



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

**CCITT**

COMITÉ CONSULTATIF  
INTERNATIONAL  
TÉLÉGRAPHIQUE ET TÉLÉPHONIQUE

**X.413**

(11/1988)

SÉRIE X: RÉSEAUX DE COMMUNICATIONS DE  
DONNÉES:

SYSTÈMES DE MESSAGERIE

---

**SYSTÈMES DE MESSAGERIE:  
DÉFINITION DU SERVICE ABSTRAIT  
D'ENREGISTREMENT DE MESSAGES (MS)**

Réédition de la Recommandation du CCITT X.413 publiée  
dans le Livre Bleu, Fascicule VIII.7 (1988)

---

## NOTES

1 La Recommandation X.413 du CCITT a été publiée dans le fascicule VIII.7 du Livre Bleu. Ce fichier est un extrait du Livre Bleu. La présentation peut en être légèrement différente, mais le contenu est identique à celui du Livre Bleu et les conditions en matière de droits d'auteur restent inchangées (voir plus loin).

2 Dans la présente Recommandation, le terme «Administration» désigne indifféremment une administration de télécommunication ou une exploitation reconnue.

© UIT 1988, 2008

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.

## Recommandation X.413

### SYSTÈMES DE MESSAGERIE: DÉFINITION DU SERVICE ABSTRAIT D'ENREGISTREMENT DE MESSAGES (MS)<sup>1)</sup>

(Melbourne, 1988)

La création dans divers pays de services télématiques et de services de messagerie informatisée en mode enregistrement et retransmission, en liaison avec les réseaux publics pour données montre qu'il faut établir des normes facilitant les échanges internationaux de messages entre abonnés à ces services.

Le CCITT,

*considérant*

- (a) que des systèmes de messagerie sont nécessaires;
- (b) qu'il est nécessaire de transférer et de stocker des messages de types différents;
- (c) que la Recommandation X.200 définit le modèle de référence pour l'interconnexion de systèmes ouverts pour les applications du CCITT;
- (d) que les Recommandations X.208, X.217, X.218 et X.219 fournissent les bases pour les applications du CCITT;
- (e) que les Recommandations de la série X.500 spécifient les services et les systèmes d'annuaire;
- (f) que les services et systèmes de messagerie sont spécifiés dans les Recommandations de la série X.400 à savoir les Recommandations X.400, X.402, X.403, X.407, X.408, X.411, X.413 et X.419;
- (g) que le service de messagerie de personne à personne est spécifié dans les Recommandations X.420 et T.330,

*déclare à l'unanimité*

- (1) que la définition du service abstrait d'enregistrement de messages est donnée dans la section 2;
- (2) que les types d'attributs généraux et les types d'actions automatiques générales sont spécifiés dans la section 3;
- (3) que les procédures applicables à l'enregistrement de messages et à la mise en œuvre des accès sont spécifiées dans la section 4.

## SOMMAIRE

### SECTION 1 – Introduction

0	Introduction
1	Portée
2	Références
3	Définitions
4	Abréviations
5	Conventions

---

<sup>1)</sup> La Recommandation X.413 et la norme ISO 10021-5 [Systèmes de traitement de l'information – Communication de texte – MOTIS – Enregistrement de messages - Définition du service abstrait] ont été élaborées en étroite collaboration et sont alignées sur le plan technique, à l'exception des différences indiquées dans l'annexe G.

## SECTION 2 – Définition du service abstrait d'enregistrement de messages

- 6 Modèle d'enregistrement de messages
- 7 Opérations abstraites rattachement et détachement
- 8 Opérations abstraites
- 9 Erreurs abstraites

## SECTION 3 – Types d'attributs généraux et types d'actions automatiques générales

- 10 Présentation générale
- 11 Types d'attributs généraux
- 12 Types d'actions automatiques générales

## SECTION 4 – Procédures applicables à l'enregistrement de messages et à la mise en œuvre des accès

- 13 Présentation générale
- 14 Utilisation du service abstrait du système de transfert de messages
- 15 Fourniture du service abstrait Enregistrement de messages
- 16 Mise en œuvre des accès

*Annexe A* – Assignment formelle des identificateurs d'objets

*Annexe B* – Définition formelle du service abstrait Enregistrement de messages

*Annexe C* – Définition formelle des types d'attributs généraux

*Annexe D* – Définition formelle des types d'actions automatiques générales

*Annexe E* – Définition formelle des limites supérieures du paramètre MS

*Annexe F* – Exemple de l'opération abstraite résumé

*Annexe G* – Différences entre la Recommandation X.413 du CCITT et le texte de la norme 10021-5 de l'ISO/CEI

## SECTION 1 – INTRODUCTION

### **0 Introduction**

La présente Recommandation fait partie d'une série de Recommandations définissant un service de messagerie (MH) dans un environnement de systèmes ouverts décentralisés.

Un service de messagerie assure l'échange de messages entre usagers en mode enregistrement et retransmission. Un message déposé par un usager (l'expéditeur) est transféré au moyen du système de transfert de messages (MTS) puis remis à un ou plusieurs autres usagers (les destinataires).

La présente Recommandation définit le service abstrait d'enregistrement de messages (service abstrait MS) qui permet la recherche de messages à partir d'une mémoire de messages (MS) et le dépôt indirect de messages par le biais du MS dans un système de messagerie (MHS). Le service-abstrait MS assure aussi un service de gestion des messages défini par le service abstrait système de transfert de messages (MTS).

La présente Recommandation a été élaborée conjointement par le CCITT et l'ISO. La norme internationale correspondante de l'ISO est ISO 10021-5. L'annexe G récapitule les différences entre les deux documents.

## 1 Portée

La présente Recommandation définit le service abstrait d'enregistrement de messages. Ce service abstrait est assuré par le protocole d'accès d'enregistrement de messages spécifié dans la Recommandation X.419 lequel est associé au service abstrait MTS défini dans la Recommandation X.411 avec les services élément de service d'opérations distantes (ROSE) définis dans la Recommandation X.219. La notation de syntaxe abstraite applicable aux protocoles de la couche-application utilisée dans la présente Recommandation est définie dans la Recommandation X.208.

D'autres Recommandations définissent d'autres aspects du MHS. La Recommandation X.400 définit les services axés sur l'utilisateur fournis par le MHS. La Recommandation X.402 donne une vue d'ensemble de l'architecture du MHS. La Recommandation X.407 décrit des conventions de définition du service abstrait utilisées dans le MHS. La Recommandation X.420 définit le service abstrait pour la messagerie de personne à personne ainsi que le format des messages de personne à personne.

La section 2 de la présente Recommandation donne la définition du service abstrait d'enregistrement de messages. Le § 6 décrit le modèle d'enregistrement de messages. Le § 7 spécifie la notation de syntaxe abstraite applicable aux opérations abstraites de rattachement et de détachement. Le § 8 spécifie la notation de syntaxe abstraite applicable aux opérations du service abstrait. Le § 9 spécifie la notation de syntaxe abstraite applicable aux erreurs du service abstrait.

La section 3 de la présente Recommandation définit les types d'attributs généraux et les types d'actions automatiques générales liées au MS. Le § 10 donne une présentation générale. Le § 11 spécifie la notation de syntaxe abstraite applicable aux types d'attributs généraux. Le § 12 spécifie la notation de syntaxe abstraite applicable aux types d'actions automatiques générales.

La section 4 de la présente Recommandation décrit les procédures applicables à l'enregistrement des messages et à la mise en œuvre des accès. Le § 13 donne une présentation générale. Le § 14 décrit comment est assuré le service abstrait d'enregistrement de messages. Le § 15 décrit comment le service abstrait de système de transfert de messages est fourni. Le § 16 décrit comment les accès au service d'enregistrement de messages sont mis en œuvre.

Aucun critère n'est spécifié pour la conformité à la présente Recommandation.

## 2 Références

On se reportera à la Recommandation X.402 pour avoir une liste des références.

## 3 Définitions

### 3.1 *Définitions communes pour le MHS*

On trouvera à la Recommandation X.402 une liste des définitions communes applicables au MHS.

### 3.2 *Définitions concernant l'enregistrement des messages*

Aux fins de la présente Recommandation, les définitions suivantes s'appliquent:

3.2.1 **abstract-association (association abstraite):** Lien abstrait entre deux interlocuteurs; dans la présente Recommandation, lien existant entre un UA et un MS pour la fourniture du service-abstrait MS ou entre un MS et un MTS pour la fourniture du service-abstrait MTS.

3.2.2 **abstract-bind-parameters (paramètres de rattachement abstrait):** Paramètres définis dans le présent document et contenus dans l'opération de rattachement abstrait.

3.2.3 **abstract-unbind-parameters (paramètres de détachement abstrait):** Paramètres définis dans le présent document et contenus dans l'opération de détachement abstrait.

3.2.4 **administration port (accès d'administration):** Accès permettant la gestion (pour le MTS) d'un ensemble de services abstraits à l'intérieur du service abstrait MS.

3.2.5 **alert abstract-operation (opération abstraite d'avertissement):** Opération abstraite permettant au MS de signaler à l'UA, moyennant des critères de sélection que des messages ou des rapports sont en attente dans le MS. Ne peut être mise en œuvre que si une association abstraite existe.

- 3.2.6 **attribute (attribut):** Information d'un type particulier apparaissant dans une entrée d'une base d'information.
- 3.2.7 **attribute-type (type d'attribut):** Composant d'un attribut indiquant la classe d'information donnée par cet attribut.
- 3.2.8 **attribute-value (valeur d'attribut):** Exemple particulier de cette classe d'information indiquée par un type d'attribut.
- 3.2.9 **attribute-value-assertion (assertion de valeur d'attribut):** Proposition vraie, fausse ou non définie concernant les valeurs des attributs d'une entrée.
- 3.2.10 **auto-action (action automatique):** Actions qui peuvent être effectuées automatiquement par le MS sur la base d'informations préalablement enregistrées depuis le propriétaire du MS via l'UA.
- 3.2.11 **auto-action-type (type d'action automatique):** Utilisé pour indiquer le type d'action automatique, par exemple avertissement.
- 3.2.12 **auto-alert (avertissement automatique):** Action automatique dans le MS qui déclenche une opération abstraite avertissement ou une autre action par le MS.
- 3.2.13 **auto-forward (retransmission automatique):** Action automatique dans le MS qui déclenche la retransmission automatique par le MS d'un message à un autre destinataire (ou à d'autres destinataires).
- 3.2.14 **child-entry (entrée secondaire):** Entrée, autre que l'entrée principale d'une base d'information. L'entrée maîtresse d'une entrée secondaire peut être soit l'entrée principale soit une autre entrée secondaire selon le nombre de niveaux d'entrée dans chaque cas.
- 3.2.15 **child-sequence-number (numéro d'ordre d'une entrée secondaire):** Numéro d'ordre d'une entrée maîtresse désignant une entrée secondaire. A une entrée maîtresse peuvent être associées plusieurs valeurs pour le numéro d'ordre d'une entrée secondaire selon le nombre d'entrées secondaires.
- 3.2.16 **conditional (C) component [élément conditionnel (C)]:** Élément ASN.1 qui doit être présent dans un exemple de sa classe comme l'indique la présente Recommandation. Voir **grade**.
- 3.2.17 **content-length (longueur du contenu):** Attribut donnant la longueur du contenu d'un message remis (ou un contenu renvoyé).
- 3.2.18 **content-returned (contenu renvoyé):** Attribut signalant qu'un rapport remis (ou un message remis) comporte un contenu renvoyé.
- 3.2.19 **converted EITs (EIT converti):** Attribut identifiant les types de codage du contenu du message après conversion.
- 3.2.20 **creation-time (date-heure de création):** Attribut donnant la date-heure de création (par le MS) d'une entrée.
- 3.2.21 **delete-abstract-operation (opération abstraite suppression):** Opération abstraite utilisée pour supprimer une ou plusieurs entrées d'une base d'information.
- 3.2.22 **delivered-EITs (EIT-remis):** Attribut à plusieurs valeurs donnant des renseignements sur les types de codage d'un message remis.
- 3.2.23 **delivered-message entry (entrée message remis):** Entrée de la base d'information de messages enregistrés provenant d'un message remis.
- 3.2.24 **delivered-report entry (entrée rapport remis):** Entrée de la base d'information de messages enregistrés provenant d'un rapport remis.
- 3.2.25 **entry (entrée):** Ensemble d'informations d'une base de données. On se reportera aux termes entrée principale, entrée maîtresse et entrée secondaire pour une classification plus poussée des entrées.
- 3.2.26 **entry-information (information d'entrée):** Paramètre utilisé dans les opérations abstraites pour transmettre une information sélectionnée à partir d'une entrée.
- 3.2.27 **entry-information-selection (sélection d'information d'entrée):** Paramètre utilisé dans les opérations abstraites pour indiquer quelle information d'une entrée est demandée.
- 3.2.28 **entry-status (état de l'entrée):** Attribut donnant des renseignements sur l'état de traitement de cette entrée. Les valeurs possibles sont: nouvelle, listée ou traitée.
- 3.2.29 **entry-type (type d'entrée):** Attribut signalant si une entrée est associée à un message remis ou à un rapport remis.

- 3.2.30 **fetch abstract-operation (opération abstraite extraction):** Opération abstraite qui permet d'extraire une entrée de la base d'information de messages enregistrés.
- 3.2.31 **fetch-restrictions (restrictions à l'extraction):** Restrictions imposées par l'UA sur le type de message qu'il est prêt à recevoir à la suite d'une extraction. Les restrictions possibles peuvent porter sur la longueur du message, le type de contenu et les types de codage.
- 3.2.32 **filter (filtre):** Paramètre utilisé dans les opérations abstraites pour vérifier une entrée particulière d'une base d'information; l'entrée est conforme ou non à ce paramètre.
- 3.2.33 **filter-item (item-filtre):** Affirmation sur la présence ou la/les valeurs d'un attribut de type particulier de l'entrée à l'étude. Cette affirmation peut être vraie ou fausse.
- 3.2.34 **forwarding-request (demande de retransmission):** Paramètre qui peut être présent dans une opération abstraite dépôt de messages sollicitée par l'UA pour demander de retransmettre un message depuis le MS.
- 3.2.35 **general-attribute (attribut général):** Ensemble d'attributs de MS valables pour tous les types de messages et de rapports, quel que soit le type de contenu. Seuls ces attributs du MS sont explicitement définis dans la présente Recommandation.
- 3.2.36 **general-auto-action (action automatique générale):** Action automatique valable pour tous les types de messages et de rapports, quel que soit le type de contenu. Seules ces actions automatiques sont définies explicitement dans la présente Recommandation.
- 3.2.37 **grade (catégorie):** Défini dans la Recommandation X.402.
- 3.2.38 **indirect-submission port (accès de dépôt indirect):** Accès permettant de fournir le service abstrait de dépôt indirect dans le cadre du service abstrait MS. Le service abstrait de dépôt indirect offre les mêmes possibilités que le service abstrait de dépôt de messages (depuis le service abstrait MTS) et permet en outre de retransmettre des messages stockés dans le MS.
- 3.2.39 **information-base (base d'information):** Objets dans le MS qui stockent les informations concernant le service abstrait MS, par exemple, base d'informations de messages enregistrés qui met en mémoire les messages et les rapports remis dans le MS.
- 3.2.40 **information-base-type (type de base d'information):** Type de base d'informations, par exemple, les messages enregistrés.
- 3.2.41 **limit (limite):** Composant du paramètre sélecteur identifiant le nombre maximal d'entrées sélectionnées à renvoyer à l'issue d'une opération abstraite.
- 3.2.42 **list-abstract-operation (opération abstraite listage):** Opération abstraite permettant de sélectionner les entrées à partir d'une base d'informations et les informations relatives aux attributs demandés à renvoyer pour ces entrées.
- 3.2.43 **listed (liste):** Valeur du paramètre état de l'entrée.
- 3.2.44 **macro (macro):** Voir la Recommandation X.208.
- 3.2.45 **main-entry (entrée principale):** Pour chaque opération abstraite réussie créant des entrées de base d'informations il existe toujours une entrée principale. On peut mettre en mémoire dans des entrées secondaires de nouvelles informations ou des informations plus détaillées résultant de la même opération abstraite.
- 3.2.46 **mandatory (M) component [élément obligatoire (O)]:** Élément ASN.1 qui doit toujours être présent dans un exemple de sa classe. Voir **grade**.
- 3.2.47 **matching (concordance):** Processus consistant à comparer la valeur fournie dans une affirmation de valeur d'attribut à la valeur du type d'attribut indiqué mise en mémoire dans le MS ou à décider si le type d'attribut indiqué est présent.
- 3.2.48 **message retrieval service element (MRSE) (élément de service recherche de message):** Élément de service d'application à l'aide duquel un UA récepteur effectue la recherche de messages provenant d'un MS ou tout autre tâche connexe.
- 3.2.49 **MS (MS):** Enregistrement de messages; sigle utilisé comme abrégé de «fournisseur du service abstrait MS».
- 3.2.50 **MS abstract-service (service abstrait MS):** Ensemble des possibilités que le MS offre à ses usagers au moyen de ses accès.

- 3.2.51 **MS abstract-service-user (usager du service abstrait MS):** Usager du service abstrait MS, c'est-à-dire l'UA.
- 3.2.52 **MS abstract-service-provider (prestataire du service abstrait MS):** MS assurant le service abstrait MS.
- 3.2.53 **MS-user (usager MS):** Abrégé pour «usager du service abstrait MS».
- 3.2.54 **message-submission abstract-operation (opération abstraite dépôt de message):** Opération abstraite permettant à l'UA de déposer un message au MTS via le MS et/ou de retransmettre un message du MS au MTS.
- 3.2.55 **multi-valued-attribute (attribut à plusieurs valeurs):** Attribut auquel peuvent être associées plusieurs valeurs.
- 3.2.56 **new (nouveau):** Valeur du paramètre état de l'entrée.
- 3.2.57 **optional (O) component [élément facultatif (F)]:** Élément ASN.1 qui doit être présent dans un exemple de sa classe à la discrétion de l'objet (c'est-à-dire l'utilisateur) fournissant cet exemple. Voir **grade**.
- 3.2.58 **original-EITs (EIT d'origine):** Attribut identifiant les types de codage d'origine du contenu du message.
- 3.2.59 **override (surclassement):** Composant du paramètre sélecteur indiquant que les restrictions précédemment enregistrées pour cette opération abstraite ne doivent pas s'appliquer à cette opération abstraite précise.
- 3.2.60 **parent-entry (entrée maîtresse):** Une entrée maîtresse à une ou plusieurs entrées secondaires créées à l'issue de la même opération abstraite. Si une entrée maîtresse n'est pas une entrée secondaire d'une autre entrée maîtresse, c'est une entrée principale.
- 3.2.61 **parent-sequence-number (numéro d'ordre d'entrée maîtresse):** Numéro d'ordre d'une entrée secondaire désignant son entrée maîtresse. Il ne peut y avoir qu'un numéro d'ordre d'une entrée maîtresse associé à une entrée secondaire.
- 3.2.62 **partial-attribute-request (demande partielle de valeurs d'attribut):** Composant de l'entrée sélection d'informations permettant de renvoyer uniquement les valeurs sélectionnées d'un attribut ayant plusieurs valeurs.
- 3.2.63 **position (position):** Les positions sont des paramètres pour spécifier la limite d'une gamme.
- 3.2.64 **processed (traité):** Valeur du paramètre état de l'entrée.
- 3.2.65 **range (portée):** Paramètre, utilisé dans les opérations abstraites, pour sélectionner une séquence continue d'entrées à partir d'une base d'informations.
- 3.2.66 **register-MS abstract-operation (opération abstraite enregistrement MS):** Opération abstraite permettant à l'UA d'enregistrer dans le MS certaines informations concernant l'interfonctionnement UA-MS.
- 3.2.67 **registration (enregistrement):** Information enregistrée dans le MS et mise en mémoire (jusqu'à ce qu'elle soit modifiée par l'opération abstraite MS-Enregistrement) entre associations abstraites (voir MS Enregistrement).
- 3.2.68 **registration-identifier (identificateur d'enregistrement):** Identificateur pour un ensemble particulier de paramètres d'enregistrement pour un type d'action automatique.
- 3.2.69 **retrieval port (accès pour recherche):** Accès permettant la recherche d'un ensemble de services abstraits dans service abstrait MS.
- 3.2.70 **returned-content-entry (entrée contenu renvoyé):** Type d'entrée de la base d'informations de messages enregistrés contenant le contenu renvoyé d'un message précédemment déposé.
- 3.2.71 **selector (sélecteur):** Paramètre utilisé dans les opérations abstraites pour sélectionner les entrées d'une base d'information.
- 3.2.72 **sequence-number (numéro d'ordre):** Attribut identifiant sans ambiguïté une entrée. Les numéros d'ordre sont attribués par ordre croissant.
- 3.2.73 **single-valued attribute (attribut à valeur unique):** Attribut auquel ne peut être associée qu'une seule valeur.
- 3.2.74 **span (portée):** Composant de l'opération abstraite Résumé contenant le numéro d'ordre le plus faible et le numéro d'ordre le plus élevé des entrées conformes aux critères de sélection.
- 3.2.75 **stored-messages (messages enregistrés):** Base d'information la plus importante de la présente Recommandation, utilisée pour mettre en mémoire les entrées contenant les messages et les rapports remis au MS par le MTS.

3.2.76 **subscription (abonnement):** Accord à long terme entre le fournisseur ou l'administrateur du MS et les usagers (propriétaires-MS) sur la disponibilité et l'utilisation des caractéristiques facultatives du MS telles que les services et les attributs facultatifs. Dans la présente Recommandation, on suppose que ce mécanisme est assuré sans toutefois prescrire ou offrir une méthode normalisée quelconque pour savoir comment assurer ce mécanisme.

3.2.77 **substring (chaîne secondaire):** Filtre-élément utilisé pour spécifier une chaîne de caractères apparaissant (dans le même ordre donné) dans une valeur d'un attribut.

3.2.78 **summarize abstract-operation (opération abstraite résumé):** Opération abstraite permettant d'avoir un bref aperçu des types et du nombre des entrées mises en mémoire dans une base d'information.

3.2.79 **Synopsis (synopsis):** Attribut spécifique au contenu qui peut être utilisé pour indiquer comment les entrées secondaires, contenant certaines parties du contenu sont liées les unes aux autres et à l'entrée principale. L'attribut doit être spécifié dans la Recommandation qui décrit le type de contenu; on se reportera par exemple au résumé IPM défini dans la Recommandation X.420.

## 4 Abréviations

On trouvera à la Recommandation X.402 une liste des abréviations.

## 5 Conventions

La présente Recommandation utilise les conventions de description énumérées dans les quatre paragraphes qui suivent.

### 5.1 Conventions applicables aux services abstraits

La présente Recommandation utilise les conventions de description suivantes fondées sur l'ASN.1 pour les objectifs indiqués:

- 1) ASN.1 elle-même pour spécifier la syntaxe abstraite des bases d'information et de leurs composants ainsi que des types de données communes.
- 2) La macro ASN.1 PORT et les conventions de définition des services abstraits correspondantes de la Recommandation X.407 pour spécifier l'accès pour recherche.
- 3) Les macros ASN.1 ABSTRACT-BIND, ABSTRACT-UNBIND, ABSTRACT-OPERATION et ABSTRACT-ERROR et les conventions de définition du service abstrait correspondantes de la Recommandation X.407 pour spécifier le service abstrait MS.

Chaque fois que la présente Recommandation décrit une classe de structure de données ayant des composants, chaque composant relève de l'une des **catégories** suivantes:

- 1) **Obligatoire (O)** – Un composant obligatoire doit figurer dans chaque exemple de la classe.
- 2) **Facultative (F)** – Un composant facultatif doit figurer dans un exemple de la classe à la discrétion de l'objet (par exemple l'utilisateur) fournissant cet exemple.
- 3) **Conditionnel (C)** – Un composant conditionnel doit figurer dans un exemple de la classe comme l'indique la présente Recommandation.

### 5.2 Conventions applicables aux types d'attributs utilisés dans le tableau 1/X.413 du § 11

La présente Recommandation utilise les conventions énumérées ci-après pour définir les types d'attributs utilisés dans le service abstrait MS.

Pour la colonne intitulée *A valeur unique/à plusieurs valeurs* on peut avoir les valeurs suivantes:

- |   |                     |
|---|---------------------|
| S | à valeur unique     |
| M | à plusieurs valeurs |

Pour la colonne intitulée *Niveau de fourniture par le MS et accès à l'UA* les valeurs suivantes peuvent se présenter:

- O obligatoire
- F facultatif

Pour les colonnes intitulées *Présence dans l'entrée message remis*, *Présence dans l'entrée rapport remis* et *Présence dans l'entrée message renvoyé*, à chaque type d'attribut est associée l'une des valeurs suivantes:

- P *toujours présent* dans l'entrée parce que:
  - cet attribut est obligatoire aux fins de création par le MS; ou
  - cet attribut est un paramètre obligatoire ou implicite de l'opération abstraite pertinente.
- C *présent de façon conditionnelle* dans l'entrée. Ce paramètre serait présent parce que:
  - il est fourni par le MS et fait l'objet d'un abonnement de la part de l'utilisateur; et
  - il figurait dans un paramètre facultatif de l'opération abstraite pertinente.
- *toujours présent* dans les autres cas.

Pour les colonnes intitulées *Disponible pour listage*, *avertissement* et *Disponible pour résumé*, on peut avoir les valeurs suivantes:

- N non
- Y oui

### 5.3 Conventions applicables aux types d'attributs utilisés dans le tableau 2/X.413 du § 11

La présente Recommandation utilise les conventions énumérées ci-dessous pour définir les types d'attributs applicables au service abstrait MS. On trouvera au § 11 le tableau 2/X.413 récapitulant les types d'attributs.

Pour la colonne intitulée *A valeur unique/à plusieurs valeurs*, on peut avoir les valeurs suivantes:

- S à valeur unique
- M à plusieurs valeurs

Pour la colonne intitulée *Source créée par*, on peut avoir les valeurs suivantes:

- MD Opération abstraite de remise des messages
- MS Enregistrement de messages
- RD Opération abstraite de remise de rapports.

### 5.4 Conventions applicables aux polices de ce caractères pour les textes en général

Dans la présente Recommandation, les termes sont **imprimés en caractères gras** dans le cadre des définitions, rien ne soulignant les autres cas. Les termes qui sont des noms propres sont imprimés en majuscules, ce qui n'est pas le cas pour les termes génériques. Les termes génériques composés de plusieurs mots sont imprimés avec des tirets.

### 5.5 Conventions applicables aux polices de caractères pour les définitions ASN.1

Dans la présente Recommandation, les définitions ASN.1 sont imprimées en caractères différents (**gras**) de ceux utilisés dans le reste du document pour souligner la différence entre le texte normal et les définitions ASN.1. La police de caractères utilisée pour les définitions ASN.1 est composée de caractères d'une taille plus petits que ceux utilisés pour le texte ordinaire. Lorsque les éléments de protocole ASN.1 et les valeurs des éléments sont décrits dans le texte d'accompagnement, leurs noms sont écrits en **caractères gras**.

### 5.6 Règles applicables aux définitions ASN.1

Les définitions ASN.1 apparaissent à la fois dans le corps même du document pour faciliter la présentation et, formellement, dans une annexe, à titre de référence. Si des différences apparaissent entre l'ASN.1 utilisé dans la présentation et celle formellement définie dans l'annexe correspondante, une erreur de spécification est signalée.

## 6 Modèle d'enregistrement de messages

L'enregistrement de messages (MS) est représenté comme un objet atomique qui joue le rôle d'un prestataire de services destinés à un usager du service abstrait MS (c'est-à-dire un agent de l'utilisateur (UA) et d'un utilisateur de services fournis par le système de transfert de messages (MTS)).

Le MS joue un rôle d'intermédiaire entre l'UA et le MTS. Sa fonction principale est d'accepter la remise de messages au nom d'un usager final unique du MHS et de les conserver pour que l'UA de l'utilisateur final puisse par la suite les rechercher. Le MS fournit également à l'UA les services de dépôt indirect de messages et de gestion de messages par l'intermédiaire d'un «passage» jusqu'au MTS. Cela permet au MS d'offrir de nouvelles possibilités en plus de celles qu'offre le dépôt direct des messages au MTA, par exemple la retransmission de messages contenus dans le MS.

Comme l'UA, le MS agit au nom d'un usager final unique du MHS, c'est-à-dire qu'il n'assure pas un service MS commun à plusieurs usagers ou utilisé en partage par eux.

Le MS est décrit à l'aide d'un modèle abstrait pour définir les services assurés par le MS – service abstrait d'enregistrement de messages. La figure 1/X.413 représente le service abstrait MS par rapport à son usager et au service abstrait de système de transfert de messages. Dans cette figure, les cases ouvertes représentent l'utilisation du service abstrait et les cases fermées la fourniture du service abstrait.

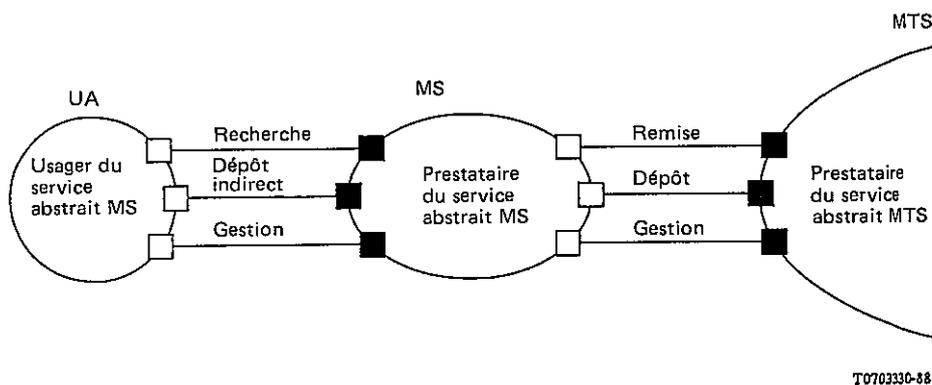


FIGURE 1/X.413

### Service abstrait d'enregistrement des messages

On trouvera à la Recommandation X.407 une introduction et une description de la notion de service abstrait et des conventions de définition qui lui sont associées.

Dans le cadre du système de messagerie, le MS est traité comme objet distinct avec identité unique et comporte une clé distincte (ou un ensemble de clés) à destination de l'UA.

