



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

CCITT

COMITÉ CONSULTATIF
INTERNATIONAL
TÉLÉGRAPHIQUE ET TÉLÉPHONIQUE

X.413

(09/92)

**RÉSEAUX DE COMMUNICATIONS
DE DONNÉES**

**SYSTÈMES DE MESSAGERIE –
ENREGISTREMENT DE MESSAGES:
DÉFINITION DU SERVICE ABSTRAIT**



Recommandation X.413

AVANT-PROPOS

Le CCITT (Comité consultatif international télégraphique et téléphonique) est un organe permanent de l'Union internationale des télécommunications (UIT). Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée plénière du CCITT, qui se réunit tous les quatre ans, détermine les thèmes d'étude et approuve les Recommandations rédigées par ses Commissions d'études. Entre les Assemblées plénières, l'approbation des Recommandations par les membres du CCITT s'effectue selon la procédure définie dans la Résolution n° 2 du CCITT (Melbourne, 1988).

La Recommandation révisée X.413, élaborée par la Commission d'études VII, a été approuvée le 10 septembre 1992 selon la procédure définie dans la Résolution n° 2.

REMARQUE

Dans cette Recommandation, le terme «Administration» désigne indifféremment une administration de télécommunication ou une exploitation privée reconnue.

© UIT 1993

Droits de reproduction réservés. Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'UIT.

TABLE DES MATIÈRES

	<i>Page</i>
SECTION 1 – CONSIDÉRATIONS GÉNÉRALES	1
1 <i>Champ d'application</i>	1
2 <i>Références normatives</i>	1
2.1 Références relatives au modèle de référence	1
2.2 Références relatives à la présentation	2
2.3 Références relatives aux opérations distantes	2
2.4 Références relatives à l'annuaire	2
2.5 Références relatives à la messagerie	2
3 <i>Définitions</i>	3
3.1 Définitions communes pour le MHS.....	3
3.2 Définitions concernant l'enregistrement des messages	3
4 <i>Abréviations</i>	10
5 <i>Conventions</i>	10
5.1 Conventions applicables aux services abstraits.....	10
5.2 Conventions applicables aux types d'attributs utilisés dans le tableau 1/X.413 du § 11.....	11
5.3 Conventions applicables aux types d'attributs utilisés dans le tableau 2/X.413 du § 11.....	11
5.4 Conventions applicables aux polices de caractères pour les textes en général	12
5.5 Conventions applicables aux polices de caractères pour les définitions en ASN.1	12
5.6 Règles applicables aux définitions en ASN.1	12
SECTION 2 – DÉFINITION DU SERVICE ABSTRAIT D'ENREGISTREMENT DE MESSAGES	12
6 <i>Modèle d'enregistrement de messages</i>	12
6.1 Objet enregistrement de messages (Message Store object).....	13
6.2 Accès pour enregistrement de messages (Message Store ports)	13
6.2.1 Accès de recherche (Retrieval Port).....	14
6.2.2 Accès de dépôt indirect (Indirect-submission Port)	14
6.2.3 Accès d'administration (Administration Port).....	14
6.3 Modèle d'information.....	14
6.3.1 Bases d'information (Information-bases)	15
6.3.2 Entrées (Entries).....	15
6.3.3 Attributs (Attributes).....	15
6.3.4 Entrées principales, entrées maîtresses et entrées secondaires (Main-entries, parent-entries and child-entries).....	17
6.4 Messages enregistrés (Stored messages).....	18
6.5 Actions automatiques (Auto-actions).....	19
6.5.1 Introduction.....	19
6.5.2 Type d'action automatique (auto-action-type)	19
6.5.3 Paramètre d'enregistrement d'action automatique (auto-action-registration)	20
6.5.4 Définition du type d'action automatique (auto-action-type) et macro AUTO-ACTION ..	20
6.6 Retransmission de messages	20
7 <i>Opérations de rattachement abstrait (abstract-bind) et de détachement abstrait (abstract-unbind)</i>	21
7.1 Opération de rattachement abstrait (Abstract-bind-operation).....	21
7.1.1 Argument de rattachement abstrait (Abstract-bind-argument)	21
7.1.2 Résultat du rattachement abstrait (Abstract-bind-result).....	23
7.1.3 Erreurs de rattachement abstrait (Abstract-bind-errors).....	23
7.2 Opération de détachement abstrait (Abstract-unbind-operation)	24
8 <i>Opérations abstraites (abstract-operations)</i>	24
8.1 Types de données communes utilisés dans les opérations abstraites	24
8.1.1 Etendue (Range).....	25
8.1.2 Filtres (Filters)	25
8.1.3 Sélecteur (selector).....	27

8.1.4	Sélection d'information d'entrée (Entry-information-selection).....	28
8.1.5	Information d'entrée (Entry-information)	29
8.2	Opération abstraite résumé (Summarize abstract-operation)	29
8.2.1	Argument de résumé (Summarize-argument)	30
8.2.2	Résultat de résumé (Summarize-result)	30
8.2.3	Erreurs abstraites de résumé (Summarize abstract-errors).....	31
8.3	Opération abstraite de listage (List abstract-operation)	31
8.3.1	Argument de listage (List-argument)	31
8.3.2	Résultat de listage (List-result)	32
8.3.3	Erreurs abstraites de listage (List abstract-errors).....	32
8.4	Opération abstraite d'extraction (Fetch abstract-operation)	32
8.4.1	Argument d'extraction (Fetch-argument).....	32
8.4.2	Résultat d'extraction (Fetch-result)	33
8.4.3	Erreurs abstraites d'extraction (Fetch abstract-errors).....	33
8.5	Opération abstraite de suppression (Delete abstract-operation).....	33
8.5.1	Argument de suppression (Delete-argument)	34
8.5.2	Résultat de suppression (Delete-result).....	34
8.5.3	Erreurs abstraites de suppression (Delete abstract-errors).....	34
8.6	Opération abstraite d'enregistrement MS (Register-MS abstract-operation)	34
8.6.1	Argument d'enregistrement MS (Register-MS-argument)	35
8.6.2	Résultat d'enregistrement MS (Register-MS-result)	36
8.6.3	Erreurs abstraites d'enregistrement MS (Register-MS abstract-errors).....	36
8.7	Opération abstraite d'avertissement (Alert abstract-operation).....	36
8.7.1	Argument d'avertissement (Alert-argument).....	37
8.7.2	Résultat d'avertissement (Alert-result).....	37
8.7.3	Erreurs abstraites d'avertissement (Alert abstract-errors)	37
9	<i>Erreurs abstraites (Abstract-errors)</i>	37
9.1	Priorité des erreurs	37
9.2	Erreur d'attribut (Attribute-error).....	37
9.3	Erreur de demande d'action automatique (Auto-action-request-error)	38
9.4	Erreur de suppression (Delete-error).....	39
9.5	Erreur de restriction d'extraction (Fetch-restriction-error).....	39
9.6	Erreur de paramètres non valides (Invalid-parameters-error)	40
9.7	Erreur d'étendue (Range-error)	40
9.8	Erreur de sécurité (Security-error)	40
9.9	Erreur de numéro d'ordre (Sequence-number-error).....	40
9.10	Erreur de service (Service-error).....	41
SECTION 3 – TYPES D'ATTRIBUTS GÉNÉRAUX ET TYPES D'ACTIONS AUTOMATIQUES GÉNÉRALES.....		42
10	<i>Présentation générale</i>	42
11	<i>Types d'attributs généraux (General-attribute-types)</i>	42
11.1	Présentation générale des types d'attributs généraux	42
11.2	Description des types d'attributs généraux.....	42
11.2.1	Numéros d'ordre secondaire (Child-sequence-numbers)	44
11.2.2	Contenu (Content).....	44
11.2.3	Identificateur d'algorithme de confidentialité du contenu (Content-confidentiality-algorithm-identifier).....	44
11.2.4	Corrélateur de contenu (Content-correlator).....	44
11.2.5	Identificateur de contenu (Content-identifier)	44
11.2.6	Vérification d'intégrité du contenu (Content-integrity-check)	44
11.2.7	Longueur du contenu (Content-length).....	45
11.2.8	Contenu renvoyé (Content-returned)	45
11.2.9	Type de contenu (Content-type)	45
11.2.10	Conversion avec perte interdite (Conversion-with-loss-prohibited).....	45

11.2.11	Types d'information codée convertis (Converted EITs)	45
11.2.12	Date-heure de création (Creation-time)	46
11.2.13	Types d'information codée remis (Delivered EITs)	46
11.2.14	Indicateurs de remise (Delivery-flags).....	46
11.2.15	Chronologie de l'allongement de la liste de distribution (DL-expansion-history)	46
11.2.16	Etat de l'entrée (Entry-status).....	46
11.2.17	Type d'entrée (Entry-type)	47
11.2.18	Nom du destinataire prévu (Intended-recipient-name)	47
11.2.19	Enveloppe de remise de message (Message-delivery-envelope)	47
11.2.20	Identificateur de remise de message (Message-delivery-identifiant).....	47
11.2.21	Date et heure de remise de message (Message-delivery-time)	47
11.2.22	Vérification d'authentification d'origine de message (Message-origin-authentication-check).....	48
11.2.23	Etiquette de sécurité de message (Message-security-label)	48
11.2.24	Date et heure de dépôt du message (Message-submission-time).....	48
11.2.25	Jeton de message (Message-token).....	48
11.2.26	Types d'information codée d'origine (Original-EITs).....	48
11.2.27	Certificat de l'expéditeur (Originator-certificate).....	48
11.2.28	Nom de l'expéditeur (Originator-name).....	49
11.2.29	Noms des autres destinataires (other-recipient-names).....	49
11.2.30	Numéro d'ordre principal (Parent-sequence-number).....	49
11.2.31	Champs de remise de rapports par destinataire (Per-recipient-report-delivery-fields)	49
11.2.32	Priorité (Priority).....	49
11.2.33	Demande de preuve de remise (Proof-of-delivery-request).....	49
11.2.34	Chronologie du réacheminement (Redirection-history).....	50
11.2.35	Enveloppe de remise de rapport (Report-delivery-envelope)	50
11.2.36	Indication du nom de la liste de distribution (Reporting-DL-name).....	50
11.2.37	Indication de certificat MTA (Reporting-MTA-certificate).....	50
11.2.38	Vérification d'authentification de l'origine du rapport (Report-origin-authentication-check).....	50
11.2.39	Classification de sécurité (Security-classification)	50
11.2.40	Numéro d'ordre (Sequence-number).....	51
11.2.41	Identificateur de dépôt du sujet (Subject-submission-identifiant).....	51
11.2.42	Nom de ce destinataire (This-recipient-name).....	51
11.3	Création des attributs généraux	51
11.4	Abonnement aux types d'attributs	51
12	<i>Types d'actions automatiques générales (General-auto-action-types)</i>	54
12.1	Retransmission automatique (Auto-forward).....	54
12.2	Avertissement automatique (Auto-alert).....	56
SECTION 4 – PROCÉDURES APPLICABLES À L'ENREGISTREMENT DE MESSAGES ET À LA MISE EN ŒUVRE DES ACCÈS		58
13	<i>Présentation générale</i>	58
14	<i>Utilisation du service abstrait du système de transfert de messages</i>	58
14.1	Utilisation des services abstraits de l'accès pour remise	58
14.1.1	Fonctionnement de l'opération abstraite de remise de messages (MessageDelivery abstract-operation)	58
14.1.2	Fonctionnement de l'opération abstraite de remise de rapports (ReportDelivery abstract-operation)	59
14.1.3	Invocation de l'opération abstraite de commande de remise (DeliveryControl abstract-operation).....	59
14.1.4	Règles de création applicables aux attributs généraux.....	59
14.2	Utilisation des services abstraits d'accès pour dépôt.....	60
14.2.1	Invocation de l'opération abstraite de dépôt de messages (MessageSubmission abstract-operation).....	60
14.2.2	Invocation de l'opération abstraite de dépôt d'essai (ProbeSubmission abstract-operation).....	60

