opération **requestShadowUpdate** ou de répondre à une demande **coordinateShadowUpdate**, ou d'effectuer ces deux actions;

- b) le ou les niveaux de sécurité auxquels est alléguée la conformité (aucune, simple, poussée);
- c) dans quelle mesure la **UnitOfReplication** est prise en charge. On doit déclarer en particulier lesquelles des fonctions facultatives suivantes sont prises en charge:
 - filtrage d'entrée sur **objectClass**;
 - sélection et exclusion des attributs au moyen d'**AttributeSelection**;
 - inclusion de connaissances subordonnées dans la zone copiée;
 - inclusion de connaissances étendues en plus des connaissances subordonnées;
 - sélection et exclusion de valeurs d'attributs fondées sur les contextes.

13.3.2 Conditions statiques

Un DSA doit:

- a) être à même de prendre en charge les contextes d'application dont les syntaxes abstraites sont définies au § 7, ainsi que les protocoles IDM définis au § 10, auxquels la conformité est revendiquée;
- b) prendre en charge les attributs opérationnels de **modifyTimestamp** et **createTimestamp**.

13.3.3 Besoins dynamiques

Un DSA doit:

- a) si la conformité à d'éventuels contextes d'application définis au § 8.2.3 est revendiquée, être conforme au mappage sur les services OSI utilisés, défini au § 8;
- b) être conforme aux procédures de la Rec. UIT-T X.525 | ISO/CEI 9594-9 qui concernent le DISP.

13.4 Conformité du consommateur de duplication

Pour être conforme à la présente Spécification d'annuaire, un DSA remplissant la fonction de consommateur de la duplication doit satisfaire aux conditions spécifiées aux § 13.4.1 à 13.4.3.

13.4.1 Conditions de déclaration

Il sera déclaré ce qui suit:

- a) le ou les contextes d'application auxquels la conformité est revendiquée en tant que consommateur de duplication:

shadowSupplierInitiatedAC, **shadowConsumerInitiatedAC**,
shadowSupplierInitiatedAsynchronousAC, **shadowConsumerInitiatedAsynchronousAC**, et **disp-ip**.

L'implémentation d'un agent DSA revendiquant la conformité en tant que fournisseur de duplication et ne prenant pas en charge le protocole **disp-ip** doit, au minimum, prendre en charge soit le contexte **shadowSupplierInitiatedAC** soit le contexte **shadowConsumerInitiatedAC**. Si l'agent DSA prend en charge le contexte **shadowSupplierInitiatedAC**, il peut, facultativement, prendre en charge le contexte **shadowSupplierInitiatedAsynchronousAC**. Si l'agent DSA prend en charge le contexte **shadowConsumerInitiatedAC**, il peut, facultativement, prendre en charge le contexte **shadowConsumerInitiatedAsynchronousAC**. S'il est allégué conformité avec le protocole **disp-ip**, il faut déclarer si l'implémentation a la capacité d'invoquer l'opération **requestShadowUpdate** ou de répondre à une demande **coordinateShadowUpdate**, ou d'effectuer ces deux actions;

- b) le ou les niveaux de sécurité auxquels est alléguée la conformité (aucune, simple, poussée);

ISO/CEI 9594-5:2005 (F)

- c) si le DSA peut fonctionner comme fournisseur secondaire de la duplication (c'est-à-dire participer à la duplication secondaire en tant que DSA intermédiaire);
- d) si le DSA prend en charge la duplication d'unités de copie superposées.

13.4.2 Conditions statiques

Un DSA doit:

- a) être à même de prendre en charge les contextes d'application dont les syntaxes abstraites sont définies au § 7 et les protocoles IDM définis au § 10, auxquels la conformité est revendiquée;
- b) fournir la prise en charge des attributs opérationnels **modifyTimestamp** et **createTimestamp** si les unités de copie superposées sont prises en charge;
- c) prendre en charge le contrôle du service **copyShallDo**.

13.4.3 Conditions dynamiques

Un DSA doit:

- a) si la conformité à d'éventuels contextes d'application est revendiquée, être conforme au mappage avec les services OSI utilisés, définis au § 8;
- b) être conforme aux procédures de la Rec. UIT-T X.525 | ISO/CEI 9594-9 qui concernent le DISP.

Annexe A

Spécifications communes des protocoles en notation ASN.1

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

CommonProtocolSpecification {joint-iso-itu-t ds(5) module (1) commonProtocolSpecification (35) 5}
DEFINITIONS ::=**BEGIN****-- EXPORTE TOUT --**

*-- Les types et les valeurs définis dans ce module sont exportés afin d'être utilisés dans les autres modules ASN.1
 -- contenus dans les Spécifications d'annuaire et en vue de l'utilisation d'autres applications pour accéder au service
 -- de l'annuaire. D'autres applications peuvent les utiliser pour leur propre compte, mais elles ne doivent pas gêner
 -- les extensions et les modifications nécessaires à la mise à jour et à l'amélioration du service d'annuaire.*

IMPORTS*-- de la Rec.UIT-T X.501 | ISO/CEI 9594-2***opBindingManagement****FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}****establishOperationalBinding, modifyOperationalBinding, terminateOperationalBinding****FROM OperationalBindingManagement opBindingManagement ;**

OPERATION ::= CLASS {
 &ArgumentType OPTIONAL,
 &ResultType OPTIONAL,
 &Errors ERROR OPTIONAL,
 &operationCode Code UNIQUE OPTIONAL }
WITH SYNTAX {
 [ARGUMENT &ArgumentType]
 [RESULT &ResultType]
 [ERRORS &Errors]
 [CODE &operationCode] }
ERROR ::= CLASS {
 &ParameterType,
 &errorCode Code UNIQUE OPTIONAL }
WITH SYNTAX {
 PARAMETER &ParameterType
 [CODE &errorCode] }
Code ::= CHOICE {
 local INTEGER,
 global OBJECT IDENTIFIER }
Invokeld ::= CHOICE {
 present INTEGER,
 absent NULL }
-- codes d'opération pour les protocoles DAP et DSP

| | |
|--------------------------|---------------------------|
| id-opcode-read | Code ::= local : 1 |
| id-opcode-compare | Code ::= local : 2 |
| id-opcode-abandon | Code ::= local : 3 |
| id-opcode-list | Code ::= local : 4 |
| id-opcode-search | Code ::= local : 5 |