#### 6.1 Objet enregistrement de messages

Le MS est représenté comme un objet atomique. Il assure les services abstraits de l'accès pour recherche MS à l'usager du service abstrait MS. En tant que «substitut» du prestataire du service abstrait MTS, le MS offre également les services abstraits de gestion et de dépôt MTS à l'usager du service abstrait MS (usager du MS) et en tant que «substitut» d'un UA, il utilise les services abstraits de l'accès pour remise MTS, de l'accès pour dépôt MTS et de l'accès pour gestion MTS, comme usager du service abstrait MTS.

La définition formelle applicable à l'objet enregistrement de messages est la suivante:

```
mS OBJECT
  PORTS { retrieval[S],
          indirectSubmission[S],
          administration[S],
          delivery[C],
          submission[C],
          administration[C] }

  ::= id-ot-ms
```

L'utilisateur du MS est également représenté comme un objet. Il utilise les services abstraits de l'accès pour recherche MS, de l'accès pour dépôt indirect et de l'accès pour gestion assurés de façon transparente par le MS.

```
msUser OBJECT
  PORTS { retrieval[C],
          indirectSubmission[C],
          administration[C] }

  ::= id-ot-ms-user
```

## 6.2 Accès pour enregistrement de messages

Un MS fournit les accès pour recherche, dépôt indirect et gestion à l'utilisateur du service abstrait MS. L'ensemble des possibilités offertes par ces accès constitue le service abstrait MS. Les possibilités de recherche de messages ne sont offertes que par le MS. Ces possibilités comprennent l'obtention d'information sur les messages, l'extraction (en totalité ou en partie) et la suppression de messages contenus dans le MS. D'autres possibilités sont offertes pour l'enregistrement de certaines actions automatiques assurées par le MS (par exemple, la retransmission automatique et l'avertissement).

*Remarque* – L'ISO projette de définir d'autres services de gestion de messages effectués par le MS au nom de l'UA pour l'enregistrement des messages entrants et sortants et pour l'autocorrélation entre les avis entrants et les informations d'enregistrement concernant les messages sortants. Ces possibilités n'entrent pas dans le cadre de la présente Recommandation.

Pour fournir à un utilisateur du MS les services décrits au § 6.1, le MS dialogue, au nom de l'utilisateur du MS, avec le service abstrait MTS et agit comme un utilisateur des accès MTS pour remise, dépôt et gestion. Les services abstraits offerts par les accès MTS sont définis au § 8 de la Recommandation X.411.

A l'aide de l'opération de rattachement abstrait, le MS s'assure de l'identité des utilisateurs du MS avant de leur offrir l'une quelconque des possibilités de recherche susmentionnées. De même, les services abstraits MTS doivent s'assurer de l'identité des utilisateurs du service abstrait MTS avant de leur offrir leurs services.

A l'exception du service d'avertissement assuré par l'accès pour recherche et du service de commande de dépôt assuré par l'accès pour dépôt indirect, tous les services fournis par le service abstrait MS sont appelés par l'utilisateur du MS et assurés par le MS.

Des étiquettes de sécurité peuvent être attribuées au MS conformément aux mesures de sécurité en vigueur. Ces mesures peuvent également définir les modalités d'utilisation des étiquettes de sécurité pour appliquer les mesures de sécurité. Si des étiquettes de sécurité sont attribuées au MS, le traitement des messages et des rapports enregistrés portant une étiquette de sécurité de message peut être affecté par les mesures de sécurité en vigueur. Si aucune étiquette de sécurité n'est attribuée au MS, le traitement des messages et des rapports enregistrés s'effectue de façon discrétionnaire.

Si des contextes de sécurité sont établis entre l'UA et le MS, ainsi qu'entre le MS et le MTA, l'étiquette de sécurité attribuée à un message ou à un essai dépend du contexte de sécurité conformément aux mesures de sécurité en vigueur. En l'absence d'un contexte de sécurité, l'attribution d'une étiquette de sécurité à un message ou à un essai est à la discrétion de l'expéditeur.

### 6.2.1 Accès pour recherche

L'**accès pour recherche** est défini comme suit:

```
retrievalPORT
  CONSUMER INVOKES {
    Summarize,
    List,
    Fetch,
    Delete,
    Register-MS }
  SUPPLIER INVOKES {
    Alert }
  ::= id-pt-retrieval
```

Les services abstraits de l'**accès pour recherche** sont décrits en détail dans les § 7 à 9.

### 6.2.2 Accès pour dépôt indirect

L'**accès pour dépôt indirect** est défini comme suit:

```
indirectSubmission PORT ::= submission
```

L'**accès pour dépôt indirect** utilise les services abstraits de l'accès pour dépôt définis au § 8.2 de la Recommandation X.411.

### 6.2.3 Accès pour gestion

L'**accès pour gestion** est défini au § 8.4 de la Recommandation X.411.

Le MS ne dialogue pas avec le service abstrait de modification des pouvoirs. Si l'utilisateur du MS a besoin de mettre à jour son document d'accréditation, l'opération abstraite MS enregistrement est utilisée. Voir le § 8.6.

## 6.3 Modèle d'information

Le présent alinéa décrit le modèle d'information utilisé par le MS. Il représente les **bases d'information** composées d'**entrées** qui elles-mêmes sont composées d'**attributs**.

### 6.3.1 Bases d'information

Le MS met en mémoire et tient à jour des **bases d'information**. Une **base d'information** du MS est une «base de données» contenant toutes les **entrées** représentant des objets constitutifs d'une ou de catégories particulières.

La présente Recommandation définit et décrit la **base d'information des messages enregistrés**. Elle contient les informations provenant des messages et des rapports remis au MS par l'intermédiaire de l'accès MTS pour remise. Elle est décrite au § 6.4.

*Remarque* – Un prochain addendum à la partie pertinente de la norme de l'ISO définira d'autres bases d'information pour l'enregistrement, appelées activation et désactivation, qui n'entrent pas dans le cadre de la présente Recommandation.

```
InformationBase ::= INTEGER {
  stored-messages      (0),
  inlog                (1),
  outlog               (2) } (0..ub-information-bases)
```

### 6.3.2 Entrées

Chaque **base d'information** est structurée en une suite d'**entrées**. Une **entrée** représente un objet unique (par exemple, un message remis) dans la **base d'information**.

Chaque entrée est identifiée à l'aide de son **numéro d'ordre**, unique dans une **base d'information**, et créé par le MS, au fur et à mesure que de nouvelles entrées sont créées. Dans une **base d'information**, le MS crée les **numéros d'ordre** par ordre croissant sans répétition; ces numéros ne sont jamais réutilisés.

```
SequenceNumber ::= INTEGER(0..ub-messages)
```

*Remarque* – Par exemple, le MS peut choisir d'attribuer des **numéros d'ordre** en utilisant le temps pour obtenir une granularité suffisante permettant de garantir l'unicité.

### 6.3.3 Attributs

#### 6.3.3.1 Introduction

Une **entrée** est composée d'un ensemble d'**attributs** (voir la description de la figure 2/X.413).

Chaque **attribut** donne une information sur les données ou provenant des données auxquelles correspond l'**entrée**. A titre d'exemple de ce type d'information on peut citer le numéro d'**ordre** de l'**entrée** proprement dite et la **date-heure** de création.

Un **attribut** comprend un **type d'attribut** identifiant la classe d'information donnée par un **attribut** et la ou les **valeurs d'attribut** correspondantes qui sont des exemples particuliers de cette classe figurant dans l'**entrée**.

```
Attribut ::= SEQUENCE {  
    type      Attribute-type,  
    values    SEQUENCE (SIZE 1 .. ub-attribute-values) OF ANY -- DEFINED BY type -- }
```

*Remarque 1* – Ainsi, par exemple, dans une entrée message remis (décrit au § 6.4) le **type d'attribut** pourrait être la **priorité** du message et la **valeur d'attribut** correspondante pourrait être **urgent**.

Tous les **attributs** d'une **entrée** doivent provenir de **types d'attributs** distincts.

Pour certains **types d'attributs**, un **attribut** peut ne contenir qu'une seule **valeur d'attribut**. Ce **type d'attribut** est dit à **valeur unique**. Pour d'autres, un **attribut** peut contenir une ou plusieurs **valeurs d'attributs**, toutes appartenant au même type de données ASN.1. Ce **type d'attribut** est dit à **plusieurs valeurs**. Le **type d'attribut**, à **valeur unique** ou à **plusieurs valeurs**, est indiqué lorsque le **type d'attribut** est défini (voir le § 6.3.3.2).

*Remarque 2* – Ainsi, par exemple, le **type d'attribut** pour l'**attribut nom de l'expéditeur** (décrit au § 11.2.28) est un attribut à **valeur unique** alors que celui pour **noms d'autres destinataires** (décrit au § 11.2.29) est un attribut à **plusieurs valeurs**.

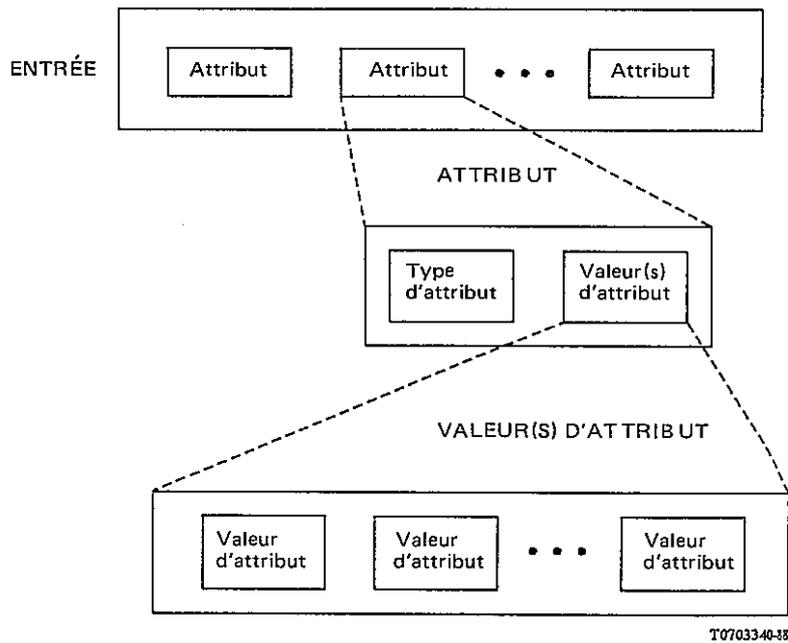


FIGURE 2/X.413  
Composants d'une entrée

### 6.3.3.2 Type d'attribut

Certains **types d'attributs** seront normalisés au niveau international. D'autres **types d'attributs** seront définis par les autorités administratives nationales et des organisations privées. C'est-à-dire qu'un certain nombre d'autorités distinctes seront responsables de l'attribution des types d'attributs et qu'elles devront s'assurer que chaque type d'attribut est distinct de tous les autres types attribués. A cet effet, chaque **type d'attribut** est identifié à l'aide d'un identificateur d'objet lorsque le **type d'attribut** est défini.

AttributeType ::= OBJECT IDENTIFIER

Certains **types d'attributs** de caractère général pour la base d'information de messages enregistrés sont définis dans le § 11 de la présente Recommandation. Ces **types d'attributs** sont appelés **types d'attributs généraux** et les attributs de ce type sont des **attributs généraux**.

### 6.3.3.3 Valeurs d'attributs

Définir un **type d'attribut** suppose également que l'on spécifie le type de données ASN.1 auquel chaque valeur de ces attributs doit être conforme. Le type de données d'une **valeur d'attribut** du **type d'attribut** est défini à l'aide de l'identificateur d'objet pour le **type d'attribut**.

### 6.3.3.4 Définition du type d'attribut et de la macro ATTRIBUTE

La définition d'un **type d'attribut** fait intervenir:

- l'affectation d'un identificateur d'objet au **type d'attribut**;
- l'indication du type de données ASN.1 d'une **valeur d'attribut**;
- le fait d'indiquer si un **attribut** de ce **type d'attribut** peut avoir plus d'une valeur;
- le fait d'indiquer si un **attribut** de ce **type d'attribut** peut être utilisé pour filtrage sur la base des relations égalité, sous-chaînes et/ou classement (voir le § 8.1.2).

*Remarque* – Un filtre peut toujours vérifier la présence ou l'absence dans une entrée d'un **attribut** d'un **type d'attribut** particulier.

La macro ASN.1 suivante est utilisée pour définir un **type d'attribut**. La définition formelle de cette macro est donnée dans la Recommandation X.501; elle est décrite ici pour aider le lecteur.

ATTRIBUTE MACRO ::= BEGIN

TYPE NOTATION	::= AttributeSyntax Multivalued   empty
VALUE NOTATION	::= value (VALUE OBJECT IDENTIFIER)
AttributeSyntax	::= "WITH ATTRIBUTE-SYNTAX" SyntaxChoice
SyntaxChoice	::= value (ATTRIBUTE-SYNTAX) Constraint   type MatchTypes
Constraint	::= ("ConstraintAlternative")   empty
ConstraintAlternative	::= StringConstraint   IntegerConstraint
StringConstraint	::= "SIZE" ("SizeConstraint")   empty
SizeConstraint::=	SingleValue   Range
SingleValue	::= value (INTEGER)
Range	::= value (INTEGER) ".." value (INTEGER)
IntegerConstraint	::= ("Range")
MatchTypes	::= "MATCHES FOR" Matches   empty
Matches	::= Match Matches   Match
Match	::= "EQUALITY"   "SUBSTRINGS"   "ORDERING"
Multivalued	::= "SINGLE VALUE"   "MULTIVALUE"   empty

END

La correspondance entre les différentes parties de la définition telles qu'elles sont indiquées ci-dessus et les différents éléments de la notation introduits par la macro **ATTRIBUTE** est la suivante:

- MACRO-value** – L'**identificateur d'objet** utilisé pour identifier un attribut.
- Attribute-syntax (syntaxe d'attribut)** – Indique la syntaxe qui a été choisie.

- c) **Syntax-choice (choix de syntaxe)** – Indique si l'attribut est défini de façon externe ou de façon interne. La syntaxe de tous les attributs définis dans la présente Recommandation est définie de façon interne, c'est-à-dire que l'on choisit le **type match types**.
- d) **Multivalued (à plusieurs valeurs)** – Indique s'il s'agit d'un attribut à valeur unique ou d'un attribut à plusieurs valeurs.
- e) **Match-types (types de concordance)** – Donnent le type de données du contenu de l'attribut et décrit si l'attribut peut être rendu compatible («MATCHES FOR») pour une relation **égalité** («EQUALITY»), pour **chaînes** («SUBSTRINGS») et **classement** («ORDERING»). Si la production est vide, alors aucune règle de mise en correspondance n'est définie.

Dans le cadre de la présente Recommandation, les mises en correspondance sont limitées comme suit:

- i) **EQUALITY (égalité)** s'applique à n'importe quelle syntaxe d'attribut. La valeur présentée doit être conforme au type de données de la syntaxe d'attribut;
- ii) **SUBSTRING (chaîne secondaire)** s'applique à n'importe quelle syntaxe d'attribut avec un type de données en **chaîne**. La valeur présentée doit être une séquence («SEQUENCE OF»), chacun de ces éléments étant conforme au type de données; et
- iii) **ORDERING (commande)** est applicable à n'importe quelle syntaxe d'attribut pour laquelle une règle peut être définie. Règle qui permettrait à une valeur présentée d'être décrite comme inférieure, égale ou plus grande qu'une valeur cible. La valeur présentée doit être conforme au type de données de la syntaxe d'attribut. Le MS utilise cela pour le type de données INTEGER (entier) et UTCTime (temps UTC).

Les choix et les paramètres restants de la macro **ATTRIBUTE** ne sont pas utilisés dans la présente Recommandation.

#### 6.3.4 Entrées principales, entrées maîtresses et entrées secondaires

Bien que les entrées d'une seule et même base de données soient généralement indépendantes les unes des autres, le modèle d'information MS leur permet d'être liées les unes aux autres. Une entrée, une **entrée secondaire**, peut être rattachée à une autre, son **entrée maîtresse**, dans une relation de type arborescente. Une entrée qui n'est pas une **entrée secondaire** est une **entrée principale**.

Cette relation est définie au moyen de deux **attributs généraux** particuliers:

- a) **parent-sequence-number (numéro d'ordre d'entrée maîtresse)** – Cet attribut à valeur unique donne le numéro d'ordre de l'**entrée maîtresse** d'une **entrée secondaire**. Il ne figure pas dans une **entrée principale**. Cette définition est donnée au § 11.2.30.
- b) **child-sequence-numbers (numéro d'ordre d'entrée secondaire)** – Cet attribut à plusieurs valeurs donne les numéros d'ordre de toutes les **entrées secondaires** d'une **entrée maîtresse**. Il ne figure pas dans une entrée qui n'est pas une **entrée maîtresse**. Cette définition est donnée au § 11.2.1.

Les opérations abstraites du service abstrait MS (voir le § 8) n'agissent par défaut que sur les **entrées principales**. Certaines peuvent être orientées pour agir sur toutes les entrées, tant les **entrées principales** que les **entrées secondaires**. En particulier, l'argument d'une opération abstraite suppression (voir le § 8.5) ne peut choisir que des **entrées principales**, auquel cas l'**entrée principale** et toutes ses entrées secondaires et les entrées secondaires de ses entrées secondaires seront également supprimées.

*Remarque* – Ce concept permet, par exemple, que les parties-corps d'un message de personne à personne qui contient un message retransmis (pour plus de détails se reporter au § 19.1 de la Recommandation X.420) soient représentées par telle ou telle **entrée secondaire**. L'**attribut général de contenu** de l'**entrée principale** comprendra la totalité du **contenu**, de sorte que les données représentant la **partie-corps** de ce message sont logiquement présentes dans plus d'une **entrée**.

#### 6.4 Messages enregistrés

La **base d'information de messages enregistrés** joue le rôle d'un dépositaire d'informations provenant des opérations abstraites remise de messages et remise de rapports de l'accès pour remise. Elle contient des entrées pour les messages remis (**entrées messages remis**) d'un nombre illimité de types de contenu ainsi que des entrées pour les rapports (**entrées rapports remis**). Une entrée d'une **base d'information de messages enregistrés** est créée par le MS chaque fois qu'un message ou un rapport est remis au MS. Pour plus de renseignements, se reporter aux § 11 et 15 sur ces entrées et sur leur mode de création.

Pour obtenir les informations sur le contenu d'un message, le MS doit connaître la syntaxe et la sémantique du contenu indiquées par l'attribut type de contenu. En général, une instance particulière du MS connaît zéro, un ou plusieurs types de contenu. Lorsqu'un MS rencontre un message dont il connaît mal le type de contenu, il n'est pas en mesure de créer des attributs spécifiques de type de contenu dans l'entrée du message.

Un message remis ou un avis entrant peut donner lieu à la création d'une entrée principale et d'un ou de plusieurs niveaux d'entrées secondaires. Le seul cas défini dans la présente Recommandation est celui où un avis de non-remise contient un contenu renvoyé (**l'entrée rapport remis** est l'entrée principale, le contenu renvoyé étant son entrée secondaire, c'est-à-dire, **l'entrée contenu renvoyé**).

Les règles de subdivision du contenu d'un message en plusieurs entrées varient selon le type de contenu. Un **attribut synopsis (résumé d'attribut)** spécifique au contenu peut être utilisé pour indiquer comment l'entrée principale et les entrées secondaires correspondantes sont liées les unes aux autres. Lorsque cet attribut est défini, il figure dans la Recommandation qui définit le type de contenu proprement dit. L'**attribut synopsis (résumé)** est créé par le MS.

*Remarque* – Pour le service de messagerie de personne à personne (Recommandation X.420) les messages emboîtés de personne à personne contenus dans un message de personne à personne sont représentés chacun par une entrée secondaire. Le type d'attribut synopsis messagerie de personne à personne est un exemple d'un **type d'attribut synopsis (résumé)** fonction du contenu.

Une propriété importante d'une entrée dans les messages enregistrés est **l'état de l'entrée**. Cet attribut est créé et mis à jour par le MS. Il peut prendre l'une des valeurs suivantes:

- a) **New (nouveau)** – Le message n'a été ni **listé** par l'UA ni traité automatiquement par le MS.
- b) **Listed (listé)** – Les informations concernant le message ont été renvoyées à l'UA par une opération abstraite listage ou extraction mais le message n'a pas encore été **traité**.
- c) **Processed (traité)** – Soit un UA «a entièrement extrait» le message, soit le MS a effectué une action automatique sur ce message. (A noter que certaines actions automatiques conduisent à la suppression du message.) La définition exacte de «entièrement extrait» est spécifique au contenu et elle est spécifiée dans l'attribut de contenu (voir la Recommandation correspondante sur le contenu).

**L'état de l'entrée** d'un avis de remise/de non-remise est **traité** lorsque l'enveloppe du rapport remis est recherchée.

La définition de l'attribut **état de l'entrée** est la suivante:

```
EntryStatus ::= INTEGER {  
    new                (0),  
    listed             (1),  
    processed          (2) }
```

## 6.5 Actions automatiques

### 6.5.1 Introduction

Ce paragraphe décrit le cadre des **actions automatiques** qui peuvent être enregistrées avec le MS.

Une **action automatique** est une action qui aura lieu automatiquement chaque fois que les critères d'enregistrement associés auront été satisfaits. Le MS peut voir extérieurement le résultat d'une action qui a été appelée. Les **actions automatiques** sont enregistrées dans le MS à l'aide de l'opération abstraite MS-enregistrement (voir le § 8.6).

Chaque classe d'**action automatique** est identifiée à l'aide d'un **type d'action automatique**. A chaque enregistrement d'une **action automatique**, correspond un **paramètre d'enregistrement d'action automatique**, ce sont les paramètres dont le MS a besoin pour effectuer automatiquement l'**action automatique** enregistrée. L'enregistrement d'une **action automatique** nécessite l'utilisation d'un **identificateur d'enregistrement d'action automatique** pour identifier l'enregistrement particulier.

```
AutoActionRegistration ::= SEQUENCE {  
    type                AutoActionType,  
    registration-identif  [0] INTEGER (1 .. ub-per-auto-action) DEFAULT 1,  
    registration-parameter [1] ANY DEFINED BY type }
```

### 6.5.2 Type d'action automatique

Certains types d'**actions automatiques** seront normalisés au niveau international. D'autres **types d'actions automatiques** seront définis par les autorités administratives nationales et des organisations privées. C'est-à-dire qu'un certain nombre d'autorités distinctes seront responsables de l'attribution des types d'actions automatiques et qu'elles devront garantir que chacun de ces types soit distinct de tous les autres **types d'actions automatiques** attribués, ce qui se fait en identifiant chaque **type d'action automatique** à l'aide d'un identificateur d'objet lorsque le **type d'action automatique** est défini.

AutoActionType ::= OBJECT IDENTIFIER

Certains **types d'actions automatiques** de caractère général sont définis au § 12 de la présente Recommandation. Ces **types d'actions automatiques** sont des **types d'actions automatiques généraux** et les **actions automatiques** de ce type sont des **actions automatiques générales**.

### 6.5.3 Paramètre d'enregistrement d'action automatique

Pour définir un **type d'action automatique**, il faut également spécifier le type de données ASN.1 auquel le **paramètre d'enregistrement d'action automatique** doit être conforme. Le type de données d'un **paramètre d'enregistrement** est défini à l'aide de l'identificateur d'objet pour le **type d'action automatique**.

### 6.5.4 Définition du type d'action automatique et macro AUTO-ACTION

La définition d'un **type d'action automatique** suppose:

- a) l'attribution d'un identificateur d'objet au **type d'action automatique**;
- b) l'indication du type de données ASN.1 du paramètre d'enregistrement d'action automatique.

On peut (mais pas nécessairement) utiliser la macro ASN.1 suivante pour définir un **type d'action automatique**:

```
AUTO-ACTION MACRO ::=
BEGIN
    TYPE NOTATION          ::= Registration
    VALUE NOTATION         ::= value (VALUE OBJECT IDENTIFIER)
    Registration           ::= "REGISTRATION PARAMETER IS" type
END
```

La correspondance entre les différentes parties de la définition, indiquées ci-dessus, et les différents éléments de la notation introduits par la macro **AUTO-ACTION** est la suivante:

- a) **Registration (enregistrement)** – Donne le type de données des paramètres d'enregistrement associés à une action automatique.
- b) **Value (valeur) – Identificateur d'objet** utilisé pour identifier l'action automatique.

*Remarque* – Rien n'est prévu dans la macro pour définir l'interaction (éventuelle) entre différents enregistrements de la même **action automatique** en des **actions automatiques** différentes.

## 6.6 Retransmission de messages

L'utilisateur du MS utilise l'opération abstraite Remise de messages et les paramètres qui lui sont associés définis au § 8.2 de la Recommandation X.411 pour demander qu'un message mis en mémoire dans le MS soit explicitement retransmis à d'autres usagers.

Le **paramètre demande de retransmission** est défini à l'aide de la macro **EXTENSION** définie elle-même au § 9 de la Recommandation X.411 comme suit:

```
forwarding-request EXTENSION
    SequenceNumber
    CRITICAL FOR SUBMISSION
    ::= 36
```

Si le **numéro de séquence** fourni ne correspond pas à celui d'une entrée dans l'**information de base** des **messages enregistrés** ou correspond à une entrée qui ne convient pas pour la retransmission, ceci est déclaré en utilisant l'erreur-abstraite **demande-incohérente** du § 8.2.2.7 de la Recommandation X.411.

## 7 Opérations abstraites rattachement et détachement

### 7.1 Opération abstraite-rattachement

L'**opération abstraite-rattachement du rattachement-MS** établit un lien entre les accès pour dépôt indirect, recherche et gestion de l'utilisateur du MS (utilisateur) et le MS (prestataire). Le demandeur (du **rattachement-MS**) est l'utilisateur du MS, le répondeur étant le MS lui-même. Le **rattachement-MS** est défini comme suit:

```
MSBind ::= ABSTRACT-BIND
        TO { indirectSubmission[S], retrieval[S], administration[S] }
        BIND
            ARGUMENT      MSBindArgument
            RESULT         MSBindResult
            BIND-ERROR    MSBindError
```

Une seule association abstraite, peut exister à un moment donné quelconque entre le MS et l'utilisateur du MS.

#### 7.1.1 Argument-abstrait-rattachement

Les paramètres de l'**argument-abstrait-rattachement** sont utilisés pour identifier, vérifier et définir le contexte de sécurité pour un usager du service abstrait MS. Ils contiennent également un ensemble de restrictions applicables aux entrées à renvoyer à l'issue de l'opération abstraite extraction et également une demande pour être informé des types d'actions automatiques, des types d'attributs et des types de contenus assurés par le MS.

La définition de ces paramètres est la suivante:

```
MSBindArgument ::= SET {
    initiator-name           ORAddressAndOrDirectoryName,
    initiator-credentials    [2] initiatorCredentials,
    security-context         [3] IMPLICIT SecurityContext OPTIONAL,
    fetch-restrictions       [4] Restrictions OPTIONAL -- default is none --,
    ms-configuration-request [5] BOOLEAN DEFAULT FALSE }
```

- 1) **Initiator-name (Nom du demandeur)** (C) – Cet argument, contenant le nom du demandeur de l'association, est fourni par le demandeur. Il est défini plus avant au § 8.1.1.1.1 de la Recommandation X.411.
- 2) **Initiator-credentials (Pouvoirs du demandeur)** (O) – Ce paramètre contient l'**accréditation** du demandeur de l'association. Il doit être créé par le demandeur de l'association abstraite.

**Les pouvoirs du demandeur** peuvent être utilisés par le répondeur pour vérifier l'identité du demandeur (voir la Recommandation X.509).

Si l'on a recours à une **authentification simple**, les **pouvoirs du demandeur** comportent un mot de passe simple.

Si l'on a recours à une **authentification forte**, l'**accréditation du demandeur** comporte un **jeton du demandeur-rattachement** et, à titre facultatif, un **certificat du demandeur**. Le **jeton du demandeur-rattachement** et le **certificat du demandeur** sont définis plus avant au § 8.1.1.1.2 de la Recommandation X.411. L'accréditation du demandeur de l'utilisateur du MS peut différer de celle qui est utilisée dans le **MTS-rattachement** défini au § 8.1.1.1.2 de la Recommandation X.411.

- 3) **Security-context (Contexte de sécurité)** (F) – Ce paramètre identifie le **contexte de sécurité** sur lequel le demandeur de l'association abstraite se propose d'agir. Il est créé par le demandeur de l'association abstraite. Le **contexte de sécurité** est défini plus avant au § 8.1.1.1.3 de la Recommandation X.411.

Le **contexte de sécurité** comporte une ou plusieurs **étiquettes de sécurité** définissant le niveau de confidentialité des interactions qui peuvent exister entre l'utilisateur du service abstrait MS et le service abstrait MS pendant la durée de l'association abstraite conformément aux **mesures de sécurité** en vigueur. Le **contexte de sécurité** sera autorisé par les **étiquettes de sécurité de l'utilisateur** enregistrées de l'utilisateur du service abstrait MS et par les **étiquettes de sécurité** associées au MS.

En l'absence de ce paramètre, il n'y a pas de **contexte de sécurité** entre l'utilisateur du service abstrait MS et le service abstrait MS et le niveau de confidentialité des interactions qui peuvent intervenir entre l'utilisateur du service abstrait MS et le service abstrait MS est laissé à la discrétion du demandeur du service abstrait.

- 4) **Fetch-restrictions (restrictions d'extraction)** (F) – Il s'agit des restrictions portant sur les entrées à renvoyer à l'issue de l'opération abstraite recherche. Les **restrictions d'extraction** restent en vigueur jusqu'au déclenchement d'une opération abstraite-détachement.

En l'absence de cet argument, la valeur par défaut est qu'il n'y a pas lieu d'appliquer des **restrictions d'extraction**.

Cet argument comprend les composants suivants:

```
Restrictions ::= SET {
    allowed-content-types          [0] SET SIZE (1 .. ub-content-types) OF OBJECT IDENTIFIER
        -- default is no restriction --,
    allowed-EITs                  [1] MS-EITs OPTIONAL -- default is no restriction --,
    maximum-content-length        [2] ContentLength OPTIONAL -- default is no restriction -- }
```

- a) **Allowed-content-types (Types de contenu autorisés)** (C) – Types de contenus que l'utilisateur du service abstrait MS est prêt à accepter à l'issue d'une opération abstraite extraction. Tout message ayant un type de contenu autre que ceux spécifiés ne sera pas retransmis et donnera lieu à la signalisation d'une erreur, sauf si l'opération abstraite extraction a explicitement passé outre à la restriction.