14.2.3	Invocation de l'opération abstraite d'annulation de remise différée (CancelDeferredDelivery abstract-operation).....	61
14.2.4	Fonctionnement de l'opération abstraite de commande de dépôt (SubmissionControl abstract-operation)	61
14.3	Utilisation des services abstraits d'accès d'administration	61
14.3.1	Invocation de l'opération abstraite d'enregistrement (Register abstract-operation)	61
14.3.2	Invocation de l'opération abstraite de modification des pouvoirs (ChangeCredentials abstract-operation)	62
14.3.3	Fonctionnement de l'opération abstraite de modification des pouvoirs (ChangeCredentials abstract-operation).....	62
15	<i>Fourniture du service abstrait d'enregistrement de messages</i>	62
15.1	Fourniture des services abstraits d'accès de recherche.....	62
15.1.1	Fonctionnement de l'opération abstraite de résumé (Summarize abstract-operation).....	62
15.1.2	Fonctionnement de l'opération abstraite de listage (List abstract-operation).....	63
15.1.3	Fonctionnement de l'opération abstraite extraction (Fetch abstract-operation)	63
15.1.4	Fonctionnement de l'opération abstraite de suppression (Delete abstract-operation)	64
15.1.5	Fonctionnement de l'opération abstraite d'enregistrement-MS (Register-MS abstract-operation)	64
15.1.6	Invocation de l'opération abstraite d'avertissement (Alert abstract-operation)	64
15.2	Fourniture des services abstraits accès pour remise indirecte.....	64
15.2.1	Fonctionnement de l'opération abstraite de dépôt de messages (MessageSubmission abstract-operation)	65
15.2.2	Fonctionnement de l'opération abstraite de dépôt d'essai (ProbeSubmission abstract-operation)	65
15.2.3	Fonctionnement de l'opération abstraite d'annulation de remise différée (CancelDeferredDelivery abstract-operation).....	65
15.2.4	Invocation de l'opération abstraite de commande de dépôt (SubmissionControl abstract-operation)	66
15.3	Fourniture des services abstraits d'accès d'administration	66
15.3.1	Fonctionnement de l'opération abstraite d'enregistrement (Register abstract-operation)..	66
15.3.2	Appel de l'opération abstraite de modification des pouvoirs (ChangeCredentials abstract-operation)	67
15.3.3	Fonctionnement de l'opération abstraite de modification des pouvoirs (ChangeCredentials abstract-operation).....	67
16	<i>Mise en œuvre des accès</i>	67
16.1	Accès de recherche	67
16.2	Accès de dépôt indirect.....	68
16.3	Accès d'administration	68
	Annexe A – Attribution formelle des identificateurs d'objets	68
	Annexe B – Définition formelle du service abstrait d'enregistrement de messages	70
	Annexe C – Définition formelle des types d'attributs généraux	77
	Annexe D – Définition formelle des types d'actions automatiques générales	82
	Annexe E – Définition formelle des limites supérieures du paramètre MS.....	83
	Annexe F – Exemple d'opération abstraite de résumé.....	84
	Annexe G – Différences entre la Recommandation X.413 (1992) et l'ISO/CEI 10021-5:1990.....	85

INTRODUCTION

Une série de Recommandations, dont la présente fait partie, définit la messagerie (MH) (*message handling*) dans un environnement de systèmes ouverts répartis.

La messagerie permet l'échange de messages entre des utilisateurs sur une base d'enregistrement et de retransmission. Un message déposé par un utilisateur donné (l'expéditeur) sera acheminé dans le système de transfert de messages (MTS) (*message-transfer-system*) et remis à un ou à plusieurs autres utilisateurs (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'un enregistrement de messages; mémoire de messages (MS) (*message store*) et le dépôt indirect de messages dans un système de messagerie (MHS) (*message handling system*). Le service abstrait MS assure aussi des services d'administration de messagerie, définis par le service abstrait de système de transfert de messages (MTS).

La présente Recommandation a été élaborée par accord mutuel du CCITT et de l'ISO/CEI. Cette Norme internationale est l'ISO/CEI 10021-5:1990 modifiée par les rectificatifs techniques 1, 2, 3 et 4. L'annexe G énumère les différences entre les documents du CCITT et de l'ISO/CEI.

Recommandation X.413

SYSTÈMES DE MESSAGERIE – ENREGISTREMENT DE MESSAGES: DÉFINITION DU SERVICE ABSTRAIT

(révisée en 1992)

SECTION 1 – CONSIDÉRATIONS GÉNÉRALES

1 Champ d'application

La présente Recommandation définit le service abstrait d'enregistrement de messages qui est assuré par le protocole d'accès à l'enregistrement de messages (spécifié dans la Rec. X.419 du CCITT | ISO/CEI 10021-6) conjointement avec le service abstrait de MTS (défini dans la Rec. X.411 du CCITT | ISO/CEI 10021-4) ainsi qu'avec les éléments de service d'opérations distantes (ROSE) (*remote operation service element*) définis dans la Rec. X.219 du CCITT | ISO/CEI 9072-1. La notation en syntaxe abstraite des protocoles de couche application utilisés dans cette Recommandation est définie dans la Rec. X.208 du CCITT | ISO 8824.

D'autres Recommandations définissent d'autres aspects des MHS. La Rec. X.400 du CCITT | ISO/CEI 10021-1 définit les services fournis aux utilisateurs par les MHS. La Rec. X.402 du CCITT | ISO/CEI 10021-2 donne une vue d'ensemble de l'architecture des MHS. La Rec. X.407 du CCITT | ISO/CEI 10021-3 décrit les conventions utilisées dans les MHS pour la définition de ce service abstrait. La Rec. X.420 du CCITT | ISO/CEI 10021-7 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 MS. Le § 7 spécifie la syntaxe abstraite pour les opérations de rattachement abstrait et de détachement abstrait. Le § 8 spécifie la notation de syntaxe abstraite pour les opérations du service abstrait. Le § 9 spécifie la notation de syntaxe abstraite pour les erreurs du service abstrait.

La section 3 de la présente Recommandation définit les types d'attributs généraux et les types d'actions automatiques générales qui sont liés au MS. Le § 10 donne une présentation générale. Le § 11 spécifie la 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 de transfert de messages. Le § 15 décrit comment le service abstrait d'enregistrement de messages est fourni. Le § 16 décrit comment les accès du service MS sont mis en œuvre.

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

2 Références normatives

Les Recommandations du CCITT et les Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation. Au moment de la publication, les éditions indiquées étaient en vigueur. Toute Recommandation ou Norme est sujette à révision et les parties prenantes aux accords fondés sur la présente Recommandation sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur. Le Secrétariat spécialisé du CCITT tient à jour une liste des Recommandations du CCITT en vigueur.

2.1 *Références relatives au modèle de référence*

- Recommandation X.200 du CCITT (1988), *Modèle de référence pour l'interconnexion des systèmes ouverts pour les applications du CCITT*.
- ISO 7498:1984/Corr. 1:1988, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Corrigendum technique 1*.

2.2 *Références relatives à la présentation*

- Recommandation X.208 du CCITT (1988), *Spécification de la syntaxe abstraite numéro un (ASN.1)*.
ISO/CEI 8824:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – Spécification de la notation de syntaxe abstraite numéro un (ASN.1)*.

2.3 *Références relatives aux opérations distantes*

- Recommandation X.219 du CCITT (1988), *Opérations distantes: modèle, notation et définition du service*.
ISO/CEI 9072-1:1989, *Systèmes de traitement de l'information – Communication de texte – Opérations à distance – Partie 1: Modèle, notation et définition du service*.

2.4 *Références relatives à l'annuaire*

- Recommandation X.500 du CCITT (1988), *L'annuaire – Aperçu général des concepts, modèles et services*.
ISO/CEI 9594-1:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 1: Présentation générale des concepts, modèles et services*.
- Recommandation X.501 du CCITT (1988), *L'annuaire – Modèles*.
ISO/CEI 9594-2:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 2: Modèles*.
- Recommandation X.509 du CCITT (1988), *L'annuaire – Cadre d'authentification*.
ISO/CEI 9594-8:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 8: Cadre général d'authentification*.
- Recommandation X.511 du CCITT (1988), *L'annuaire – Définition du service abstrait*.
ISO/CEI 9594-3:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 3: Définition du service abstrait*.
- Recommandation X.518 du CCITT (1988), *L'annuaire – Procédures de fonctionnement réparti*.
ISO/CEI 9594-4:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 4: Procédures pour le fonctionnement réparti*.
- Recommandation X.519 du CCITT (1988), *L'annuaire – Spécifications du protocole*.
ISO/CEI 9594-5:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 5: Spécifications du protocole*.
- Recommandation X.520 du CCITT (1988), *L'annuaire – Types d'attributs sélectionnés*.
ISO/CEI 9594-6:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 6: Types d'attributs sélectionnés*.
- Recommandation X.521 du CCITT (1988), *L'annuaire – Catégories d'objets sélectionnés*.
ISO/CEI 9594-7:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – L'annuaire – Partie 7: Classes d'objets sélectionnés*.

2.5 *Références relatives à la messagerie*

- Recommandation X.400 du CCITT (1992), *Système de messagerie – Principes du système et du service de messagerie*.
ISO/CEI 10021-1:1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 1: Présentation générale du système et des services*.
- Recommandation X.402 du CCITT (1992), *Systèmes de messagerie – Architecture globale*.
ISO/CEI 10021-2:1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 2: Architecture générale*.

- Recommandation X.407 du CCITT (1988), *Systèmes de messagerie – Conventions pour la définition des services abstraits*.
ISO/CEI 10021-3:1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 3: Conventions relatives à la définition de service abstrait*.
- Recommandation X.411 du CCITT (1992), *Systèmes de messagerie – Système de transfert de messages – Définition des services abstraits et procédures*.
ISO/CEI 10021-4:1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 4: Système de transfert de message – Procédures et définition de service abstrait*.
- Recommandation X.419 du CCITT (1992), *Systèmes de messagerie – Spécifications de protocoles*.
ISO/CEI 10021-6:1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 6: Spécifications de protocole*.
- Recommandation X.420 du CCITT (1992), *Systèmes de messagerie – Système de messagerie de personne à personne*.
ISO/CEI 10021-7:1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 7: Système de messagerie de personne à personne*.

3 Définitions

3.1 Définitions communes pour le MHS

On trouvera à la Rec. X.402 du CCITT | ISO/CEI 10021-2 une liste des définitions communes applicables au MHS.

3.2 Définitions concernant l'enregistrement des messages

Pour les besoins de la présente Recommandation, on utilise les définitions suivantes:

3.2.1 **association abstraite** (*abstract-association*)

Lien abstrait entre deux interlocuteurs; dans la présente Recommandation, lien existant entre un agent d'utilisateur et une mémoire de messages pour la fourniture du service abstrait mémoire de messages ou entre une mémoire de messages et un agent de transfert de messages pour la fourniture du service abstrait système de transfert de messages.

3.2.2 **paramètres de rattachement abstrait** (*abstract-bind-parameters*)

Paramètres définis dans la présente Recommandation et contenus dans l'opération de rattachement abstrait.

3.2.3 **paramètres de détachement abstrait** (*abstract-unbind-parameters*)

Paramètres définis dans la présente Recommandation et contenus dans l'opération de détachement abstrait.

3.2.4 **accès d'administration** (*administration port*)

Accès permettant la gestion (pour le système de transfert de messages) d'un ensemble de services abstraits à l'intérieur du service abstrait mémoire de messages.

3.2.5 **opération abstraite d'avertissement** (*alert abstract-operation*)

Opération abstraite permettant à la mémoire de messages de signaler à l'agent d'utilisateur, moyennant des critères de sélection, que des messages ou des rapports sont en attente dans la mémoire de messages. Cette opération ne peut être mise en œuvre que si une association abstraite existe.

3.2.6 **attribut** (*attribute*)

Information d'un type particulier apparaissant dans une entrée d'une base d'information.

3.2.7 **type d'attribut** (*attribute-type*)

Composant d'un attribut indiquant la classe d'information donnée par cet attribut.

3.2.8 **valeur d'attribut** (*attribute-value*)

Instance particulière de la classe d'information indiquée par un type d'attribut.