ISO/CEI 9594-5:2005 (F)

id-opcode-addEntry Code ::= local : 6
id-opcode-removeEntry Code ::= local : 7
id-opcode-modifyEntry Code ::= local : 8
id-opcode-modifyDN Code ::= local : 9

-- codes d'opération pour le protocole DISP

id-opcode-requestShadowUpdate Code ::= local : 1
id-opcode-updateShadow Code ::= local : 2
id-opcode-coordinateShadowUpdate Code ::= local : 3

-- codes d'opération pour le protocole DOP

id-op-establishOperationalBinding Code ::= local : 100
id-op-modifyOperationalBinding Code ::= local : 102
id-op-terminateOperationalBinding Code ::= local : 101

-- codes d'erreur pour les protocoles DAP et DSP

id-errcode-attributeError Code ::= local : 1
id-errcode-nameError Code ::= local : 2
id-errcode-serviceError Code ::= local : 3
id-errcode-referral Code ::= local : 4
id-errcode-abandoned Code ::= local : 5
id-errcode-securityError Code ::= local : 6
id-errcode-abandonFailed Code ::= local : 7
id-errcode-updateError Code ::= local : 8
id-errcode-dsaReferral Code ::= local : 9

-- code d'erreur pour le protocole DISP

id-errcode-shadowError Code ::= local : 1

-- code d'erreur pour le protocole DOP

id-err-operationalBindingError Code ::= local : 100

DOP-Invokable OPERATION ::= { establishOperationalBinding |
modifyOperationalBinding |
terminateOperationalBinding }

DOP-Returnable OPERATION ::= { establishOperationalBinding |
modifyOperationalBinding |
terminateOperationalBinding }

END -- CommonProtocolSpecification

Annexe B

Protocole OSI en notation ASN.1

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

OSIProtocolSpecification {joint-iso-itu-t ds(5) module (1) oSIProtocolSpecification (36) 5}

DEFINITIONS ::=

BEGIN

-- EXPORTE TOUT --

-- Les types et les valeurs définis dans ce module sont exportés afin d'être utilisés dans les autres modules ASN.1
 -- contenus dans les Spécifications d'annuaire et en vue de l'utilisation d'autres applications pour accéder au service
 -- de l'annuaire. D'autres applications peuvent les utiliser pour leur propre compte, mais elles ne doivent pas gêner
 -- les extensions et les modifications nécessaires à la mise à jour et à l'amélioration du service d'annuaire.

IMPORTS

-- de la Rec.UIT-T X.501 | ISO/CEI 9594-2

commonProtocolSpecification, directoryAbstractService, directoryOSIProtocols,
 enhancedSecurity, informationFramework
 FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}

Name, RelativeDistinguishedName
 FROM InformationFramework informationFramework

OPTIONALLY-PROTECTED
 FROM EnhancedSecurity enhancedSecurity

-- de la Rec.UIT-T X.511 | ISO/CEI 9594-3

SecurityProblem, ServiceProblem, Versions
 FROM DirectoryAbstractService directoryAbstractService

-- de la Rec.UIT-T X.519 | ISO/CEI 9594-5

Invokeld, OPERATION
 FROM CommonProtocolSpecification commonProtocolSpecification

APPLICATION-CONTEXT
 FROM DirectoryOSIProtocols directoryOSIProtocols ;

-- protocole OSI --

OSI-PDU {APPLICATION-CONTEXT:protocol} ::= TYPE-IDENTIFIER.&Type (
 OsiBind { {protocol} } |
 OsiBindResult { {protocol} } |
 OsiBindError { {protocol} } |
 OsiOperation { {protocol.&Operations} } |
 PresentationAbort)

OsiBind {APPLICATION-CONTEXT:Protocols} ::= SET {
 mode-selector [0] IMPLICIT SET {mode-value [0] IMPLICIT INTEGER (1)},
 normal-mode-parameters [2] IMPLICIT SEQUENCE {
 protocol-version [0] IMPLICIT BIT STRING {version-1(0)}
 DEFAULT {version-1},
 calling-presentation-selector [1] IMPLICIT Presentation-selector OPTIONAL,
 called-presentation-selector [2] IMPLICIT Presentation-selector OPTIONAL,

```

presentation-context-definition-list
user-data [4] IMPLICIT Context-list,
  fully-encoded-data CHOICE {
    transfer-syntax-name Transfer-syntax-name OPTIONAL,
    presentation-context-identifier Presentation-context-identifier,
    presentation-data-values CHOICE {
      single-ASN1-type [0] AARQ-apdu {{(Protocols)}} } } } }

```

Presentation-selector ::= OCTET STRING(SIZE (1..4, ..., 5..MAX))

```

Context-list ::= SEQUENCE SIZE (2) OF
SEQUENCE {
  presentation-context-identifier Presentation-context-identifier,
  abstract-syntax-name Abstract-syntax-name,
  transfer-syntax-name-list SEQUENCE OF Transfer-syntax-name }

```

Presentation-context-identifier ::= INTEGER(1..127, ..., 128..MAX)