En l'absence de ce composant, la valeur par défaut est qu'il n'y a pas lieu d'appliquer de restrictions d'extraction aux types de contenus.

- b) **Allowed-EITs (EIT autorisés)** (C) – Types de codage que l'utilisateur du service abstrait MS est prêt à accepter à l'issue d'une opération abstraite extraction. Si un message contient des types de codage autres que ceux spécifiés, il est procédé à un filtrage afin de ne pas renvoyer les parties du type de codage non autorisées avec le texte du message. Si le message ne contient que des types de codage non autorisés, on signalera une erreur. Il n'est procédé à aucun filtrage si l'opération abstraite extraction a explicitement passé outre à la restriction.

```
MS-EITs ::= SET SIZE (1 .. ub-encoded-information-types) OF MS-EIT
```

```
MS-EIT ::= OBJECT IDENTIFIER
```

En l'absence de ce composant, la valeur par défaut est qu'il n'y a pas lieu d'appliquer de **restrictions d'extraction** aux types de codage.

- c) **Maximum-content-length (longueur maximale du contenu)** (C) – Longueur maximale du contenu que l'utilisateur du service abstrait MS est prêt à accepter à l'issue d'une opération abstraite extraction. Tout message dont la **longueur du contenu** dépasse celle spécifiée ne sera pas retransmis et donnera lieu à la signalisation d'une erreur, sauf si l'opération abstraite extraction a explicitement passé outre à la restriction.

En l'absence de ce composant, la valeur par défaut est qu'aucune **restriction d'extraction** ne doit s'appliquer à la **longueur de contenu**.

- 5) **MS-configuration-request (demande de configuration MS)** (C) – La **demande de configuration MS** est faite pour obtenir des informations concernant les actions automatiques et les attributs facultatifs assurés par le MS.

En l'absence de ce composant, la valeur par défaut est fausse, indiquant qu'aucune demande de ce type n'est faite.

### 7.1.2 *Résultat abstrait-rattachement*

Les paramètres **résultat abstrait-rattachement** sont les suivants:

```
MSBindResult ::= SET {
    responder-credentials          [2] ResponderCredentials,
    available-auto-actions         [3] SET SIZE (1 .. ub-auto-actions) OF AutoActionType OPTIONAL,
    available-attribute-types      [4] SET SIZE (1 .. ub-attributes-supported) OF Attribute Type
        OPTIONAL,
    alert-indication              [5] BOOLEAN DEFAULT FALSE,
    content-types-supported        [6] SET SIZE (1 .. ub-content-types) OF OBJECT IDENTIFIER
    content-types-supported        [6] OPTIONAL }
```

- 1) **Responder-credentials (pouvoirs du répondeur)** (O) – Ce paramètre contient les pouvoirs du répondeur de l'association abstraite. Il doit être créé par le répondeur de l'association abstraite.  
Le paramètre **pouvoirs du répondeur** peut être utilisé par le demandeur pour vérifier l'identité du répondeur (voir la Recommandation X.509).  
Si l'on a recours à une **authentification simple**, les **pouvoirs du répondeur** comportent un **mot de passe simple** associé au répondeur.  
Si l'on a recours à une **authentification forte**, les **pouvoirs du répondeur** comportent un **jeton du répondeur-rattachement** et, à titre facultatif, un **certificat du répondeur** créés tous les deux par le répondeur de l'association abstraite. Le **jeton du répondeur-rattachement** et le **certificat du répondeur** sont définis plus avant au § 8.1.1.1.2.2 de la Recommandation X.411.
- 2) **Available-auto-actions (actions automatiques disponibles)** (C) – Spécifient l'ensemble de toutes les **actions automatiques** possibles assurées par le MS (pas uniquement celles demandées par l'utilisateur du service abstrait MS). Ce paramètre n'existe que si une **demande de configuration MS** est faite.
- 3) **Available-attribute-types (types d'attributs disponibles)** (C) – Spécifient l'ensemble de tous les attributs facultatifs fournis par le MS. Ce paramètre n'est présent que si une **demande de configuration MS** est faite.
- 4) **Alert-indication (indication d'avertissement)** (C) – Si la valeur de ce paramètre est vraie, c'est qu'il y a eu un **avertissement** depuis la dernière **indication d'avertissement** positive.
- 5) **Content-types-supported (types de contenus acceptés)** (C) – Spécifient un ensemble d'identificateurs d'objets définissant les **types de contenus** que connaît le MS. Ce paramètre n'est présent que si une **demande de configuration MS** est faite.

### 7.1.3 Erreurs-abstraites-rattachement

Il y a deux erreurs possibles définies par l'**accès pour recherche**, à savoir: une **erreur d'authentification** et un **contexte de sécurité inacceptable**.

La définition des erreurs est la suivante:

```
MSBindError ::= ENUMERATED {
    authentication-error           (0),
    unacceptable-security-context  (1),
    unable-to-establish-association (2) }
```

- 1) **Authentication-error (erreur d'authentification)** (C) – Cette erreur indique qu'une association abstraite ne peut être établie parce que les **pouvoirs** du demandeur ne sont pas acceptables ou mal spécifiés.  
L'**erreur d'authentification** n'a pas de paramètres.
- 2) **Unacceptable-security-context (contexte de sécurité inacceptable)** (C) – Cette erreur indique que le répondeur ne peut accepter le **contexte de sécurité** proposé par le demandeur de l'association abstraite.  
L'erreur contexte de sécurité inacceptable n'a pas de paramètres.
- 3) **Unable-to-establish-association (incapacité à créer une association)** (C) – Cette erreur indique que le répondeur a rejeté la tentative du demandeur visant à créer une association abstraite.  
L'erreur d'incapacité à créer une association n'a pas de paramètres.

### 7.2 Opération abstraite-détachement

L'**opération abstraite-détachement-détachement MS** met fin à l'association abstraite. Le déclenchement d'une **opération abstraite-détachement** conduit à l'assouplissement de toutes les **restrictions d'extraction** spécifiées dans l'argument de l'**opération abstraite-liaison**. Il n'y a pas d'argument, de résultat ou d'erreur associés à l'**opération abstraite-détachement**.

```
MSUnbind ::= ABSTRACT-UNBIND
    FROM { indirectSubmission[S], retrieval[S], administration[S] }
```

## 8 Opérations abstraites

Le présent paragraphe définit les **opérations abstraites** suivantes disponibles à l'accès pour Recherche:

- a) résumé;
- b) listage;
- c) extraction;
- d) suppression;
- e) enregistrement-MS;
- f) avertissement.

Le MS est le prestataire du service abstrait MS de chacune de ces **opérations abstraites**. En ce qui concerne la définition formelle de l'accès pour recherche on se reportera au § 6.2.

Les opérations abstraites peuvent être exécutées de manière asynchrone, sous réserve des conditions suivantes. Les opérations abstraites suppression et enregistrement-MS ne seront pas exécutées tant que toutes les opérations abstraites en suspens n'auront pas été menées à bien. En outre, ces opérations abstraites sont exécutées dans l'ordre selon lequel elles sont appelées et doivent être menées à bien avant que d'autres opérations abstraites ne soient exécutées. Cela étant et sachant que les opérations abstraites listage et extraction modifient l'état de l'entrée d'un message, les résultats des opérations abstraites résumé, listage et extraction peuvent ne pas avoir un caractère déterministe.

### 8.1 Types de données communes utilisées dans les opérations abstraites

Ce paragraphe définit un certain nombre de types de données communes utilisées dans plusieurs des **opérations abstraites** définies au § 8. Beaucoup de ces **opérations abstraites** utilisent également des entrées et des attributs définis au § 6.3.

Les types de données communes définis dans la présente Recommandation sont les suivants:

- a) gamme;
- b) filtre;
- c) sélecteur;
- d) sélection d'information d'entrée;
- e) information d'entrée.

#### 8.1.1 Gamme

Un paramètre **gamme** est utilisé pour choisir une séquence continue d'entrées d'une base d'information.

```
Range ::= CHOICE {
    sequence-number-range    [0] NumberRange,
    creation-time-range      [1] TimeRange }

NumberRange ::= SEQUENCE {
    from[0] SequenceNumber OPTIONAL -- omitted means no lower bound --,
    to      [1] SequenceNumber OPTIONAL -- omitted means no upper bound -- }

TimeRange ::= SEQUENCE {
    from[0] CreationTime OPTIONAL -- omitted means no lower bound --,
    to      [1] CreationTime OPTIONAL -- omitted means no upper bound -- }

CreationTime ::= UTCTime
```

Les composants du paramètre **gamme** ont la signification suivante:

- 1) **Sequence-number-range (gamme-numéro d'ordre)** (C); et
- 2) **Creation-time-range (gamme-date-heure de création)** (C) – Ces deux paramètres identifient la séquence continue d'entrées à sélectionner. Le paramètre **gamme-numéro d'ordre** est exprimé sans forme de **numéros d'ordre** et le paramètre **gamme-date-heure de création** est exprimé sans forme de **dates-heures de création**. Le paramètre **date-heure de création** indique la date et l'heure auxquelles le MS a créé l'entrée en cours. Les paramètres **numéro d'ordre** d'entrées successives sont toujours dans l'ordre ascendant mais plusieurs entrées adjacentes peuvent avoir la même **date et la même heure de création**. Les paramètres **gamme-numéro** et **gamme-date-heure** ont la signification suivante:

- a) **From (à partir de) (F)** – Il s'agit de la limite inférieure de la **gamme**.

En l'absence de ce composant, la valeur par défaut est **pas de limite inférieure**; la sélection commence par le premier message reçu (**numéro d'ordre** le moins élevé) dans la base d'information.

- b) **To (jusqu'à) (F)** – Il s'agit de la limite supérieure de la **gamme**.

En l'absence de ce composant, la valeur par défaut est **pas de limite supérieure**; la sélection s'achève par le dernier message reçu (**numéro d'ordre** le plus élevé) dans la base d'information.

## 8.1.2 *Filtres*

### 8.1.2.1 *Filtre*

Un paramètre **filtre** fait subir un essai à une entrée donnée; l'essai est soit positif, soit négatif. Le **filtre** est exprimé sous forme d'affirmations concernant la présence ou la valeur de certains attributs de l'entrée; le résultat est positif uniquement si la valeur est **vraie**.

```
Filter ::= CHOICE {
    item      [0] FilterItem,
    and       [1] SET SIZE (1 .. ub-nested-filters) OF Filter,
    or        [2] SET SIZE (1 .. ub-nested-filters) OF Filter,
    not       [3] Filter }
```

Un **filtre** est soit un **filter-item (item-filtre)**, soit une expression faisant intervenir des **filtres** plus simples assemblés qui utilisent des opérateurs logiques **and (et)**, **or (ou)** et **not (non)**.

Lorsque le **filtre** est:

- a) un **item**, il est **vrai** si et uniquement si le **filter-item (item-filtre)** correspondant est **vrai**;

- b) un **and (et)**, il est vrai sauf si l'un des **filtres** du **SET (ensemble)** est **faux**;

*Remarque* – Ainsi, s'il n'y a pas de **filtres** dans le **SET (ensemble)** le **and (et)** a comme résultat **vrai**.

- c) un **or (ou)**, il est **faux** sauf si l'un des **filtres** de **SET (ensemble)** est **vrai**;

*Remarque* – Ainsi, s'il n'y a pas de **filtres** dans le **SET (ensemble)**, le **or (ou)** a comme résultat **faux**.

- d) un **not (non)**, il est **vrai** si et uniquement si le **filtre** est **faux**.

### 8.1.2.2 *Item-filtre*

Un **item-filtre** est une affirmation sur la présence ou la/les valeur(s) d'un attribut d'un type particulier de l'entrée soumise à l'essai. Chaque affirmation de ce type est soit **vraie** soit **fausse**.

```
FilterItem ::= CHOICE {
    equality      [0] AttributeValueAssertion,
    substrings   [1] SEQUENCE {
        type      AttributeType,
        strings   SEQUENCE SIZE (1 .. ub-attribute-values) OF CHOICE {
            initial [0] ANY -- DEFINED BY type --,
            any     [1] ANY -- DEFINED BY type --,
            final   [2] ANY -- DEFINED BY type -- },
        greater-or-equal [2] AttributeValueAssertion,
        less-or-equal  [3] AttributeValueAssertion,
        present       [4] AttributeType,
        approximate-match [5] AttributeValueAssertion }
```

Chaque item-filtre comporte un type d'attribut identifiant l'attribut particulier concerné.

Toute affirmation sur la valeur de cet attribut n'est évaluée que si le type d'attribut est défini et si la/les valeur(s) d'attributs prétendue(s) sont du type de données défini pour les valeurs d'attributs de cet attribut.

Les affirmations sur la valeur d'un attribut sont évaluées en mettant l'attribut sur EQUALITY (égalité), SUBSTRINGS (chaînes secondaires) et ORDERING (commande) comme défini au § 6.3.3.4.

Lorsque le **item-filtre** affirme:

- a) **equality (égalité)**, il est **vrai** si et uniquement s'il y a une valeur de l'attribut égale à celle indiquée;

- b) **substrings (chaînes secondaires)**, il est **vrai** si et uniquement s'il y a une valeur de l'attribut dans laquelle les **chaînes secondaires** spécifiées apparaissent dans l'ordre donné. Les **chaînes secondaires** ne doivent pas se chevaucher et peuvent (mais pas nécessairement) être séparées des extrémités de la valeur d'attribut et l'une de l'autre par zéro, un ou plusieurs éléments de **chaîne**.

Le premier caractère au **début**, s'il existe, doit concorder avec le premier caractère de la valeur d'attribut; le dernier caractère à la **fin**, s'il existe, doit concorder avec le dernier caractère de la valeur d'attribut. **N'importe quel** caractère, s'il existe, peut concorder avec n'importe quelle chaîne secondaire de la valeur d'attribut;

- c) **greater-or-equal (plus grand ou égal)**, il est **vrai** si et uniquement si le classement relatif place la valeur fournie *après* l'une quelconque des valeurs de l'attribut;
- d) **less-or-equal (inférieur ou égal)**, il est **vrai** si et uniquement si le classement relatif place la valeur fournie *avant* l'une quelconque des valeurs de l'attribut;
- e) **present (présent)**, il est **vrai** si et uniquement si cet attribut est présent dans l'entrée;
- f) **approximate-match (concordance approximative)**, il est **vrai** si et uniquement s'il y a une valeur de l'attribut qui concorde avec celle qui est affirmée par quelques algorithmes de concordance approximative localement définis (par exemple, variations pour l'épellation, concordance phonétique, etc.). Il n'y a aucune directive spécifique pour la concordance approximative dans la présente Recommandation. Si la concordance approximative n'est pas assurée, ce **filter-item (item-filtre)** doit être traité comme concordant par **equality (égalité)**.

*Remarque* – Si aucune règle de concordance n'est indiquée dans la définition de l'attribut, cela signifie que seule la présence d'un attribut peut être vérifiée dans un **item-filtre**.

### 8.1.2.3 Affirmation de valeur d'attribut

Une **affirmation de valeur d'attribut** est une proposition - **vraie, fausse, ou non définie** - concernant les valeurs d'une entrée. Elle fait intervenir un type d'attribut et une valeur d'attribut:

```
AttributeValueAssertion ::= SEQUENCE {
    type      AttributeType,
    value     ANY DEFINED BY type }
```

Elle est:

- a) **undefined (non définie)**, dans les cas suivants:
- 1) le type d'attribut n'est pas présent dans l'entrée;
  - 2) la définition du type d'attribut ne peut être rendue compatible pour une relation égalité ou classement;
  - 3) la valeur d'attribut n'est pas conforme aux types de données des valeurs d'attributs;
- b) **true (vraie)**, si l'entrée contient un attribut de ce type d'attributs dont l'une des valeurs d'attributs correspond à cette valeur d'attribut;
- c) **false (fausse)**, dans tous les autres cas.

### 8.1.3 Sélecteur

Un paramètre **sélecteur** est utilisé pour sélectionner les entrées d'une base de données. La sélection se fait en trois temps. Dans un premier temps, l'ensemble de toutes les entrées de la base d'information peut être limité à un ensemble contigu particulier en spécifiant sa gamme. Dans un deuxième temps, les entrées de cet ensemble peuvent être sélectionnées en spécifiant un filtre auquel doit se conformer l'entrée retenue. Dans un troisième temps, une limite peut être fixée pour le nombre d'entrées ainsi sélectionnées; dans ce cas, ce sont les entrées dont les numéros d'ordre sont les moins élevés qui sont sélectionnées.

```
Selector ::= SET {
    child-entries [0] BOOLEAN DEFAULT FALSE,
    range         [1] Range OPTIONAL -- default is unbounded --,
    filter        [2] Filter OPTIONAL -- default is all entries within the specified range --,
    limit         [3] INTEGER (1 .. ub-messages) OPTIONAL,
    override      [4] OverrideRestrictions OPTIONAL -- default is that any fetch-restrictions in
                    force do apply -- }
```

Les composantes d'un **sélecteur** ont les significations suivantes:

- 1) **Child-entries (entrées secondaires)** (F) – Si la valeur de ce paramètre est **fausse**, seules les entrées principales sont prises en compte pour la sélection. Si la valeur de ce paramètre est  **vraie**, les entrées principales et secondaires sont prises en compte pour la sélection.

En l'absence de ce composant, la valeur par défaut est *seules les entrées principales sont prises en compte*.

- 2) **Range (gamme)** (F) – La notation de syntaxe abstraite du paramètre **gamme** est donnée au § 8.1.1.

En l'absence de ce composant, la valeur par défaut est *détaché*.

- 3) **Filter (filtre)** (F) – La notation de syntaxe abstraite du paramètre **filtre** est donnée au § 8.1.2.

En l'absence de ce composant, la valeur par défaut est *toutes les entrées de la gamme spécifiée*.

- 4) **Limit (limite)** (F) – Ce paramètre permet de fixer une limite supérieure pour le nombre d'entrées à sélectionner.

En l'absence de ce composant, toutes les entrées sélectionnées seront renvoyées.

*Remarque* – Le rôle essentiel de la limite est d'éviter d'avoir des résultats trop volumineux dus à une mauvaise sélection à l'issue d'une opération abstraite. Cette limite peut également être utilisée pour renvoyer un nombre exact d'ensembles d'informations convenant à tel ou tel dispositif de sortie.

- 5) **Override (annulation)** (F) – Si l'annulation de l'une des **restrictions d'extraction** est demandée, le/les composant(s) correspondant(s) des **restrictions d'extraction** doit (doivent) être présent(s).

```
OverrideRestrictions ::= BIT STRING {  
    overrideContentTypesRestriction      (0),  
    overrideEITsRestriction              (1),  
    overrideContentLengthRestriction     (2) } (SIZE (1 .. ub-information-bases))
```

Les bits des **annulations de restrictions** ont la signification suivante:

- a) **Override-content-types-restriction (annulation de restriction sur les types de contenu)** (O) – Ce bit doit être mis sur 1 pour annuler la **restriction portant sur les types de contenus**.

Si ce bit est mis à 0 (zéro), les **restrictions portant sur les types de contenus** telles qu'elles sont spécifiées dans l'opération abstraite-rattachement s'appliqueront.

- b) **Override-EITs-restriction (annulation de restriction sur EIT)** (O) – Ce bit doit être mis sur 1 pour annuler la **restriction portant sur les types de codage**.

Si ce bit est mis à 0 (zéro), les **restrictions portant sur les types de codage**, telles qu'elles sont spécifiées dans l'opération abstraite-rattachement s'appliqueront.

- c) **Override-content-length-restriction (annulation de restriction sur la longueur du contenu)** (O) – Ce bit doit être mis sur 1 pour annuler la **restriction portant sur la longueur du contenu**.

Si ce bit est mis à 0 (zéro), les **restrictions portant sur la longueur du contenu**, telles qu'elles sont spécifiées dans l'opération abstraite-rattachement s'appliqueront.

En l'absence du paramètre **annulation de restrictions** la valeur par défaut est toutes les **restrictions d'extraction**, telles qu'elles sont spécifiées dans l'opération abstraite-rattachement, s'appliqueront.

#### 8.1.4 Sélection de l'information d'une entrée

Un paramètre **sélection de l'information d'une entrée** indique quelle information d'une entrée est demandée.

```
EntryInformationSelection ::= SET SIZE (0..ub-per-entry) OF AttributeSelection
```

Un ensemble vide indique qu'une information est demandée au sujet de l'entrée proprement dite, plutôt que sur les attributs de celle-ci.

```
AttributeSelection ::= SET {  
    type      AttributeType,  
    from[0] INTEGER (1 .. ub-attribute-values) OPTIONAL -- used if type is multi valued --,  
    count    [1] INTEGER (1 .. ub-attribute-values) OPTIONAL -- used if type is multi valued -- }
```

Les composants du paramètre **sélection d'information d'une entrée** ont la signification suivante:

- 1) **Type (type) (O)** – Indique le type d'attribut de l'attribut.
- 2) **From (à partir de) (C)** – Lorsqu'un attribut a plusieurs valeurs, ce nombre entier donne la position relative de la première valeur à renvoyer. S'il spécifie une valeur supérieure à celle qui est présente dans l'attribut, aucune valeur n'est renvoyée. Ce composant n'est présent que si le type d'attribut est du type à plusieurs valeurs. S'il est omis, les valeurs commençant à la première valeur sont renvoyées.
- 3) **Count (compte) (C)** – Lorsqu'un attribut a plusieurs valeurs, ce nombre entier donne le nombre de valeurs à renvoyer. S'il y a moins de valeurs que de valeurs **compte** présentes dans l'attribut, toutes les valeurs sont renvoyées. Cet attribut n'est présent que si le type d'attribut est du type à plusieurs valeurs. S'il est omis, le nombre de valeurs n'est pas limité.

#### 8.1.5 Informations d'une entrée

Un paramètre **informations d'une entrée** achemine les informations sélectionnées provenant d'une entrée.

```
EntryInformation ::= SEQUENCE {
    sequence-number    SequenceNumber,
    attributes          SET SIZE (1..ub-per-entry) OF Attribute OPTIONAL }
```

Les composants du paramètre **informations d'une entrée** ont la signification suivante:

- 1) **Sequence-number (numéro d'ordre) (O)** – Numéro d'ordre identifiant l'entrée (voir le § 6.3.2.2).
- 2) **Attributes (attributs) (F)** – Ensemble des attributs sélectionnés d'une entrée. Lorsque cela est explicitement demandé au moyen d'une demande d'attribut partiel, un attribut sélectionné, défini comme étant à plusieurs valeurs, peut contenir un sous-ensemble de toutes les valeurs d'attribut de cet attribut, telles qu'elles sont mises en mémoire dans l'entrée. Ce paramètre est absent si les informations provenant des messages sélectionnés ne sont pas demandées, par exemple, lorsque l'utilisateur du service abstrait MS veut connaître uniquement les numéros d'ordre des messages sélectionnés.

#### 8.2 Opération abstraite résumé

L'**opération abstraite résumé** renvoie les comptes résumés des entrées sélectionnées dans une base d'information. En plus de ces résumés, l'opération renvoie également un compte des entrées sélectionnées, ainsi que leur numéro d'ordre le moins élevé et leur numéro d'ordre le plus élevé. Zéro, un ou plusieurs résumés individuels peuvent être demandés.

L'**opération abstraite résumé** n'aboutira que si la base d'information autorise l'accès conformément au contexte de sécurité et aux mesures de sécurité en vigueur.

Les attributs qui peuvent être utilisés pour les résumés sont limités. Pour les attributs généraux de la base d'information de messages enregistrés, les restrictions sont données dans le tableau 1/X.413.

```
Summarize ::= ABSTRACT-OPERATION
    ARGUMENT      SummarizeArgument
    RESULT        SummarizeResult
    ERRORS {
        AttributeError,
        InvalidParametersError,
        RangeError,
        SecurityError,
        SequenceNumberError,
        ServiceError }
```

*Remarque* – On trouvera à l'annexe F un exemple d'opération abstraite résumé.

##### 8.2.1 Argument résumé

```
SummarizeArgument ::= SET {
    information-base-type [0] InformationBase DEFAULT stored-messages,
    selector              [1] Selector,
    summary-requests     [2] SEQUENCE SIZE (1..ub-summaries) OF AttributeType OPTIONAL
    -- absent if no summaries are requested -- }
```

Les composants de l'**argument résumé** ont la signification suivante:

- 1) **Information-base-type (type de base d'information)** (F) – Spécifie la **base d'information** appelée par l'opération abstraite (voir le § 6.3.1).  
En l'absence du composant **type de base d'information**, la valeur par défaut est messages enregistrés.
- 2) **Selector (sélecteur)** (O) – Ensemble de critères de sélection pour déterminer les entrées qui devront être résumées (voir le § 8.1.3).
- 3) **Summary-requests (demandes de résumés)** (F) – Séquence de types d'attributs pour lesquels des résumés sont demandés. Ce paramètre n'est présent que si un résumé est demandé.

### 8.2.2 *Résultat de l'opération résumé*

Si la demande aboutit, le **résultat de l'opération résumé** doit être renvoyé.

```
SummarizeResult ::= SET {  
    next          [0] SequenceNumber OPTIONAL,  
    count         [1] INTEGER (0 . . ub-messages) -- of the entries selected --,  
    span          [2] Span OPTIONAL -- of the entries selected, omitted if count is zero --,  
    summaries     [3] SEQUENCE SIZE (1 . . ub-summaries) OF Summary OPTIONAL }
```

Les composants du paramètre **résultat de l'opération résumé** ont les significations suivantes:

- 1) **Next (suivant)** (C) – Ce paramètre est renvoyé dans le cas où le nombre d'entrées sélectionnées aurait été plus grand en l'absence de la limite spécifiée dans le sélecteur. Le composant contient le numéro d'ordre de l'entrée suivante qui aurait été sélectionnée.
- 2) **Count (compte)** (O) – Il s'agit d'un nombre entier donnant le compte des entrées répondant aux critères de sélection.
- 3) **Span (fourchette)** (C) – Ce paramètre contient le numéro d'ordre le moins élevé et le numéro d'ordre le plus élevé des entrées répondant au critère de sélection. Il est absent s'il n'y a pas de telles entrées.

```
Span ::= SEQUENCE {  
    lowest        [0] SequenceNumber,  
    highest       [1] SequenceNumber }
```

Les composants du paramètre **fourchette** ont la signification suivante:

- a) **Lowest (inférieur)** (O) – Il s'agit du point de départ de la **fourchette**, exprimé sous forme d'un numéro d'ordre (voir le § 6.3.2.2).
  - b) **Highest (supérieur)** (O) – Il s'agit du point final de la **fourchette**, exprimé sous forme d'un numéro d'ordre (voir le § 6.3.2.2).
- 4) **Summaries (résumés)** (C) – Un **résumé** est renvoyé pour chaque **demande de résumé**. Les **résumés** sont renvoyés dans l'ordre dans lequel ils ont été demandés.

```
Summary ::= SET {  
    absent        [0] INTEGER (1 . . ub-messages) OPTIONAL -- count of entries where the  
                  attribute is absent --,  
    present       [1] SET SIZE (1 . . ub-attribute-values) OF -- one for each attribute value present -  
                  SEQUENCE {  
                        type      AttributeType,  
                        value     ANY DEFINED BY type,  
                        count     INTEGER (1 . . ub-messages) } OPTIONAL }
```

Les composants du paramètre **résumé** ont la signification suivante:

- a) **Absent** (C) – Compte des entrées qui ne contient pas un attribut du type d'attribut spécifié dans la demande. Ce paramètre est omis s'il n'y a pas d'entrées de ce type.
- b) **Présent** (C) – Résumé des entrées contenant un attribut du type d'attribut spécifié, ventilé selon les valeurs d'attributs réellement présentes. Ce paramètre est omis s'il n'y a pas d'entrées de ce type.

Les composants du paramètre **présent** ont la signification suivante:

- i) **Type (type)** (O) – Type d'attribut.
- ii) **Value (valeur)** (O) – Valeur d'attribut pour laquelle le compte est donné.
- iii) **Count (compte)** (O) – Compte des entrées auxquelles est associée cette valeur d'attribut.

### 8.2.3 Erreurs abstraites résumé

Si la demande n'aboutit pas, une des erreurs abstraites listées sera signalée. Les conditions dans lesquelles telle ou telle erreur abstraite sera signalée sont définies au § 9.

### 8.3 Opération abstraite-listage

L'**opération abstraite-listage** est utilisée pour rechercher dans une base d'information sélectionnée les entrées présentant un intérêt et pour renvoyer des informations choisies provenant de ces entrées.

L'**opération abstraite-listage** n'aboutit que si la base d'information autorise l'accès conformément au contexte de sécurité et aux mesures de sécurité en vigueur.

Les informations qui peuvent être sélectionnées pour les entrées d'une base d'information peuvent être restreintes. Les restrictions portant sur les attributs généraux de la base d'information de messages enregistrés sont données dans le tableau 1/X.413.

```
List ::= ABSTRACT-OPERATION
        ARGUMENT ListArgument
        RESULT ListResult
        ERRORS {
            AttributeError,
            InvalidParametersError,
            RangeError,
            SecurityError,
            SequenceNumberError,
            ServiceError }
```

#### 8.3.1 Argument-listage

```
ListArgument ::= SET {
    Information-base-type [0] InformationBase DEFAULT stored-messages,
    selector [1] Selector,
    requested-attributes [3] EntryInformationSelection OPTIONAL }
```

Les composants du paramètre **argument-listage** ont la signification suivante:

- 1) **Information-base-type (type de base d'information)** (F) – Spécifie la base d'information appelée par l'opération abstraite (voir le § 6.3.1).  
En l'absence du composant **type de base d'information**, la valeur par défaut est messages enregistrés.
- 2) **Selector (sélecteur)** (O) – Ensemble de critères de sélection pour déterminer les entrées qui seront renvoyées (voir le § 8.1.3).
- 3) **Requested-attributes (attributs demandés)** (F) – Indique les informations des entrées sélectionnées qui seront renvoyées dans le résultat (voir le § 8.1.4).

Si ce paramètre est absent, l'ensemble enregistré des **valeurs d'attribut par défaut du paramètre listage** est utilisé. Se reporter au § 8.6.1 pour de plus amples informations sur ces valeurs par défaut.