3.2.9 **assertion de valeur d'attribut** (*attribute-value-assertion*)

Proposition vraie, fausse ou non définie concernant les valeurs des attributs d'une entrée.

3.2.10 **action automatique** (*auto-action*)

Actions qui peuvent être effectuées automatiquement par la mémoire de messages sur la base d'informations préalablement enregistrées depuis le propriétaire de la mémoire de messages via l'agent d'utilisateur.

3.2.11 **type d'action automatique** (*auto-action-type*)

Qualificatif utilisé pour indiquer le type d'action automatique, par exemple avertissement.

3.2.12 **avertissement automatique** (*auto-alert*)

Action automatique dans la mémoire de messages qui déclenche une opération abstraite d'avertissement ou une autre action par la mémoire de messages.

3.2.13 **retransmission automatique** (*auto-forward*)

Action automatique dans la mémoire de messages qui déclenche la retransmission automatique par la mémoire de messages d'un message à un autre destinataire (ou à d'autres destinataires).

3.2.14 **entrée secondaire** (*child-entry*)

Entrée autre que l'entrée principale d'une base d'information. L'entrée maîtresse d'une entrée secondaire peut être soit l'entrée principale ou une autre entrée secondaire, selon le nombre de niveaux d'entrée dans chaque cas.

3.2.15 **numéro d'ordre d'une entrée secondaire** (*child-sequence-number*)

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 **élément conditionnel** [*conditional (C) component*]

Élément ASN.1 qui doit être présent dans une instance de sa classe comme l'indique la présente Recommandation. Voir catégorie **grade**.

3.2.17 **longueur du contenu** (*content-length*)

Attribut donnant la longueur du contenu d'un message remis (ou un contenu renvoyé).

3.2.18 **contenu renvoyé** (*content-returned*)

Attribut signalant qu'un rapport remis (ou un message remis) comporte un contenu renvoyé.

3.2.19 **types d'information codée convertis** (*converted EITs*)

Attribut identifiant les types de codage du contenu du message après conversion.

3.2.20 **date-heure de création** (*creation-time*)

Attribut donnant la date-heure de création (par la mémoire de messages) d'une entrée.

3.2.21 **opération abstraite de suppression** (*delete abstract-operation*)

Opération abstraite utilisée pour supprimer une ou plusieurs entrées d'une base d'information.

3.2.22 **types d'information codée remis** (*delivered-EITs*)

Attribut à plusieurs valeurs donnant des renseignements sur les types d'information codée d'un message remis.

3.2.23 **entrée message remis** (*delivered-message entry*)

Entrée, dans la base d'information, de messages enregistrés provenant d'un message remis.

3.2.24 **entrée rapport remis** (*delivered-report entry*)

Entrée, dans la base d'information, de messages enregistrés provenant d'un rapport remis.

3.2.25 **entrée** (*entry*)

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 **information d'entrée** (*entry-information*)

Paramètre utilisé dans les opérations abstraites pour transmettre une information sélectionnée à partir d'une entrée.

3.2.27 **sélection d'information d'entrée** (*entry-information-selection*)

Paramètre utilisé dans les opérations abstraites pour indiquer quelle information d'une entrée est demandée.

3.2.28 **état de l'entrée** (*entry-status*)

Attribut donnant des renseignements sur l'état de traitement de cette entrée. Les valeurs possibles sont: nouvelle, listée ou traitée.

3.2.29 **type d'entrée** (*entry-type*)

Attribut signalant si une entrée est associée à un message remis ou à un rapport remis.

3.2.30 **opération abstraite extraction** (*fetch abstract-operation*)

Opération abstraite qui permet d'extraire une entrée de la base d'information de messages enregistrés.

3.2.31 **restrictions à l'extraction** (*fetch-restrictions*)

Restrictions imposées par l'agent d'utilisateur 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 d'information codée.

3.2.32 **filtre** (*filter*)

Paramètre utilisé dans les opérations abstraites pour vérifier une entrée particulière d'une base d'information; l'entrée est ou non conforme à ce paramètre.

3.2.33 **item-filtre** (*filter-item*)

Affirmation sur la présence ou sur la/les valeur(s) d'un attribut de type particulier d'une entrée contrôlée. Cette affirmation peut être vraie ou indéfinie.

3.2.34 **demande de retransmission** (*forwarding-request*)

Paramètre qui peut être présent dans une opération abstraite de dépôt de messages sollicitée par l'agent d'utilisateur pour demander de retransmettre un message depuis la mémoire de messages.

3.2.35 **attribut général** (*general-attribute*)

Ensemble d'attributs de mémoire de messages qui sont valables pour tous les types de messages et de rapports, quel que soit le type de contenu. Seuls ces attributs de la mémoire de messages sont explicitement définis dans la présente Recommandation.

3.2.36 **action automatique générale** (*general-auto-action*)

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 **catégorie** (*grade*)

Défini dans la Rec. X.402 du CCITT | ISO/CEI 10021-2.

3.2.38 **accès de dépôt indirect** (*indirect-submission port*)

Accès permettant de fournir le service abstrait de dépôt indirect dans le cadre du service abstrait mémoire de messages. Le service abstrait de dépôt indirect offre les mêmes possibilités que le service abstrait de dépôt de messages (depuis le service abstrait système de transfert de messages). Il permet en outre de retransmettre des messages stockés dans la mémoire de messages.

3.2.39 **base d'information** (*information-base*)

Objets dans la mémoire de messages qui stockent les informations concernant le service abstrait mémoire de messages, par exemple, la base d'information de messages enregistrés qui met en mémoire les messages et les rapports remis dans la mémoire de messages.

3.2.40 **type de base d'information** (*information-base-type*)

Type de base d'information, par exemple les messages enregistrés.

3.2.41 **limite** (*limit*)

Composant du paramètre sélecteur identifiant le nombre maximal d'entrées sélectionnées à renvoyer à l'issue d'une opération abstraite.

3.2.42 **opération abstraite de listage** (*list abstract-operation*)

Opération abstraite permettant de sélectionner les entrées à partir d'une base d'information et les informations relatives aux attributs demandés à renvoyer pour ces entrées.

3.2.43 **listé** (*listed*)

Valeur du paramètre état de l'entrée.

3.2.44 **macro** (*macro*)

Voir la Rec. X.208 du CCITT | ISO/CEI 8824.

3.2.45 **entrée principale** (*main-entry*)

Pour chaque opération abstraite réussie, créant des entrées de base d'information, 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 **élément obligatoire** [*mandatory (M) component*]

Élément ASN.1 qui doit toujours être présent dans une instance de sa classe. Voir catégorie **grade**.

3.2.47 **concordance** (*matching*)

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 la mémoire de messages ou à décider si le type d'attribut indiqué est présent.

3.2.48 **élément de service recherche de message** (*message retrieval service element*)

Élément de service d'application à l'aide duquel un agent d'utilisateur récepteur effectue la recherche de messages provenant d'une mémoire de messages ou toute autre tâche connexe.

3.2.49 **mémoire de messages** (*message store*)

Enregistrement de messages; sigle utilisé comme abrégé du terme "MS abstract-service-provider" (fournisseur du service abstrait mémoire de messages).

3.2.50 **service abstrait mémoire de messages** (*MS abstract-service*)

Ensemble des possibilités que la mémoire de messages offre à ses utilisateurs au moyen de ses accès.

3.2.51 **utilisateur du service abstrait mémoire de messages** (*MS abstract-service-user*)

Utilisateur du service abstrait mémoire de messages, c'est-à-dire l'agent d'utilisateur.

3.2.52 **fournisseur du service abstrait mémoire de messages** (*MS abstract-service-provider*)

Mémoire de messages assurant le service abstrait mémoire de messages.

3.2.53 **utilisateur mémoire de messages** (*MS-user*)

Abrégé pour «utilisateur du service abstrait mémoire de messages».

3.2.54 **opération abstraite de dépôt de message** (*message-submission abstract-operation*)

Opération abstraite permettant à l'agent d'utilisateur de déposer un message au système de transfert de messages via la mémoire de messages et/ou de retransmettre un message de la mémoire de messages au système de transfert de messages.

3.2.55 **attribut à plusieurs valeurs** (*multi-valued-attribute*)

Attribut auquel peuvent être associées plusieurs valeurs.

3.2.56 **nouveau** (*new*)

Valeur du paramètre état de l'entrée.

3.2.57 **élément facultatif** [*optional (O) component*]

Élément ASN.1 qui doit être présent dans une instance de sa classe à la discrétion de l'objet (par exemple l'utilisateur) fournissant cette instance. Voir catégorie **grade**.

3.2.58 **type d'information codée d'origine** (*original-EITs*)

Attribut identifiant les types de codage d'origine du contenu du message.

3.2.59 **surclassement** (*override*)

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 **entrée maîtresse** (*parent-entry*)

Une entrée maîtresse possède 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 **numéro d'ordre d'entrée maîtresse** (*parent-sequence-number*)

Numéro d'ordre d'une entrée secondaire désignant son entrée maîtresse. Il ne peut y avoir qu'un seul numéro d'ordre d'une entrée maîtresse associé à une entrée secondaire.

3.2.62 **demande partielle de valeurs d'attribut** (*partial-attribute-request*)

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 utilisés pour spécifier la limite d'une étendue.

3.2.64 **traité** (*processed*)

Valeur du paramètre état de l'entrée.

3.2.65 **étendue** (*range*)

Paramètre utilisé dans les opérations abstraites pour sélectionner une séquence continue d'entrées à partir d'une base d'information.

3.2.66 **opération abstraite d'enregistrement mémoire de messages** (*register-MS abstract-operation*)

Opération abstraite permettant à l'agent d'utilisateur d'enregistrer dans la mémoire de messages certaines informations concernant l'interfonctionnement agent d'utilisateur-mémoire de messages.

3.2.67 **enregistrement** (*registration*)

Information enregistrée dans la mémoire de messages et mise en mémoire (jusqu'à ce qu'elle soit modifiée par l'opération abstraite d'enregistrement mémoire de messages) entre associations abstraites (voir "Register-MS abstract-operation" ci-dessus).

3.2.68 **identificateur d'enregistrement** (*registration-identifier*)

Identificateur pour un ensemble particulier de paramètres d'enregistrement pour un type d'action automatique.

3.2.69 **accès de recherche** (*retrieval port*)

Accès permettant la recherche d'un ensemble de services abstraits dans le service abstrait mémoire de messages.

3.2.70 **entrée contenu renvoyé** (*returned-content entry*)

Type d'entrée de la base d'information de messages enregistrés contenant le contenu renvoyé d'un message précédemment déposé.

3.2.71 **sélecteur** (*selector*)

Paramètre utilisé dans les opérations abstraites pour sélectionner les entrées d'une base d'information.

3.2.72 **numéro d'ordre** (*sequence-number*)

Attribut identifiant sans ambiguïté une entrée. Les numéros d'ordre sont attribués par ordre croissant.

3.2.73 **attribut à valeur unique** (*single-valued attribute*)

Attribut auquel ne peut être associée qu'une seule valeur.

3.2.74 **portée** (*span*)

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 **messages enregistrés** (*stored-messages*)

Base d'information la plus importante de la présente Recommandation, utilisée pour mettre en mémoire les entrées contenant les messages et les rapports remis à la mémoire de messages par le système de transfert de messages.

3.2.76 **abonnement** (*subscription*)

Accord à long terme entre le fournisseur ou l'administrateur de la mémoire de messages et les utilisateurs (propriétaires-mémoire de messages) sur la disponibilité et l'utilisation des caractéristiques facultatives de la mémoire de messages 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 **chaîne secondaire** (*substring*)

Filtre-élément utilisé pour spécifier une chaîne de caractères apparaissant (dans le même ordre donné) dans une valeur d'attribut.

3.2.78 **opération abstraite de résumé** (*summarize abstract-operation*)

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 du 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. Cet attribut doit être spécifié dans la Recommandation qui décrit le type de contenu; on se reportera par exemple au résumé messagerie de personne à personne défini dans la Rec. X.420 du CCITT | ISO/CEI 10021-7.