Abstract-syntax-name ::= OBJECT IDENTIFIER

Transfer-syntax-name ::= OBJECT IDENTIFIER

```

AARQ-apdu {APPLICATION-CONTEXT:Protocols} ::= [APPLICATION 0] IMPLICIT SEQUENCE {
  protocol-version [0] IMPLICIT BIT STRING {version1(0)} DEFAULT {version1},
  application-context-name [1] Application-context-name,
  called-AP-title [2] Name OPTIONAL,
  called-AE-qualifier [3] RelativeDistinguishedName OPTIONAL,
  called-AP-invocation-identifier [4] AP-invocation-identifier OPTIONAL,
  called-AE-invocation-identifier [5] AE-invocation-identifier OPTIONAL,
  calling-AP-title [6] Name OPTIONAL,
  calling-AE-qualifier [7] RelativeDistinguishedName OPTIONAL,
  calling-AP-invocation-identifier [8] AP-invocation-identifier OPTIONAL,
  calling-AE-invocation-identifier [9] AE-invocation-identifier OPTIONAL,
  implementation-information [29] IMPLICIT Implementation-data OPTIONAL,
  user-information [30] IMPLICIT Association-informationBind {{(Protocols)}} }

```

```

Association-informationBind {APPLICATION-CONTEXT:Protocols} ::= SEQUENCE SIZE(1) OF EXTERNAL (
  WITH COMPONENTS {
    identification ( WITH COMPONENTS { syntax ABSENT } ),
    data-value-descriptor ABSENT,
    data-value (CONTAINING TheOsiBind {{(Protocols)}} ) )

```

Application-context-name ::= OBJECT IDENTIFIER

AP-invocation-identifier ::= INTEGER

AE-invocation-identifier ::= INTEGER

Implementation-data ::= GraphicString

```

TheOsiBind {APPLICATION-CONTEXT:Protocols} ::=
  [16] APPLICATION-CONTEXT.&bind-operation.&ArgumentType {{(Protocols)}}

```

```

OsiBindResult {APPLICATION-CONTEXT:Protocols} ::= SET {
  mode-selector [0] IMPLICIT SET {mode-value [0] IMPLICIT INTEGER (1) },
  normal-modeparameters [2] IMPLICIT SEQUENCE {
    protocol-version [0] IMPLICIT BIT STRING {version-1(0)} DEFAULT {version-1},
    responding-presentation-selector [3] IMPLICIT Presentation-selector OPTIONAL,
    presentation-context-definition-result-list [5] IMPLICIT SEQUENCE SIZE (2) OF SEQUENCE {
      result [0] IMPLICIT Result (acceptance),
      transfer-syntax-name [1] IMPLICIT Transfer-syntax-name },
    user-data CHOICE {
      fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        transfer-syntax-name Transfer-syntax-name OPTIONAL,
        presentation-context-identifier Presentation-context-identifier,

```

presentation-data-values
single-ASN1-type CHOICE {
[0] AARE-apdu {{Protocols}} } } }

Result ::= INTEGER {
acceptance (0),
user-rejection (1),
provider-rejection (2) }

AARE-apdu {APPLICATION-CONTEXT:Protocols} ::= [APPLICATION 1] IMPLICIT SEQUENCE {
protocol-version [0]
IMPLICIT BIT STRING {version1(0)} DEFAULT {version1},
application-context-name [1] Application-context-name,
result [2] Associate-result (accepted),
result-source-diagnostic [3] Associate-source-diagnostic,
responding-AP-title [4] Name OPTIONAL,
responding-AE-qualifier [5] RelativeDistinguishedName OPTIONAL,
responding-AP-invocation-identifier [6] AP-invocation-identifier OPTIONAL,
responding-AE-invocation-identifier [7] AE-invocation-identifier OPTIONAL,
implementation-information [29] IMPLICIT Implementation-data OPTIONAL,
user-information [30] IMPLICIT Association-informationBindRes {{Protocols}} }

Association-informationBindRes {APPLICATION-CONTEXT:Protocols} ::= SEQUENCE SIZE(1) OF EXTERNAL (WITH COMPONENTS {
identification (WITH COMPONENTS { syntax ABSENT }),
data-value-descriptor ABSENT,
data-value (CONTAINING TheOsiBindRes {{Protocols}}) })

Associate-result ::= INTEGER {
accepted (0),
rejected-permanent (1),
rejected-transient (2) } (0..2, ...)

Associate-source-diagnostic ::= CHOICE {
acse-service-user [1] INTEGER {
null (0),
no-reason-give (1),
application-context-name-not-supported (2),
calling-AP-title-not-recognized (3),
calling-AP-invocation-identifier-not-recognized (4),
calling-AE-qualifier-not-recognized (5),
calling-AE-invocation-identifier-not-recognized (6),
called-AP-title-not-recognized (7),
called-AP-invocation-identifier-not-recognized (8),
called-AE-qualifier-not-recognized (9),
called-AE-invocation-identifier-not-recognized (10) } (0..10, ...),
acse-service-provider [2] INTEGER {
null (0),
no-reason-given (1),
no-common-acse-version (2) } (0..2, ...)

TheOsiBindRes {APPLICATION-CONTEXT:Protocols} ::=
[17] APPLICATION-CONTEXT.&bind-operation.&ResultType ({Protocols})

OsiBindError {APPLICATION-CONTEXT:Protocols} ::= CHOICE {
normal-mode-parameters SEQUENCE {
protocol-version [0] IMPLICIT BIT STRING {version-1(0)} DEFAULT {version-1},
responding-presentation-selector [3] IMPLICIT Presentation-selector OPTIONAL,
presentation-context-definition-result-list [5] IMPLICIT Result-list OPTIONAL,
provider-reason [10] IMPLICIT Provider-reason OPTIONAL,
user-data CHOICE {
fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
transfer-syntax-name Transfer-syntax-name OPTIONAL,
presentation-context-identifier Presentation-context-identifier,
presentation-data-values CHOICE {
single-ASN1-type [0] AAREerr-apdu {{Protocols}} } } } OPTIONAL } }