#### 8.3.2 Résultat-listage

Si la demande aboutit, le paramètre **résultat-listage** doit être renvoyé.

```
ListResult ::= SET {
    next [0] SequenceNumber OPTIONAL,
    requested [1] SEQUENCE SIZE (1 . . ub-messages) OF EntryInformation OPTIONAL -- omitted
    if none found -- }
```

Les composants du paramètre **résultat-listage** ont la signification suivante:

- 1) **Next (suivant)** (C) – Ce paramètre est renvoyé lorsque le nombre d'entrées sélectionnées aurait été plus grand, en l'absence de la limite spécifiée dans le sélecteur. Le composant contient le numéro d'ordre de l'entrée suivante qui aurait été sélectionnée.
- 2) **Requested (demandé)** (C) – Ce paramètre achemine les informations sur l'entrée requises (voir le § 8.1.5) provenant de chaque entrée sélectionnée (une ou plus) par ordre croissant des numéros d'ordre. Il n'est pas présent lorsqu'une recherche a été effectuée et qu'aucune entrée n'a été sélectionnée.

### 8.3.3 Erreur abstraite-listage

Si la demande n'aboutit pas, une des erreurs abstraites énumérées sera signalée. Les conditions dans lesquelles telle ou telle erreur abstraite sera signalée sont définies au § 9.

### 8.4 Opération abstraite extraction

L'**opération abstraite extraction** sert à renvoyer des informations sélectionnés provenant d'une entrée spécifique d'une base d'information. Elle peut aussi être utilisée pour renvoyer des informations sélectionnées provenant de la première entrée parmi plusieurs entrées présentant un intérêt; dans ce cas, les numéros d'ordre des autres entrées sélectionnées sont également renvoyés. L'**opération abstraite extraction** n'aboutit que si l'on demande des bases d'information autorisées par le contexte de sécurité et les mesures de sécurité en vigueur.

Les informations provenant d'une entrée peuvent être extraites plusieurs fois jusqu'à ce que l'entrée ait été explicitement supprimée à l'aide de l'opération abstraite suppression.

```
Fetch ::= ABSTRACT-OPERATION
  ARGUMENT      FetchArgument
  RESULT        FetchResult
  ERRORS {
    AttributeError,
    FetchRestrictionError,
    InvalidParametersError,
    RangeError,
    SecurityError,
    SequenceNumberError,
    ServiceError }
```

#### 8.4.1 Argument de l'opération extraction

```
FetchArgument ::= SET {
  information-base-type      [0] InformationBase DEFAULT stored-messages,
  item                       CHOICE {
    search                   [1] Selector,
    precise                  [2] SequenceNumber },
  requested-attributes      [3] EntryInformationSelection OPTIONAL }
```

Les composants de l'**argument de l'opération extraction** ont la signification suivante:

- 1) **Information-base-type (type de base d'information)** (F) – Ce paramètre spécifie la base d'information appelée par l'opération abstraite (voir le § 6.3.1).  
En l'absence du composant type de base d'information, la valeur par défaut est messages enregistrés.
- 2) **Item (item)** (O) – L'une des composantes décrites ci-après doit être spécifiée pour déterminer l'entrée à extraire:
  - a) **Search (recherche)** (C) – Il s'agit d'un sélecteur spécifiant un ensemble d'entrées, celle qui a le numéro d'ordre le moins élevé est l'entrée à extraire (voir le § 8.1.3).
  - b) **Precise (précis)** (C) – Il s'agit du numéro d'ordre de l'entrée à extraire (voir le § 6.3.2.2).
- 3) **Requested-attributes (attributs demandés)** (F) – Ce paramètre indique les informations provenant de l'entrée sélectionnée à renvoyer dans le résultat (voir le § 8.1.4).  
Si ce paramètre est absent, l'ensemble enregistré des **valeurs d'attribut par défaut du paramètre extraction**, est utilisé. On se reportera au § 8.6.1 pour avoir de plus amples renseignements sur ces valeurs par défaut.

#### 8.4.2 Résultat de l'opération extraction

Si la demande aboutit, le paramètre **résultat de l'opération extraction** doit être renvoyé.

```
FetchResult ::= SET {
  entry-information [0] EntryInformation OPTIONAL -- if an entry was selected --,
  list              [1] SEQUENCE SIZE (1 .. ub-messages) OF SequenceNumber OPTIONAL,
  next              [2] SequenceNumber OPTIONAL }
```

Les composants du paramètre **résultat de l'opération extraction** ont la signification suivante:

- 1) **Entry-information (information-entrée)** (C) – Il s'agit d'un ensemble d'attributs provenant d'une entrée demandé dans l'argument (voir le § 8.1.5). Ce paramètre n'est pas présent lorsqu'une recherche a été effectuée et qu'aucune entrée n'a été sélectionnée.
- 2) **List (listage)** (C) – Ce paramètre est renvoyé lorsqu'une recherche a été effectuée et que plusieurs entrées ont été sélectionnées par le sélecteur de recherche. La liste donne, par ordre croissant, les numéros d'ordre de ces nouvelles entrées.
- 3) **Next (suivant)** (C) – Ce paramètre est renvoyé lorsque le nombre d'entrées sélectionnées aurait été plus grand en l'absence de la limite spécifiée dans le sélecteur. Le composant comprend le numéro d'ordre de l'entrée suivante qui aurait été sélectionnée.

#### 8.4.3 Erreurs opération abstraite extraction

Si la demande échoue, une des erreurs abstraites listées sera signalée. Les conditions dans lesquelles telle ou telle erreur abstraite sera signalée sont définies au § 9.

#### 8.5 Opération abstraite-suppression

L'**opération abstraite-suppression** est utilisée pour supprimer les entrées sélectionnées d'une base d'information. Une entrée principale et toutes les entrées secondaires qui lui sont rattachées ne peuvent être supprimées qu'ensemble. Cette opération est effectuée en spécifiant uniquement l'entrée principale comme argument. L'opération abstraite-suppression n'aboutira que dans le cas de bases d'information autorisées par le contexte de sécurité et les mesures de sécurité en vigueur.

Pour certaines bases d'information, il peut y avoir des restrictions sur les entrées à supprimer. De plus, des actions spécifiques au contenu peuvent être prises comme indiqué dans la Recommandation correspondante qui définit le type de contenu. Pour les messages enregistrés, aucune entrée ne peut être supprimée si la valeur de l'état de cette entrée (voir le § 6.4) est «nouveau».

```
Delete ::= ABSTRACT-OPERATION
  ARGUMENT          DeleteArgument
  RESULT            DeleteResult
  ERRORS {
    DeleteError,
    InvalidParametersError,
    RangeError,
    SecurityError,
    SequenceNumberError,
    ServiceError }
```

##### 8.5.1 Argument de l'opération suppression

```
DeleteArgument ::= SET {
  information-base-type [0] InformationBase DEFAULT stored-messages,
  items CHOICE {
    selector [1] Selector
    sequence-numbers [2] SET SIZE (1 . . ub-messages) OF SequenceNumber } }
```

Les composants de l'**argument de l'opération suppression** ont la signification suivante:

- 1) **Information-base-type (type de base d'information)** (F) – Spécifie la base d'information appelée par l'opération abstraite. Voir le § 6.3.1.  
En l'absence du composant **type de base d'information**, la valeur par défaut est messages enregistrés.
- 2) **Items (items)** (O) – L'une des composantes décrites ci-après doit être spécifiée pour déterminer les entrées à supprimer:
  - a) **Selector (sélecteur)** (C) – Voir le § 8.1.3.
  - b) **Sequence-numbers (numéros d'ordre)** (C) – Liste non ordonnée des numéros d'ordre. Voir le § 6.3.2.2.

### 8.5.2 *Résultat de l'opération suppression*

Si la demande aboutit, le **résultat de l'opération suppression** doit être renvoyé. Il n'y a pas de paramètres.

DeleteResult ::= NULL

### 8.5.3 *Erreurs opération abstraite suppression*

Si la demande échoue, une des erreurs abstraites listées sera signalée. Les conditions dans lesquelles telle ou telle erreur abstraite sera signalée sont définies au § 9.

## 8.6 *Opération abstraite enregistrement-MS*

L'opération abstraite enregistrement-MS est utilisée pour enregistrer ou effacer diverses informations avec le MS:

- a) actions automatiques;
- b) liste des valeurs par défaut des types d'attributs;
- c) nouvelle accréditation;
- d) nouvel ensemble d'étiquettes de sécurité de l'utilisateur.

```
Register-MS ::= ABSTRACT-OPERATION
  ARGUMENT Register-MSArgument
  RESULT Register-MSResult
  ERRORS {
    AttributeError,
    AutoActionRequestError,
    InvalidParametersError,
    SecurityError,
    ServiceError }
```

### 8.6.1 *Argument de l'opération abstraite enregistrement-MS*

```
Register-MS Arguments := SET {
  auto-action-registrations [0] SET SIZE (1 .. ub-auto-registrations) OF AutoActionRegistration OPTIONAL,
  auto-action-deregistrations [1] SET SIZE (1 .. ub-auto-registrations) OF AutoActionDeregistration OPTIONAL,
  list-attribute-defaults [2] SET SIZE (1 .. ub-default-registrations) OF Attribute Type OPTIONAL,
  fetch-attribute-defaults [3] SET SIZE (1 .. ub-default-registrations) OF Attribute Type OPTIONAL,
  change-credentials [4] SEQUENCE {
    old-credentials [0] IMPLICIT Credentials,
    new-credentials [1] IMPLICIT Credentials } OPTIONAL
  -- same CHOICE as for old-credentials --,
  user-security-labels [5] SET SIZE (1 .. ub-labels-and-redirections) OF SecurityLabel OPTIONAL }
```

Les composants de l'argument de l'opération enregistrement-MS ont la signification suivante:

- 1) **Auto-action-registrations (enregistrement d'actions automatiques)** (F) – Ensemble **d'enregistrements d'actions automatiques** (voir le § 6.5.1), un enregistrement pour chaque action automatique. Le nouveau **paramètre enregistrement d'actions automatiques** annule toute action automatique précédemment enregistrée (le cas échéant) à l'aide d'un **identificateur d'enregistrement** et du type d'actions automatiques.

En l'absence du paramètre **enregistrements d'actions automatiques**, la valeur par défaut est pas de nouvelles actions automatiques enregistrées.

- 2) **Auto-action-deregistrations (désenregistrements d'actions automatiques)** (F) – Ensemble de **désenregistrements d'actions automatiques**, un pour chaque action automatique à désenregistrer. Toute action automatique ayant un **identificateur d'enregistrement** et un **type d'action automatique** conformes à ceux spécifiés dans le paramètre **désenregistrement d'actions automatiques** est désenregistrée.

AutoActionDeregistration ::= AutoActionRegistration  
(WITH COMPONENTS { . . . , registration-parameter ABSENT } )

En l'absence du paramètre **désenregistrements d'actions automatiques**, la valeur par défaut est aucune action automatique enregistrée n'est désenregistrée.

- 3) **List-attribute-defaults (liste de valeurs par défaut d'attributs)** (F) – Spécifie un ensemble de valeurs par défaut de types d'attributs pour indiquer les attributs à renvoyer pour toute opération abstraite ultérieure listage ou avertissement si l'argument sélection d'information d'entrée est absent.

En l'absence du paramètre **liste des valeurs par défaut d'attributs**, la valeur par défaut est aucun changement à la valeur par défaut enregistrée (le cas échéant). Les paramètres **liste des valeurs par défaut d'attributs** forment l'ensemble vide jusqu'à ce qu'ils soient modifiés explicitement par l'utilisateur MS par l'intermédiaire de l'opération abstraite enregistrement MS.

- 4) **Fetch-attribute-defaults (liste des valeurs par défaut d'extraction)** (F) – Spécifie un ensemble de valeurs par défaut de types d'attributs pour indiquer les attributs à renvoyer pour toute opération abstraite ultérieure extraction si l'argument sélection d'information d'entrée est absent.

En l'absence du paramètre **liste des valeurs par défaut d'extraction**, la valeur par défaut est aucun changement à la valeur par défaut enregistrée (le cas échéant). Les paramètres **listes des valeurs par défaut d'extraction** forment l'ensemble vide jusqu'à ce qu'ils soient modifiés explicitement par l'utilisateur MS par l'intermédiaire de l'opération abstraite enregistrement-MS.

- 5) **Change-credentials (modification des pouvoirs)** (F) – Ancienne et nouvelle accréditation si une **modification des pouvoirs** est demandée.

Les **anciens pouvoirs** sont les pouvoirs de l'utilisateur final actuel; les **nouveaux pouvoirs** sont les pouvoirs que souhaiterait obtenir l'utilisateur final.

En l'absence de cet argument, la valeur par défaut est que les pouvoirs précédemment enregistrés restent en l'état.

Les pouvoirs de l'utilisateur MS peuvent différer des **pouvoirs du demandeur** spécifiés au § 8.1.1.1.2 de la Recommandation X.411.

- 6) **User-security-labels (étiquettes de sécurité de l'utilisateur)** (F) – **Étiquette(s) de sécurité** de l'utilisateur du service abstrait MS si elles doivent être modifiées. Ce paramètre peut être créé par l'utilisateur du service abstrait MS.

En l'absence de cet argument, les **étiquettes de sécurité de l'utilisateur** restent inchangées.

A noter que selon certaines **mesures de sécurité** les **étiquettes de sécurité** ne peuvent être ainsi modifiées que si une liaison sûre est utilisée. D'autres moyens locaux pour modifier les **étiquettes de sécurité** de l'utilisateur de façon sûre peuvent être fournis. Les **étiquettes de sécurité** de l'utilisateur sont définies au § 8.4.1.1.7 de la Recommandation X.411.

L'**étiquette de sécurité** est définie au § 9 de la Recommandation X.411.

#### 8.6.2 *Résultat de l'opération enregistrement-MS*

Si la demande aboutit, le résultat de l'opération enregistrement-MS doit être renvoyé. Il n'y a pas de paramètres.

Register-MSResult ::= NULL

#### 8.6.3 *Erreurs de l'opération abstraite enregistrement-MS*

Si la demande échoue, une des erreurs abstraites listées doit être signalée. Les conditions dans lesquelles telle ou telle erreur abstraite sera signalée sont définies au § 9.

#### 8.7 *Opération abstraite-avertissement*

L'**opération abstraite-avertissement** permet au prestataire du service abstrait MS d'informer immédiatement l'utilisateur du service abstrait MS de l'introduction d'une nouvelle entrée dans le MS, entrée dont les attributs sont conformes aux critères de sélection de l'un des **enregistrements d'avertissements automatiques** (voir le § 12.2), précédemment fournis à l'aide de l'opération abstraite MS-enregistrement (voir le § 8.6).

L'**opération abstraite-avertissement** peut être appelée lorsqu'il existe une association abstraite établie par l'UA, mais seulement après la création de nouvelles entrées, une fois l'association abstraite établie.

Les entrées conformes aux critères de sélection qui ont été créées entre associations abstraites seront indiquées à l'issue de la prochaine opération abstraite-liaison pour l'association abstraite. Aucune **opération abstraite-avertissement** ne doit être appelée pour ces entrées (voir le § 7).

L'**opération abstraite-avertissement** n'aboutira que si la base d'information autorise l'accès conformément au contexte de sécurité et aux mesures de sécurité en vigueur.

```
Alert := ABSTRACT-OPERATION
        ARGUMENT      AlertArgument
        RESULT AlertResult
        ERRORS {
            SecurityError }
```

#### 8.7.1 *Argument de l'opération abstraite-avertissement*

```
AlertArgument ::= SET {
    alert-registration-identifieur [0] INTEGER (1 .. ub-auto-actions),
    new-entry                       [2] EntryInformation OPTIONAL }
```

Les composants de l'**argument de l'opération abstraite-avertissement** ont la signification suivante:

- 1) **Alert-registration-identifieur (identificateur d'enregistrement d'avertissement)** (O) – Identifie lequel des **enregistrements d'avertissements automatiques** a donné lieu à l'avertissement (voir les § 6.4 et 12.2).
- 2) **New-entry (nouvelle entrée)** (F) – Achemine l'information provenant de la nouvelle entrée qui a été demandée dans le **paramètre d'enregistrement d'avertissement automatique** (voir le § 12.2). Ce paramètre est absent lorsque l'utilisateur du service abstrait MS n'a pas spécifié un **paramètre d'enregistrement d'avertissement automatique**.

#### 8.7.2 *Résultat de l'opération abstraite-avertissement*

Si la demande aboutit, le résultat de l'opération abstraite-avertissement doit être renvoyé.

```
AlertResult ::= NULL
```

#### 8.7.3 *Erreurs de l'opération abstraite-avertissement*

Si la demande échoue, une des erreurs abstraites listées sera signalée. Les conditions dans lesquelles telle ou telle erreur abstraite doit être signalée sont définies au § 9.

## 9 Erreurs abstraites

Le présent paragraphe définit les erreurs abstraites suivantes associées à l'utilisation d'opérations abstraites à l'accès pour recherche:

- a) erreur d'attribut;
- b) erreur de demande d'action automatique;
- c) erreur de suppression;
- d) erreur de restriction d'extraction;
- e) erreur paramètres non valables;
- f) erreur de gamme;
- g) erreur de sécurité;
- h) erreur de service;
- i) erreur de numéro d'ordre.

#### 9.1 *Priorité des erreurs*

Le responsable d'une opération abstraite n'est pas tenu de poursuivre le traitement du message au-delà du point auquel une erreur a été détectée. Cela permet, lors de la mise en œuvre, de choisir si l'on poursuit le traitement des erreurs.

*Remarque* – Il découle de cette règle que la première erreur détectée peut être différente pour des mises en œuvre répétées de la même opération abstraite, étant donné qu'il n'y a pas nécessairement d'ordre logique particulier pour traiter ces erreurs.

## 9.2 Erreur d'attribut

Une **erreur d'attribut** indique un problème lié à l'attribut.

```

AttributeError ::= ABSTRACT-ERROR
  PARAMETER SET {
    Problems [0] SET SIZE (1 .. ub-per-entry) OF SET {
      problem [0] AttributeProblem,
      type [1] AttributeType,
      value [2] ANY DEFINED BY type OPTIONAL } }

AttributeProblem ::= INTEGER {
  invalid-attribute-value (0),
  unavailable-attribute-type (1),
  inappropriate-matching (2),
  attribute-type-not-subscribed (3),
  inappropriate-for-operation (4) } (0 .. ub-error-reasons)
  
```

Ce paramètre a la signification suivante:

- 1) **Problems (problèmes) (O)** – Problèmes particuliers rencontrés. Un nombre quelconque de problèmes spécifiques peut être signalé; avec chaque problème on indique le type avec chaque problème on indique le type d'attribut et, si nécessaire, pour éviter toute ambiguïté, la valeur qui est à l'origine du problème:
  - a) **Invalid-attribute-value (type d'attribut invalide) (C)** – Une prétendue valeur d'attribut spécifiée comme argument de l'opération abstraite n'est pas conforme au type de données défini pour le type d'attribut en question.
  - b) **Unavailable-attribute-type (type d'attribut indisponible) (C)** – Un prétendu type d'attribut utilisé comme argument de l'opération abstraite ne fait pas partie de ceux fournis par le prestataire du service abstrait MS. Si le prestataire du service abstrait MS peut néanmoins effectuer l'opération, il n'en est pas empêché.
  - c) **Inappropriate-matching (concordance non appropriée) (C)** – Le filtre contient un item-filtre dans lequel un attribut est mis en correspondance à l'aide d'une opération (égalité, classement ou sous-chaînes) non définie pour cet attribut.
  - d) **Attribute-type-not-subscribed (type d'attribut non souscrit) (C)** – Un type d'attribut utilisé comme argument de l'opération abstraite ne fait pas partie de ceux auxquels l'utilisateur du service abstrait MS s'est abonné.
 

*Remarque* – Une modification de l'abonnement n'est pas nécessairement reflétée dans les attributs présents dans une entrée créée avant la modification.
  - e) **Inappropriate-for-operation (inapproprié pour l'utilisation) (C)** – Un type d'attribut utilisé comme argument de l'opération abstraite ne convient pas pour l'utilisation requise.

## 9.3 Erreur de demande d'action automatique

Une **erreur de demande d'action automatique** indique un problème lié à l'enregistrement d'une action automatique.

```

AutoActionRequestError ::= ABSTRACT-ERROR
  PARAMETER SET {
    problems [0] SET SIZE (1 .. ub-auto-registrations) OF SET {
      problem [0] Auto-ActionRequestProblem,
      type [1] AutoActionType } }

AutoActionRequestProblem ::= INTEGER {
  unavailable-auto-action-type (0),
  auto-action-type-not-subscribed (1) } (0 .. ub-error-reasons)
  
```

Le paramètre a la signification suivante:

- 1) **Problems (problèmes)** (O) – Problèmes particuliers rencontrés. Un nombre quelconque de problèmes spécifiques peut être signalé; avec chaque problème, on indique le **type d'action automatique** à l'origine du problème:
  - a) **Unavailable-auto-action-type (type d'action automatique indisponible)** – Un type d'action automatique utilisé comme argument de l'opération abstraite ne fait pas partie de ceux assurés par le prestataire du service abstrait MS.
  - b) **Action-type-not-subscribed (type d'action non souscrit)** – Un type d'action utilisé comme argument de l'opération abstraite ne fait pas partie de ceux auxquels l'utilisateur du service abstrait MS s'est abonné.

#### 9.4 Erreur de suppression

Un paramètre **erreur de suppression** indique un problème lié à une tentative de suppression d'une ou de plusieurs entrées d'une base d'information.

```
DeleteError ::= ABSTRACT-ERROR
  PARAMETER SET {
    problems [0] SET SIZE (1 .. ub-messages) OF SET {
      problem [0] DeleteProblem,
      sequence-number [1] SequenceNumber } }
```

```
DeleteProblem ::= INTEGER {
  child-entry-specified (0),
  delete-restriction-problem (1) } (0 .. ub-error-reasons)
```

Le paramètre a la signification suivante:

- 1) **Problems (problèmes)** (O) – problèmes particuliers rencontrés. Un nombre quelconque de problèmes spécifiques peut être signalé, avec chaque problème on indique le numéro d'ordre de l'entrée à l'origine du problème:
  - a) **Child-entry-specified (entrée secondaire spécifiée)** – Une tentative a été faite pour supprimer une entrée secondaire.
  - b) **Delete-restriction-problem (problème de restriction de suppression)** – On a essayé de transgresser une restriction spécifiée pour l'opération abstraite-suppression (voir le § 8.5).

#### 9.5 Erreur de restriction d'extraction

Le paramètre **erreur de restriction d'extraction** indique une tentative de transgression d'une restriction liée à l'opération abstraite-extraction.

```
FetchRestrictionError ::= ABSTRACT-ERROR
  PARAMETER SET {
    problems [0] SET SIZE (1 .. ub-default-registrations) OF SET {
      problem [3] FetchRestrictionProblem,
      restriction CHOICE {
        content-type [0] ContentType,
        eit [1] MS-EITs,
        content-length [2] ContentLength } } }
```

```
FetchRestrictionProblem ::= INTEGER {
  content-type-problem (1),
  eit-problem (2),
  content-length-problem (3) } (0 .. ub-error-reasons)
```

Le paramètre a la signification suivante:

- 1) **Problems (Problèmes)** (O) – Problèmes particuliers rencontrés. Un nombre quelconque de problèmes particuliers peut être signalé, avec chaque problème, on indique le type de contenu, le type de codage ou la longueur du contenu à l'origine du problème:
  - a) **Content-type-problem (problème de type de contenu)** (F) – Le type de contenu du message extrait est interdit par les restrictions d'extraction en vigueur.

- b) **EIT-problem (problème-EIT)** (F) – Les types de codages requis dans l'opération abstraite-extraction sont interdits par les restrictions d'extraction en vigueur.
- c) **Content-length-problem (problème de longueur de contenu)** (C) – La longueur de contenu du message extrait est supérieure à celle autorisée par les restrictions d'extraction en vigueur.

#### 9.6 *Erreur-paramètre-non valable*

Un paramètre **erreur-paramètres-non valables** indique un problème sémantique dans l'ensemble des paramètres reçus. Cette erreur serait utilisée, par exemple, pour indiquer qu'un paramètre facultatif a été utilisé dans un mauvais contexte ou qu'une valeur de l'un des paramètres est inappropriée.

```
InvalidParametersError ::= ABSTRACT-ERROR
    PARAMETER NULL
```

Cette erreur n'a pas de paramètres.

#### 9.7 *Erreur de gamme*

Un paramètre **erreur de gamme** indique un problème lié à la limite spécifiée dans un sélecteur comme argument d'une opération abstraite.

```
RangeError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problem      [0] RangeProblem }
RangeProblem ::= INTEGER {
    reversed      (0) } (0) .. ub-error-reasons }
```

Le paramètre a la signification suivante:

- 1) **Problems (problèmes)** (0) - Problèmes particuliers rencontrés:
  - a) **Reversed** (C) – Limite supérieure indique un numéro d'ordre ou une date-heure de création plus faible que celui ou celle indiqué par la limite inférieure.

#### 9.8 *Erreur de sécurité*

Le paramètre **erreur de sécurité** indique que l'opération abstraite demandée ne peut être assurée car elle transgresserait les mesures de sécurité en vigueur. Cette erreur est définie dans la Recommandation X.411.

#### 9.9 *Erreur de numéro d'ordre*

Un paramètre **erreur de numéro d'ordre** indique un problème lié au numéro d'ordre spécifié dans un argument d'une opération abstraite.

```
SequenceNumberError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problems [1] SET SIZE (1 .. ub-messages) OF SET {
            problem      [0] SequenceNumberProblem,
            sequence-number [1] SequenceNumber } }
SequenceNumberProblem ::= INTEGER {
    no-such-entry      (0) } (0 .. ub-error-reasons)
```

Ce paramètre a la signification suivante:

- 1) **Problems (problèmes)** (0) – Problèmes particuliers rencontrés. Un nombre quelconque de problèmes spécifiques peut être signalé; avec chaque problème on indique le numéro d'ordre à l'origine du problème:
  - a) **No-such-entry (nulle entrée semblable)** – Le numéro d'ordre fourni ne correspond à aucune entrée de la base d'information.

## 9.10 Erreur de service

Un paramètre **erreur de service** indique une erreur liée à la fourniture du service.

ServiceError ::= ABSTRACT-ERROR

PARAMETER SET {  
    problem [0] ServiceProblem }

ServiceProblem ::= INTEGER {

    busy (0),  
    unavailable (1),  
    unwilling-to-perform (2) } (0 .. ub-error-reasons)

Ce paramètre a la signification suivante:

- 1) **Problem (problème)** (0) – Problème particulier rencontré:
  - a) **Busy (occupé)** (C) – Le MS, ou une partie de ce service, est trop occupé pour effectuer l'opération abstraite demandée mais pourra le faire un peu plus tard.
  - b) **Unavailable (indisponible)** (C) – Le MS, ou une partie de ce service, est à l'heure actuelle non disponible.
  - c) **Unwilling to perform (non prêt à exécuter)** (C) – Le MS n'est pas prêt à donner suite à cette demande car elle engendrerait une utilisation excessive de ressources.

## SECTION 3 – TYPES D'ATTRIBUTS GÉNÉRAUX ET TYPES D'ACTIONS AUTOMATIQUES GÉNÉRALES

### 10 Présentation générale

Le «modèle d'information MS» et les notions d'**attributs** et d'**actions automatiques** ont été présentés aux § 6.3.3 et 6.5. Le § 11 définit les **types d'attributs généraux** spécifiés pour le MS. Le § 12 définit les **types d'actions automatiques générales** spécifiés pour le MS.

### 11 Types d'attributs généraux

Les **types d'attributs généraux** sont valables pour tous les types de contenus de message. D'autres types d'attributs, propres à tel ou tel contenu, sont définis dans les Recommandations appropriées; à titre d'exemple, les types d'attributs pour le service de messagerie de personne à personne (IPMS) pour le MS sont définis dans l'annexe C à la Recommandation X.420.

#### 11.1 Présentation générale des types d'attributs généraux

Les **attributs généraux** qui peuvent apparaître dans une entrée de base d'information de messages enregistrés sont énumérés dans le tableau 1/X.413. Ils proviennent essentiellement des informations relatives aux paramètres des opérations abstraites de remise de messages et de remise de rapports du service abstrait MTS définies au § 8 de la Recommandation X.411; ces attributs sont désignés en conséquence. Certains **attributs généraux** sont créés, certains également mis à jour par la MS.

Le tableau 1/X.413 définit les divers **attributs généraux** et donne les caractéristiques suivantes pour chaque type d'attribut:

- s'il s'agit d'un attribut à valeur unique ou à plusieurs valeurs;
- si le type d'attribut est assuré ou non par la MS et si l'accès de l'UA est obligatoire ou facultatif;
- si le type d'attribut est toujours présent, présent de façon conditionnelle ou absent dans l'entrée message remis, rapport remis ou contenu renvoyé respectivement;
- si le type d'attribut peut ou non être renvoyé dans une opération abstraite listage ou avertissement;
- si le type d'attribut peut être utilisé ou non dans une opération abstraite-résumé.

*Remarque* – Uniquement pour les types de données ASN.1 simples.

On trouvera dans les conventions du § 5.2 une description plus détaillée des classifications du tableau 1/X.413.

Un type d'attribut facultatif n'est assuré par un MS que si ce type d'attribut a fait l'objet d'un abonnement (ce qui implique que le MS et l'UA qui a accès à ce service fournissent cet attribut). L'abonnement au type d'attribut facultatif peut se faire par type d'attribut et par UA.

## 11.2 *Description des types d'attributs généraux*

Les alinéas suivants décrivent brièvement chaque **type d'attribut général** ainsi que la syntaxe abstraite utilisant la macro ATTRIBUTE définie au § 6.3.

A noter que certains **attributs généraux** sont utilisés essentiellement pour le listage et le filtrage, alors que d'autres peuvent contenir des informations plus complexes, voire plus importantes (types de données ASN.1 plus structurés). Seuls quelques **attributs généraux** conviennent pour les résumés.