4 Abréviations

ASN.1	Notation de syntaxe abstraite numéro un (<i>abstract syntax notation one</i>)
AVA	Affirmation de valeur d'attribut (<i>attribute-value-assertion</i>)
C	Conditionnel
DL	Liste de distribution (<i>distribution-list</i>)
EIT	Type d'information codée (<i>encoded-information-type</i>)
IPMS	Service de messagerie de personne à personne (<i>interpersonal messaging system</i>)
M	Obligatoire (<i>mandatory</i>)
M	à plusieurs valeurs (<i>multi-valued</i>)
MASE	Élément de service de gestion de messages (<i>message administration service element</i>)
MDSE	Élément de service de remise de messages (<i>message delivery service element</i>)
MH	Messagerie (<i>message handling</i>)
MHS	Système de messagerie (<i>message handling system</i>)
MOTIS	Système d'échange de texte en mode message (<i>message oriented text interchange system</i>)
MRSE	Élément de service de recherche de message (<i>message retrieval service element</i>)
MS	Mémoire de messages; enregistrement de messages (<i>message store</i>)
MSSE	Élément de service de dépôt de messages (<i>message submission service element</i>)
MT	Transfert de message (<i>message transfer</i>)
MTA	Agent de transfert de messages (<i>message transfer agent</i>)
MTS	Système de transfert de messages (<i>message transfer system</i>)
N	Non
O	Optionnel
O/R	Expéditeur/destinataire (<i>originator/recipient</i>)
P	Présent
ROS	Service d'opérations distantes (<i>remote operation service</i>)
ROSE	Élément de service d'opérations distantes (<i>remote operation service element</i>)
S	à valeur unique (<i>single-valued</i>)
UA	Agent d'utilisateur; agent d'usager (<i>user agent</i>)
UTC	Temps universel coordonné (<i>coordinated universal time</i>)
Y	Oui (<i>yes</i>)

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) L'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 Rec. X.407 du CCITT | ISO/CEI 10021-3 pour spécifier l'accès de 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 Rec. X.407 du CCITT | ISO/CEI 10021-3 pour spécifier le service abstrait MS.

Chaque fois que la présente Recommandation décrit une classe de structure de données ayant des composants, dont chacun relève de l'un des **grades** ci-après:

- 1) **Obligatoire (M) (Mandatory)** – Un composant obligatoire doit figurer dans chaque instance de la classe.
- 2) **Optionnel (O)** – Un composant facultatif doit figurer dans une instance de la classe à la discrétion de l'objet (par exemple l'utilisateur) fournissant cette instance.
- 3) **Conditionnel (C)** – Un composant conditionnel doit figurer dans une instance 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 (*single-valued*)
- M à plusieurs valeurs (*multi-valued*)

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

- M obligatoire (*mandatory*)
- O optionnel

Pour les colonnes intitulées *Présence dans l'entrée de message remis*, *Présence dans l'entrée de rapport remis* et *Présence dans l'entrée de 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 sera 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 absent* 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 (*yes*)

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 (*single-valued*)
- M à plusieurs valeurs (*multi-valued*)

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

MD	Opération abstraite de remise de messages (<i>message delivery</i>)
MS	Mémoire de messages; enregistrement de messages (<i>message store</i>)
RD	Opération abstraite de remise de rapports (<i>ReportDelivery</i>)

5.4 Conventions applicables aux polices de 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 en ASN.1

Dans la présente Recommandation, les définitions ASN.1 sont imprimées en caractères différents (**gras**) de ceux qui sont 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 plus petits que ceux qui sont 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 en ASN.1

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

SECTION 2 – DÉFINITION DU SERVICE ABSTRAIT D'ENREGISTREMENT DE MESSAGES

6 Modèle d'enregistrement de messages

L'enregistrement de messages, mémoire de messages (MS) est représenté comme un objet atomique qui joue le rôle d'un fournisseur de services destinés à un utilisateur du service abstrait (*abstract-service-user*) MS [c'est-à-dire un agent d'utilisateur (UA) (*user agent*)] 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 utilisateur 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 utilisateur final unique du MHS, c'est-à-dire qu'il n'assure pas un service MS commun à plusieurs utilisateurs ou partagé par eux.

Le MS est décrit à l'aide d'un modèle abstrait pour définir les services assurés par le service abstrait d'enregistrement de messages. La figure 1/X.413 représente le service abstrait MS par rapport à son utilisateur 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.

On trouvera à la Rec. X.407 du CCITT | ISO/CEI 10021-3 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.

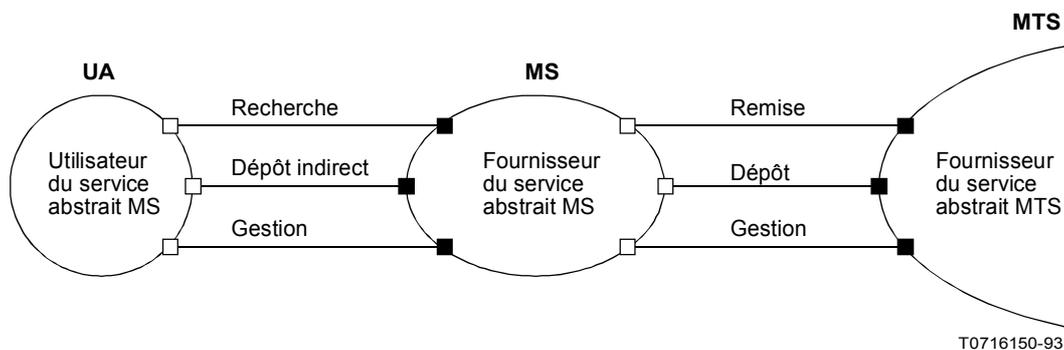


FIGURE 1/X.413

Service abstrait d'enregistrement des messages

6.1 *Objet enregistrement de messages (Message Store object)*

Le MS est représenté comme un objet atomique. Il assure les services abstraits de l'accès de recherche MS à l'utilisateur du service abstrait MS. En tant que «substitut» du fournisseur du service abstrait MTS, le MS offre également les services abstraits de gestion et de dépôt MTS à l'utilisateur 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 de dépôt MTS et de l'accès d'administration MTS, comme utilisateur du service abstrait MTS.

La définition formelle applicable à l'objet enregistrement de messages **Message Store object** 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 **MS-User** est également représenté comme un objet. Il utilise les services abstraits de l'accès de recherche MS, de l'accès de dépôt indirect et de l'accès d'administration 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 (Message Store ports)*

Un MS fournit les accès de recherche **Retrieval Port**, accès de dépôt indirect **Indirect-submission Port** et accès d'administration **Administration Port** à l'usager 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 – Une future version de la présente Recommandation pourra 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 des notifications avec les informations d'enregistrement concernant les messages sortants.

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 de remise, dépôt et administration. Les services abstraits offerts par les accès MTS sont définis au § 8 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

A l'aide de l'opération de rattachement abstrait (*abstract-bind*), 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 de recherche et du service de commande de dépôt assuré par l'accès de 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 de recherche (Retrieval Port)*

L'accès de recherche **Retrieval Port** 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 de recherche **Retrieval Port** sont décrits en détail dans les § 7 à 9.

6.2.2 *Accès de dépôt indirect (Indirect-submission Port)*

L'accès de dépôt indirect **Indirect-submission Port** est défini comme suit:

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

L'accès de dépôt indirect **Indirect-submission Port** utilise les services abstraits de l'accès pour dépôt définis au § 8.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

6.2.3 *Accès d'administration (Administration Port)*

L'accès d'administration **Administration Port** est défini au § 8.4 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

Le service abstrait de modification des pouvoirs fonctionne de bout en bout entre l'utilisateur du MS et le fournisseur du service de MTS. Il est acheminé par le MS. Celui-ci enregistre les nouveaux pouvoirs à utiliser lors de ses prochains rattachements au MTA. Si l'utilisateur du MS a besoin de mettre à jour les pouvoirs qu'il utilise pour se rattacher au MS, celui-ci utilise l'opération abstraite d'enregistrement MS (voir le § 8.6).

6.3 *Modèle d'information*

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

6.3.1 Bases d'information (Information-bases)

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

La présente Recommandation définit et décrit la base d'information des messages enregistrés **stored-messages information-base**. Cette base 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 – Une future version de la présente Recommandation pourra définir d'autres bases d'information pour l'enregistrement, appelées activation et désactivation.

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

6.3.2 Entrées (Entries)

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

Chaque entrée est identifiée à l'aide de son numéro d'ordre **sequence-number**, unique dans une base d'information **information-base** et créé par le MS au fur et à mesure que de nouvelles entrées sont créées. Dans une base d'information **information-base**, le MS crée les numéros d'ordre **sequence-numbers** 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 **sequence-numbers** en utilisant le temps pour obtenir une granularité suffisante permettant de garantir l'unicité.

6.3.3 Attributs (Attributes)

6.3.3.1 Introduction

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

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

Un attribut **attribute** comprend un type d'attribut **attribute-type** identifiant la classe d'information donnée par un attribut **attribute** et la ou les valeurs d'attribut **attribute-value(s)** correspondantes, qui sont des instances particulières de cette classe figurant dans l'entrée **entry**.

```
Attribute ::= SEQUENCE {  
    type AttributeType,  
    values SEQUENCE SIZE (1 .. ub-attribute-values) OF AttributeValue }
```

Remarque – Ainsi, par exemple, dans une entrée de message remis (décrite au § 6.4) le type d'attribut **attribute-type** pourrait être la priorité **priority** du message et la valeur d'attribut **attribute-value** correspondante pourrait être: urgent **urgent**.

Tous les attributs **attributes** d'une entrée **entry** doivent provenir de types d'attribut **attribute-types** distincts.

Pour certains types d'attribut **attribute-types**, un attribut **attribute** ne doit contenir qu'une seule valeur d'attribut **attribute-value**. Un tel type d'attribut **attribute-type** est dit à valeur unique **single-valued**. Pour d'autres, un attribut **attribute** peut contenir une ou plusieurs valeurs d'attributs **attributes-values**, toutes appartenant au même type de données ASN.1. Ce type d'attribut **attribute-type** est dit à plusieurs valeurs **multi-valued**. Le type d'attribut **attribute-type** est déclaré à valeur unique **single-valued** ou à plusieurs valeurs **multi-valued** lorsque le type d'attribut **attribute-type** est défini (voir le § 6.3.3.2).

Remarque – Ainsi, par exemple, le type d'attribut **attribute-type** pour l'attribut nom de l'expéditeur **attribute originator-name** (décrit au § 11.2.28) est à valeur unique **single-valued** alors que l'attribut **attribute** pour noms des autres destinataires **other-recipient-names** (décrit au § 11.2.29) est à plusieurs valeurs **multi-valued**.

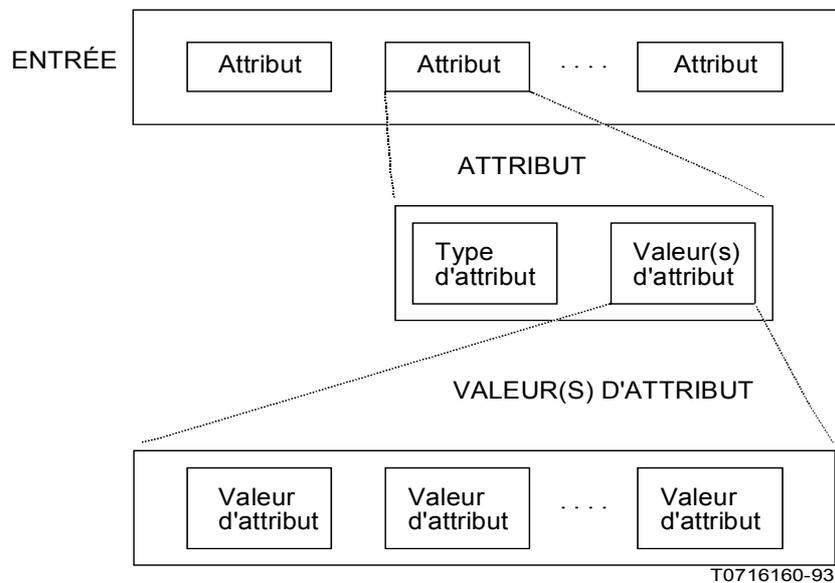


FIGURE 2/X.413
Composants d'une entrée

6.3.3.2 Type d'attribut (Attribute-type)

Certains types d'attribut **attribute-types** seront normalisés au niveau international. D'autres seront définis par les autorités administratives nationales et par 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 **attribute-type** est identifié à l'aide d'un identificateur d'objet lorsque le type d'attribut **attribute-type** est défini.