AAREerr-apdu {APPLICATION-CONTEXT:Protocols} ::= [APPLICATION 1] IMPLICIT SEQUENCE {
protocol-version [0] IMPLICIT BIT STRING {version1(0)}

| | | |
|-------------------------------------|--|------------|
| | DEFAULT {version1}, | |
| application-context-name | [1] Application-context-name, | |
| result | [2] Associate-result (rejected-permanent..rejected-transient), | |
| result-source-diagnostic | [3] Associate-source-diagnostic, | |
| responding-AP-title | [4] Name | OPTIONAL, |
| responding-AE-qualifier | [5] RelativeDistinguishedName | OPTIONAL, |
| responding-AP-invocation-identifier | [6] AP-invocation-identifier | OPTIONAL, |
| responding-AE-invocation-identifier | [7] AE-invocation-identifier | OPTIONAL, |
| implementation-information | [29] IMPLICIT Implementation-data | OPTIONAL, |
| user-information | [30] | |
| | IMPLICIT Association-informationBindErr {{Protocols}} | OPTIONAL } |

Association-informationBindErr {APPLICATION-CONTEXT:Protocols} ::= SEQUENCE SIZE(1) OF EXTERNAL (WITH COMPONENTS {
 identification (WITH COMPONENTS { syntax ABSENT }),
 data-value-descriptor ABSENT,
 data-value (CONTAINING TheOsiBindErr {{Protocols}}) })

TheOsiBindErr {APPLICATION-CONTEXT:Protocols} ::= [18] APPLICATION-CONTEXT.&bind-operation.&Errors.&ParameterType {{Protocols}}

Result-list ::= SEQUENCE SIZE (2) OF SEQUENCE {
 result [0] IMPLICIT Result,
 transfer-syntax-name [1] IMPLICIT Transfer-syntax-name OPTIONAL,
 provider-reason [2] IMPLICIT INTEGER {
 reason-not-specified (0),
 abstract-syntax-not-supported (1),
 proposed-transfer-syntaxes-not-supported (2) } OPTIONAL }

Provider-reason ::= INTEGER {
 reason-not-specified (0),
 temporary-congestion (1),
 local-limit-exceeded (2),
 called-presentation-address-unknown (3),
 protocol-version-not-supported (4),
 default-context-not-supported (5),
 user-data-not-readable (6),
 no-PSAP-available (7) }

OsiUnbind ::= CHOICE {
 fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
 presentation-context-identifier Presentation-context-identifier,
 presentation-data-values CHOICE {
 single-ASN1-type [0] TheOsiUnbind } } }

TheOsiUnbind ::= [APPLICATION 2] IMPLICIT SEQUENCE {
 reason [0] IMPLICIT Release-request-reason OPTIONAL }

Release-request-reason ::= INTEGER {
 normal (0) }

OsiUnbindResult ::= CHOICE {
 fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
 presentation-context-identifier Presentation-context-identifier,
 presentation-data-values CHOICE {
 single-ASN1-type [0] TheOsiUnbindRes } } }

TheOsiUnbindRes ::= [APPLICATION 3] IMPLICIT SEQUENCE {
 reason [0] IMPLICIT Release-response-reason OPTIONAL }

Release-response-reason ::= INTEGER {
 normal (0) }

OsiOperation {OPERATION:Operations} ::= CHOICE {
 fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
 presentation-context-identifier Presentation-context-identifier,
 presentation-data-values CHOICE {
 single-ASN1-type [0] CHOICE {
 request OsiReq {{Operations}},
 result OsiRes {{Operations}},

```

error    OsiErr {{Operations}},
reject   OsiRej } } }

```

```

OsiReq {OPERATION:Operations} ::= [1] IMPLICIT SEQUENCE {
  invokeld    Invokeld,
  opcode      OPERATION.&operationCode ({Operations}),
  argument    OPERATION.&ArgumentType ({Operations} {@opcode}) }

```

```

OsiRes { OPERATION:Operations} ::= [2] IMPLICIT SEQUENCE {
  invokeld    Invokeld,
  result      SEQUENCE {
    opcode    OPERATION.&operationCode ({Operations}),
    result    OPERATION.&ResultType ({Operations} {@opcode}) } }

```

```

OsiErr {OPERATION:Operations} ::= [3] IMPLICIT SEQUENCE {
  invokeld    Invokeld,
  errcode     OPERATION.&Errors.&errorCode ({Operations}),
  error       OPERATION.&Errors.&ParameterType ({Operations} {@.errcode}) }

```

```

OsiRej ::= [4] IMPLICIT SEQUENCE {
  invokeld    Invokeld,
  problem     CHOICE {
    general    [0] GeneralProblem,
    invoke     [1] InvokeProblem,
    returnResult [2] ReturnResultProblem,
    returnError [3] ReturnErrorProblem } }

```

```

GeneralProblem ::= INTEGER {
  unrecognizedPDU      (0),
  mistypedPDU          (1),
  badlyStructuredPDU  (2) }

```

```

InvokeProblem ::= INTEGER {
  duplicateInvocation      (0),
  unrecognizedOperation    (1),
  mistypedArgument        (2),
  resourceLimitation       (3),
  releaseInProgress        (4) }

```

```

ReturnResultProblem ::= INTEGER {
  unrecognizedInvocation    (0),
  resultResponseUnexpected  (1),
  mistypedResult            (2) }

```

```

ReturnErrorProblem ::= INTEGER {
  unrecognizedInvocation    (0),
  errorResponseUnexpected    (1),
  unrecognizedError         (2),
  unexpectedError           (3),
  mistypedParameter         (4) }

```

```

PresentationAbort ::= CHOICE {
  aru-ppdu ARU-PPDU,
  arp-ppdu ARP-PPDU }

```

```

ARU-PPDU ::= CHOICE {
  normal-mode-parameters [0] IMPLICIT SEQUENCE {
    presentation-context-identifier-list [0] IMPLICIT Presentation-context-identifier-list,
    user-data                             CHOICE {
      fully-encoded-data [APPLICATION 1] IMPLICIT SEQUENCE SIZE (1) OF SEQUENCE {
        presentation-context-identifier Presentation-context-identifier,
        presentation-data-values        CHOICE {
          single-ASN1-type [0] ABRT-apdu } } } } }