### 11.2.1 *Numéros d'ordre secondaire*

Cet attribut général à plusieurs valeurs, contient une ou plusieurs «flèches» indiquant, le cas échéant, le niveau suivant des entrées secondaires. Il est créé par le MS. Cet attribut est présent dans une entrée principale à laquelle sont rattachées une ou plusieurs entrées secondaires. Il est absent dans une entrée à laquelle n'est rattachée aucune entrée secondaire.

```
ms-child-sequence-number ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX-SequenceNumber
    MULTI VALUE
    ::= id-att-child-sequence-numbers
```

TABLEAU 1/X.413

## Types d'attributs généraux pour les messages enregistrés

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Niveau de fourniture par le MS et UA-accès	Présence dans l'entrée message remis	Présence dans l'entrée rapport remis	Présence dans l'entrée contenu renvoyé	Disponible pour listage avertissement	Disponible pour résumé
Child-sequence-numbers	M	M	C	C	C	Y	N
Content	S	M	P	-	P	N	N
Content-confidentiality-algorithm-identifiant	S	O	C	-	-	Y	N
Content-correlator	S	O	-	C	-	Y	N
Content-identifiant	S	O	C	C	-	Y	N
Content-integrity-check	S	O	C	-	-	Y	N
Content-length	S	O	P	-	P	Y	N
Content-returned	S	O	-	P	-	Y	Y
Content-type	S	M	P	C	C	Y	Y
Conversion-with-loss-prohibited	S	O	C	-	-	Y	N
Converted-EITs	M	O	C	-	-	Y	N
Creation-time	S	M	P	P	P	Y	N
Delivered-EITs	M	O	P	-	Y	N	
Delivery-flags	S	O	P	-	-	Y	N
DL-expansion-history	M	O	C	C	-	Y	N
Entry-status	S	M	P	P	P	Y	Y
Entry-type	S	M	P	P	P	Y	Y
Intended-recipient-name	S	O	C	-	-	Y	N
Message-delivery-envelope	S	M	P	-	-	N	N
Message-delivery-identifiant	S	O	P	-	-	Y	N
Message-delivery-time	S	O	P	-	-	Y	N
Message-origin-authentication-check	S	O	C	-	-	Y	N
Message-security-label	S	O	C	C	-	Y	N
Message-submission-time	S	O	P	-	-	Y	N
Message-token	S	O	C	-	-	Y	N
Original-EITs	M	O	C	C	-	Y	N
Originator-certificate	S	O	C	-	-	Y	N
Originator-name	S	O	P	-	-	Y	N
Other-recipient-names	M	O	C	-	-	Y	N
Parent-sequence-number	S	M	C	-	P	Y	N
Per-recipient-report-delivery-fields	M	M	-	P	-	Y	N
Priority	S	O	P	-	-	Y	Y
Proof-of-delivery-request	S	O	C	-	-	Y	N
Redirection-history	M	O	C	-	-	Y	N
Report-delivery-envelope	S	M	-	P	-	N	N
Reporting-DL-name	S	O	-	C	-	Y	N
Reporting-MTA-certificate	S	O	-	C	-	Y	N
Report-origin-authentication-check	S	O	C	C	-	Y	Y
Security-classification	S	O	C	C	-	Y	Y
Sequence-number	S	M	P	P	P	Y	N
Subject-submission-identifiant	S	M	-	P	-	Y	N
This-recipient-name	S	O	P	-	-	Y	N

### 11.2.2 *Contenu*

Cet attribut général contient le contenu complet d'un message tel qu'il est remis à l'aide de l'opération abstraite remise de messages ou sous forme de contenu renvoyé à l'aide de l'opération abstraite remise de rapports. On trouvera de plus amples détails aux § 8.2.1.1.1.37 et 8.3.1.2.1.14 de la Recommandation X.411.

```
ms-contentATTRIBUTE
  WITH ATTRIBUTE-SYNTAX Content
  SINGLE VALUE
  ::= id-att-content
```

### 11.2.3 *Identificateur d'algorithme de confidentialité du contenu*

Cet attribut général contient l'**identificateur d'algorithme** utilisé par l'expéditeur du message pour coder le contenu du message. Il peut être créé par l'expéditeur du message. On trouvera de plus amples détails au § 8.5.10 de la Recommandation X.411.

```
mt-content-confidentiality-algorithm-identifiant ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX AlgorithmIdentifier
  SINGLE VALUE
  ::= id-att-content-confidentiality-algorithm-identifiant
```

### 11.2.4 *Corrélateur de contenu*

Cet attribut général contient les informations permettant la corrélation du contenu du message. Il peut être créé par l'UA expéditeur. On trouvera de plus amples détails au § 8.2.1.1.1.36 de la Recommandation X.411.

```
mt-content-correlator ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ContentCorrelator
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-content-correlator
```

### 11.2.5 *Identificateur de contenu*

Cet attribut général contient un identificateur de contenu du message. Il peut être créé par l'UA expéditeur. On trouvera de plus amples détails au § 8.2.1.1.1.35 de la Recommandation X.411.

```
mt-content-identifiant ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ContentIdentifier
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-content-identifiant
```

### 11.2.6 *Vérification d'intégrité du contenu*

Cet attribut général donne au(x) destinataire(s) du message un moyen de contrôler que le contenu du message n'a pas été modifié. Il peut être créé par l'expéditeur du message et spécifier une valeur différente pour chaque destinataire du message. On trouvera de plus amples détails au § 8.2.1.1.28 de la Recommandation X.411.

```
mt-content-integrity-check ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ContentIntegrityCheck
  SINGLE VALUE
  ::= id-att-content-integrity-check
```

### 11.2.7 *Longueur du contenu*

Cet attribut général donne la longueur du contenu en octets d'un message tel qu'il est remis à l'aide de l'opération abstraite remise de messages ou d'un contenu renvoyé (le cas échéant) notifié par l'opération abstraite remise de rapports. Lorsqu'il n'y a pas de contenu renvoyé, cet attribut est absent. Il est créé par la MS.

```
ms-content ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ContentLength
  MATCHES FOR ORDERING
  SINGLE VALUE
  ::= id-att-content-length
```

### 11.2.8 *Contenu renvoyé*

Cet attribut général indique si le contenu d'un message a été renvoyé à l'aide de l'opération abstraite remise de rapports. Il est créé par le MS.

```
ms-content-returned ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX BOOLEAN
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-content-returned
```

### 11.2.9 *Type de contenu*

Cet attribut général est créé à partir du type de contenu à l'aide de l'opération abstraite remise de messages ou remise de rapports. Se reporter également au § 8.2.1.1.1.34 de la Recommandation X.411.

```
mt-content-type ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX OBJECT IDENTIFIER
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-content-type
```

### 11.2.10 *Conversion avec perte interdite*

Cet attribut général contient des informations permettant de savoir si la conversion avec perte d'information a été autorisée ou non. Se reporter au § 8.2.1.1.1.10 de la Recommandation X.411 pour de plus amples détails.

```
mt-conversion-with-loss-prohibited ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ConversionWithLossProhibited
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-conversion-with-loss-prohibited
```

### 11.2.11 *Types de codages convertis*

Cet attribut général à plusieurs valeurs, identifie les types de codages du contenu après conversion indiqués par l'opération abstraite remise de messages ou remise de rapports. Il est créé par le MS. Il est absent si aucune conversion n'a eu lieu. On trouvera de plus amples détails aux § 8.3.1.1.1.8 et 8.3.1.2.1.5 de la Recommandation X.411.

```
ms-converted-EITs ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX MS-EIT
    MATCHES FOR EQUALITY
    MULTI VALUE
    ::= id-att-converted-EITs
```

### 11.2.12 *Date-heure de création*

Cet attribut général indique la date et l'heure auxquelles l'entrée a été créée dans le MS. Il est produit par le MS. On trouvera de plus amples détails au § 6.3.2.

*Remarque* – Deux entrées consécutives ou plus peuvent avoir la même date et la même heure de création.

```
ms-creation-time ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX CreationTime
    MATCHES FOR EQUALITY ORDERING
    SINGLE VALUE
    ::= id-att-creation-time
```

### 11.2.13 *Types de codages remis*

Cet attribut général, à plusieurs valeurs, identifie les types de codages dans le contenu du message remis. Il est créé par le MS sur la base des informations relatives aux types de codages d'origine et aux types de codages convertis pendant l'opération abstraite remise de messages.

```
ms-delivered-EITs ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX-EIT
    MATCHES FOR EQUALITY
    MULTI VALUE
    ::= id-att-delivered-EITs
```

#### 11.2.14 *Indicateurs de remise*

Cet attribut général contient des informations concernant la remise. Actuellement, il n'est utilisé que pour indiquer une conversion implicite du contenu. On trouvera de plus amples détails au § 8.2.1.1.1.9 de la Recommandation X.411.

```
mt-delivery-flags ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX DeliveryFlags
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-delivery-flags
```

#### 11.2.15 *Chronologie de l'allongement de la liste de distribution*

Cet attribut général, à plusieurs valeurs, est utilisé pour décrire la chronologie de l'allongement de la liste de distribution. Il contient un ou plusieurs noms de liste de distribution utilisés pendant le processus d'allongement. Il est absent si la remise à ce destinataire-ci n'a nécessité aucun allongement de la liste de distribution. On trouvera de plus amples détails au § 8.3.1.1.1.7 de la Recommandation X.411.

```
mt-dl-expansion-history ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX DLExpansionHistory
  MULTI VALUE
  ::= id-att-expansion-history
```

#### 11.2.16 *Etat de l'entrée*

Cet attribut général contient l'état actuel d'une entrée de la base d'information de messages enregistrés. Il est créé et mis à jour par le MS. On trouvera de plus amples détails au § 6.4.

```
ms-entry-status ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX EntryStatus
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-entry-status
```

#### 11.2.17 *Type d'entrée*

Cet attribut général contient des informations indiquant si une entrée a trait à un message remis ou à un rapport remis. Il est créé par le MS.

```
ms-entry-type ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX Entry-Type
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-entry-type

EntryType ::= INTEGER {
  delivered-message      (0),
  delivered-report      (1),
  returned-content      (2) (0 . . ub-entry-types) }
```

#### 11.2.18 *Nom du destinataire prévu*

Cet attribut général contient le nom O/R du destinataire prévu à l'origine si le message a été réacheminé, chaque valeur représentant un réacheminement. On trouvera de plus amples détails au § 8.3.1.1.1.4.

```
mt-intended-recipient-name ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ORName
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-intended-recipient-name
```

#### 11.2.19 *Enveloppe de remise de message*

Cet attribut général contient l'**enveloppe** complète **de remise de message** d'un message tel qu'il a été remis grâce à l'opération abstraite remise de messages. On trouvera de plus amples détails au § 9 de la Recommandation X.411.

mt-message-delivery-envelope ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageDeliveryEnvelope  
SINGLE VALUE  
::= id-att-message-delivery-envelope

#### 11.2.20 *Identificateur de remise de message*

Cet attribut général contient l'**identificateur de remise de message** provenant de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.1 de la Recommandation X.411.

mt-message-delivery-identifiant ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageDeliveryIdentifier  
SINGLE VALUE  
::= id-att-message-delivery-identifiant

#### 11.2.21 *Date-heure de remise de message*

Cet attribut général contient la **date et l'heure de remise du message** provenant de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.2 de la Recommandation X.411.

*Remarque* – Il n'y a pas d'attribut général correspondant au paramètre date et heure de remise de l'opération abstraite remise de rapports car, pour être utile, ce paramètre doit être corrélé avec le nom du destinataire auquel le message a été remis. Ces informations sont incluses dans l'attribut général informations concernant le rapport.

mt-message-delivery-time ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageDeliveryTime  
MATCHES FOR EQUALITY ORDERING  
SINGLE VALUE  
::= id-att-message-delivery-time

#### 11.2.22 *Vérification d'authentification d'origine de message*

Cet attribut général est calculé à l'aide de l'algorithme identifié par l'identificateur d'authentification d'origine du message. Il permet au(x) destinataire(s) du message d'authentifier l'origine du message et peut être créé par l'expéditeur du message. On trouvera de plus amples détails au § 8.2.1.1.1.29 de la Recommandation X.411.

mt-message-origin-authentication-check ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageOriginAuthenticationCheck  
SINGLE VALUE  
::= id-att-message-origin-authentication-check

#### 11.2.23 *Etiquette de sécurité de message*

Cet attribut général comprend un ensemble d'attributs de sécurité parmi lesquels un identificateur de sécurité, une classification de sécurité, une marque de confidentialité et un ensemble de catégories de sécurité. On trouvera de plus amples détails au § 8.2.1.1.1.30 de la Recommandation X.411.

mt-message-security-label ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageSecurityLabel  
SINGLE VALUE  
::= id-att-message-security-label

#### 11.2.24 *Date et heure de dépôt du message*

Cet attribut général contient la **date et l'heure de dépôt du message** provenant de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.2.1.1.2.2 de la Recommandation X.411.

mt-message-submission-time ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageSubmissionTime  
MATCHES FOR EQUALITY ORDERING  
SINGLE VALUE  
::= id-att-message-submission-time

#### 11.2.25 *Symbole de message*

Cet attribut général contient le symbole associé au message. Il est créé par l'expéditeur du message et peut contenir une valeur différente pour chaque destinataire du message. On trouvera de plus amples détails au § 8.2.1.1.1.26 de la Recommandation X.411.

mt-message-token ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MessageToken  
SINGLE VALUE  
::= id-att-message-token

#### 11.2.26 *Types de codages d'origine*

Cet attribut général, à plusieurs valeurs, contient les **types de codages d'origine** provenant de l'opération abstraite remise de messages. Il est créé par le MS. On trouvera de plus amples détails au § 8.2.1.1.1.33 de la Recommandation X.411.

ms-original-EITs ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX MS-EIT  
MATCHES FOR EQUALITY  
MULTI VALUE  
::= id-att-original-EITs

#### 11.2.27 *Certificat de l'expéditeur*

Cet attribut général contient le certificat de l'expéditeur du message. Il est créé par une personne de confiance (par exemple une autorité d'authentification) et peut être fourni par l'expéditeur du message. On trouvera de plus amples détails au § 8.2.1.1.1.25 de la Recommandation X.411.

mt-originator-certificate ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX Originator-Certificate  
SINGLE VALUE  
::= id-att-originator-certificate

#### 11.2.28 *Nom de l'expéditeur*

Cet attribut général contient le nom O/R de l'expéditeur provenant de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.2.1.1.1.1 de la Recommandation X.411.

mt-originator-name ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX ORName  
MATCHES FOR EQUALITY  
SINGLE VALUE  
::= id-att-originator-name

#### 11.2.29 *Noms des autres destinataires*

Cet attribut général, à plusieurs valeurs, contient les noms O/R de tous les autres destinataires spécifiés (le cas échéant) du message provenant de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.6 de la Recommandation X.411.

mt-other-recipient-names ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX ORName  
MATCHES FOR EQUALITY  
MULTI VALUE  
::= id-att-other-recipient-names

#### 11.2.30 *Numéro d'ordre principal*

Cet attribut général est associé à une entrée principale. Il est créé par le MS. Il est toujours présent dans une entrée secondaire et absent dans une entrée principale.

ms-parent-sequence-number ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX SequenceNumber  
MATCHES FOR EQUALITY ORDERING  
SINGLE VALUE  
::= id-att-parent-sequence-number

#### 11.2.31 *Champs de remise de rapports par destinataire*

Cet attribut général, à plusieurs valeurs, contient des informations sur les rapports, pour chaque destinataire, provenant de l'opération abstraite remise de rapports. On trouvera de plus amples détails au § 8.3.1.2 de la Recommandation X.411.

mt-per-recipient-report-delivery-fields ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX PerRecipientReportDeliveryFields  
MULTI VALUE  
::= id-att-per-recipient-report-delivery-fields

#### 11.2.32 *Priorité*

Cet attribut général indique la **priorité** relative du message issu de l'opération abstraite remise de messages. Si aucune valeur n'est fournie dans le paramètre opération abstraite de remise de messages, le MS utilise sa valeur par défaut lorsqu'il crée cet attribut. On trouvera de plus amples détails au § 8.2.1.1.1.8 de la Recommandation X.411.

mt-priority ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX Priority  
MATCHES FOR EQUALITY  
SINGLE VALUE  
::= id-att-priority

#### 11.2.33 *Demande de preuve de remise*

Cet attribut général indique si l'expéditeur du message exige une **preuve de remise** du message du destinataire. Il peut être créé par l'expéditeur du message et avoir une valeur différente pour chaque destinataire du message. On trouvera de plus amples détails au § 8.2.1.1.1.32 de la Recommandation X.411.

mt-proof-of-delivery-request ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX ProofOfDeliveryRequest  
SINGLE VALUE  
::= id-att-proof-of-delivery-request

#### 11.2.34 *Chronologie du réacheminement*

Cet attribut général, à plusieurs valeurs, contient une chronologie du/des réacheminements vers le destinataire, ainsi que le/les motif(s) de ce réacheminement provenant de l'opération abstraite remise de messages ou remise de rapports. On trouvera de plus amples détails au § 8.3.1.1.1.5 de la Recommandation X.411.

mt-redirect-history ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX RedirectHistory  
MULTI VALUE  
::= id-att-redirect-history

#### 11.2.35 *Enveloppe de remise de rapport*

Cet attribut général contient tous les paramètres de l'opération abstraite remise de rapports à l'exception du paramètre contenu renvoyé (s'il existe). On trouvera de plus amples détails au § 8.3.1.2 de la Recommandation X.411.

mt-report-delivery-envelope ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX ReportDeliveryEnvelope  
SINGLE VALUE  
::= id-att-report-delivery-envelope

#### 11.2.36 *Indication du nom de la liste de distribution*

Cet attribut général contient le nom O/R de la liste de distribution qui a retransmis le rapport au propriétaire de cette liste de distribution. On trouvera de plus amples détails au § 8.3.1.2.1.4 de la Recommandation X.411.

mt-reporting-DL-name ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX ReportingDLName  
SINGLE VALUE  
::= id-att-reporting-DL-name

#### 11.2.37 *Indication de certificat MTA*

Cet attribut général contient le certificat du MTA qui a créé le rapport. On trouvera de plus amples détails au § 8.3.1.2.1.12 de la Recommandation X.411.

mt-reporting-MTA-certificate-ATTRIBUTE  
WITH ATTRIBUTE-SYNTAX ReportingMTACertificate  
SINGLE VALUE  
::= id-att-reporting-MTA-certificate

#### 11.2.38 *Vérification d'authentification de l'origine du rapport*

Cet attribut général offre un moyen d'authentifier l'origine du rapport. On trouvera de plus amples détails au § 8.3.1.2.1.13 de la Recommandation X.411.

```
mt-report-origin-authentication-check ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ReportOriginAuthenticationCheck
  SINGLE VALUE
  ::= id-att-report-origin-authentication-check
```

#### 11.2.39 *Classification de sécurité*

Cet attribut général comprend le paramètre classification de sécurité provenant de l'étiquette de sécurité du message. Il est défini comme un attribut distinct afin de pouvoir être utilisé dans l'opération abstraite résumé. On trouvera de plus amples détails au § 8.5.9 de la Recommandation X.411.

```
mt-security-classification ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX SecurityClassification
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-security-classification
```

#### 11.2.40 *Numéro d'ordre*

Cet attribut général est utilisé pour identifier l'entrée proprement dite. Il est attribué par le MS lorsque l'entrée est créée. On trouvera de plus amples détails au § 6.3.2.

```
ms-sequence-number ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX SequenceNumber
  MATCHES FOR EQUALITY ORDERING
  SINGLE VALUE
  ::= id-att-sequence-number
```

#### 11.2.41 *Identificateur de dépôt du sujet*

Cet attribut général contient l'identificateur de dépôt du message ou de l'essai du sujet du rapport. On trouvera de plus amples détails au § 8.3.1.2.1.1 de la Recommandation X.411.

```
mt-subject-submission-identifiant ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX SubjectSubmissionIdentifier
  SINGLE VALUE
  ::= id-att-subject-submission-identifiant
```

#### 11.2.42 *Nom de ce destinataire*

Cet attribut général contient le nom O/R de ce destinataire (MS) de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.3 de la Recommandation X.411.

```
mt-this-recipient-name ATTRIBUTE
  WITH ATTRIBUTE-SYNTAX ORName
  MATCHES FOR EQUALITY
  SINGLE VALUE
  ::= id-att-this-recipient-name
```

#### 11.3 *Création des attributs généraux*

Ce paragraphe décrit de quelle manière sont créés les attributs généraux. Les informations sont présentées dans le tableau 2/X.413. On trouvera une description des classifications utilisées au § 5.3.

TABLEAU 2/X.413

## Création des types d'attributs généraux

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Paramètre source	Source créée par	Règles de création
<b>Child-sequence-numbers</b>	M	–	MS	Une valeur est créée pour chaque entrée secondaire correspondant à une entrée principale
<b>Content</b>	S	<b>contenu</b> <b>contenu renvoyé</b>	MD RD	La valeur du paramètre est la valeur d'attribut
<b>Content-confidentiality-algorithm-identifier</b>	S	<b>identificateur de l'algorithme de confidentialité du contenu</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Content-correlator</b>	S	<b>corrélateur de contenu</b>	RD	La valeur du paramètre est la valeur d'attribut
<b>Content-identifier</b>	S	<b>identificateur de contenu</b>	MD RD	La valeur du paramètre est la valeur d'attribut
<b>Content-integrity-check</b>	S	<b>vérification d'intégrité du contenu</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Content-length</b>	S	–	MS	La taille (approximative) en octets du contenu mis en mémoire basée sur le contenu remis ou le contenu renvoyé
<b>Content-returned</b>	S	–	MS	La valeur est mise sur <b>vrai</b> si le paramètre contenu renvoyé est présent dans l'opération remise de rapports et sur <b>faux</b> si ce paramètre n'est pas présent
<b>Content-type</b>	S	<b>type de contenu</b>	MD RD	La valeur du paramètre, si elle est représentée par IDENTIFICATEUR D'OBJET. Si elle est représentée par NOMBRE ENTIER, convertir à l'IDENTIFICATEUR D'OBJET correspondant
<b>Conversion-with-loss-prohibited</b>	S	<b>conversion avec perte interdite</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Converted-EITs</b>	M	<b>types de codages convertis</b>	MD	Une valeur correspondante est créée pour chaque bit mis à <b>1</b> dans le paramètre <b>type de codage implicite</b> et à partir de chaque <b>type de codage externe</b> présent dans le paramètre <b>type de codage externe</b>
<b>Creation-time</b>	S	–	MS	La date et l'heure de création de l'entrée
<b>Delivered-EITs</b>	M	<b>types de codages convertis et types de codages d'origine</b>	MS	Union des deux autres types d'attributs généraux

TABLEAU 2/X.413 (suite)

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Paramètre source	Source créée par	Règles de création
<b>Delivery-flags</b>	S	<b>indicateurs de remise</b>	MD	La valeur du paramètre est la valeur d'attribut. Si aucun <b>indicateur de remise</b> ne figure dans le MD, il est créé une valeur par défaut avec aucun ensemble d'indicateurs
<b>DL-expansion-history</b>	M	<b>chronologie de l'allongement de la liste de distribution</b>	MD RD	Une valeur correspondante est créée à partir de chaque composant de la SÉQUENCE
<b>Entry-status</b>	S	–	MS	Créé lorsque l'entrée est créée avec la valeur «nouveau»
<b>Entry-type</b>	S	<b>ARGUMENT remise de message</b> <b>ARGUMENT remise de rapport</b>	MS MS	La valeur est mise à «message remis» La valeur est mise à «rapport remis». Si un paramètre contenu renvoyé est présent, une entrée secondaire contenant le paramètre contenu renvoyé est créée. Pour l'entrée secondaire, la valeur de cet attribut est mise à «contenu renvoyé»
<b>Intended-recipient-name</b>	S	<b>nom du destinataire prévu</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Message-delivery-envelope</b>	S	<b>enveloppe</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Message-delivery-time</b>	S	<b>date et heure de remise du message</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Message-delivery-identifier</b>	S	<b>identificateur de remise du message</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Message-origin-authentication-check</b>	S	<b>vérification d'authentification d'origine du message</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Message-security-label</b>	S	<b>étiquette de sécurité du message</b>	MD RD	La valeur du paramètre est la valeur d'attribut
<b>Message-submission-time</b>	S	<b>date et heure de dépôt du message</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Message-token</b>	S	<b>symbole de message</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Original-EITs</b>	M	<b>types de codage d'origine</b>	MD RD	Une valeur correspondante est créée pour chaque bit mis à <b>1</b> dans le paramètre <b>types de codages implicites</b> et à partir de chaque <b>type de codage externe</b> présent dans le paramètre <b>type de codage externe</b>
<b>Originator-certificate</b>	S	<b>certificat de l'expéditeur</b>	MD	La valeur du paramètre est la valeur d'attribut

TABLEAU 2/X.413 (fin)

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Paramètre source	Source créée par	Règles de création
<b>Originator-name</b>	S	<b>nom de l'expéditeur</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Other-recipient-names</b>	M	<b>noms des autres destinataires</b>	MD	Une valeur correspondante est créée pour chaque composant de la SÉQUENCE
<b>Parent-sequence-number</b>	S	–	MS	Lorsque l'on crée une entrée secondaire, cet attribut est créé avec le numéro d'ordre correspondant de l'entrée principale comme valeur
<b>Per-recipient-report-delivery-fields</b>	M	<b>champs par destinataire</b>	RD	Une valeur correspondante est créée à partir de chaque composant de la SÉQUENCE
<b>Priority</b>	S	<b>priorité</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Proof-of-delivery-request</b>	S	<b>demande de preuve de remise</b>	MD	La valeur du paramètre est la valeur d'attribut
<b>Redirection-history</b>	M	<b>chronologie du réacheminement</b>	MD	Une valeur correspondante est créée à partir de chaque composant de la SÉQUENCE
<b>Report-delivery-envelope</b>	S	<b>enveloppe</b>	RD	La valeur du paramètre est la valeur d'attribut
<b>Reporting-DL-name</b>	S	<b>indication du nom de la liste de distribution</b>	RD	La valeur du paramètre est la valeur d'attribut
<b>Reporting-MTA-certificate</b>	S	<b>indication de certificat MTA</b>	RD	La valeur du paramètre est la valeur d'attribut
<b>Report-origin-authentication-check</b>	S	<b>vérification d'authentification d'origine du rapport</b>	RD	La valeur du paramètre est la valeur d'attribut
<b>Security-classification</b>	S	<b>classification de sécurité</b>	MD RD	La valeur du paramètre est la valeur d'attribut
<b>Sequence-number</b>	S	–	MS	Lorsque l'on crée une entrée, le MS attribue une valeur unique à cet attribut par ordre croissant
<b>Subject-submission-identifier</b>	S	<b>identificateur de dépôt de sujet</b>	RD	La valeur du paramètre est la valeur d'attribut
<b>This-recipient-name</b>	S	<b>nom de ce destinataire</b>	MD	La valeur du paramètre est la valeur d'attribut

*Remarque* – Lorsqu'une entrée remise de message est créée, il n'y a pas d'attributs généraux distincts créés pour les arguments remise physique et méthode de remise car les informations contenues dans ces arguments ne sont pas pertinentes pour le MS. Toutefois, l'agent de l'utilisateur (UA) peut rechercher toutes les informations contenues dans ces arguments à l'aide de l'attribut général enveloppe de remise de messages.

## 11.4 Abonnement aux types d'attributs

L'abonnement aux types d'attributs est une question locale. Si l'abonnement à un type d'attribut est modifié, l'agent de l'utilisateur (UA) peut recevoir tous les attributs de l'abonnement d'origine aux messages présents dans le MS au moment où l'abonnement a été modifié. Le traitement de ces attributs qui ne font pas l'objet d'un abonnement est une question locale. De même, lorsque l'agent de l'utilisateur s'abonne à un nouvel attribut, il peut ne pas recevoir cet attribut pour les messages du MS lorsque l'abonnement a été souscrit.

## 12 Types d'actions automatiques générales

Les **types d'actions automatiques générales** sont valables pour tous les types de contenu de message. Toutefois, leur effet précis peut varier selon le contenu; il s'ensuit donc que les descriptions de procédures données dans la présente Recommandation sont susceptibles d'être complétées dans les Recommandations respectives; à titre d'exemple, la procédure spécifique au service de messagerie de personne à personne (IPMS) pour le **type d'action automatique générale** retransmission automatique est décrite au § 19.4 de la Recommandation X.420. D'autres **types d'actions automatiques** ayant un contenu bien précis peuvent être définis dans leurs Recommandations respectives.

Les **actions automatiques**, présentées au § 6.5, sont enregistrées et désenregistrées à l'aide de l'opération abstraite enregistrement-MS décrite au § 8.6.

Les **types suivants d'actions automatiques générales** sont définis:

- a) retransmission automatique;
- b) avertissement automatique.

Le fonctionnement des **actions automatiques** peut être affecté par l'application de mesures de sécurité.

Les alinéas suivants donnent une brève description de chaque **type d'action automatique générale** ainsi que de la syntaxe abstraite qui leur est associée à l'aide de la macro **AUTO-ACTION** définie au § 6.5.

### 12.1 *Retransmission automatique*

L'**action automatique retransmission automatique** permet au prestataire du service abstrait MS de **retransmettre** automatiquement tout message remis dans la base d'information de messages enregistrés. La définition exacte du processus de retransmission est fonction du contenu du message, mais elle suppose toujours le dépôt auprès du service abstrait MTS d'un nouveau message contenant le contenu remis.

Le **type d'action automatique retransmission automatique** permet d'enregistrer avec le MS un ou plusieurs ensembles de paramètres de **retransmission automatique**, dont chacun est identifié par son **identificateur d'enregistrement de retransmission automatique**. Chaque **paramètre d'enregistrement de retransmission automatique** spécifie des critères permettant de déterminer s'il concerne tel ou tel message remis; dans l'affirmative, une copie du message est **retransmise automatiquement** à l'aide de l'opération abstraite dépôt de messages. Un message conforme à plusieurs ensembles de critères est **retransmis automatiquement** autant de fois qu'il y a d'ensembles.