AttributeType ::= OBJECT IDENTIFIER

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

6.3.3.3 Valeurs d'attributs (Attribute-values)

Définir un type d'attribut **attribute-type** 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 **attribute-value** pour le type d'attribut **attribute-type** est défini à l'aide de l'identificateur d'objet pour le type d'attribut **attribute-type**.

AttributeValue ::= ANY

6.3.3.4 Définition d'un type d'attribut (attribute-type) et de la macro ATTRIBUT (ATTRIBUTE)

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

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

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

La macro ASN.1 suivante est utilisée pour définir un type d'attribut **attribute-type**. La définition formelle de cette macro est donnée dans la Rec. X.501 du CCITT | ISO/CEI 9594-2; 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 ”)
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” “MULTI VALUE” 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 **ATTRIBUT (ATTRIBUTE)** est la suivante:

- a) valeur **MACRO MACRO value**: identificateur d'objet utilisé pour identifier un attribut;
- b) syntaxe d'attribut **Attribute-syntax**: indique la syntaxe qui a été choisie;
- c) choix de syntaxe **Syntax-choice**: 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 types** de concordance **MatchTypes**;
- d) à plusieurs valeurs **Multivalued**: indique s'il s'agit d'un attribut à valeur unique ou d'un attribut à plusieurs valeurs;
- e) types de concordance **Match-types**: donne 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 secondaires** (**“SUBSTRINGS”**) et pour **classement** (**“ORDERING”**).

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

- i) égalité **EQUALITY**: 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) chaînes secondaires **SUBSTRINGS**: s'applique à n'importe quelle syntaxe d'attribut avec un type de données **string**. La valeur présentée doit être une séquence («SEQUENCE OF»), chacun de ces éléments étant conforme au type de données; et
- iii) classement **ORDERING**: 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 ou égale à – ou bien supérieure ou égale à – une valeur cible. La valeur présentée doit être conforme au type de données de la syntaxe d'attribut. Le MS utilise cela pour les types de données **INTEGER** (entier) et **UTCTime** (temps UTC). Pour ce dernier, le classement est chronologique et non pas alphabétique.

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

6.3.4 Entrées principales, entrées maîtresses et entrées secondaires (Main-entries, parent-entries and child-entries)

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 même entrée, ou entrée secondaire **child-entry**, peut être rattachée à une autre, son entrée maîtresse **parent-entry**, dans une relation de type arborescente. Une entrée qui n'est pas une entrée secondaire **child-entry** est une entrée principale **main-entry**.

Cette relation est définie au moyen de deux attributs généraux (*general-attributes*) particuliers:

- a) numéro d'ordre d'entrée maîtresse **parent-sequence-number**: cet attribut à valeur unique donne le numéro d'ordre de l'entrée maîtresse **parent-entry** d'une entrée secondaire **child-entry**. Il ne figure pas dans une entrée principale **main-entry**. Cette définition est donnée au § 11.2.30;
- b) numéros d'ordre d'entrée secondaire **child-sequence-numbers**: cet attribut à plusieurs valeurs donne les numéros d'ordre de toutes les entrées secondaires **child-entries** d'une entrée maîtresse **parent-entry**. Il ne figure pas dans une entrée qui n'est pas une entrée maîtresse **parent-entry**. 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 **main-entries**. Certaines peuvent être orientées pour agir sur toutes les entrées, tant les entrées principales **main-entries** que les entrées secondaires **child-entries**. En particulier, l'argument d'une opération abstraite suppression (voir le § 8.5) ne peut choisir que des entrées principales **main-entries**, auquel cas l'entrée principale **main-entry** 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 Rec. X.420 du CCITT | ISO/CEI 10021-7) soient représentées par telle ou telle entrée secondaire **child-entry**. L'attribut général de contenu **content-general-attribute** de l'entrée principale **main-entry** comprendra la totalité du contenu **content**, de sorte que les données représentant la partie-corps **body-part** de ce message seront logiquement présentes dans plus d'une entrée **entry**.

6.4 Messages enregistrés (*Stored messages*)

La base d'information des messages enregistrés **stored-messages information-base** 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 (**delivered-message-entries**) d'un nombre illimité de types de contenu ainsi que des entrées pour les rapports (**delivered-report-entries**). Une entrée d'une base d'information des messages enregistrés **stored-messages information-base** 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 des 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 (*content-type*). 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 une notification entrante 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 une notification de non-remise contient un contenu renvoyé (l'**entry delivered-report** est l'entrée principale, le contenu renvoyé étant son entrée secondaire, c'est-à-dire une entrée contenu renvoyé **returned-content entry**).

Les règles de subdivision du contenu d'un message en plusieurs entrées varient selon le type de contenu. Un attribut synopsis **synopsis-attribute** spécifique du 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 **synopsis-attribute** est créé par le MS.

Remarque – Pour le service de messagerie de personne à personne (Rec. X.420 du CCITT | ISO/CEI 10021-7) 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 de messagerie de personne à personne **ipm-synopsis** est un exemple de type d'attribut synopsis **synopsis-attribute-type** spécifique du contenu.

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

- a) nouveau **New**: le message n'a été ni listé **listed** par l'UA ni traité **processed** automatiquement par le MS;
- b) listé **Listed**: les informations concernant le message ont été renvoyées à l'UA par une opération abstraite de listage ou d'extraction, mais le message n'a pas encore été entièrement traité **processed**;

- c) traité **Processed**: soit un UA a «entièrement extrait» le message, soit le MS a effectué une action automatique sur ce message et la définition de cette action automatique provoque un changement d'état de l'entrée. (A noter que certaines actions automatiques conduisent à la suppression du message.) La définition exacte de «entièrement extrait» est spécifique au contenu et est spécifiée dans l'attribut contenu (voir la Recommandation correspondante sur le contenu).

L'état d'une entrée **entry-status** d'une notification de remise/de non-remise devient traité **processed** lorsque l'enveloppe du rapport remis est recherchée.

La définition de l'attribut état d'une entrée **entry-status** est la suivante:

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

L'état d'une entrée secondaire est géré selon les mêmes règles qu'une entrée principale. La modification de l'état d'une entrée peut être le résultat d'opérations effectuées sur l'entrée maîtresse ou secondaire de cette entrée. Si une entrée secondaire est logiquement présente dans un attribut de son entrée maîtresse, la recherche de cet attribut est considérée comme équivalant à celle de tous les attributs de cette entrée secondaire. Si un attribut d'entrée est présent logiquement dans plusieurs entrées secondaires, la recherche de toutes ces entrées équivaut à celle de cet attribut.

6.5 Actions automatiques (*Auto-actions*)

6.5.1 Introduction

Ce paragraphe décrit le cadre des actions automatiques (**auto-actions**) qui peuvent être enregistrées avec le MS ou commandées par abonnement.

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

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

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

Même si une action automatique (*auto-action*) possède un paramètre d'enregistrement d'action automatique **auto-action-registration parameter** défini, tous les critères et autres paramètres nécessaires pour son exécution peuvent être indiqués au MS par abonnement. Certaines actions automatiques **auto-actions** peuvent cependant exiger que le MS gère l'enregistrement de son paramètre d'enregistrement **registration-parameter** au moyen de l'opération abstraite d'enregistrement MS (voir le § 8.6).

6.5.2 Type d'action automatique (*auto-action-type*)

Certains types d'action automatique **auto-action-types** seront normalisés au niveau international. D'autres types d'action automatique **auto-action-types** seront définis par les autorités administratives nationales et par 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'action automatique **auto-action-types** attribués, ce qui se fait en identifiant chaque type d'action automatique **auto-action-type** à l'aide d'un identificateur d'objet lorsque le type d'action automatique **auto-action-type** est défini.

```
AutoActionType ::= OBJECT IDENTIFIER
```

Certains types d'action automatique **auto-action-types** de caractère général sont définis au § 12. Ces types d'action automatique **auto-action-types** sont des types d'action automatique générale **general-auto-action-types** et les actions automatiques **auto-actions** de ce type sont des actions automatiques générales **general-auto-actions**.

6.5.3 Paramètre d'enregistrement d'action automatique (auto-action-registration)

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

6.5.4 Définition du type d'action automatique (auto-action-type) et macro AUTO-ACTION

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

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

On peut (le cas échéant) utiliser la macro ASN.1 suivante pour définir un type d'action automatique **auto-action-type**:

```
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) enregistrement **Registration**: donne le type de données des paramètres d'enregistrement associés à une action automatique;
- b) valeur **Value**: 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 **auto-action** (ou d'actions automatiques **auto-actions** différentes).

6.6 Retransmission de messages

L'utilisateur du MS utilise l'opération abstraite de remise de messages (*Message-submission*) et les paramètres qui lui sont associés, définis au § 8.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4, pour demander qu'un message mis en mémoire dans le MS soit explicitement retransmis à d'autres utilisateurs.

Le paramètre demande de retransmission **forwarding-request parameter** est défini à l'aide de la macro EXTENSION, définie elle-même au § 9 de la Rec. X.411 du CCITT | ISO/CEI 10021-4 comme suit:

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

Si le numéro d'ordre **SequenceNumber** fourni ne correspond pas à celui d'une entrée dans la base d'information des messages enregistrés **stored-messages information-base** ou correspond à une entrée qui ne convient pas pour la retransmission, cela est déclaré en utilisant l'erreur abstraite du type demande incohérente **Inconsistent-request** du § 8.2.2.7 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

7 Opérations de rattachement abstrait (**abstract-bind**) et de détachement abstrait (**abstract-unbind**)

7.1 Opération de rattachement abstrait (*Abstract-bind-operation*)

L'opération de rattachement abstrait du rattachement MS (*MS-Bind abstract-bind-operation*) établit un lien entre les accès de dépôt indirect, de recherche et d'administration de l'utilisateur du MS (consommateur) et le MS (fournisseur). Le demandeur [du rattachement MS (*MS-Bind*)] est l'utilisateur du MS, le répondeur étant le MS lui-même. Les informations échangées dans l'argument et le résultat du rattachement MS doivent s'appliquer pendant la durée de l'association abstraite. Le rattachement MS (*MS-Bind*) 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.

Remarque – Les mécanismes de traitement des associations multiples feront l'objet d'un complément d'étude.

7.1.1 Argument de rattachement abstrait (*Abstract-bind-argument*)

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

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 -- aucune restriction par défaut --,  
ms-configuration-request [5] BOOLEAN DEFAULT FALSE }
```

- 1) Nom du demandeur **Initiator-name** (C): cet argument, contenant le nom du demandeur de l'association, est fourni par le demandeur. Il est défini plus avant au § 8.1.1.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.
- 2) Pouvoirs du demandeur **Initiator-credentials** (M): ce paramètre contient les pouvoirs **credentials** du demandeur de l'association. Il doit être créé par le demandeur de l'association abstraite.

Les pouvoirs du demandeur **initiator-credentials** peuvent être utilisés par le répondeur pour vérifier l'identité du demandeur (voir la Rec. X.509 du CCITT | ISO/CEI 9594-8).

Si l'on a recours à une authentification simple **simple-authentication**, les pouvoirs du demandeur **initiator-credentials** comportent un mot de passe (*password*) simple, dont une définition plus complète est donnée au § 8.5.11 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

Si l'on a recours à une authentification forte **strong-authentication**, les pouvoirs du demandeur **initiator-credentials** comportent un jeton de rattachement demandeur **initiator-bind-token** et, à titre facultatif, un certificat du demandeur **initiator-certificate**. Le jeton de rattachement demandeur **initiator-bind-token** et le certificat du demandeur **initiator-certificate** sont définis plus avant au § 8.1.1.1.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4. Les pouvoirs du demandeur **initiator-credentials** de l'utilisateur du MS peuvent différer de ceux qui sont utilisés dans le rattachement MTS comme indiqué au § 8.1.1.1.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

- 3) Contexte de sécurité **Security-context** (O): ce paramètre identifie le contexte de sécurité **security-context** 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 **security-context** est défini plus avant au § 8.1.1.1.1.3 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

Le contexte de sécurité **security-context** comporte une ou plusieurs étiquettes de sécurité **security-labels** 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 **security-policy** en vigueur. Le contexte de sécurité **security-context** sera autorisé par les étiquettes de sécurité de l'utilisateur **user-security-labels** enregistrées de l'utilisateur du service abstrait MS et par les étiquettes de sécurité **security-labels** associées au MS.