```

```

Presentation-context-identifier-list ::=
  SEQUENCE SIZE (1) OF SEQUENCE {
    presentation-context-identifier Presentation-context-identifier,
    transfer-syntax-name            Transfer-syntax-name}

```

ABRT-apdu ::= [APPLICATION 4] IMPLICIT SEQUENCE {
 abort-source ABRT-source }

ABRT-source ::= INTEGER {
 acse-service-user (0),
 acse-service-provider (1) }

ARP-PPDU ::= SEQUENCE {
 provider-reason [0] IMPLICIT Abort-reason OPTIONAL,
 event-identifier [1] IMPLICIT Event-identifier OPTIONAL }

Abort-reason ::= INTEGER {
 reason-not-specified (0),
 unrecognized-ppdu (1),
 unexpected-ppdu (2),
 unexpected-session-service-primitive (3),
 unrecognized-ppdu-parameter (4),
 unexpected-ppdu-parameter (5),
 invalid-ppdu-parameter-value (6) }

Event-identifier ::= INTEGER {
 cp-PPDU (0),
 cpa-PPDU (1),
 cpr-PPDU (2),
 aru-PPDU (3),
 arp-PPDU (4),
 td-PPDU (7),
 s-release-indication (14),
 s-release-confirm (15) }

END --OSIProtocolSpecification

Annexe C

Protocoles d'annuaire OSI en notation ASN.1

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

DirectoryOSIProtocols {joint-iso-itu-t ds(5) module(1) directoryOSIProtocols(37) 5}

DEFINITIONS ::=

BEGIN

-- EXPORTE TOUT --

-- Les types et les valeurs définis dans ce module sont exportés afin d'être utilisés dans les autres modules ASN.1
 -- contenus dans les Spécifications d'annuaire et en vue de l'utilisation d'autres applications pour accéder au service
 -- de l'annuaire. D'autres applications peuvent les utiliser pour leur propre compte, mais elles ne doivent pas gêner
 -- les extensions et les modifications nécessaires à la mise à jour et à l'amélioration du service d'annuaire.

IMPORTS

-- de la Rec.UIT-T X.501 | ISO/CEI 9594-2

commonProtocolSpecification, directoryAbstractService, distributedOperations,
 directoryShadowAbstractService, id-ac, id-as, id-idm, iDMProtocolSpecification,
 opBindingManagement, oSIProtocolSpecification
 FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}

dSAOperationalBindingManagementBind, establishOperationalBinding, modifyOperationalBinding,
 terminateOperationalBinding
 FROM OperationalBindingManagement opBindingManagement

-- de la Rec.UIT-T X.511 | ISO/CEI 9594-3

abandon, addEntry, compare, directoryBind, list, modifyDN, modifyEntry, read, removeEntry, search
 FROM DirectoryAbstractService directoryAbstractService

-- de la Rec.UIT-T X.518 | ISO/CEI 9594-4

chainedAbandon, chainedAddEntry, chainedCompare, chainedList, chainedModifyDN,
 chainedModifyEntry, chainedRead, chainedRemoveEntry, chainedSearch, dSABind
 FROM DistributedOperations distributedOperations

-- de la Rec.UIT-T X.519 | ISO/CEI 9594-5

OPERATION
 FROM CommonProtocolSpecification commonProtocolSpecification

OSI-PDU {}
 FROM OSIProtocolSpecifications oSIProtocolSpecification

-- de la Rec.UIT-T X.525 | ISO/CEI 9594-9

coordinateShadowUpdate, dSAShadowBind, requestShadowUpdate, updateShadow
 FROM DirectoryShadowAbstractService directoryShadowAbstractService ;

-- OSI protocols --

DAP-OSI-PDUs ::= OSI-PDU { directoryAccessAC }

DSP-OSI-PDUs ::= OSI-PDU { directorySystemAC }

DOP-OSI-PDUs ::= OSI-PDU { directoryOperationalBindingManagementAC }

ShadowSupplierInitiatedDISP-OSI-PDUs ::= OSI-PDU { shadowSupplierInitiatedAC }

```

ShadowSupplierInitiatedAsynchronousDISP-OSI-PDUs ::=
    OSI-PDU { shadowSupplierInitiatedAsynchronousAC }

ShadowConsumerInitiatedDISP-OSI-PDUs ::= OSI-PDU { shadowConsumerInitiatedAC }

ShadowConsumerInitiatedAsynchronousDISP-OSI-PDUs ::=
    OSI-PDU { shadowConsumerInitiatedAsynchronousAC }

APPLICATION-CONTEXT ::= CLASS {
    &bind-operation          OPERATION,
    &Operations              OPERATION,
    &applicationContextName OBJECT IDENTIFIER UNIQUE }
WITH SYNTAX {
    BIND-OPERATION          &bind-operation
    OPERATIONS              &Operations
    APPLICATION CONTEXT NAME &applicationContextName }

directoryAccessAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          directoryBind
    OPERATIONS              { read | compare | abandon | list | search
                            | addEntry | removeEntry | modifyEntry | modifyDN }
    APPLICATION CONTEXT NAME id-ac-directoryAccessAC }

directorySystemAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSABind
    OPERATIONS              { chainedRead | chainedCompare | chainedAbandon
                            | chainedList | chainedSearch
                            | chainedAddEntry | chainedRemoveEntry
                            | chainedModifyEntry | chainedModifyDN }
    APPLICATION CONTEXT NAME id-ac-directorySystemAC }

shadowSupplierInitiatedAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSAShadowBind
    OPERATIONS              { updateShadow
                            | coordinateShadowUpdate }
    APPLICATION CONTEXT NAME id-ac-shadowSupplierInitiatedAC }

shadowConsumerInitiatedAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSAShadowBind
    OPERATIONS              { requestShadowUpdate
                            | updateShadow }
    APPLICATION CONTEXT NAME id-ac-shadowConsumerInitiatedAC }

shadowSupplierInitiatedAsynchronousAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSAShadowBind
    OPERATIONS              { updateShadow
                            | coordinateShadowUpdate }
    APPLICATION CONTEXT NAME id-ac-shadowSupplierInitiatedAsynchronousAC }

shadowConsumerInitiatedAsynchronousAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSAShadowBind
    OPERATIONS              { requestShadowUpdate
                            | updateShadow }
    APPLICATION CONTEXT NAME id-ac-shadowConsumerInitiatedAsynchronousAC }

directoryOperationalBindingManagementAC APPLICATION-CONTEXT ::= {
    BIND-OPERATION          dSAOperationalBindingManagementBind
    OPERATIONS              { establishOperationalBinding
                            | modifyOperationalBinding
                            | terminateOperationalBinding}
    APPLICATION CONTEXT NAME id-ac-directoryOperationalBindingManagementAC }

-- syntaxes abstraites --

id-as-directoryAccessAS          OBJECT IDENTIFIER ::= {id-as 1}
id-as-directorySystemAS         OBJECT IDENTIFIER ::= {id-as 2}
id-as-directoryShadowAS         OBJECT IDENTIFIER ::= {id-as 3}
id-as-directoryOperationalBindingManagementAS OBJECT IDENTIFIER ::= {id-as 4}
-- id-as-directoryReliableShadowAS OBJECT IDENTIFIER ::= {id-as 5}