Le **paramètre d'enregistrement de retransmission automatique** spécifie si l'entrée principale (et toute entrée secondaire qui lui est associée) correspondant au message doit être supprimée après la **retransmission automatique**. Si l'un quelconque des paramètres intervenant indique une non-suppression (ou si l'un quelconque des dépôts échoue), l'entrée n'est pas supprimée.

```
auto-forward AUTO-ACTION
  REGISTRATION PARAMETER IS AutoForwardRegistrationParameter
  ::= id-act-auto-forward

AutoForwardRegistrationParameter ::= SET {
  filter [0] Filter OPTIONAL,
  auto-forward-arguments [1] AutoForwardArguments,
  delete-after-auto-forwarding [2] BOOLEAN DEFAULT FALSE,
  other-parameters [3] OCTET STRING OPTIONAL }

AutoForwardArguments ::= SET {
  COMPONENTS OF PerMessageAutoForwardFields,
  per-recipient-fields [1] IMPLICIT SEQUENCE (1..ub-recipients) OF
    PerRecipient-AutoForwardFields }
```

```

PerMessageAutoForwardFields ::= SET {
    originator-name           OriginatorName,
    content-identifiant      ContentIdentifier OPTIONAL,
    priority                  Priority DEFAULT normal,
    per-message-indicators   PerMessageIndicators DEFAULT { },
    deferred-delivery-time   [0] IMPLICIT DeferredDeliveryTime OPTIONAL,
    extensions                [2] IMPLICIT PerMessageSubmissionExtensions DEFAULT { } }

PerRecipientAutoForwardFields ::= SET {
    recipient-name           RecipientName,
    originator-report-request [0] IMPLICIT OriginatorReportRequest,
    explicit-conversion      [1] IMPLICIT ExplicitConversion OPTIONAL,
    extensions                [2] IMPLICIT PerRecipientMessageSubmissionExtensions
                                DEFAULT { } }

```

Les paramètres du **paramètre d'enregistrement de retransmission automatique** ont la signification suivante:

- 1) **Filter (filtre) (F)** – Il s'agit d'un ensemble de critères auxquels une nouvelle entrée représentant un message remis doit se conformer pour que le prestataire du service abstrait MS puisse **retransmettre automatiquement** cette entrée à l'aide de cet ensemble de paramètres.

L'absence de ce paramètre indique que toutes les nouvelles entrées sont **retransmises automatiquement**.

- 2) **Auto-forward-arguments (arguments de retransmission automatique) (O)** – Il s'agit d'un ensemble d'arguments enregistrés à utiliser pour chaque opération abstraite dépôt de messages (voir le § 8.2.1.1.1 de la Recommandation X.411). Tout argument qui n'est ni enregistré, ni obligatoire, ni mentionné spécifiquement ci-dessous, sera absent de chaque dépôt de messages.

Si les arguments suivants ne sont pas enregistrés, ou s'ils sont enregistrés avec leurs valeurs par défaut, les valeurs utilisées pour chaque opération abstraite dépôt de messages sont celles des arguments remise de messages correspondants: **priorité, conversion implicite interdite et conversion avec perte interdite**.

Si les arguments suivants ne sont pas enregistrés, ou s'ils sont enregistrés avec leurs valeurs par défaut, leur présence en tant qu'arguments dépôt de messages dépend de la présence des arguments remise de messages correspondants, leurs valeurs étant transformées, le cas échéant: **jeton de message, identificateur de l'algorithme de confidentialité du contenu, vérification d'intégrité du contenu, vérification d'authentification d'origine du message et étiquette de sécurité du message**.

Certains arguments de dépôt de messages peuvent ne pas être enregistrés, à savoir: **demande de preuve de dépôt, types de codage d'origine, type de contenu et contenu**.

- 3) **Delete-after-auto-forwarding (supprimer après retransmission automatique) (F)** – Ce paramètre indique si une entrée doit être supprimée ou non après le dépôt.

L'absence de ce paramètre indique que le message ne doit pas être supprimé.

- 4) **Other-parameters (autres paramètres) (F)** – Ce paramètre, qui dépend du contenu du message, n'est pas nécessairement présent. Lorsqu'il est présent, les informations qu'il contient seront utilisées pendant la procédure de **retransmission automatique**.

*Remarque* – Ainsi, par exemple, dans le cas du service de messagerie de personne à personne, ce paramètre peut contenir l'**observation retransmission automatique** renvoyée dans l'avis de non-réception, un préfixe spécifié d'utilisateur et une note de couverture accompagnant le message IP étant retransmis automatiquement. L'utilisation de l'**observation retransmission automatique** est décrite au § 19.4 de la Recommandation X.420.

## 12.2 Avertissement automatique

L'**action automatique avertissement automatique** permet au prestataire du service abstrait MS d'*avertir* automatiquement l'utilisateur en aval de l'utilisateur du service abstrait MS de la remise de tout message remis dans la base d'information de messages enregistrés. L'**avertissement automatique** n'aura lieu que pour les entrées message remis.

Le **type d'action automatique avertissement automatique** permet d'enregistrer avec le MS un ou plusieurs ensembles de paramètres d'**avertissement automatique**, chacun étant identifié par son **identificateur d'enregistrement d'avertissement automatique**. Chaque **paramètre d'enregistrement d'avertissement automatique** spécifie des critères permettant de déterminer s'il s'applique à tel ou tel message remis. Si un message est conforme au filtre de plus d'un enregistrement d'avertissement automatique, on procède à l'enregistrement de compatibilité avec l'identificateur

d'enregistrement d'avertissement automatique le moins élevé, et si une adresse au moins (ou l'UA) a été avertie, aucun autre enregistrement n'aura lieu. Si aucune de ces adresses ne peut être avertie, on procédera à l'enregistrement d'avertissement automatique à l'aide de l'identificateur suivant le plus élevé. Ce processus se poursuit jusqu'à ce qu'une adresse au moins ou plusieurs adresses d'un enregistrement aient été averties ou jusqu'à ce que la liste des enregistrements soit épuisée.

L'**opération abstraite avertissement** ne sera appelée que si parmi les adresses d'avertissement de l'enregistrement d'avertissement automatique il y a celle de l'UA [voir le point 2) ci-dessous]. Si cette opération abstraite d'avertissement aboutit, aucune autre adresse contenue dans l'enregistrement d'avertissement automatique ne sera avertie.

auto-alert AUTO-ACTION

REGISTRATION PARAMETER IS AutoAlertRegistrationParameter

::= id-act-auto-alert

AutoAlertRegistrationParameter ::= SET {

filter [0] Filter OPTIONAL,

alert-addresses [1] SEQUENCE SIZE (1..ub-alert-addresses) OF AlertAddress  
OPTIONAL,

requested-attributes [2] EntryInformationSelection OPTIONAL }

Les paramètres du **paramètre d'enregistrement d'avertissement automatique** ont la signification suivante:

- 1) **Filter (filtre) (F)** – Il s'agit d'un ensemble de critères auxquels une nouvelle entrée représentant un message remis doit se conformer pour que le prestataire du service abstrait MS **retransmette automatiquement** cette nouvelle entrée à l'aide de cet ensemble de paramètres.

L'absence de ce paramètre indique que l'**avertissement automatique** sera effectué pour toutes les nouvelles entrées message remis.

- 2) **Alert-addresses (adresses d'avertissement) (F)** – Cet argument identifie les types de services d'**avertissement** à appeler, ainsi que toute information nécessaire pour accéder à des branches particulières de ces services d'**avertissement**, et toute autre information qui doit être acheminée pendant ces **avertissements**.

L'absence de cet argument remplacera l'opération abstraite d'avertissement pour informer l'utilisateur du service abstrait MS de l'existence d'un état d'avertissement, soit en utilisant l'opération abstraite d'avertissement (voir le § 8.7), (ce qui n'est possible que si une association abstraite existe déjà entre l'utilisateur du service abstrait MS et le prestataire du service abstrait MS), soit en signalant à l'aide d'un indicateur dans l'opération abstraite liaison la prochaine fois que l'utilisateur du service abstrait MS établira une association abstraite (voir le § 7). Si le paramètre attributs requis est présent, l'utilisateur du service abstrait MS (UA) sera considéré comme faisant partie des adresses à avertir.

Certains types d'**avertissements** seront normalisés au niveau international. D'autres seront définis par les autorités administratives nationales ou par des organisations privées. Il s'ensuit que plusieurs autorités distinctes seront responsables de l'affectation des types d'avertissement et devront s'assurer que chaque type est distinct de tous les autres types affectés. Cette opération s'effectue en identifiant chaque type à l'aide d'un identificateur d'objet lorsque le type est défini, et en définissant le type de données ASN.1 de l'information d'adressage auxiliaire.

Le **qualificateur d'avertissement** contient toute autre information devant être acheminée pendant l'**avertissement automatique**. L'absence de ce paramètre signifie qu'aucune autre information ne sera transmise à l'utilisateur du service abstrait MS.

AlertAddress ::= SEQUENCE {

address EXTERNAL,

alert-qualifier OCTET STRING OPTIONAL }

- 3) **Requested-attributes (attributs requis) (F)** – Ce paramètre indique les informations des entrées sélectionnées à inclure dans l'avertissement automatique (voir le § 8.1.4).

L'absence de ce paramètre indique que seul l'identificateur d'enregistrement d'avertissement sera présent dans l'argument d'avertissement.

### 13 Présentation générale

Le présent paragraphe décrit les procédures applicables au MS et à la mise en œuvre des accès. On trouve au § 14 une description de l'utilisation du service abstrait MTS. La mise en œuvre du service abstrait MS est décrite au § 15. La mise en œuvre des accès sous forme d'éléments de service est décrite au § 16.

Le fonctionnement des opérations abstraites décrites aux § 14 et 15 sera soumis aux exigences des mesures de sécurité (si de telles mesures sont en vigueur) qui s'appliquent au service abstrait MTS et au service abstrait MS.

### 14 Utilisation du service abstrait du système de transfert de messages (MTS)

Le présent paragraphe indique comment un MS utilisera le service abstrait MTS défini au § 8 de la Recommandation X.411. Il couvre également l'utilisation des accès MTS pour remise, dépôt et gestion.

#### 14.1 *Utilisation des services abstraits de l'accès pour remise*

Ce paragraphe traite des opérations abstraites remise de messages et remise de rapports et de l'appel de l'opération abstraite commande de remise. L'utilisation par le MS des services abstraits de l'accès pour remise suppose qu'il existe une association abstraite entre le fournisseur de l'accès pour remise (le MTA) et l'utilisateur de l'accès pour remise (le MS). Les opérations abstraites sont exécutées séquentiellement et il n'y a pas de traitement parallèle. Les cas d'erreur ne sont pas décrits.

##### 14.1.1 *Fonctionnement de l'opération abstraite remise de messages*

Lorsque le MS reçoit du MTA une opération abstraite remise de messages, il effectue les opérations suivantes:

- 1) Il renvoie un résultat remise de messages au MTA pour indiquer que la remise du message a eu lieu. Le résultat remise de messages doit contenir une preuve de remise si le message remis contient un argument de demande de preuve de remise. Cette preuve de remise peut être calculée à l'aide du code d'accès public MS; on trouvera de plus amples détails aux § 8.5.7 et 8.3.1.1.2.2 de la Recommandation X.411.
- 2) L'étape suivante consiste à voir si l'une quelconque des actions automatiques a été activée. Les actions automatiques sont en partie fonction du contenu et sont donc également décrites dans les Recommandations portant sur le contenu. La description selon le contenu doit comporter des règles sur l'ordre dans lequel les actions automatiques doivent être effectuées. Les actions automatiques peuvent donner lieu à des avertissements, à des dépôts, à la création de nouvelles entrées, voire à la suppression du message remis ou d'autres messages du MS (voir le § 12.1).
  - a) Si des critères de retransmission automatique sont enregistrés, à l'aide de l'opération abstraite enregistrement-MS, la nouvelle entrée est comparée aux critères spécifiés. La mise en correspondance s'effectue séquentiellement pour chaque ensemble spécifié de critères de sélection. En cas de «concordance» un nouveau message est créé puis présenté du MS au MTA à l'aide de l'opération abstraite dépôt de messages (voir le § 15.2.1).

Les règles applicables à l'élaboration du nouveau message retransmis dépendent de nouveau du contenu et sont donc décrites dans les Recommandations pertinentes. D'autres actions ayant un contenu bien précis peuvent également être effectuées à ce stade (par exemple, la suppression du bouclage des messages retransmis automatiquement et l'envoi d'un avis de non-réception) comme c'est le cas pour le service de messagerie de personne à personne (IPMS) décrit au § 19.4 de la Recommandation X.420. Selon les valeurs des arguments de l'opération abstraite enregistrement-MS pour la retransmission automatique, une copie du message remis peut être conservée dans le MS. Si la tentative de retransmission automatique échoue, une copie est toujours conservée pour éviter la perte des messages.

*Remarque* – Le traitement d'un résultat ou d'une erreur provenant du dépôt d'un message est une question locale.

- b) Si des enregistrements d'avertissements automatiques ont été effectués à l'aide de l'opération abstraite enregistrement-MS, la nouvelle entrée est comparée au filtre de chaque enregistrement spécifié. La mise en correspondance se fait séquentiellement pour chaque enregistrement. En cas de

«correspondance», on essaie d'appeler une opération abstraite avertissement du MS à l'UA. Cela n'est possible que s'il existe une association abstraite entre le MS et l'UA. S'il n'y a pas d'association abstraite, le MS pourra disposer d'autres moyens locaux ou non normalisés de solliciter un avertissement. Lorsque des tentatives ont été faites pour avertir l'ensemble des adresses enregistrées pour le premier paramètre d'enregistrement de mise en correspondance et qu'au moins un des avertissements a abouti, l'action automatique avertissement a été menée à bien et il n'est plus procédé à de nouveaux enregistrements d'avertissement. Si aucun moyen n'a été trouvé pour donner l'avertissement, le MS utilise l'indicateur d'avertissement lequel est signalé à l'agent de l'utilisateur lorsque l'agent de l'utilisateur déclenche l'association abstraite suivante en direction du MS.

*Remarque* – Si le message remis a été supprimé à l'issue d'une action de retransmission automatique spécifiée à l'alinéa a), l'avertissement automatique n'a bien entendu pas lieu.

- 3) Ce n'est qu'une fois que ces actions ont été exécutées que la nouvelle entrée est rendue visible en dehors du MS grâce à l'accès pour recherche. Si le message remis a été supprimé à l'issue d'une action automatique, aucun numéro d'ordre attribué au cours de l'opération 2) n'est réutilisé (pour ne pas se heurter aux extensions d'enregistrement des opérations de l'ISO). Le paramètre état de l'entrée est mis sur nouveau.

#### 14.1.1.1 Règles de création applicables aux attributs généraux

Des attributs facultatifs ne sont créés que s'ils sont mis en œuvre par le MS et s'ils font l'objet d'un abonnement de la part de l'utilisateur. Les attributs créés constituent une nouvelle entrée du MS (dans certains cas, une entrée principale et des entrées secondaires, voir le § 6).

On trouvera au tableau 1/X.413 et au § 11.3 les règles de création des attributs généraux. Il convient de noter que pour les attributs généraux absents de l'enveloppe de remise correspondante, il est créé dans l'entrée un attribut auquel est associée une valeur par défaut.

#### 14.1.2 Fonctionnement de l'opération abstraite remise de rapports

Lorsque le MS reçoit du MTA une opération abstraite remise de rapports, il effectue les opérations suivantes:

- 1) il envoie un résultat remise de rapports au MTA pour indiquer que la remise a eu lieu. Le résultat remise de rapports n'a pas de paramètres. On trouvera de plus amples détails au § 8.3.1.2.2 de la Recommandation X.411;
- 2) ensuite, si l'une quelconque des actions automatiques ou toute autre procédure interne est activée, elle est effectuée. Elles varient selon le contenu et sont décrites dans les Recommandations pertinentes.

#### 14.1.2.1 Règles de création applicables aux attributs généraux

Les attributs peuvent être créés soit lorsqu'un message est reçu soit lorsqu'une opération abstraite est effectuée dans le MS après avoir été déclenchée à la demande de l'agent de l'utilisateur.

Tous les attributs obligatoires (voir le tableau 1/X.413) sont créés. Les attributs facultatifs ne sont créés que s'ils sont mis en œuvre par le MS et s'ils font l'objet d'un abonnement de la part de l'utilisateur. Les attributs créés constituent une nouvelle entrée (dans certains cas une entrée principale et des entrées secondaires, voir le § 6) du MS. Les types suivants d'attributs généraux peuvent être créés dans le cadre de ce processus:

- a) attributs généraux créés par le MS même (par exemple, numéro d'ordre);
- b) attributs généraux créés à partir des composants de l'enveloppe de remise de messages. Pour les composants qui ne sont pas présents, mais pour lesquels des valeurs par défaut sont spécifiées, un attribut général contenant la valeur par défaut est créé.

Les règles de création applicables aux cas a) et b) sont décrites au § 14.1.1.1. Les règles de création applicables aux attributs spécifiques de contenu sont décrites dans les Recommandations pertinentes portant sur le contenu. A titre d'exemple, les attributs spécifiques au service de messagerie de personne à personne (IPMS) sont décrits dans l'annexe C de la Recommandation X.420.

On trouvera au tableau 1/X.413 et au § 11.3 les règles de création des attributs généraux. A noter que pour les attributs généraux absents de l'enveloppe de rapport correspondante, un attribut contenant la valeur par défaut est créé dans l'entrée.

### 14.1.3 *Appel de l'opération abstraite commande de remise*

Si le MS souhaite empêcher temporairement le MTA soit de transmettre des messages et des rapports, soit de modifier la longueur maximale du contenu ou le rang de priorité minimal des messages et des rapports provenant du MTA, il effectue les opérations suivantes:

- 1) il appelle une opération abstraite commande de remise contenant les paramètres à modifier. On trouvera de plus amples détails au § 8.3.1.3 de la Recommandation X.411;
- 2) il obtient un résultat en retour lorsque le service abstrait MTS a accepté les modifications. Le résultat contient des informations permettant de savoir si des messages et/ou des rapports sont en attente dans le MTA, en raison des restrictions en vigueur. On trouvera de plus amples détails au § 8.3.1.3.2 de la Recommandation X.411;
- 3) lorsque le MS est en mesure d'accepter de nouveau tout message et/ou tout rapport en attente, il doit appeler une nouvelle opération abstraite commande de remise pour annuler les restrictions. Les effets d'une opération abstraite commande de remise sont annulés lorsqu'une nouvelle opération abstraite commande de remise modifie les restrictions ou lorsque l'association abstraite est libérée.

### 14.2 *Utilisation des services abstraits accès pour dépôt*

Ce paragraphe traite de l'appel des opérations abstraites, dépôt de messages, dépôt d'essai et annulation de remise différée ainsi que de l'utilisation de l'opération abstraite commande de dépôt. L'utilisation par le service abstrait MS des services abstraits de l'accès pour dépôt suppose qu'une association abstraite existe entre le fournisseur de l'accès pour dépôt (MTA) et l'utilisateur de l'accès pour dépôt (le MS). Les opérations abstraites sont exécutées séquentiellement et aucun traitement parallèle n'a lieu. Les cas d'erreur ne sont pas décrits.

#### 14.2.1 *Appel de l'opération abstraite dépôt de messages*

Le déclenchement d'une association abstraite dépôt de messages peut avoir lieu soit par une action automatique dans le MS soit par l'agent de l'utilisateur qui a fait appel au MS pour une opération abstraite dépôt de messages. Pour déposer le message auprès du MTA, le MS effectue les opérations suivantes:

- 1) si l'argument dépôt de messages ne contient pas l'extension demande de retransmission (voir le § 6.6), il sollicite une opération abstraite dépôt de messages contenant le message à déposer ainsi que les paramètres qui lui sont associés. On trouvera de plus amples détails au § 8.2.1.1 de la Recommandation X.411. Par ailleurs, il faut vérifier que l'entrée est un message remis et qu'elle comprend des informations provenant d'une entrée message remis de la base d'information de messages enregistrés, puis appeler l'opération abstraite dépôt de messages avec le nouveau contenu. La retransmission d'entrées qui ne sont pas des messages remis nécessite un complément d'étude.  
A noter que cette demande de retransmission, tout en étant générique, n'est pas nécessairement significative pour tous les types de contenu. Lorsqu'elle est significative, le type de contenu de l'entrée message remis en question doit pouvoir être inclus dans l'argument de contenu;
- 2) il obtient un résultat dépôt de messages en retour lorsque le MTA a accepté le dépôt. Le résultat dépôt de messages contient, entre autres, des informations sur l'identification, la date et l'heure de dépôt du message déposé. On trouvera de plus amples détails au § 8.2.1.1.2 de la Recommandation X.411;
- 3) si l'opération abstraite dépôt de messages a été déclenchée par une opération abstraite correspondante dépôt de messages de l'UA à destination du MS, le résultat de l'opération abstraite est transmis en retour à l'UA sous forme d'un résultat dépôt de messages émis par le MS. Cette procédure garantit que le message a bien été accepté par le MTA avant que le résultat ne soit transmis en retour à l'UA;
- 4) si le MTA n'a pas accepté le dépôt du message en raison de problèmes tels qu'un numéro d'ordre non valable ou un type de contenu inapproprié, le MS doit créer une erreur de demande incohérente. A noter que toutes les erreurs créées par le MTA sont communiquées à l'UA;
- 5) si des mesures de sécurité sont en vigueur, le MS compare l'étiquette de sécurité du message au contexte de sécurité pour s'assurer que ces mesures ne sont pas transgressées pendant le dépôt du message. Si le dépôt de message est interdit soit par les mesures de sécurité soit par des restrictions de sécurité temporaires, une erreur de sécurité doit être signalée.

#### 14.2.2 *Appel de l'opération abstraite dépôt d'essai*

Une opération abstraite dépôt d'essai est déclenchée lorsque l'UA a fait appel au MS pour une opération abstraite dépôt d'essai. Pour déposer l'essai auprès du MTA, le MS effectue les opérations suivantes:

- 1) il appelle une opération abstraite dépôt d'essai contenant l'essai à déposer ainsi que les paramètres qui lui sont associés. On trouvera de plus amples détails au § 8.2.1.2.1 de la Recommandation X.411;
- 2) il obtient un résultat dépôt d'essai en retour lorsque le MTA a accepté le dépôt. Le résultat contient, entre autres, des informations sur l'identification, la date et l'heure de dépôt de l'essai déposé. On trouvera de plus amples détails au § 8.2.1.2.2 de la Recommandation X.411;
- 3) le résultat de l'opération abstraite est transmis en retour à l'UA sous forme d'un résultat dépôt d'essai émis par le MS. Cette procédure garantit que l'essai a bien été accepté par le MTA avant que le résultat ne soit transmis en retour à l'UA;
- 4) si des mesures de sécurité sont en vigueur, le MS compare l'étiquette de sécurité du message de l'essai au contexte de sécurité pour s'assurer que ces mesures ne sont pas transgressées pendant le dépôt d'essai. Si le dépôt d'essai est interdit soit par les mesures de sécurité soit par des restrictions de sécurité temporaires, une erreur dépôt d'essai est créée.

#### 14.2.3 *Appel de l'opération abstraite annulation de remise différée*

Une opération abstraite annulation de remise différée est déclenchée lorsque l'UA a fait appel au MS pour une opération abstraite annulation de remise différée. Pour transmettre l'annulation au MTA, le MS effectue les opérations suivantes:

- 1) il appelle une opération abstraite annulation de remise différée, contenant l'annulation à déposer ainsi que les paramètres qui lui sont associés. On trouvera de plus amples détails au § 8.2.1.3.1 de la Recommandation X.411;
- 2) il obtient un résultat en retour lorsque le MTA a accepté l'annulation. Le résultat renvoyé, s'il est vide, est une indication de succès;
- 3) le résultat de l'opération abstraite est transmis en retour à l'UA sous forme d'un résultat annulation de remise différée soumis par le MS. Cette procédure garantit que l'essai a bien été accepté (ou n'a pas été accepté) par le MTA avant que le résultat ne soit transmis en retour à l'UA.

#### 14.2.4 *Fonctionnement de l'opération abstraite commande de dépôt*

Si le MTA souhaite empêcher temporairement le MS soit de transmettre des messages ou des essais, soit de modifier la longueur maximale du contenu ou le rang de priorité minimal de messages provenant du MS, il appelle une opération abstraite commande de dépôt. On trouvera de plus amples détails au § 8.2.1.4.1 de la Recommandation X.411. Le MS réagit en effectuant les opérations suivantes:

- 1) il fait appel à l'UA pour une opération abstraite correspondante commande de dépôt provenant du MS;
- 2) il attend que l'UA envoie un résultat commande de dépôt contenant des informations permettant de savoir si des messages ou des essais sont en attente dans l'UA en raison de restrictions en vigueur. On trouvera de plus amples détails au § 8.2.1.4.2 de la Recommandation X.411;
- 3) le MS envoie en retour au MTA un résultat commande de dépôt contenant des informations provenant de l'UA;
- 4) lorsque le MTS est en mesure d'accepter de nouveau un message ou un essai, il doit appeler une nouvelle opération abstraite commande de dépôt pour annuler les restrictions. Les effets d'une opération abstraite commande de dépôt sont annulés lorsqu'une nouvelle opération abstraite commande de dépôt modifie les restrictions ou lorsque l'association abstraite est libérée. Le MS fait ensuite appel à l'UA pour une opération abstraite correspondante commande de dépôt et attend le résultat commande de dépôt.

#### 14.3 *Utilisation des services abstraits accès pour gestion*

Ce paragraphe traite du fonctionnement des opérations abstraites enregistrement et modification des pouvoirs. L'utilisation des services abstraits accès pour gestion suppose qu'une association abstraite existe entre le fournisseur de l'accès pour gestion (le MTA) et l'utilisateur de l'accès pour gestion (le MS). Les opérations abstraites sont exécutées séquentiellement et aucun traitement parallèle n'a lieu. Les cas d'erreur ne sont pas décrits.

L'utilisation par le MS de l'accès pour gestion est soumise aux mesures de sécurité en vigueur.

#### 14.3.1 *Appel de l'opération abstraite enregistrement*

Une opération abstraite enregistrement est déclenchée parce que l'UA a fait appel au MS pour une opération abstraite enregistrement. Pour envoyer l'enregistrement au MTA, le MS effectue les opérations suivantes:

- 1) il appelle une opération abstraite enregistrement, contenant les nouvelles données à enregistrer. On trouvera de plus amples détails au § 8.4.1.1.1 de la Recommandation X.411;
- 2) il obtient un résultat en retour lorsque le MTA a accepté l'enregistrement. Le résultat renvoyé, s'il est vide, est une indication de succès;
- 3) l'ampleur des modifications que l'UA peut apporter, par l'intermédiaire du MS, aux arguments des étiquettes de sécurité de l'utilisateur doit respecter les mesures de sécurité en vigueur.

#### 14.3.2 *Appel de l'opération abstraite modification des pouvoirs*

Une opération abstraite modification des pouvoirs est déclenchée parce que l'UA a fait appel au MS pour une opération abstraite modification des pouvoirs. Pour transmettre les nouveaux pouvoirs de l'UA au MTA, le MS effectue les opérations suivantes:

- 1) il fait appel au MTA pour une opération abstraite modification des pouvoirs contenant les nouveaux pouvoirs à enregistrer. On trouvera de plus amples détails au § 8.4.1.2.1 de la Recommandation X.411;
- 2) il obtient en retour un résultat modification des pouvoirs lorsque le MTA a accepté la modification et il enregistre les nouveaux pouvoirs. Le résultat modification des pouvoirs ou l'erreur résultante provenant du MTA est transmis à l'UA. Un résultat vide indique un succès.

#### 14.3.3 *Fonctionnement de l'opération abstraite modification des pouvoirs*

Lorsque le MS reçoit du MTA une opération abstraite modification des pouvoirs et les arguments qui lui sont associés, il effectue les opérations suivantes:

- 1) il s'assure que les informations relatives à l'argument sont valables pour une opération abstraite modification des pouvoirs. On trouvera de plus amples détails au § 8.4.1.2 de la Recommandation X.411;
- 2) il vérifie qu'il existe déjà une association abstraite entre le MS et l'UA. Si une association abstraite entre le MS et l'UA n'existe pas, le MTA est informé par une erreur indiquant que la modification des pouvoirs ne peut avoir lieu pour l'instant et qu'aucune autre action n'est entreprise;
- 3) si l'association abstraite entre le MS et l'UA existe, le MS fait appel à l'UA pour une opération abstraite modification des pouvoirs;
- 4) si l'UA envoie en retour un résultat modification des pouvoirs vide, indiquant un succès, le MS envoie en retour au MTA un résultat modification des pouvoirs correspondant indiquant un succès et enregistre les pouvoirs. Si l'UA renvoie une erreur, elle est transmise au MTA pour indiquer cette erreur. A noter que le MS ne renvoie jamais une indication de succès au MTA avant d'avoir reçu le résultat correspondant de l'UA.

## 15 **Fourniture du service abstrait enregistrement de messages**

Ce paragraphe indique comment un MS fournit le service abstrait MS. Il traite encore de la mise en œuvre des accès pour recherche, pour dépôt indirect et pour gestion.

### 15.1 *Fourniture des services abstraits accès pour recherche*

Ce paragraphe traite de la fourniture des opérations abstraites, résumé, listage, extraction, suppression, enregistrement-MS et avertissement. La fourniture par le service abstrait MS des services abstraits accès pour recherche suppose qu'une association abstraite existe entre le fournisseur de l'accès pour recherche (le MS) et l'utilisateur de l'accès pour recherche (l'UA). Les opérations abstraites sont effectuées séquentiellement et aucun traitement parallèle n'a lieu. Tous les cas d'erreur ne sont pas décrits.

#### 15.1.1 *Fonctionnement de l'opération abstraite résumé*

Lorsque le MS reçoit de l'UA une opération abstraite résumé, il effectue les opérations suivantes:

- 1) il établit à quelle base d'information s'adresse l'opération abstraite résumé;

- 2) il vérifie s'il y a des entrées dans la base d'information. Si cette base d'information est vide, un résultat résumé de longueur zéro est renvoyé et aucune autre action n'est effectuée;
- 3) il s'assure que les attributs généraux d'arguments fournis et quelques attributs spécifiques de contenu reconnus par le MS sont valables pour une opération abstraite résumé. On trouvera de plus amples détails au § 8.2.1;
- 4) il totalise les comptes conformément aux attributs généraux de l'argument fourni et quelques attributs spécifiques de contenu reconnus par le MS;
- 5) il renvoie le résultat résumé à l'UA. On trouvera de plus amples détails au § 8.2.2;
- 6) si des mesures de sécurité sont en vigueur, le MS compare la classification de sécurité de l'étiquette de sécurité au contexte de sécurité pour s'assurer que ces mesures ne sont pas transgressées pendant l'opération abstraite résumé. Si une opération abstraite résumé est interdite par les mesures de sécurité, elle doit être abandonnée et une erreur de sécurité doit être signalée.