En l'absence de ce paramètre, aucun contexte de sécurité **security-context** n'est établi 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 lanceur du service abstrait.

- 4) Restrictions d'extraction **Fetch-restrictions** (O): il s'agit des restrictions portant sur les entrées à renvoyer à l'issue de l'opération abstraite de recherche. Les restrictions d'extraction **fetch-restrictions** restent en vigueur jusqu'au déclenchement d'une opération de détachement abstrait.

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

Cet argument comprend les composants suivants:

```
Restrictions ::= SET {
allowed-content-types [0] SET SIZE (1 . . ub-content-types) OF OBJECT IDENTIFIER
OPTIONAL
-- pas de restriction par défaut --,
allowed-EITs [1] MS-EITs OPTIONAL -- pas de restriction par défaut --,
maximum-content-length [2] ContentLength OPTIONAL -- pas de restriction par défaut -- }
```

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

- b) Types d'information codée (EIT) autorisés **Allowed-EITs** (C): types d'information codée (*encoded information type*) 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 d'information codée autres que ceux qui sont spécifiés, il est procédé à un filtrage afin de ne pas renvoyer les parties du type d'information codée non autorisées avec le texte du message. Si le message ne contient que des types d'information codée non autorisés, on signalera une erreur. Il n'est procédé à aucun filtrage si l'opération abstraite d'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 **fetch-restrictions** aux types d'information codée.

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

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

- 5) Demande de configuration MS **MS-configuration-request** (C): ce paramètre spécifie si le MS est requis de renvoyer l'information qui identifie les types d'action automatique, les types d'attributs facultatifs et les types de contenu auxquels l'utilisateur du MS s'est abonné et qui sont donc à disposition pendant l'association abstraite.

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 du rattachement abstrait (*Abstract-bind-result*)

Les paramètres du résultat du rattachement abstrait **abstract-bind-result** 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 AttributeType**
 OPTIONAL,
 alert-indication **[5] BOOLEAN DEFAULT FALSE,**
 content-types-supported **[6] SET SIZE (1 .. ub-content-types) OF OBJECT IDENTIFIER**
 OPTIONAL }

- 1) Pouvoirs du répondeur **Responder-credentials** (M): ce paramètre contient les pouvoirs du répondeur de l'association abstraite. Il doit être créé par le répondeur de l'association abstraite.

Les pouvoirs du répondeur **responder-credentials** peuvent être utilisés par le demandeur pour vérifier l'identité du répondeur (voir la Rec. X.509 du CCITT | ISO/CEI 9594-8).

Si l'on a recours à une authentification simple **simple-authentication**, les pouvoirs du répondeur **responder-credentials** comportent un mot de passe **password** simple associé au répondeur. Une définition plus complète du mot de passe **password** est donnée au § 8.5.11 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

Si l'on a recours à une authentification forte **strong-authentication**, les pouvoirs du répondeur **responder-credentials** comportent un jeton de rattachement répondeur **responder-bind-token** créé par le répondeur de l'association abstraite. Le jeton de rattachement répondeur **responder-bind-token** est défini plus avant au § 8.1.1.1.2.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

- 2) Actions automatiques disponibles **Available-auto-actions** (C): ce paramètre indique l'ensemble des types d'action automatique auxquels l'utilisateur du MS s'est abonné. Il est présent si l'argument de rattachement abstrait le demande et qu'au moins un type d'action automatique soit disponible; sinon, il est absent.
- 3) Types d'attributs disponibles **Available-attribute-types** (C): ce paramètre indique l'ensemble des types d'attributs auxquels l'utilisateur du MS s'est abonné. Il est présent si l'argument de rattachement abstrait le demande et qu'au moins un type d'attribut soit disponible; sinon, il est absent.
- 4) Indication d'avertissement **Alert-indication** (C): ce paramètre, si sa valeur est vraie, indique qu'une situation d'avertissement s'est produite depuis la dernière indication d'avertissement **Alert-indication**.
- 5) Types de contenus gérés **Content-types-supported** (C): ce paramètre indique l'ensemble des types de contenu auxquels l'utilisateur du MS s'est abonné et auxquels le MS offre une gestion spécifique, comme défini dans la Recommandation particulière concernant le contenu. Un message dont le type de contenu est absent de cet ensemble peut encore être déposé, ou peut être remis et recherché par la suite. Dans ce dernier cas, aucun des types d'attributs spécifiques du contenu ou des types d'action automatique définis pour ce type de contenu ne sera à la disposition de l'utilisateur MS.

Ce paramètre est présent s'il a été demandé dans l'argument du rattachement abstrait et qu'au moins un type de contenu soit géré. Sinon, il est absent.

7.1.3 Erreurs de rattachement abstrait (*Abstract-bind-errors*)

Une erreur de rattachement MS **MS-bind-error** signale qu'un problème s'est posé lors d'une tentative d'établissement d'une association abstraite.

La définition des erreurs est la suivante:

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

- 1) Erreur d'authentification **Authentication-error** (C): cette erreur indique qu'une association abstraite ne peut être établie parce que les pouvoirs **credentials** du demandeur ne sont pas acceptables ou sont mal spécifiés.

L'erreur d'authentification **authentication-error** n'a pas de paramètres.

- 2) Contexte de sécurité inacceptable **Unacceptable-security-context** (C): cette erreur indique que le répondeur ne peut accepter le contexte de sécurité **security-context** proposé par le demandeur de l'association abstraite.

Le contexte de sécurité inacceptable **unacceptable-security-context** n'a pas de paramètres.

- 3) Incapacité à créer une association **Unable-to-establish-association** (C): cette erreur indique que le répondeur a rejeté la tentative du demandeur visant à créer une association abstraite.

L'incapacité à créer une association **unable-to-establish-association** n'a pas de paramètres.

7.2 Opération de détachement abstrait (*Abstract-unbind-operation*)

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

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

8 Opérations abstraites (*abstract-operations*)

Le présent paragraphe définit les opérations abstraites **abstract-operations** suivantes, disponibles à l'accès pour recherche:

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

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

Les opérations abstraites peuvent être exécutées de manière asynchrone, sous réserve des conditions suivantes. Les opérations abstraites de suppression et d'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 invoquées et doivent être menées à bien avant que d'autres opérations abstraites soient exécutées. Cela étant et sachant que les opérations abstraites de listage et d'extraction modifient l'état de l'entrée d'un message, les résultats des opérations abstraites de résumé, listage et extraction peuvent ne pas avoir un caractère déterministe.

8.1 Types de données communes utilisés dans les opérations abstraites

Ce paragraphe définit un certain nombre de types de données communes **common data-types** utilisés dans plusieurs opérations abstraites **abstract-operations** définies au § 8. Beaucoup de ces opérations abstraites **abstract-operations** 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) étendue;
- b) filtre;
- c) sélecteur;
- d) sélection d'information d'entrée;
- e) information d'entrée.

8.1.1 *Étendue (Range)*

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

```
Range ::= CHOICE {  
    sequence-number-range [0] NumberRange,  
    creation-time-range[1] TimeRange }  
  
NumberRange ::= SEQUENCE {  
    from [0] SequenceNumber OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas  
                                         -- de limite inférieure --,  
    to [1] SequenceNumber OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas  
                                         -- de limite supérieure -- }  
  
TimeRange ::= SEQUENCE {  
    from [0] CreationTime OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas  
                                         -- de limite inférieure --,  
    to [1] CreationTime OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas  
                                         -- de limite supérieure -- }  
  
CreationTime ::= UTCTime
```

Les composants du paramètre étendue **Range** ont la signification suivante:

- 1) Étendue des numéros d'ordre **Sequence-number-range** (C); et
- 2) Étendue des dates-heures de création **Creation-time-range** (C): ces deux paramètres identifient la séquence continue des entrées à sélectionner. Le composant étendue des numéros d'ordre **sequence-number-range** est exprimé sous forme de numéros d'ordre **sequence-numbers** et le composant étendue des dates-heures de création **creation-time-range** est exprimé sous forme de dates-heures de création **creation-times**. Le composant date-heure de création **creation-time** d'une entrée indique la date et l'heure auxquelles le MS a créé l'entrée en cours. Les numéros d'ordre **sequence-numbers** d'entrées successives sont toujours dans l'ordre ascendant mais plusieurs entrées adjacentes peuvent avoir la même date-heure de création **creation-time**. Les composants de étendue des numéros **number-range** et étendue des dates-heures **time-range** ont la signification suivante:

- a) à partir de **From** (O): il s'agit de la limite inférieure de étendue **range**.

En l'absence de ce composant, la valeur par défaut est pas de limite inférieure **no lower bound**; la sélection commence par le premier message reçu (numéro d'ordre **sequence-number** le moins élevé) dans la base d'information.

- b) jusqu'à **To** (O): il s'agit de la limite supérieure de étendue **range**.

En l'absence de ce composant, la valeur par défaut est pas de limite supérieure **no upper bound**; la sélection s'achève par le dernier message reçu (numéro d'ordre **sequence-number** le plus élevé) dans la base d'information.

8.1.2 *Filtres (Filters)*

8.1.2.1 *Filtre*

Un paramètre filtre **filter** fait subir un essai à une entrée donnée; l'essai est soit positif, soit négatif. Le filtre **filter** 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 évaluée est vraie **true**. Un filtre **filter** peut être vrai **true**, faux **false** ou non défini **undefined**.

```
Filter ::= CHOICE {  
    item [0] FilterItem,  
    and [1] SET OF Filter,  
    or [2] SET OF Filter,  
    not [3] Filter }
```

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

- a) un filtre **filter** qui est un item-filtre **filter-item** a la valeur de l'item-filtre **filter-item** (c'est-à-dire vraie **true**, fausse **false** ou non définie **undefined**);
- b) un filtre **filter** qui est le «et» (**and**) d'un ensemble de filtres **filters** est vrai **true** si cet ensemble est vide ou si chaque filtre **filter** est vrai **true**; il est faux **false** si au moins un filtre **filter** est faux **false**; sinon, il est non défini **undefined** (c'est-à-dire qu'au moins un filtre **filter** est non défini **undefined** et qu'aucun filtre **filter** n'est faux **false**);
- c) un filtre **filter** qui est le «ou» (**or**) d'un ensemble de filtres **filters** est faux **false** si cet ensemble est vide ou si chaque filtre **filter** est faux **false**; il est vrai **true** si au moins un filtre **filter** est vrai **true**; sinon, il est non défini **undefined** (c'est-à-dire qu'au moins un filtre **filter** est non défini **undefined** et qu'aucun filtre **filter** n'est vrai **true**);
- d) un filtre **filter** qui est le «non» (**not**) d'un filtre **filter** est vrai **true** si ce filtre **filter** est faux **false**; faux **false** s'il est vrai **true**; et non défini **undefined** si ce filtre **filter** est non défini **undefined**.

8.1.2.2 Item-filtre (Filter-item)

Un item-filtre **filter-item** 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 vraie **true**, fausse **false** ou non définie **undefined**.

```

FilterItem ::= CHOICE {
    equality          [0]  AttributeValueAssertion,
    substrings       [1]  SEQUENCE {
        type          AttributeType,
        strings       SEQUENCE OF CHOICE {
            initial    [0]  AttributeValue,
            any         [1]  AttributeValue,
            final      [2]  AttributeValue } },
    greater-or-equal [2]  AttributeValueAssertion,
    less-or-equal[3] AttributeValueAssertion,
    present          [4]  AttributeType,
    approximate-match [5] AttributeValueAssertion }

```

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

Toute affirmation sur la valeur d'un tel attribut n'est définie que si le type d'attribut est connu par le mécanisme d'évaluation et que la/les valeur(s) visée(s) de l'attribut soit (soient) conforme(s) à la syntaxe attributive définie pour ce type d'attribut. Si ces conditions ne sont pas satisfaites, l'item-filtre **filter-item** est non défini.