```

```
-- id-as-reliableShadowBindingAS          OBJECT IDENTIFIER ::= {id-as 6}
-- id-as-2or3se                            OBJECT IDENTIFIER ::= {id-as 7}
```

```
id-acseAS                                OBJECT IDENTIFIER ::=
{ joint-iso-itu-t association-control(2) abstract-syntax(1) apdus(0) version(1) }
```

```
-- identificateurs d'objets de contextes d'application
```

```
id-ac-directoryAccessAC                 OBJECT IDENTIFIER ::= {id-ac 1}
id-ac-directorySystemAC                 OBJECT IDENTIFIER ::= {id-ac 2}
id-ac-directoryOperationalBindingManagementAC OBJECT IDENTIFIER ::= {id-ac 3}
id-ac-shadowConsumerInitiatedAC        OBJECT IDENTIFIER ::= {id-ac 4}
id-ac-shadowSupplierInitiatedAC        OBJECT IDENTIFIER ::= {id-ac 5}
-- id-ac-reliableShadowSupplierInitiatedAC OBJECT IDENTIFIER ::= {id-ac 6}
-- id-ac-reliableShadowConsumerInitiatedAC OBJECT IDENTIFIER ::= {id-ac 7}
id-ac-shadowSupplierInitiatedAsynchronousAC OBJECT IDENTIFIER ::= {id-ac 8}
id-ac-shadowConsumerInitiatedAsynchronousAC OBJECT IDENTIFIER ::= {id-ac 9}
-- id-ac-directoryAccessWith2or3seAC      OBJECT IDENTIFIER ::= {id-ac 10}
-- id-ac-directorySystemWith2or3seAC      OBJECT IDENTIFIER ::= {id-ac 11}
-- id-ac-shadowSupplierInitiatedWith2or3seAC OBJECT IDENTIFIER ::= {id-ac 12}
-- id-ac-shadowConsumerInitiatedWith2or3seAC OBJECT IDENTIFIER ::= {id-ac 13}
-- id-ac-reliableShadowSupplierInitiatedWith2or3seAC OBJECT IDENTIFIER ::= {id-ac 14}
-- id-ac-reliableShadowConsumerInitiatedWith2or3seAC OBJECT IDENTIFIER ::= {id-ac 15}
-- id-ac-directoryOperationalBindingManagementWith2or3seAC OBJECT IDENTIFIER ::= {id-ac 16}
```

```
END -- DirectoryOSIProtocols
```

Annexe D

Protocole IDM en notation ASN.1

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

La présente annexe comporte toutes les définitions de type et de valeur ASN.1 applicables dans la présente Spécification d'annuaire, sous la forme du module ASN.1, **IDMProtocolSpecification**.

IDMProtocolSpecification {joint-iso-itu-t ds(5) module (1) iDMProtocolSpecification (30) 5}

DEFINITIONS ::=

BEGIN

-- EXPORTE TOUT --

*-- Les types et les valeurs définis dans ce module sont exportés afin d'être utilisés
-- dans les autres modules ASN.1 contenus dans les Spécifications d'annuaire et en vue de
-- leur utilisation dans d'autres applications pour accéder au service de l'annuaire.
-- D'autres applications peuvent les utiliser pour leur propre compte, mais elles
-- ne doivent pas gêner les extensions et les modifications nécessaires à la mise à jour
-- et à l'amélioration du service d'annuaire.*

IMPORTS

-- de la Rec.UIT-T X.501 | ISO/CEI 9594-2

**certificateExtensions, commonProtocolSpecification, directoryAbstractService, directoryIDMProtocols,
enhancedSecurity**
FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}

-- de la Rec.UIT-T X.509 | ISO/CEI 9594-8

GeneralName
FROM CertificateExtensions certificateExtensions

-- de la Rec.UIT-T X.511 | ISO/CEI 9594-3

SecurityProblem, ServiceProblem, Versions
FROM DirectoryAbstractService directoryAbstractService

-- de la Rec.UIT-T X.519 | ISO/CEI 9594-5

Invokeld, OPERATION
FROM CommonProtocolSpecification commonProtocolSpecification ;

-- classe d'objets informationnels du protocole IDM --

IDM-PROTOCOL ::= CLASS {
 &bind-operation **OPERATION,**
 &Operations **OPERATION,**
 &id **OBJECT IDENTIFIER UNIQUE }**