#### 15.1.2 *Fonctionnement de l'opération abstraite listage*

Lorsque le MS reçoit de l'UA une opération abstraite listage, il effectue les opérations suivantes:

- 1) il établit à quelle base d'information s'adresse l'opération abstraite listage;
- 2) il vérifie que les attributs généraux de l'argument fourni et quelques attributs spécifiques de contenu reconnus par le MS sont valables pour une opération abstraite listage. On trouvera de plus amples détails au § 8.3.1;
- 3) il identifie zéro, une ou plusieurs entrées selon l'argument de l'opération abstraite, jusqu'à la limite indiquée. Les entrées secondaires rattachées à une entrée principale sont exclues à moins qu'elles ne soient explicitement retenues dans l'argument;
- 4) si un ensemble d'attributs généraux requis ont été spécifiés comme arguments de l'opération abstraite, ces attributs généraux, s'ils sont présents, sont renvoyés à l'UA pour chaque entrée sélectionnée. Si aucune demande n'a été faite, les valeurs par défaut de l'opération abstraite listage, telles qu'elles ont été précédemment spécifiées à l'aide d'une opération abstraite enregistrement-MS, sont renvoyées si elles sont présentes. On trouvera de plus amples détails au § 8.3.2. La valeur du paramètre état de l'entrée pour chaque message sélectionné est mise sur liste;
- 5) si des mesures de sécurité sont en vigueur, le MS compare l'étiquette de sécurité du message au contexte de sécurité pour s'assurer que ces mesures ne sont pas transgressées pendant l'opération abstraite listage. Si l'opération abstraite listage est interdite soit par des mesures de sécurité soit par des restrictions de sécurité temporaires, elle doit être abandonnée et une erreur de sécurité doit être signalée.

#### 15.1.3 *Fonctionnement de l'opération abstraite extraction*

Lorsque le MS reçoit de l'UA une opération abstraite extraction, il effectue les opérations suivantes:

- 1) il établit à quelle base d'information s'adresse l'opération abstraite extraction;
- 2) il vérifie que les attributs généraux de l'argument fourni et quelques attributs spécifiques de contenu reconnus par le MS sont valables pour une opération abstraite extraction. On trouvera de plus amples détails au § 8.4.1;
- 3) il identifie zéro, une ou plusieurs entrées selon l'argument de l'opération abstraite, jusqu'à la limite indiquée. Les entrées secondaires rattachées à une entrée principale sont exclues, à moins qu'elles ne soient explicitement retenues dans l'argument;
- 4) si un ensemble d'attributs généraux requis ont été spécifiés comme arguments de l'opération abstraite, ces attributs généraux, s'ils sont présents, sont renvoyés à l'UA pour la première entrée sélectionnée. Si aucune demande n'a été faite, les valeurs par défaut de l'opération abstraite extraction telles qu'elles ont été précédemment spécifiées à l'aide de l'opération abstraite enregistrement-MS, sont renvoyées, si elles sont présentes. Si plusieurs entrées répondent aux critères de recherche, les numéros d'ordre affectés à la deuxième entrée et aux entrées suivantes sont renvoyés dans l'ordre croissant. Si le nombre d'entrées répondant aux critères est supérieur à la limite spécifiée, le numéro d'ordre qui suit immédiatement la limite est également renvoyé. On trouvera de plus amples détails au § 8.4.2;
- 5) si des mesures de sécurité sont en vigueur, le MS compare l'étiquette de sécurité du message au contexte de sécurité pour s'assurer que ces mesures ne sont pas transgressées pendant l'opération abstraite extraction. Si l'opération abstraite extraction est interdite soit par des mesures de sécurité, soit par des restrictions de sécurité temporaires, elle doit être abandonnée et une erreur de sécurité doit être signalée.

#### 15.1.4 *Fonctionnement de l'opération abstraite suppression*

Lorsque le MS reçoit de l'UA une opération abstraite suppression, il effectue les opérations suivantes:

- 1) il établit à quelle base d'information s'adresse l'opération abstraite suppression;
- 2) il vérifie que les arguments fournis sont valables pour une opération abstraite suppression. On trouvera de plus amples détails au § 8.5.1;
- 3) il identifie l'entrée ou la liste d'entrées requises dans l'argument de l'opération abstraite;
- 4) si l'une quelconque des entrées comporte des restrictions de suppression (voir le § 8.5), aucune des suppressions n'a lieu. Dans tous les autres cas, toutes les suppressions sont effectuées et un résultat suppression vide est renvoyé à l'UA comme indication de succès.

#### 15.1.5 *Fonctionnement de l'opération abstraite enregistrement-MS*

Lorsque le MS reçoit de l'UA une opération abstraite enregistrement-MS, il effectue les opérations suivantes:

- 1) il vérifie que les arguments fournis sont valables pour une opération abstraite enregistrement-MS. On trouvera de plus amples détails au § 8.6.1;
- 2) il remplace les anciens paramètres par les nouveaux paramètres correspondants. Les actions automatiques ont une incidence sur des opérations comme les remises de messages et les remises de rapports qui ont lieu après le déclenchement ou la suppression de demandes d'actions automatiques; il n'y a, à ce stade, aucun traitement des entrées déjà contenues dans le MS;
- 3) il renvoie à l'UA un résultat enregistrement-MS vide pour indiquer que l'opération abstraite a été menée à bien;
- 4) si des mesures de sécurité sont en vigueur, l'opération abstraite enregistrement-MS devra les respecter. Selon certaines mesures de sécurité, on ne peut modifier les étiquettes de sécurité de l'utilisateur que si l'on utilise une liaison sûre. Il existe d'autres moyens locaux pour modifier de façon sûre les étiquettes de sécurité de l'utilisateur.

#### 15.1.6 *Appel de l'opération abstraite avertissement*

L'appel de l'opération abstraite avertissement résulte de l'utilisation du service abstrait accès pour remise (voir le § 14.1.1).

Si l'action automatique avertissement est déclenchée par l'UA au moyen d'une opération abstraite enregistrement-MS antérieure, le service abstrait MS effectue les opérations suivantes:

- 1) il s'assure qu'une association abstraite existe. Dans le cas contraire, le MS n'établit jamais une association abstraite et il est alors impossible d'appeler une opération abstraite avertissement;
- 2) s'il existe une association abstraite, le MS appelle une opération abstraite contenant des informations d'argument pertinentes (on trouvera de plus amples détails au § 8.7.1) et attend que l'UA renvoie un résultat avertissement vide comme indication de succès;
- 3) s'il n'existe pas d'association abstraite, on peut utiliser un protocole non normalisé pour informer l'utilisateur. Le signal d'avertissement peut dans ce cas être donné sur le terminal de l'utilisateur mais également par téléphone, à l'aide d'un avertisseur sonore ou de tout autre équipement terminal approprié associé à l'utilisateur. Cette dernière méthode peut également être utilisée si l'opération abstraite avertissement n'a pas été mise en œuvre;
- 4) si des mesures de sécurité sont en vigueur, le MS compare l'étiquette de sécurité du message au contexte de sécurité pour s'assurer que ces mesures ne sont pas transgressées pendant l'opération avertissement. Si l'opération abstraite avertissement est interdite soit par les mesures de sécurité soit par des restrictions de sécurité temporaires, l'action entreprise doit respecter les mesures de sécurité en vigueur.

#### 15.2 *Fourniture des services abstraits accès pour remise indirecte*

Ce paragraphe traite du fonctionnement des opérations abstraites, dépôt de messages, dépôt d'essai et annulation de remise différée ainsi que de l'appel de l'opération abstraite commande de dépôt. La fourniture par le service abstrait MS des services abstraits accès pour dépôt indirect suppose qu'il existe une association abstraite entre le fournisseur de l'accès pour dépôt indirect (le MS) et l'utilisateur de l'accès pour dépôt indirect (l'UA). Les opérations abstraites sont exécutées séquentiellement et aucun traitement parallèle n'a lieu. Tous les cas d'erreur ne sont pas décrits.

### 15.2.1 *Fonctionnement de l'opération abstraite dépôt de messages*

Lorsque le MS reçoit de l'UA une opération abstraite dépôt de messages et les arguments qui lui sont associés, il effectue les opérations suivantes:

- 1) il s'assure que les informations relatives à l'argument sont valables pour une opération abstraite dépôt de messages. On trouvera de plus amples détails au § 8.2.1.1.1 de la Recommandation X.411;
- 2) il vérifie les arguments pour savoir si le contenu du message a été fourni par l'UA ou s'il doit être inséré par le MS (c'est-à-dire si le paramètre extension de la demande de retransmission est présent). Dans ce dernier cas, si l'entrée est une entrée message remis, le message correspondant est inséré et les arguments relatifs au MS sont supprimés. La retransmission d'entrées autres que des entrées message remis doit faire l'objet d'un complément d'étude;
- 3) il vérifie s'il existe déjà une association abstraite entre le MS et le MTA. Dans le cas contraire, le MS déclenche une telle association abstraite. Si une association abstraite ne peut être établie, l'UA en est informé à l'aide d'une erreur indiquant que le dépôt ne peut avoir lieu pour l'instant et qu'aucune autre action n'est effectuée;
- 4) si l'association abstraite entre le MS et le MTA existe, le MS fait appel au MTA pour une opération abstraite dépôt de messages après toute modification indiquée au point 2);
- 5) si le MTA envoie en retour un résultat dépôt de messages (on trouvera de plus amples détails au § 8.2.1.1.2 de la Recommandation X.411) indiquant un succès, le MS envoie en retour à l'UA un résultat correspondant dépôt de messages indiquant le succès. A noter que le MS ne renvoie jamais une indication de succès à l'UA avant d'avoir reçu le résultat correspondant du MTA, ce qui permet de garantir un service cohérent du point de vue de l'utilisateur car le fait de déposer un message signifie toujours que la responsabilité de ce message a été acceptée par le MTA lorsque le résultat est renvoyé;
- 6) le MS peut choisir de mettre fin à l'association abstraite avec le MTA soit après une certaine période d'activité soit lorsque l'UA met fin à son association abstraite correspondante avec le MS.

### 15.2.2 *Fonctionnement de l'opération abstraite dépôt d'essai*

Lorsque le MS reçoit de l'UA une opération abstraite dépôt d'essai et les arguments qui lui sont associés, il effectue les opérations suivantes:

- 1) il s'assure que les informations relatives à l'argument sont valables pour une opération abstraite dépôt d'essai. On trouvera de plus amples détails au § 8.2.1.2.1 de la Recommandation X.411;
- 2) il vérifie s'il existe déjà une association abstraite entre le MS et le MTA. Si ce n'est pas le cas, le MS déclenche cette association abstraite. Si une association abstraite ne peut être établie, l'UA en est informé à l'aide d'une erreur indiquant que le dépôt ne peut avoir lieu pour l'instant et qu'aucune autre opération n'est effectuée;
- 3) si une association abstraite entre le MS et le MTA existe, le MS fait appel au MTA pour une opération abstraite dépôt d'essai;
- 4) si le MTA envoie en retour un résultat dépôt d'essai (on trouvera de plus amples détails au § 8.2.1.2.2 de la Recommandation X.411) indiquant un succès, le MS envoie en retour à l'UA un résultat correspondant dépôt d'essai indiquant un succès. A noter que le MS ne renvoie jamais une indication de succès à l'UA avant d'avoir reçu le résultat correspondant du MTA, ce qui permet de garantir un service cohérent du point de vue de l'utilisateur, car le fait de déposer un essai signifie toujours que la responsabilité de cet essai a été acceptée par le MTS lorsque le résultat est renvoyé;
- 5) le MS peut choisir de mettre fin à l'association abstraite avec le MTA soit après une certaine période d'activité soit lorsque l'UA met fin à son association abstraite correspondante avec le MS.

### 15.2.3 *Fonctionnement de l'opération abstraite annulation de remise différée*

Lorsque le MS reçoit de l'UA une opération abstraite annulation de remise différée et les arguments qui lui sont associés, il effectue les opérations suivantes:

- 1) il s'assure que les informations relatives à l'argument sont valables pour une opération abstraite annulation de remise différée. On trouvera de plus amples détails au § 8.2.1.3.1 de la Recommandation X.411;
- 2) il vérifie s'il existe déjà une association abstraite entre le MS et le MTA. Si ce n'est pas le cas, le MS déclenche cette association abstraite. Si une association abstraite ne peut être établie, l'UA en est informé par une erreur indiquant que l'annulation de remise différée ne peut avoir lieu pour l'instant et qu'aucune autre opération n'est effectuée;

- 3) si l'association abstraite entre le MS et le MTA existe, le MS fait appel au MTA pour une opération abstraite annulation de remise différée;
- 4) si le MTA envoie en retour un résultat annulation de remise différée (on trouvera de plus amples détails au § 8.2.1.3.2 de la Recommandation X.411) indiquant un succès, le MS envoie en retour à l'UA un résultat correspondant annulation de remise différée indiquant un succès. A noter que le MS ne renvoie jamais une indication de succès à l'UA avant d'avoir reçu le résultat correspondant du MTA, ce qui permet de garantir un service cohérent du point de vue de l'utilisateur, car le fait de déposer un essai signifie toujours que la responsabilité de cet essai a été acceptée par le MTS lorsque le résultat est renvoyé;
- 5) le MS peut choisir de mettre fin à l'association abstraite avec le MTA soit après une certaine période d'activité soit lorsque l'UA met fin à son association abstraite correspondante avec le MS.

#### 15.2.4 *Appel de l'opération abstraite commande de dépôt*

Si le MS reçoit du MTA une opération abstraite commande de dépôt ou si, pour diverses raisons internes, le MS veut empêcher temporairement l'UA soit de déposer des messages ou des essais soit de modifier la longueur maximale ou le rang de priorité minimal des messages provenant de l'UA, le MS effectue les opérations suivantes:

- 1) il fait appel à l'UA pour une opération abstraite commande de dépôt. On trouvera de plus amples détails au § 8.2.1.4.1 de la Recommandation X.411;
- 2) il attend que l'UA lui renvoie un résultat commande de dépôt (on trouvera de plus amples détails au § 8.2.1.4.2 de la Recommandation X.411) confirmant l'acceptation de l'opération abstraite commande de dépôt;
- 3) si l'opération abstraite commande de dépôt a été déclenchée par une opération abstraite correspondante demandée au MTA par le MS, le résultat commande de dépôt provenant de l'UA est communiqué du MS au MTA et le MS attend que l'UA lui renvoie un résultat commande de dépôt.

#### 15.3 *Fourniture des services abstraits accès pour gestion*

Ce paragraphe traite du fonctionnement des opérations abstraites enregistrement et modification des pouvoirs. La fourniture par le service abstrait MS des services abstraits accès pour gestion suppose qu'il existe une association abstraite entre le fournisseur de l'accès pour dépôt indirect (le MS) et l'utilisateur de l'accès pour dépôt indirect (l'UA). Les opérations abstraites sont exécutées séquentiellement et aucun traitement parallèle n'a lieu. Tous les cas d'erreur ne sont pas décrits.

##### 15.3.1 *Fonctionnement de l'opération abstraite enregistrement*

Lorsque le MS reçoit de l'UA une opération abstraite enregistrement et les arguments qui lui sont associés, il effectue les opérations suivantes:

- 1) il s'assure que les informations relatives à l'argument sont valables pour une opération abstraite enregistrement. On trouvera de plus amples détails au § 8.4.1.1.1 de la Recommandation X.411;
- 2) il vérifie qu'il existe déjà une association abstraite entre le MS et le MTA. Si ce n'est pas le cas, le MS déclenche cette association abstraite. Si une association abstraite ne peut être établie, l'UA en est informé par une erreur indiquant que l'enregistrement ne peut avoir lieu pour l'instant et qu'aucune autre action n'est effectuée;
- 3) si l'association abstraite entre le MS et le MTA existe, le MS fait appel au MTA pour une opération abstraite enregistrement;
- 4) si le MTA envoie en retour un résultat enregistrement (on trouvera de plus amples détails au § 8.4.1.1.2 de la Recommandation X.411) indiquant un succès, le MS renvoie à l'UA un résultat enregistrement correspondant indiquant un succès. A noter que le MS ne renvoie jamais une indication de succès à l'UA avant d'avoir reçu le résultat correspondant du MTA, ce qui permet de garantir un service cohérent du point de vue de l'utilisateur car la responsabilité de l'enregistrement a été acceptée par le MTS lorsque le résultat est renvoyé;
- 5) le MS peut choisir de mettre fin à l'association abstraite avec le MTA soit après une certaine période d'inactivité soit lorsque l'UA met fin à son association abstraite correspondante avec le MS;
- 6) l'importance des modifications que l'UA peut apporter aux étiquettes de sécurité de l'utilisateur devra respecter les mesures de sécurité en vigueur. Selon certaines mesures de sécurité, on ne peut modifier les étiquettes de sécurité de l'utilisateur que si l'on utilise une liaison sûre. Il existe d'autres moyens locaux pour modifier de façon sûre les étiquettes de sécurité de l'utilisateur.

### 15.3.2 *Appel de l'opération abstraite modification des pouvoirs*

Une opération abstraite modification des pouvoirs est déclenchée parce que le MTA a fait appel au MS pour une opération abstraite modification des pouvoirs. Pour transmettre les nouveaux pouvoirs du MTA à l'UA, le MS effectue les opérations suivantes:

- 1) il s'assure que les informations relatives à l'argument sont valables pour une opération abstraite modification des pouvoirs. On trouvera de plus amples détails au § 8.4.1.2 de la Recommandation X.411. Si les anciens pouvoirs sont incorrects et si les nouveaux ne sont pas acceptables, une erreur est renvoyée et aucune autre action n'a lieu;
- 2) il fait appel à l'UA pour une opération abstraite modification des pouvoirs contenant les nouveaux pouvoirs à enregistrer. On trouvera de plus amples détails au § 8.4.1.2 de la Recommandation X.411;
- 3) il obtient en retour un résultat modification des pouvoirs lorsque l'UA a accepté la modification et il met en mémoire les nouveaux pouvoirs. Le résultat modification des pouvoirs ou l'erreur résultante provenant de l'UA est transmis au MTA.

### 15.3.3 *Fonctionnement de l'opération abstraite modification des pouvoirs*

Lorsque le MS reçoit de l'UA une opération abstraite modification des pouvoirs et les arguments qui lui sont associés, il effectue les opérations suivantes:

- 1) il s'assure que l'information d'argument est valable pour une opération abstraite modification des pouvoirs. On trouvera de plus amples détails au § 8.4.1.2 de la Recommandation X.411;
- 2) il vérifie s'il existe déjà une association abstraite entre le MS et le MTA. Dans le cas contraire, le MS déclenche une telle association abstraite. S'il est impossible d'établir une association abstraite, l'UA est informé à l'aide d'une erreur indiquant que la modification des pouvoirs ne peut avoir lieu pour l'instant et que l'on ne procède à aucune autre opération;
- 3) si l'association abstraite entre le MS et le MTA existe, le MS fait appel au MTA pour une opération abstraite modification des pouvoirs;
- 4) si le MTA envoie en retour un résultat modification des pouvoirs vide, indiquant un succès, le MS envoie en retour un résultat correspondant modification des pouvoirs informant l'UA du succès et enregistre les pouvoirs. Si le MTA renvoie une erreur, elle est transmise à l'UA pour indiquer cette erreur. A noter que le MS ne renvoie jamais une indication de succès à l'UA avant d'avoir reçu le résultat correspondant du MTA;
- 5) le MS peut choisir de mettre fin à l'association abstraite avec le MTA soit après une certaine période d'inactivité, soit lorsque l'UA met fin à son opération abstraite correspondante avec le MS.

## **16 Mise en œuvre des accès**

Ce paragraphe décrit comment les accès pour recherche, pour dépôt et pour gestion du service abstrait MS sont fournis. On trouvera au § 8 de la Recommandation X.411 une description de la fourniture par le service abstrait MTS des accès pour recherche, pour dépôt et pour gestion.

### 16.1 *Accès pour recherche*

Les services abstraits accès pour recherche sont fournis un à un entre les opérations abstraites et les opérations réelles de l'élément de service de recherche de messages (MRSE) dont il est question dans la Recommandation X.419.

### 16.2 *Accès pour dépôt indirect*

Les services abstraits accès pour dépôt indirect sont fournis un à un entre les opérations abstraites et les opérations réelles de l'élément de service de dépôt de message (MSSE) dont il est question dans la Recommandation X.419.

### 16.3 *Accès pour gestion*

Les services abstraits accès pour gestion sont fournis un à un entre les opérations abstraites et les opérations réelles de l'élément de service de gestion de message (MASE) dont il est question dans la Recommandation X.419.

## ANNEXE A

(à la Recommandation X.413)

### Attribution formelle des identificateurs d'objets

La présente annexe fait partie intégrante de la Recommandation.

Tous les identificateurs d'objets, que la présente Recommandation attribue, sont formellement attribués dans la présente annexe à l'aide de la notation de syntaxe abstraite 1 (ASN.1). Les valeurs spécifiées sont indiquées dans les modules de l'ASN.1 des annexes ultérieures.

La présente annexe est définitive pour toutes les valeurs, à l'exception de celles applicables aux modules ASN.1 et à l'ensemble du thème de la présente Recommandation. Les attributions définitives dans le premier cas apparaissent dans les modules eux-mêmes. Dans le second cas, les attributions sont fixes. D'autres références aux valeurs attribuées aux modules apparaissent dans les paragraphes IMPORT.

MXObjectIdentifiers

```
{ joint-iso-ccitt mhs-motis(6) ms(4) modules(0) object-identifiers(0) }
```

DEFINITIONS ::=

BEGIN

-- Prologue

-- Exporte tout

IMPORTS

ID, id-ms

```
FROM MHSObjectIdentifiers { joint-iso-ccitt mhs-motis(6) arch(5) modules(0) object-identifiers(0) };
```

-- Catégories

id-mod -- *modules* -- ID ::= { id-ms 0 }

id-ot -- *objects* -- ID ::= { id-ms 1 }

id-pt -- *port types* -- ID ::= { id-ms 2 }

id-att -- *attribute types* -- ID ::= { id-ms 3 }

id-act -- *auto-action types* -- ID ::= { id-ms 4 }

-- Modules

id-mod-object-identifiers ID ::= { id-mod 0 } -- *not definitive*

id-mod-abstract-service ID ::= { id-mod 1 } -- *not definitive*

id-mod-attribute-types ID ::= { id-mod 2 } -- *not definitive*

id-mod-action-types ID ::= { id-mod 3 } -- *not definitive*

id-mod-upper-bounds ID ::= { id-mod 4 } -- *not definitive*

-- Objets

id-ot-ms ID ::= { id-ot 0 }

id-ot-ms-user ID ::= { id-ot 1 }

-- Types d'accès

id-pt-retrieval ID ::= { id-pt 0 }

-- Types d'attributs

id-att-child-sequence-numbers ID ::= { id-att 0 }

id-att-content ID ::= { id-att 1 }

id-att-content-confidentiality-algorithm-identifier ID ::= { id-att 2 }

id-att-content-correlator ID ::= { id-att 3 }

id-att-content-identifier ID ::= { id-att 4 }

id-att-content-integrity-check ID ::= { id-att 5 }

id-att-content-length ID ::= { id-att 6 }

id-att-content-returned ID ::= { id-att 7 }

id-att-content-type ID ::= { id-att 8 }

id-att-conversion-with-loss-prohibited	ID ::= { id-att 9 }
id-att-converted-EITs	ID ::= { id-att 10 }
id-att-creation-time	ID ::= { id-att 11 }
id-att-delivered-EITs	ID ::= { id-att 12 }
id-att-delivery-flags	ID ::= { id-att 13 }
id-att-dl-expansion-history	ID ::= { id-att 14 }
id-att-entry-status	ID ::= { id-att 15 }
id-att-entry-type	ID ::= { id-att 16 }
id-att-intended-recipient-name	ID ::= { id-att 17 }
id-att-message-delivery-envelope	ID ::= { id-att 18 }
id-att-message-delivery-identifier	ID ::= { id-att 19 }
id-att-message-delivery-time	ID ::= { id-att 20 }
id-att-message-origin-authentication-check	ID ::= { id-att 21 }
id-att-message-security-label	ID ::= { id-att 22 }
id-att-message-submission-time	ID ::= { id-att 23 }
id-att-message-token	ID ::= { id-att 24 }
id-att-original-EITs	ID ::= { id-att 25 }
id-att-originator-certificate	ID ::= { id-att 26 }
id-att-originator-name	ID ::= { id-att 27 }
id-att-other-recipient-names	ID ::= { id-att 28 }
id-att-parent-sequence-number	ID ::= { id-att 29 }
id-att-per-recipient-report-delivery-fields	ID ::= { id-att 30 }
id-att-priority	ID ::= { id-att 31 }
id-att-priority-of-delivery-request	ID ::= { id-att 32 }
id-att-redirect-history	ID ::= { id-att 33 }
id-att-report-delivery-envelope	ID ::= { id-att 34 }
id-att-reporting-DL-name	ID ::= { id-att 35 }
id-att-reporting-MTA-certificate	ID ::= { id-att 36 }
id-att-report-origin-authentication-check	ID ::= { id-att 37 }
id-att-security-classification	ID ::= { id-att 38 }
id-att-sequence-number	ID ::= { id-att 39 }
id-att-subject-submission-identifier	ID ::= { id-att 40 }
id-att-this-recipient-name	ID ::= { id-att 41 }
<i>-- Types d'actions automatiques</i>	
id-act-auto-forward	ID ::= { id-att 0 }
id-act-auto-alert	ID ::= { id-att 1 }
END -- des identificateurs objets MS	

## ANNEXE B

(à la Recommandation X.413)

### Définition formelle du service abstrait enregistrement de messages

La présente annexe fait partie intégrante de la Recommandation.

La présente annexe, qui complète la section, définit formellement le service abstrait enregistrement de messages. Ce service utilise la notation de syntaxe abstraite 1 (ASN.1) et les macros OBJECT, PORT, ABSTRACT-BIND, ABSTRACT-UNBIND, ABSTRACT-OPERATION et ABSTRACT-ERROR définies dans la Recommandation X.407.