Les affirmations sur la valeur d'un attribut sont évaluées au moyen des règles de concordance associées à la syntaxe attributive définie pour ce type d'attribut (voir le § 6.3.3.4). Une règle de concordance non définie pour une syntaxe attributive particulière ne peut pas être utilisée pour faire des affirmations sur l'attribut. Si cette condition n'est pas satisfaite, l'item-filtre **filter-item** est non défini.

Un item-filtre **filter-item** peut être non défini, comme décrit ci-dessus. Sinon, lorsque l'item-filtre **filter-item** affirme:

- a) égalité **equality**, il est vrai **true** s'il y a une valeur de l'attribut égale à celle qui est affirmée et uniquement dans ce cas;
- b) chaînes secondaires **substrings**, il est vrai **true** s'il y a une valeur de l'attribut dans laquelle les chaînes secondaires **substrings** spécifiées apparaissent dans l'ordre donné et uniquement dans ce cas. Les chaînes secondaires **substrings** ne doivent pas se chevaucher et peuvent (le cas échéant) être séparées des extrémités de la valeur d'attribut et l'une de l'autre par zéro, un ou plusieurs éléments de chaîne **string**.

Si l'élément **initial** est présent, la chaîne secondaire doit concorder avec la chaîne secondaire initiale de la valeur d'attribut; si l'élément **final** est présent, la chaîne secondaire doit concorder avec la chaîne secondaire finale de la valeur d'attribut; si l'élément quelconque **any** est présent, la chaîne secondaire doit concorder avec l'élément quelconque de la valeur d'attribut;

- c) supérieur ou égal **greater-or-equal**, il est vrai **true** si le classement relatif (tel qu'il est défini par l'algorithme de classement approprié) place une valeur de l'attribut après, c'est-à-dire plus grand que ou égal à la valeur fournie;
- d) inférieur ou égal **less-or-equal**, il est vrai **true** si le classement relatif (tel qu'il est défini par l'algorithme de classement approprié) place une valeur de l'attribut avant, c'est-à-dire inférieur ou égal à la valeur fournie;
- e) présent **present**, il est vrai **true** si cet attribut est présent dans l'entrée et seulement dans ce cas;
- f) concordance approximative **approximate-match**, il est vrai **true** 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 d'orthographe, concordance phonétique, etc.) et seulement dans ce cas. Il n'y a aucune directive spécifique pour la concordance approximative dans la présente version de cette Recommandation. Si la concordance approximative **approximate-match** n'est pas assurée, cet item-filtre **filter-item** doit être traité comme s'il était concordant par égalité **equality**.

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 **filter-item**.

8.1.2.3 Affirmation de valeur d'attribut (*Attribute-value-assertion*)

Une affirmation de valeur d'attribut **attribute-value-assertion** est une proposition – vraie **true**, fausse **false** ou non définie **undefined** – concernant les valeurs d'une entrée. Elle est évaluée au moyen d'une règle de concordance spécifiée pour le type approprié au contexte dans lequel l'affirmation de valeur d'attribut **attribute-value-assertion** est évaluée. Elle fait intervenir un type d'attribut et une valeur d'attribut:

```
AttributeValueAssertion ::= SEQUENCE {
    type      AttributeType,
    value     AttributeValue }
```

Elle est:

- a) non définie **undefined**, dans les cas suivants:
 - 1) le type d'attribut est inconnu;
 - 2) la syntaxe attributive du type ne possède pas de règle de concordance;
 - 3) la valeur n'est pas conforme aux types de données de la syntaxe attributive;

Remarque – Les cas 2) et 3) indiquent normalement que l'affirmation de valeur d'attribut (AVA) (*attribute-value-assertion*) est erronée; le cas 1) peut toutefois se produire en tant que situation locale (par exemple si un MS particulier ne gère pas ce type d'attribut particulier).
- b) vraie **true**, 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) fausse **false**, dans tous les autres cas.

8.1.3 Sélecteur (*selector*)

Un paramètre sélecteur **selector** est utilisé pour sélectionner les entrées d'une base d'information. 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 son étendue. 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 -- pas de limites par défaut --,
    filter        [2] Filter OPTIONAL -- toutes les entrées sont censées être dans l'étendue
                    spécifiée --,
    limit         [3] INTEGER (1 . . ub-messages) OPTIONAL,
    override     [4] OverrideRestrictions OPTIONAL -- toutes restrictions d'extraction
                    éventuellement applicables sont censées s'appliquer par défaut -- }
```

Les composants d'un sélecteur ont les significations suivantes:

- 1) entrées secondaires **Child-entries** (O): si la valeur de ce composant est fausse **false**, seules les entrées principales sont prises en compte pour la sélection. Si la valeur de ce composant est vraie **true**, 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) étendue **Range** (O): la syntaxe abstraite du composant étendue **range** est donnée au § 8.1.1.

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

- 3) filtre **Filter** (O): la notation de syntaxe abstraite du composant filtre **filter** est donnée au § 8.1.2.

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

- 4) limite **Limit** (O): ce composant permet de fixer une limite supérieure pour le nombre d'entrées à sélectionner.

En l'absence de ce composant, aucune limite n'est fixée quant au nombre d'entrées choisies;

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) annulation **Override** (O): si l'annulation de l'une des restrictions d'extraction **fetch-restrictions** est demandée, le/les composant(s) correspondant(s) des restrictions d'annulation **override-restrictions** doit (doivent) être présent(s).

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

Les bits des restrictions d'annulation **override-restrictions** ont la signification suivante:

- a) annulation de restriction sur les types de contenu **Override-content-types-restriction** (M): ce bit doit être mis sur 1 pour annuler les restrictions d'extraction **fetch-restrictions** portant sur les types de contenus.

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

- b) annulation de restriction sur les EIT **Override-EITs-restriction** (M): ce bit doit être mis sur 1 pour annuler les restrictions d'extraction **fetch-restrictions** portant sur les types d'information codée (EIT).

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

- c) annulation de restriction sur la longueur du contenu **Override-content-length-restriction** (M): ce bit doit être mis sur 1 pour annuler les restrictions d'extraction **fetch-restrictions** portant sur la longueur du contenu.

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

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

8.1.4 *Sélection d'information d'entrée (Entry-information-selection)*

Un paramètre sélection d'information d'entrée **entry-information-selection** 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 -- à utiliser si le type a plusieurs
              valeurs --,
count     [1] INTEGER (1 .. ub-attribute-values) OPTIONAL -- à utiliser si le type a plusieurs
              valeurs -- }

```

Les composants du paramètre sélection d'attribut **attribute-selection** ont la signification suivante:

- 1) type **Type** (M): indique le type d'attribut de l'attribut;
- 2) à partir de **From** (O): lorsqu'un attribut a plusieurs valeurs, ce nombre entier donne la position relative de la première valeur à renvoyer. S'il spécifie une valeur postérieure à celles qui sont présentes dans l'attribut, aucune valeur n'est renvoyée. Ce composant ne peut être 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) décompte **Count** (O): 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 de décompte **count** 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 renvoyées n'est pas limité.

8.1.5 Information d'entrée (Entry-information)

Un paramètre information d'entrée **entry-information** 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 information d'entrée **entry-information** ont la signification suivante:

- 1) numéro d'ordre **Sequence-number** (M): numéro d'ordre identifiant l'entrée (voir le § 6.3.2.2);
- 2) attributs **Attributes** (O): ensemble des attributs sélectionnés de l'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 (Summarize abstract-operation)

L'opération abstraite de résumé **Summarize abstract-operation** renvoie les décomptes résumés des entrées sélectionnées dans une base d'information. En plus de ces résumés, l'opération renvoie également un décompte 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 de résumé **Summarize abstract-operation** 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 de résumé.

8.2.1 *Argument de résumé (Summarize-argument)*

```
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 si aucun résumé n'est demandé -- }
```

Les composants de argument de résumé **summarize-argument** ont la signification suivante:

- 1) type de base d'information **Information-base-type** (O): 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 **information-base-type**, la valeur par défaut est messages enregistrés;
- 2) sélecteur **Selector** (M): ensemble de critères de sélection pour déterminer les entrées qui devront être résumées (voir le § 8.1.3);
- 3) demandes de résumés **Summary-requests** (O): 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 résumé (Summarize-result)*

Si la demande aboutit, le résultat de résumé **summarize-result** sera renvoyé.

```
SummarizeResult ::= SET {  
    next [0] SequenceNumber OPTIONAL,  
    count [1] INTEGER (0 .. ub-messages) -- décompte des entrées sélectionnées --,  
    span [2] Span OPTIONAL -- décompte des entrées sélectionnées, supprimé si nul --,  
    summaries [3] SEQUENCE SIZE (1 .. ub-summaries) OF Summary OPTIONAL }
```

Les composants de résultat de résumé **summarize-result** ont les significations suivantes:

- 1) suivant **Next** (C): ce composant 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. Ce composant contient le numéro d'ordre de l'entrée suivante qui aurait alors été sélectionnée;
- 2) décompte **Count** (M): il s'agit d'un nombre entier donnant le décompte des entrées répondant aux critères de sélection;
- 3) portée **Span** (C): ce composant 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 portée **span** ont la signification suivante:

- a) inférieur **Lowest** (M): il s'agit du point de départ de portée **span**, exprimé sous forme d'un numéro d'ordre (voir le § 6.3.2.2);
 - b) supérieur **Highest** (M): il s'agit du point final de portée **span**, exprimé sous forme d'un numéro d'ordre (voir le § 6.3.2.2);
- 4) résumés **Summaries** (C): un seul résumé **summary** est renvoyé pour chaque demande de résumé **summary-request**. Les résumés **summaries** sont renvoyés dans l'ordre dans lequel ils ont été demandés.

```
Summary ::= SET {  
    absent [0] INTEGER (1 .. ub-messages) OPTIONAL -- décompte des entrées pour lesquelles  
    l'attribut est absent --,  
    present [1] SET SIZE (1 .. ub-attribute-values) OF -- un résumé pour chaque valeur d'attribut  
    présente --  
    SEQUENCE {  
        type AttributeType,  
        value ANY DEFINED BY type,  
        count INTEGER (1 .. ub-messages) } OPTIONAL }
```

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

- a) **absent** (C): décompte des entrées qui ne contiennent pas un attribut du type d'attribut spécifié dans la demande. Ce composant est omis s'il n'y a pas d'entrées de ce type;
- b) **présent present** (C): résumé des entrées contenant un attribut du type d'attribut spécifié, ventilé selon les valeurs d'attributs réellement présentes. Ce composant est omis s'il n'y a pas d'entrées de ce type.

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

- i) type **Type** (M): type d'attribut;
- ii) valeur **Value** (M): valeur d'attribut pour laquelle le compte est donné;
- iii) décompte **Count** (M): décompte des entrées auxquelles est associée cette valeur d'attribut.

8.2.3 Erreurs abstraites de résumé (*Summarize abstract-errors*)

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 de listage (*List abstract-operation*)

L'opération abstraite de listage **List abstract-operation** 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 de listage **List abstract-operation** n'aboutira 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 de listage (*List-argument*)

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

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

- 1) type de base d'information **Information-base-type** (O): 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 **information-base-type**, la valeur par défaut est messages enregistrés;

- 2) sélecteur **Selector** (M): ensemble de critères de sélection pour déterminer les entrées qui seront renvoyées (voir le § 8.1.3);
- 3) attributs demandés **Requested-attributes** (O): indique les informations des entrées sélectionnées qui devront être renvoyées dans le résultat (voir le § 8.1.4).

Si ce paramètre est absent, l'ensemble enregistré des listes de valeurs par défaut d'attributs **list-attribute-defaults** est utilisé. Se reporter au § 8.6.1 pour de plus amples informations sur ces valeurs par défaut.