WITH SYNTAX {
 BIND-OPERATION **&bind-operation**
 OPERATIONS **&Operations**
 ID **&id }**

-- Protocole IDM --

```

IDM-PDU {IDM-PROTOCOL:protocol} ::= CHOICE {
    bind                [0] IdmBind{ {protocol} },
    bindResult          [1] IdmBindResult{ {protocol} },
    bindError           [2] IdmBindError{ {protocol} },
    request             [3] Request{ {protocol.&Operations} },
    result              [4] IdmResult{ {protocol.&Operations} },
    error               [5] Error{ {protocol.&Operations} },
    reject              [6] IdmReject,
    unbind              [7] Unbind,
    abort               [8] Abort,
    startTLS            [9] StartTLS,
    tLSResponse         [10] TLSResponse }

IdmBind {IDM-PROTOCOL:Protocols} ::= SEQUENCE {
    protocolID          IDM-PROTOCOL.&id ({Protocols}),
    callingAETitle      [0] GeneralName OPTIONAL,
    calledAETitle       [1] GeneralName OPTIONAL,
    argument            [2] IDM-PROTOCOL.&bind-operation.&ArgumentType
                        ({Protocols} {@protocolID}) }

IdmBindResult {IDM-PROTOCOL:Protocols} ::= SEQUENCE {
    protocolID          IDM-PROTOCOL.&id ({Protocols}),
    respondingAETitle   [0] GeneralName OPTIONAL,
    result              [1] IDM-PROTOCOL.&bind-operation.&ResultType
                        ({Protocols} {@protocolID}) }

IdmBindError {IDM-PROTOCOL:Protocols} ::= SEQUENCE {
    protocolID          IDM-PROTOCOL.&id ({Protocols}),
    errcode             IDM-PROTOCOL.&bind-operation.&Errors.&errorCode
                        ({Protocols} {@protocolID}),
    respondingAETitle   [0] GeneralName OPTIONAL,
    aETitleError        ENUMERATED {
        callingAETitleNotAccepted (0),
        calledAETitleNotRecognized (1) } OPTIONAL,
    error               [1] IDM-PROTOCOL.&bind-operation.&Errors.&ParameterType
                        ({Protocols} {@protocolID, @errcode}) }

Unbind ::= NULL

Request {OPERATION:Operations} ::= SEQUENCE {
    invokeID           INTEGER,
    opcode             OPERATION.&operationCode ({Operations}),
    argument            OPERATION.&ArgumentType ({Operations} {@opcode}) }

IdmResult {OPERATION:Operations} ::= SEQUENCE {
    invokeID           Invokeld,
    opcode             OPERATION.&operationCode ({Operations}),
    result             OPERATION.&ResultType ({Operations} {@opcode}) }

Error {OPERATION:Operations} ::= SEQUENCE {
    invokeID           INTEGER,
    errcode            OPERATION.&Errors.&errorCode ({Operations}),
    error              OPERATION.&Errors.&ParameterType
                        ({Operations} {@errcode}) }

IdmReject ::= SEQUENCE {
    invokeID           INTEGER,
    reason             ENUMERATED {
        mistypedPDU (0),
        duplicateInvokeIDRequest (1),
        unsupportedOperationRequest (2),
        unknownOperationRequest (3),
        mistypedArgumentRequest (4),
        resourceLimitationRequest (5),
        unknownInvokeIDResult (6),
        mistypedResultRequest (7),
        unknownInvokeIDError (8),
        unknownError (9),
    }
}

```

mistypedParameterError (10) } }

Abort ::= ENUMERATED {
 mistypedPDU (0),
 unboundRequest (1),
 invalidPDU (2),
 resourceLimitation (3),
 connectionFailed (4),
 invalidProtocol (5),
 reasonNotSpecified (6) }

StartTLS ::= NULL

TLSResponse ::= ENUMERATED {
 success (0),
 operationsError (1),
 protocolError (2),
 unavailable (3) }

END -- *IDMProtocolSpecification*

Annexe E

Protocoles IDM d'annuaire en notation ASN.1

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

La présente annexe comporte toutes les définitions de type et de valeur ASN.1 applicables dans la présente Spécification d'annuaire, sous la forme du module ASN.1, **DirectoryIDMProtocols**.

DirectoryIDMProtocols {joint-iso-itu-t ds(5) module(1) directoryIDMProtocols(31) 5}

DEFINITIONS ::=

BEGIN

-- EXPORTE TOUT --

*-- Les types et les valeurs définis dans ce module sont exportés afin d'être utilisés dans les autres modules ASN.1
 -- contenus dans les Spécifications d'annuaire et en vue de leur utilisation dans d'autres applications pour accéder
 -- au service de l'annuaire. D'autres applications peuvent les utiliser pour leur propre compte, mais elles ne doivent
 -- pas gêner les extensions et les modifications nécessaires à la mise à jour et à l'amélioration du service d'annuaire.*

IMPORTS

-- de la Rec.UIT-T X.501 | ISO/CEI 9594-2

**directoryAbstractService, distributedOperations, directoryShadowAbstractService, id-idm,
 iDMProtocolSpecification, opBindingManagement
 FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}**

**establishOperationalBinding, modifyOperationalBinding, terminateOperationalBinding
 FROM OperationalBindingManagement opBindingManagement**

-- de la Rec.UIT-T X.511 | ISO/CEI 9594-3

**abandon, addEntry, compare, directoryBind, list, modifyDN, modifyEntry, read, removeEntry, search
 FROM DirectoryAbstractService directoryAbstractService**

-- de la Rec.UIT-T X.518 | ISO/CEI 9594-4

**chainedAbandon, chainedAddEntry, chainedCompare, chainedList, chainedModifyDN,
 chainedModifyEntry, chainedRead, chainedRemoveEntry, chainedSearch
 FROM DistributedOperations distributedOperations**