*Remarque* – L'utilisation des macros ABSTRACT-BIND, ABSTRACT-UNBIND, ABSTRACT-OPERATION et ABSTRACT-ERROR qui découlent des macros BIND, UNBIND, OPERATION et ERROR du service des opérations distantes (ROS) n'implique pas que les opérations abstraites et les erreurs abstraites sont dans chaque cas appelées et signalées au-delà de la limite entre systèmes ouverts. Toutefois, fréquemment, cela sera fait. La Recommandation X.419 traite des modalités de cette opération.

```

MSAbstractService { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) abstract-service(1) }
DEFINITIONS ::=

BEGIN

-- Prologue
-- Exporte tout

IMPORTS

-- Macros des services abstraits

ABSTRACT-BIND, ABSTRACT-ERROR, ABSTRACT-OPERATION, ABSTRACT-UNBIND, OBJECT, PORT
    FROM AbstractServiceNotation { joint-iso-ccitt mhs-motis(6) asdc(2) modules(0) notation(1) }

-- Accès MS
administration, delivery, submission,

-- Macro MTS

EXTENSION,

-- Types de données du service abstrait MTS
ContentLength, ContentType, Credentials, InitiatorCredentials, ORAddressAndOrDirectoryName,
ResponderCredentials, SecurityContext, SecurityError, SecurityLabel
    FROM MTSAbstractService { joint-iso-ccitt mhs-motis(6) mts(3) modules(0) mts-abstract-service(1) }

-- Objets MS
id-ot-ms, id-ot-ms-user, id-pt-retrieval
    FROM MSObjectIdentifiers { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) object-identifiers(0) }

-- Limite supérieure du service abstrait MS
ub-attributes-supported, ub-attribute-values, ub-auto-actions, ub-auto-registrations,
ub-default-registrations, ub-error-reasons, ub-information-bases, ub-messages,
ub-nested-filters, ub-per-auto-action, ub-per-entry, ub-summaries
    FROM MSUpperBounds { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) upper-bounds(4) }

-- Limite supérieure du service abstrait MTS
ub-content-types, ub-encoded-information-types, ub-labels-and-redirections
    FROM MTSUpperBounds { joint-iso-ccitt mhs-motis(6) mts(3) modules(0) upper-bounds(3) };

-- Objets abstraits MS

mS OBJECT
    PORTS { retrieval[S],
            indirectSubmission[S],
            administration[S],
            delivery[C],
            submission[C],
            administration[C] }
    ::= id-ot-ms

msUser OBJECT
    PORTS { retrieval[C],
            indirectSubmission[C],
            administration[C] }
    ::= id-ot-ms-user

-- Types d'accès
indirectSubmission PORT ::= submission

```

```

retrieval PORT
    CONSUMER INVOKES {
        Summarize,
        List,
        Fetch,
        Delete,
        Register-MS }
    SUPPLIER INVOKES {
        Alert }
    ::= id-pt-retrieval

-- Macros

AUTO-ACTION MACRO ::=
BEGIN
    TYPE NOTATION      ::=      Registration
    VALUE NOTATION     ::=      value (VALUE OBJECT IDENTIFIER)
    Registration       ::=      "REGISTRATION PARAMETER IS" type

END

-- Types de données communes relatives au modèle d'information

InformationBase ::= INTEGER {
    stored-messages (0),
    inlog           (1),
    outlog          (2) } (0..ub-information-bases)

SequenceNumber ::= INTEGER (0..ub-messages)

CreationTime ::= UTCTime

Attribute ::= SEQUENCE {
    type      AttributeType,
    values    SEQUENCE SIZE (1..ub-attribute-values) OF ANY -- DEFINED BY type -- }

AttributeType ::= OBJECT IDENTIFIER

AutoActionRegistration ::= SEQUENCE {
    type                AutoActionType,
    registration-identifier [0] INTEGER (1..ub-per-auto-action) DEFAULT 1,
    registration-parameter [1] ANY DEFINED BY type }

AutoActionType ::= OBJECT IDENTIFIER

EntryStatus ::= INTEGER {
    new           (0),
    listed        (1),
    processed     (2) }

-- Abstrait-Liaison

MSBind ::= ABSTRACT-BIND
    TO { indirectSubmission[S], retrieval[S], administration[S] }
    BIND
        ARGUMENT      MSBindArgument
        RESULT         MSBindResult
        BIND-ERROR    MSBindError

MSUnbind ::= ABSTRACT-UNBIND
    FROM { indirectSubmission[S], retrieval[S], administration[S] }

MSBindArgument ::= SET {
    initiator-name      ORAddressAndOrDirectoryName
    initiator-credentials [2] InitiatorCredentials,
    security-context    [3] IMPLICIT SecurityContext OPTIONAL,
    fetch-restrictions[4] Restrictions OPTIONAL -- default is none --,
    ms-configuration-request [5] BOOLEAN DEFAULT FALSE }

```

```

Restrictions ::= SET {
    allowed-content-types    [0] SET SIZE (1. .ub-content-types) OF OBJECT IDENTIFIER OPTIONAL
        -- default is no restriction --,
    allowed-EITs            [1] MS-EITs OPTIONAL -- default is no restriction --,
    maximum-content-length [2] ContentLength OPTIONAL -- default is no restriction -- }

MS-EITs ::= SET SIZE (1. .ub-encoded-information-types) OF MS-EIT

MS-EIT ::= OBJECT IDENTIFIER

MSBindResult ::= SET {
    responder-credentials    [2] ResponderCredentials,
    available-auto-actions    [3] SET SIZE (1. .ub-auto-actions) OF AutoActionType OPTIONAL,
    available-attribute-types [4] SET SIZE (1. .ub-attributes-supported) OF AttributeType OPTIONAL,
    alert-indication         [5] BOOLEAN DEFAULT FALSE,
    content-types-supported   [6] SET SIZE (1. .ub-content-types) OF OBJECT IDENTIFIER OPTIONAL

MSBindError ::= ENUMERATED {
    authentication-error      (0),
    unacceptable-security-context (1),
    unable-to-establish-association (2) }

-- Types de données communes pour les opérations abstraites

Range ::= CHOICE {
    sequence-number-range [0] NumberRange,
    creation-time-range   [1] TimeRange }

NumberRange ::= SEQUENCE {
    from    [0] SequenceNumber OPTIONAL -- omitted means no lower bound --,
    to      [1] SequenceNumber OPTIONAL -- omitted means no upper bound -- }

TimeRange ::= SEQUENCE {
    From    [0] CreationTime OPTIONAL -- omitted means no lower bound --,
    to      [1] CreationTime OPTIONAL -- omitted means no upper bound -- }

Filter ::= CHOICE {
    item    [0] FilterItem,
    and     [1] SET SIZE (1. .ub-nested-filters) OF Filter,
    or      [2] SET SIZE (1. .ub-nested-filters) OF Filter,
    not     [3] Filter }

FilterItem ::= CHOICE {
    equality          [0] AttributeValueAssertion,
    substrings       [1] SEQUENCE {
        type          AttributeType,
        strings       SEQUENCE SIZE (1. .ub-attribute-values) OF CHOICE {
            initial    [0] ANY -- DEFINED BY type --,
            any        [1] ANY -- DEFINED BY type --,
            final      [2] ANY -- DEFINED BY type -- } },
    greater-or-equal [2] AttributeValueAssertion,
    less-or-equal    [3] AttributeValueAssertion,
    present          [4] AttributeType,
    approximate-match [5] AttributeValueAssertion }

AttributeValueAssertion ::= SEQUENCE {
    type    AttributeType,
    value   ANY DEFINED BY type }

Selector ::= SET {
    child-entries [0] BOOLEAN DEFAULT FALSE,
    range         [1] Range OPTIONAL -- default is unbounded --,
    filter        [2] Filter OPTIONAL -- default is all entries within the specified range --,
    limit         [3] INTEGER (1. .ub-messages) OPTIONAL,
    override      [4] OverrideRestrictions OPTIONAL -- default is that any fetch-restrictions in force do
        apply -- }

```

```

OverrideRestrictions ::= BIT STRING {
    overrideContentTypesRestriction      (0),
    overrideEITsRestriction              (1),
    overrideContentLengthRestriction     (2) } (SIZE (1..ub-information-bases))

EntryInformationSelection ::= SET SIZE (0..ub-per-entry) OF AttributeSelection

AttributeSelection ::= SET {
    type      AttributeType,
    from      [0] INTEGER (1..ub-attribute-values) OPTIONAL -- used if type is multi valued --,
    count     [1] INTEGER (1..ub-attribute-values) OPTIONAL -- used if type is multi valued -- }

EntryInformation ::= SEQUENCE {
    sequence-number      SequenceNumber,
    attributes           SET SIZE (1..ub-per-entry) OF Attribute OPTIONAL }

-- Paramètre demande de retransmission pour dépôt indirect

forwarding-request EXTENSION
    SequenceNumber
    CRITICAL FOR SUBMISSION
    ::= 36

-- Opérations abstraites

Summarize ::= ABSTRACT-OPERATION
    ARGUMENT      SummarizeArgument
    RESULT        SummarizeResult
    ERRORS {
        AttributeError,
        InvalidParametersError,
        RangeError,
        SecurityError,
        SequenceNumberError,
        ServiceError }

SummarizeArgument ::= SET {
    information-base-type [0] InformationBase DEFAULT stored-messages,
    selector              [1] Selector,
    summary-requests     [2] SEQUENCE SIZE (1..ub-summaries) OF AttributeType OPTIONAL
    -- absent if no summaries are requested -- }

SummarizeResult ::= SET {
    next      [0] SequenceNumber OPTIONAL,
    count     [1] INTEGER (0..ub-messages) -- of the entries selected --,
    span      [2] Span OPTIONAL -- of the entries selected, omitted if count is zero --,
    summaries [3] SEQUENCE SIZE (1..ub-summaries) OF Summary OPTIONAL }

Span ::= SEQUENCE {
    lowest [0] SequenceNumber,
    highest [1] SequenceNumber }

Summary ::= SET {
    absent [0] INTEGER (1..ub-messages) OPTIONAL -- count of entries where the attribute is absent --,
    present [1] SET SIZE (1..ub-attribute-values) OF -- one for each attribute value present --
        SEQUENCE {
            type      AttributeType,
            value     ANY DEFINED BY type,
            count     INTEGER (1..ub-messages) } OPTIONAL }

```

```

List ::= ABSTRACT-OPERATION
    ARGUMENT      ListArgument
    RESULT        ListResult
    ERRORS {
        AttributeError,
        InvalidParametersError,
        RangeError,
        SecurityError,
        SequenceNumberError,
        ServiceError }

ListArgument ::= SET {
    information-base-type [0] InformationBase DEFAULT stored-messages,
    selector              [1] Selector,
    requested-attributes [3] EntryInformationSelection OPTIONAL }

ListResult ::= SET {
    next          [0] SequenceNumber OPTIONAL,
    requested     [1] SEQUENCE SIZE (1. .ub-messages) OF EntryInformation OPTIONAL -- omitted if
                 none found -- }

--

Fetch ::= ABSTRACT-OPERATION
    ARGUMENT      FetchArgument
    RESULT        FetchResult
    ERRORS {
        AttributeError,
        FetchRestrictionError,
        InvalidParametersError,
        RangeError,
        SecurityError,
        SequenceNumberError,
        ServiceError }

FetchArgument ::= SET {
    information-base-type [0] InformationBase DEFAULT stored-messages,
    item                 CHOICE {
        search           [1] Selector,
        precise          [2] SequenceNumber },
    requested-attributes [3] EntryInformationSelection OPTIONAL }

FetchResult ::= SET {
    entry-information [0] EntryInformation OPTIONAL -- if an entry was selected --,
    list              [1] SEQUENCE SIZE (1. .ub-messages) OF SequenceNumber OPTIONAL,
    next              [2] SequenceNumber OPTIONAL }

--

Delete ::= ABSTRACT-OPERATION
    ARGUMENT      DeleteArgument
    RESULT        DeleteResult
    ERRORS {
        DeleteError,
        InvalidParametersError,
        RangeError,
        SecurityError,
        SequenceNumberError,
        ServiceError }

DeleteArgument ::= SET {
    information-base-type [0] InformationBase DEFAULT stored-messages,
    items                 CHOICE {
        selector          [1] Selector
        sequence-numbers [2] SET SIZE (1. .ub-messages) OF SequenceNumber } }

```

DeleteResult ::= NULL

--

Register-MS ::= ABSTRACT-OPERATION  
ARGUMENT Register-MSArgument  
RESULT Register-MSResult  
ERRORS {  
AttributeError,  
AutoActionRequestError,  
InvalidParametersError,  
SecurityError,  
ServiceError }

Register-MSArgument ::= SET {  
auto-action-registrations [0] SET SIZE (1..ub-auto-registrations) OF AutoActionRegistration  
OPTIONAL,  
auto-action-deregistrations [1] SET SIZE (1..ub-auto-registrations) OF AutoActionDeregistration  
OPTIONAL,  
list-attribute-defaults [2] SET SIZE (1..ub-default-registrations) OF AttributeType  
OPTIONAL,  
fetch-attribute-defaults [3] SET SIZE (1..ub-default-registrations) OF AttributeType OPTIONAL,  
change-credentials [4] SEQUENCE {  
old-credentials [0] IMPLICIT Credentials,  
new-credentials [1] IMPLICIT Credentials } OPTIONAL  
-- same CHOICE as for old-credentials --,  
user-security-labels [5] SET SIZE (1..ub-labels-and-redirections) OF SecurityLabel  
OPTIONAL }

AutoActionDeregistration ::= AutoActionRegistration  
(WITH COMPONENTS { .., registration-parameter ABSENT } )

Register-MSResult ::= NULL

--

Alert ::= ABSTRACT-OPERATION  
ARGUMENT AlertArgument  
RESULT AlertResult  
ERRORS {  
SecurityError }

AlertArgument ::= SET {  
alert-registration-identifier [0] INTEGER (1..ub-auto-actions),  
new-entry [2] EntryInformation OPTIONAL }

AlertResult ::= NULL

-- Erreurs abstraites

AttributeError ::= ABSTRACT-ERROR  
PARAMETER SET {  
problems [0] SET SIZE (1..ub-per-entry) OF SET {  
problem [0] AttributeProblem,  
type [1] AttributeType,  
value [2] ANY DEFINED BY type OPTIONAL } }

AttributeProblem ::= INTEGER {  
invalid-attribute-value (0),  
unavailable-attribute-type (1),  
inappropriate-matching (2),  
attribute-type-not-subscribed (3),  
inappropriate-for-operation (4) } (0..ub-error-reasons)

--

```

AutoActionRequestError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problems [0]          SET SIZE (1. .ub-auto-registrations) OF SET {
            problem          [0] AutoActionRequestProblem,
            type              [1] AutoActionType } }

AutoActionRequestProblem ::= INTEGER {
    unavailable-auto-action-type (0),
    auto-action-type-not-subscribed (1) } (0. .ub-error-reasons)

--

DeleteError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problems [0]          SET SIZE (1. .ub-messages) OF SET {
            problem          [0] DeleteProblem,
            sequence-number  [1] SequenceNumber } }

DeleteProblem ::= INTEGER {
    child-entry-specified (0),
    delete-restriction-problem (1) } (0. .ub-error-reasons)

--

FetchRestrictionError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problems [0]          SET SIZE (1. .ub-default-registrations) OF SET {
            problem          [3] FetchRestrictionProblem,
            restriction      CHOICE {
                content-type [1] ContentType,
                eit           [2] MS-EITs,
                content-length [3] ContentLength } } }

FetchRestrictionProblem ::= INTEGER {
    content-type-problem (1),
    eit-problem (2),
    content-length-problem (3) } (0. .ub-error-reasons)

--

InvalidParametersError ::= ABSTRACT-ERROR
    PARAMETER NULL

--

RangeError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problem [0] RangeProblem }

RangeProblem ::= INTEGER {
    reversed (0) } (0. .ub-error-reasons)

--

SequenceNumberError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problems [1]          SET SIZE (1. .ub-messages) OF SET {
            problem          [0] SequenceNumberProblem,
            sequence-number  [1] SequenceNumber } }

SequenceNumberProblem ::= INTEGER {
    no-such-entry (0) } (0. .ub-error-reasons)

--

ServiceError ::= ABSTRACT-ERROR
    PARAMETER SET {
        problem [0] ServiceProblem }

```

```

ServiceProblem ::= INTEGER {
    busy (0),
    unavailable (1),
    unwilling-to-perform (2) } (0..ub-error-reasons)
END -- du service abstrait MS

```

## ANNEXE C

(à la Recommandation X.413)

### Définition formelle des types d'attributs généraux

La présente annexe fait partie intégrante de la Recommandation.

La présente annexe, qui complète la section 3, définit formellement les types d'attributs généraux applicables à toutes les formes de messagerie et non à une seule. Elle utilise la notation de syntaxe abstraite n<sup>o</sup> 1 (ASN.1) et la macro ATTRIBUTE.

```

MSGeneralAttributeTypes { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) general-attribute-types(2) }
DEFINITIONS ::=
BEGIN
-- Prologue
-- Exporte tout
IMPORTS
-- Identificadores de objeto de tipos de atributos generales
id-att-child-sequence-numbers, id-att-content, id-att-content-confidentiality-algorithm-identifier,
id-att-content-correlator, id-att-content-identifier, id-att-content-integrity-check, id-att-content-length,
id-att-content-returned, id-att-content-type, id-att-conversion-with-loss-prohibited, id-att-converted-EITs,
id-att-creation-time, id-att-delivered-EITs, id-att-delivery-flags, id-att-dl-expansion-history,
id-att-entry-status, id-att-entry-type, id-intended-recipient-name, id-att-message-delivery-envelope,
id-att-message-delivery-identifier, id-att-message-delivery-time, id-att-message-origin-authentication-check,
id-att-message-security-label, id-att-message-submission-time, id-att-message-token, id-att-original-EITs,
id-att-originator-certificate, id-att-originator-name, id-att-other-recipient-names,
id-att-parent-sequence-number, id-att-priority, id-att-proof-of-delivery-request, id-att-redirect-history,
id-att-report-delivery-envelope, id-att-reporting-DL-name, id-att-reporting-MTA-certificate,
id-att-report-origin-authentication-check, id-att-sequence-number, id-att-subject-submission-identifier,
id-att-this-recipient-name
    FROM MSObjectIdentifiers { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) object-identifiers(0) }
-- Macros attribut
ATTRIBUTE, ATTRIBUTE-SYNTAX
    FROM InformationFramework { joint-iso-ccitt ds(5) modules(1) informationFramework(1) }
-- Types de données du service abstrait MS
CreationTime, EntryStatus, MS-EIT, SequenceNumber
    FROM MSAbstractService { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) abstract-service(1) }
-- Types de données du service authentification
AlgorithmIdentifier
    FROM AuthenticationFramework { joint-iso-ccitt ds(5) modules(1) authentication-framework(7) }
-- Types de données du service abstrait MTS
Content, ContentCorrelator, ContentIdentifier, ContentIntegrityCheck, ContentLength,
ConversionWithLossProhibited, DeliveryFlags, DLExpansionHistory, MessageDeliveryEnvelope,
MessageDeliveryIdentifier, MessageDeliveryTime, MessageOriginAuthenticationCheck,
MessageSecurityLabel, MessageSubmissionTime, MessageToken, OriginatorCertificate, ORName,
PerRecipientReportDeliveryFields, Priority, ProofOfDeliveryRequest, RedirectHistory,

```

```

ReportDeliveryEnvelope, ReportingDLName, ReportingMTACertificate,
ReportOriginAuthenticationCheck, SecurityClassification, subjectSubmissionIdentifier
    FROM MTSAbstractService { joint-iso-ccitt mhs-motis(6) mts(3) modules(0)
    mts-abstract-service(1) }

-- Limite supérieure du service abstrait MS
ub-entry-types
    FROM MSUpperBounds { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) upper-bounds(4) };

-- Types d'attributs

ms-child-sequence-numbers ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX SequenceNumber
    MULTI VALUE
    ::= id-att-child-sequence-numbers

ms-content ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX Content
    SINGLE VALUE
    ::= id-att-content

mt-content-confidentiality-algorithm-identifier ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX AlgorithmIdentifier
    SINGLE VALUE
    ::= id-att-content-confidentiality-algorithm-identifier

mt-content-correlator ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ContentCorrelator
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-content-correlator

mt-content-identifier ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ContentIdentifier
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-content-identifier

mt-content-integrity-check ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ContentIntegrityCheck
    SINGLE VALUE
    ::= id-att-content-integrity-check

ms-content-length ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ContentLength
    MATCHES FOR ORDERING
    SINGLE VALUE
    ::= id-att-content-length

ms-content-returned ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX BOOLEAN
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-content-returned

mt-content-type ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX OBJECT IDENTIFIER
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-content-type

mt-conversion-with-loss-prohibited ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ConversionWithLossProhibited
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-conversion-with-loss-prohibited

```

ms-converted-EITs ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MS-EIT  
 MATCHES FOR EQUALITY  
 MULTI VALUE  
 ::= id-att-converted-EITs

ms-creation-time ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX CreationTime  
 MATCHES FOR EQUALITY ORDERING  
 SINGLE VALUE  
 ::= id-att-creation-time

ms-delivered-EITs ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MS-EIT  
 MATCHES FOR EQUALITY  
 MULTI VALUE  
 ::= id-att-delivered-EITs

mt-delivery-flags ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX DeliveryFlags  
 MATCHES FOR EQUALITY  
 SINGLE VALUE  
 ::= id-att-delivery-flags

mt-dl-expansion-history ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX DLExpansionHistory  
 MULTI VALUE  
 ::= id-att-dl-expansion-history

ms-entry-status ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX EntryStatus  
 MATCHES FOR EQUALITY  
 SINGLE VALUE  
 ::= id-att-entry-status

ms-entry-type ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX EntryType  
 MATCHES FOR EQUALITY  
 SINGLE VALUE  
 ::= id-att-entry-type

EntryType ::= INTEGER {  
 delivered-message (0),  
 delivered-report (1),  
 returned-content (2) (0..ub-entry-types) }

mt-intended-recipient-name ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX ORName  
 MATCHES FOR EQUALITY  
 SINGLE VALUE  
 ::= id-att-intended-recipient-name

mt-message-delivery-envelope ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageDeliveryEnvelope  
 SINGLE VALUE  
 ::= id-att-message-delivery-envelope

mt-message-delivery-identifier ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageDeliveryIdentifier  
 SINGLE VALUE  
 ::= id-att-message-delivery-identifier

mt-message-delivery-time ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageDeliveryTime  
 MATCHES FOR EQUALITY ORDERING  
 SINGLE VALUE  
 ::= id-att-message-delivery-time

mt-message-origin-authentication-check ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageOriginAuthenticationCheck  
 SINGLE VALUE  
 ::= id-att-message-origin-authentication-check

mt-message-security-label ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageSecurityLabel  
 SINGLE VALUE  
 ::= id-att-message-security-label

mt-message-submission-time ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageSubmissionTime  
 MATCHES FOR EQUALITY ORDERING  
 SINGLE VALUE  
 ::= id-att-message-submission-time

mt-message-token ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MessageToken  
 SINGLE VALUE  
 ::= id-att-message-token

ms-original-EITs ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX MS-EIT  
 MATCHES FOR EQUALITY  
 MULTI VALUE  
 ::= id-att-original-EITs

mt-originator-certificate ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX OriginatorCertificate  
 SINGLE VALUE  
 ::= id-att-originator-certificate

mt-originator-name ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX ORName  
 MATCHES FOR EQUALITY  
 SINGLE VALUE  
 ::= id-att-originator-name

mt-other-recipient-names ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX ORName  
 MATCHES FOR EQUALITY  
 MULTI VALUE  
 ::= id-att-other-recipient-names

ms-parent-sequence-number ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX SequenceNumber  
 MATCHES FOR EQUALITY ORDERING  
 SINGLE VALUE  
 ::= id-att-parent-sequence-number

mt-per-recipient-report-delivery-fields ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX PerRecipientReportDeliveryFields  
 MULTI VALUE  
 ::= id-att-per-recipient-report-delivery-fields

mt-priority ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX Priority  
 MATCHES FOR EQUALITY  
 SINGLE VALUE  
 ::= id-att-priority

mt-proof-of-delivery-request ATTRIBUTE  
 WITH ATTRIBUTE-SYNTAX ProofOfDeliveryRequest  
 SINGLE VALUE  
 ::= id-att-proof-of-delivery-request

```

mt-redirectation-history ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX RedirectionHistory
    MULTI VALUE
    ::= id-att-redirectation-history

mt-report-delivery-envelope ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ReportDeliveryEnvelope
    SINGLE VALUE
    ::= id-att-report-delivery-envelope

mt-reporting-DL-name ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ReportingDLName
    SINGLE VALUE
    ::= id-att-reporting-DL-name

mt-reporting-MTA-certificate ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ReportingMTACertificate
    SINGLE VALUE
    ::= id-att-reporting-MTA-certificate

mt-report-origin-authentication-check ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ReportOriginAuthenticationCheck
    SINGLE VALUE
    ::= id-att-report-origin-authentication-check

mt-security-classification ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX SecurityClassification
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-security-classification

ms-sequence-number ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX SequenceNumber
    MATCHES FOR EQUALITY ORDERING
    SINGLE VALUE
    ::= id-att-sequence-number

mt-subject-submission-identifier ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX SubjectSubmissionIdentifier
    SINGLE VALUE
    ::= id-att-subject-submission-identifier

mt-this-recipient-name ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX ORName
    MATCHES FOR EQUALITY
    SINGLE VALUE
    ::= id-att-this-recipient-name

END -- des types d'attributs généraux du MS

```

## ANNEXE D

(à la Recommandation X.413)

### **Définition formelle des types d'actions automatiques générales**

La présente annexe fait partie intégrante de la Recommandation.

La présente annexe, qui complète la section 3, définit formellement les types d'actions automatiques générales applicables à toutes les formes de messagerie et non à une seule. Elle utilise la notation de syntaxe abstraite n° 1 (ASN.1) et la macro AUTO-ACTION.

```

MSGeneralAutoActionTypes { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) general-auto-action-types(3) }
DEFINITIONS ::=
BEGIN
-- Prologue
EXPORTS
    -- Types d'actions automatiques g n rales
    auto-forward, auto-alert;
IMPORTS
    -- Identificateurs d'objets des types d'actions automatiques g n rales
    id-act-auto-forward, id-act-auto-alert
        FROM MSObjectIdentifiers { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) object-identifiers(0) }
    -- Macro action automatique
    AUTO-ACTION,
    -- Types de donn es du service abstrait MS
    Content, Filter, EntryInformationSelection
        FROM MSAbstractService { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) abstract-service(1) }
    -- Types de donn es du service abstrait MTS
    ContentIdentifier, DeferredDeliveryTime, ExplicitConversion, OriginatorName, OriginatorReportRequest,
    PerMessageIndicators, PerMessageSubmissionExtensions, PerRecipientMessageSubmissionExtensions,
    Priority, RecipientName
        FROM MTSAbstractService { joint-iso-ccitt mhs-motis(6) mts(3) modules(0) mts-abstract-service(1) }
    -- Limite sup rieure du service abstrait MS
    ub-alert-addresses
        FROM MSUpperBounds { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) upper-bounds(4) };
    -- Types d'actions
    auto-forward AUTO-ACTION
        REGISTRATION PARAMETER IS AutoForwardRegistrationParameter
        ::= id-act-auto-forward
AutoForwardRegistrationParameter ::= SET {
    filter [0] Filter OPTIONAL,
    auto-forward-arguments [1] AutoForwardArguments,
    delete-after-auto-forwarding [2] BOOLEAN DEFAULT TRUE,
    other-parameters [3] OCTET STRING OPTIONAL }
AutoForwardArguments ::= SET {
    COMPONENTS OF PerMessageAutoForwardFields,
    per-recipient-fields [1] IMPLICIT SEQUENCE SIZE (1..ub-recipients) OF PerRecipient-
        AutoForwardFields }
PerMessageAutoForwardFields ::= SET {
    originator-name OriginatorName,
    content-identifier ContentIdentifier OPTIONAL,
    priority Priority DEFAULT normal,
    per-message-indicators PerMessageIndicators DEFAULT { },
    deferred-delivery-time [0] IMPLICIT DeferredDeliveryTime OPTIONAL,
    extensions [2] IMPLICIT PerMessageSubmissionExtensions DEFAULT { } }
PerRecipientAutoForwardFields ::= SET {
    recipient-name RecipientName,
    originator-report-request [0] IMPLICIT OriginatorReportRequest,
    explicit-conversion [1] IMPLICIT ExplicitConversion OPTIONAL,
    extensions [2] IMPLICIT PerRecipientMessageSubmissionExtensions DEFAULT { } }
auto-alert AUTO-ACTION
    REGISTRATION PARAMETER IS AutoAlertRegistrationParameter
    ::= id-act-auto-alert

```

```

AutoAlertRegistrationParameter ::= SET {
    filter [0] Filter OPTIONAL,
    alert-addresses [1] SEQUENCE SIZE (1..ub-alert-addresses) OF AlertAddress OPTIONAL,
    requested-attributes [2] EntryInformationSelection OPTIONAL }

AlertAddress ::= SEQUENCE {
    address EXTERNAL,
    alert-qualifier OCTET STRING OPTIONAL }

END -- des types d'actions automatiques g n rales du MS

```

## ANNEXE E

(  la Recommandation X.413)

### D finition formelle des limites sup rieures du param tre enregistrement de messages

La pr sente annexe fait partie int grante de la Recommandation.

La pr sente annexe d finit,   titre de r f rence, les limites sup rieures des divers types de donn es de longueur variable dont les syntaxes abstraites sont d finies dans les modules ASN.1 figurant dans le corps de la pr sente Recommandation.

```

MSUpperBounds { joint-iso-ccitt mhs-motis(6) ms(4) modules(0) upper-bounds(4) }
DEFINITIONS IMPLICIT TAGS ::=

```

```

BEGIN

```

```

-- Prologue

```

```

-- Exporte tout

```

```

IMPORTS -- rien --;

```

```

-- Limites sup rieures

```

```

ub-alert-addresses    INTEGER ::= 16

```

```

ub-attribute-values   INTEGER ::= 32767    -- (215 - 1) the largest integer representable in 16 bits --

```

```

ub-attributes-supported INTEGER ::= 1024

```

```

ub-auto-actions       INTEGER ::= 16

```

```

ub-auto-registrations INTEGER ::= 1024

```

```

ub-default-registrations INTEGER ::= 1024

```

```

ub-entry-types        INTEGER ::= 16

```

```

ub-error-reasons      INTEGER ::= 16

```

```

ub-information-bases  INTEGER ::= 16

```

```

ub-messages           INTEGER ::= 2147483647 -- (231 - 1) the largest integer representable in 32 bits --

```

```

ub-nested-filters     INTEGER ::= 32

```

```

ub-per-auto-action    INTEGER ::= 32767    -- (215 - 1) the largest integer representable in 16 bits --

```

```

ub-per-entry          INTEGER ::= 1024

```

```

ub-summaries          INTEGER ::= 16

```

```

END -- des limites sup rieures du MS

```

ANNEXE F

**Exemple de l'opération abstraite résumé**

La présente annexe ne fait pas partie de la Recommandation.

La présente annexe donne un exemple d'utilisation de l'opération abstraite résumé.

F.1 *Entrées du MS dans l'exemple choisi*

Prenons, par exemple, un MS contenant les entrées suivantes, une par ligne. Les colonnes montrent les valeurs des types d'attributs indiqués. «-» signifie que l'attribut ne figure pas dans l'entrée.

TABLEAU F-1/X.413

**Messages enregistrés dans l'exemple choisi**

Numéro d'ordre	Type d'entrée	Etat de l'entrée	Priorité
3	message	listé	urgent
5	message	listé	faible
8	rapport	listé	-
10	message	listé	normal
15	rapport	nouveau	-
18	message	nouveau	normal
20	message	nouveau	urgent
22	message	nouveau	normal
23	message	nouveau	normal

*Remarque* – Même si le paramètre priorité est omis dans l'enveloppe de remise d'un message et remplacé par «normal», l'attribut correspondant est présent, sa valeur étant mise sur valeur par défaut.

F.2 *Exemple de demande de résumé*

Supposons que l'on veuille résumer toutes les «nouvelles» entrées à l'aide de l'attribut priorité. Le résultat demandé est la liste des comptes ci-après. Les chiffres entre parenthèses sont les numéros d'ordre des messages intervenant dans ce compte (voir le tableau F-2/X.413).

TABLEAU F-2/X.413

**Résultat escompté du résumé Listage**

Priorité	Compte
-	1 (15)
urgent	1 (20)
normal	3 (18,22,23)
faible	0

Les composants de l'argument résumé devraient avoir la valeur suivante:

sélecteur:

    filtre:                   état de l'entrée = nouveau

demandes-résumé:        type d'attribut = Priorité

Les composants du résultat résumé pourraient être les suivants:

compte:                   5

fourchette:

    limite inférieure:   15

    limite supérieure:  23

résumés:

    { absent:             1

      présent:           { valeur = normale, compte = 3 }

                          { valeur = urgent, compte = 1 } }

## ANNEXE G

### **Différences entre la Recommandation X.413 et le texte de la norme 10021-5 de l'ISO/CEI**

La présente annexe ne fait pas partie de la Recommandation.

Il n'y a que deux différences connues entre la Recommandation X.413 et le MOTIS pour la norme 10021-5 dans l'ISO/CEI.

- 1) Le texte du CCITT contient une restriction au § 7.1; cette restriction porte sur le fait qu'une seule association abstraite peut exister à un moment donné quelconque entre le MS et l'utilisateur du MS. Cette restriction n'est *pas* incluse dans le texte ISO/CEI.
- 2) Les parties de la notation ASN.1 exprimant les limites supérieures font l'objet de l'annexe E. Elles ne sont pas considérées comme faisant partie de la norme MOTIS, mais elles font partie intégrante de la Recommandation X.413.

A l'ISO, ce niveau de fonctionnalité relève de la compétence du Groupe spécial sur la normalisation fonctionnelle, qui publie des profils normalisés au niveau international (ISP), contenant, par exemple, les limites supérieures des éléments de protocole.



## SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série B	Moyens d'expression: définitions, symboles, classification
Série C	Statistiques générales des télécommunications
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	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	RGT et maintenance des réseaux: systèmes de transmission, de télégraphie, de télécopie, circuits téléphoniques et circuits loués internationaux
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
<b>Série X</b>	<b>Réseaux de données et communication entre systèmes ouverts</b>
Série Y	Infrastructure mondiale de l'information et protocole Internet
Série Z	Langages et aspects informatiques généraux des systèmes de télécommunication