-- de la Rec.UIT-T X.519 | ISO/CEI 9594-5

**IDM-PDU, IDM-PROTOCOL
 FROM iDMProtocolSpecification iDMProtocolSpecification**

-- de la Rec.UIT-T X.525 | ISO/CEI 9594-9

**coordinateShadowUpdate, requestShadowUpdate, updateShadow
 FROM DirectoryShadowAbstractService directoryShadowAbstractService ;**

-- Protocoles IDM --

DAP-IDM-PDUs ::= IDM-PDU {dap-ip}

ISO/CEI 9594-5:2005 (F)

```
dap-ip IDM-PROTOCOL ::= {  
  BIND-OPERATION      directoryBind  
  OPERATIONS          { read | compare | abandon | list | search  
                      | addEntry | removeEntry | modifyEntry | modifyDN }  
  ID                  id-idm-dap }
```

DSP-IDM-PDUs ::= IDM-PDU {dsp-ip}

```
dsp-ip IDM-PROTOCOL ::= {  
  BIND-OPERATION      directoryBind  
  OPERATIONS          { chainedRead | chainedCompare | chainedAbandon  
                      | chainedList | chainedSearch  
                      | chainedAddEntry | chainedRemoveEntry  
                      | chainedModifyEntry | chainedModifyDN }  
  ID                  id-idm-dsp }
```

DISP-IDM-PDUs ::= IDM-PDU {disp-ip}

```
disp-ip IDM-PROTOCOL ::= {  
  BIND-OPERATION      directoryBind  
  OPERATIONS          { requestShadowUpdate  
                      | updateShadow  
                      | coordinateShadowUpdate }  
  ID                  id-idm-disp }
```

DOP-IDM-PDUs ::= IDM-PDU {dop-ip}

```
dop-ip IDM-PROTOCOL ::= {  
  BIND-OPERATION      directoryBind  
  OPERATIONS          { establishOperationalBinding  
                      | modifyOperationalBinding  
                      | terminateOperationalBinding }  
  ID                  id-idm-dop }
```

-- identificateurs d'objet de protocole --

```
id-idm-dap      OBJECT IDENTIFIER ::= {id-idm 0}  
id-idm-dsp      OBJECT IDENTIFIER ::= {id-idm 1}  
id-idm-disp     OBJECT IDENTIFIER ::= {id-idm 2}  
id-idm-dop      OBJECT IDENTIFIER ::= {id-idm 3}
```

END -- DirectoryIDMProtocols

Annexe F

Types de liens opérationnels pour l'annuaire

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

La présente annexe comprend tous les identificateurs d'objets ASN.1 assignés pour identifier les types de liens opérationnels utilisés dans la présente Spécification d'annuaire sous la forme du module ASN.1, "DirectoryOperationalBindingTypes".

DirectoryOperationalBindingTypes

{ joint-iso-itu-t ds(5) module (1) directoryOperationalBindingTypes(25) 5 }

DEFINITIONS ::=

BEGIN

-- EXPORTE TOUT --

-- Les types et les valeurs définis dans ce module sont exportés afin d'être utilisés dans les autres modules ASN.1
 -- contenus dans les Spécifications d'annuaire et en vue de l'utilisation d'autres applications pour accéder au service
 -- de l'annuaire. D'autres applications peuvent les utiliser pour leur propre compte, mais elles ne doivent pas gêner les
 -- extensions et les modifications nécessaires à la mise à jour et à l'amélioration du service d'annuaire.

IMPORTS

-- de la Rec.UIT-T X.501 | ISO/CEI 9594-2

id-ob

FROM UsefulDefinitions { joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5 } ;

| | | | |
|---|-------------------|-----|-------------|
| id-op-binding-shadow | OBJECT IDENTIFIER | ::= | { id-ob 1 } |
| id-op-binding-hierarchical | OBJECT IDENTIFIER | ::= | { id-ob 2 } |
| id-op-binding-non-specific-hierarchical | OBJECT IDENTIFIER | ::= | { id-ob 3 } |

END -- DirectoryOperationalBindingTypes

Annexe G

Amendements et corrigenda

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

Cette édition de la présente Spécification d'annuaire inclut le projet d'amendement suivant qui avait été approuvé par l'ISO/CEI:

- Amendement 3 pour maximiser l'alignement entre le système X.500 et le protocole LDAP.

Cette édition de la présente Spécification n'inclut aucun corrigendum technique.

SÉRIES DES RECOMMANDATIONS UIT-T

| | |
|----------------|--|
| Série A | Organisation du travail de l'UIT-T |
| Série D | Principes généraux de tarification |
| Série E | Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains |
| Série F | Services de télécommunication non téléphoniques |
| Série G | Systèmes et supports de transmission, systèmes et réseaux numériques |
| Série H | Systèmes audiovisuels et multimédias |
| Série I | Réseau numérique à intégration de services |
| Série J | Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias |
| Série K | Protection contre les perturbations |
| Série L | Construction, installation et protection des câbles et autres éléments des installations extérieures |
| Série M | Gestion des télécommunications y compris le RGT et maintenance des réseaux |
| Série N | Maintenance: circuits internationaux de transmission radiophonique et télévisuelle |
| Série O | Spécifications des appareils de mesure |
| Série P | Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux |
| Série Q | Commutation et signalisation |
| Série R | Transmission télégraphique |
| Série S | Equipements terminaux de télégraphie |
| Série T | Terminaux des services télématiques |
| Série U | Commutation télégraphique |
| Série V | Communications de données sur le réseau téléphonique |
| Série X | Réseaux de données, communication entre systèmes ouverts et sécurité |
| Série Y | Infrastructure mondiale de l'information, protocole Internet et réseaux de prochaine génération |
| Série Z | Langages et aspects généraux logiciels des systèmes de télécommunication |