8.3.2 *Résultat de listage (List-result)*

Si la demande aboutit, le paramètre résultat de listage **list-result** sera renvoyé.

```
ListResult ::= SET {  
  next [0] SequenceNumber OPTIONAL,  
  requested [1] SEQUENCE SIZE (1 .. ub-messages) OF EntryInformation OPTIONAL  
  -- composant absent si aucune entrée n'est trouvée -- }
```

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

- 1) entrée suivante **Next** (C): ce composant est renvoyé lorsque le nombre d'entrées sélectionnées aurait été plus grand en l'absence de la limite spécifiée dans le sélecteur. Ce composant contient le numéro d'ordre de l'entrée suivante qui aurait été sélectionnée;
- 2) informations demandées **Requested** (C): ce composant 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 *Erreurs abstraites de listage (List abstract-errors)*

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 d'extraction (Fetch abstract-operation)*

L'opération abstraite d'extraction **Fetch abstract-operation** sert à renvoyer des informations sélectionnées 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 d'extraction **Fetch abstract-operation** n'aboutit que si l'on demande des bases d'information autorisées par le contexte de sécurité et par 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 de suppression.

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

8.4.1 *Argument d'extraction (Fetch-argument)*

```
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 d'extraction **fetch-argument** ont la signification suivante:

- 1) type de base d'information **Information-base-type** (O): ce composant 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 **information-base-type**, la valeur par défaut est messages enregistrés;

- 2) **item** (M): l'un des composants décrits ci-après doit être spécifié pour déterminer l'entrée à extraire:
 - a) recherche **Search** (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) arrêt **Precise** (C): il s'agit du numéro d'ordre de l'entrée à extraire (voir le § 6.3.2.2);
- 3) attributs demandés **Requested-attributes** (O): ce composant 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 listes des valeurs par défaut d'extraction **fetch-attribute-defaults** est utilisé. On se reportera au § 8.6.1 pour avoir de plus amples renseignements sur ces valeurs par défaut.

8.4.2 *Résultat d'extraction (Fetch-result)*

Si la demande aboutit, le paramètre résultat d'extraction **fetch-result** doit être renvoyé.

FetchResult ::= SET {

entry-information	[0]	EntryInformation OPTIONAL -- si une entrée a été sélectionnée --,
list	[1]	SEQUENCE SIZE (1 .. ub-messages) OF SequenceNumber OPTIONAL,
next	[2]	SequenceNumber OPTIONAL }

Les composants du paramètre résultat d'extraction **fetch-result** ont la signification suivante:

- 1) information-entrée **Entry-information** (C): il s'agit de l'ensemble de tous les attributs qui sont présents dans l'entrée sélectionnée (voir le § 8.1.5). Ce composant n'est pas présent lorsqu'une recherche a été effectuée et qu'aucune entrée n'a été sélectionnée;
- 2) listage **List** (C): ce composant 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) entrée suivante **Next** (C): ce composant est renvoyé lorsque le nombre d'entrées sélectionnées aurait été plus grand en l'absence de la limite spécifiée dans le sélecteur. Ce composant comprend le numéro d'ordre de l'entrée suivante qui aurait été sélectionnée.

8.4.3 *Erreurs abstraites d'extraction (Fetch abstract-errors)*

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 de suppression (Delete abstract-operation)*

L'opération abstraite de suppression **Delete abstract-operation** 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 de suppression **Delete abstract-operation** n'aboutira que dans le cas de bases d'information autorisées par le contexte de sécurité et par 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. On ne doit pas tenir compte de l'état d'entrée d'une quelconque entrée secondaire associée à une entrée principale lors de l'exécution de l'opération abstraite de suppression **Delete abstract-operation**.

```

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

```

8.5.1 *Argument de suppression (Delete-argument)*

```

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 argument de suppression **delete-argument** ont la signification suivante:

- 1) type de base d'information **Information-base-type** (O): 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 **information-base-type**, la valeur par défaut est messages enregistrés;
- 2) **items** (M): l'un des composants décrits ci-après doit être spécifié pour déterminer les entrées à supprimer:
 - a) sélecteur **Selector** (C): voir le § 8.1.3;
 - b) numéros d'ordre **Sequence-numbers** (C): liste non ordonnée de numéros d'ordre **sequence-numbers**. Voir le § 6.3.2.2.

8.5.2 *Résultat de suppression (Delete-result)*

Si la demande aboutit, le résultat de suppression **delete-result** sera renvoyé. Il n'y a pas de paramètres.

```
DeleteResult ::= NULL
```

8.5.3 *Erreurs abstraites de suppression (Delete abstract-errors)*

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 d'enregistrement MS (Register-MS abstract-operation)*

L'opération abstraite d'enregistrement MS **Register-MS abstract-operation** 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) nouveaux pouvoirs;
- 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 d'enregistrement MS (Register-MS-argument)

```

Register-MS-Argument ::= 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] Credentials,
    new-credentials [1] Credentials } OPTIONAL
    -- même CHOIX (CHOICE) que pour les anciens pouvoirs --,
  user-security-labels [5] SET SIZE (1 .. ub-labels-and-redirections) OF SecurityLabel
OPTIONAL }

```

Les composants de argument d'enregistrement MS **register-MS-argument** ont la signification suivante:

- 1) enregistrements d'actions automatiques **Auto-action-registrations** (O): ensemble d'enregistrements d'actions automatiques **auto-action-registrations** (voir le § 6.5.1), un enregistrement pour chaque action automatique. Le nouveau paramètre d'enregistrement **registration-parameter** d'actions automatiques annule toute action automatique précédemment enregistrée (le cas échéant) à l'aide de l'identificateur d'enregistrement **registration-identifiant** et du **type** d'action automatique.

En l'absence du composant d'enregistrements d'actions automatiques **auto-action-registrations**, la valeur par défaut est pas de nouvelles actions automatiques enregistrées;

- 2) désenregistrements d'actions automatiques **Auto-action-deregistrations** (O): ensemble de désenregistrements d'actions automatiques **auto-action-deregistrations**, un désenregistrement pour chaque action automatique à désenregistrer. Toute action automatique ayant un identificateur de désenregistrement **registration-identifiant** et un type d'action automatique **auto-action-type** conformes à ceux qui sont spécifiés dans le composant désenregistrement d'action automatique **auto-action-deregistration** est désenregistrée.

```

AutoActionDeregistration ::= SEQUENCE {
  type AutoActionType,
  registration-identifiant [0] INTEGER (1 .. ub-per-auto-action) DEFAULT 1 }

```

En l'absence du composant désenregistrement d'action automatique **auto-action-deregistrations**, la valeur par défaut est aucune action automatique enregistrée n'est désenregistrée;

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

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

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

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

- 5) modification des pouvoirs **Change-credentials** (O): anciens et nouveaux pouvoirs si une opération modification des pouvoirs **change-credentials** est demandée.

Les anciens pouvoirs **old-credentials** sont ceux de l'utilisateur final actuel; les nouveaux pouvoirs **new-credentials** sont ceux que l'utilisateur final souhaiterait obtenir.

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 **initiator-credentials** spécifiés au § 8.1.1.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;

- 6) étiquettes de sécurité de l'utilisateur **User-security-labels** (O): ce composant contient les étiquettes de sécurité **security-label(s)** de l'utilisateur du service abstrait MS, si elles doivent être modifiées. Ce composant peut être créé par l'utilisateur du service abstrait MS.

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

A noter que, selon certaines mesures de sécurité **security-policies**, les étiquettes de sécurité de l'utilisateur **user-security-labels** 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 **user-security-labels** de façon sûre peuvent être fournis. Les étiquettes de sécurité de l'utilisateur **user-security-labels** sont définies au § 8.4.1.1.7 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

L'étiquette de sécurité **security-label** est définie au § 9 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

8.6.2 *Résultat d'enregistrement MS (Register-MS-result)*

Si la demande aboutit, le résultat d'enregistrement MS **register-MS-result** sera renvoyé. Il n'y a pas de paramètres.

Register-MSResult ::= NULL

8.6.3 *Erreurs abstraites d'enregistrement MS (Register-MS abstract-errors)*

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 d'avertissement (Alert abstract-operation)*

L'opération abstraite d'avertissement **Alert abstract-operation** permet au fournisseur 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'avertissement automatique **auto-alert-registrations** (voir le § 12.2), précédemment fournis à l'aide de l'opération abstraite d'enregistrement MS (voir le § 8.6).

L'opération abstraite d'avertissement **Alert abstract-operation** 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 de rattachement abstrait pour l'association abstraite. Aucune opération abstraite d'avertissement **Alert abstract-operation** ne doit être appelée pour ces entrées (voir le § 7).

L'opération abstraite d'avertissement **Alert abstract-operation** 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 d'avertissement (Alert-argument)*

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

Les composants de l'argument d'avertissement **alert-argument** ont la signification suivante:

- 1) identificateur d'enregistrement d'avertissement **Alert-registration-identif**ier (M): identifie lequel des enregistrements d'avertissement automatique **auto-alert-registration** a donné lieu à l'avertissement (voir les § 6.5 et 12.2);
- 2) nouvelle entrée **New-entry** (O): achemine l'information provenant de la nouvelle entrée qui a été demandée dans le paramètre d'enregistrement d'avertissement automatique **auto-alert-registration-parameter** (voir le § 12.2). Ce composant est absent lorsque l'utilisateur du service abstrait MS n'a pas spécifié les attributs demandés **requested-attributes** dans le paramètre d'enregistrement d'avertissement automatique **auto-alert-registration-parameter**.

8.7.2 *Résultat d'avertissement (Alert-result)*

Si la demande aboutit, le résultat d'avertissement **alert-result** sera renvoyé.

AlertResult ::= NULL

8.7.3 *Erreurs abstraites d'avertissement (Alert abstract-errors)*

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 (Abstract-errors)

Ce paragraphe définit les erreurs abstraites **abstract-errors** suivantes, associées à l'utilisation d'opérations abstraites à l'accès de recherche:

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

9.1 *Priorité des erreurs*

Le responsable d'une opération abstraite (*abstract-operation*) 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 (*abstract-operation*), étant donné qu'il n'y a pas nécessairement d'ordre logique particulier pour traiter ces erreurs.

9.2 *Erreur d'attribut (Attribute-error)*

Une erreur d'attribut **Attribute-error** 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) problèmes **Problems** (M): problèmes particuliers rencontrés. Un nombre quelconque de problèmes spécifiques peut être signalé; 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) type d'attribut non valide **Invalid-attribute-value** (C): une valeur visée d'attribut, spécifiée comme argument de l'opération abstraite (*abstract-operation*), n'est pas conforme au type de données défini pour le type d'attribut en question;
- b) type d'attribut indisponible **Unavailable-attribute-type** (C): un type visé d'attribut, utilisé comme argument de l'opération abstraite (*abstract-operation*), ne fait pas partie de ceux qui sont gérés par le fournisseur du service abstrait MS. Si le fournisseur du service abstrait MS peut néanmoins effectuer l'opération, il n'en est pas empêché;
- c) concordance non appropriée **Inappropriate-matching** (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) type d'attribut non souscrit **Attribute-type-not-subscribed** (C): un type d'attribut utilisé comme argument de l'opération abstraite (*abstract-operation*) 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) inapproprié pour l'utilisation **Inappropriate-for-operation** (C): un type d'attribut utilisé comme argument de l'opération abstraite (*abstract-operation*) ne convient pas pour l'utilisation requise.

9.3 *Erreur de demande d'action automatique (Auto-action-request-error)*

Une erreur de demande d'action automatique **Auto-action-request-error** 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] AutoActionRequestProblem,
type [1] ActionType } }

AutoActionRequestProblem ::= INTEGER {

unavailable-auto-action-type (0),
auto-action-type-not-subscribed (1) } (0 .. ub-error-reasons)

Ce paramètre a la signification suivante:

- 1) problèmes **Problems** (M): 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) type d'action automatique indisponible **Unavailable-auto-action-type**: un type d'action automatique utilisé comme argument de l'opération abstraite (*abstract-operation*) ne fait pas partie de ceux qui sont assurés par le fournisseur du service abstrait MS;
 - b) type d'action non souscrit **Action-type-not-subscribed**: un type d'action utilisé comme argument de l'opération abstraite (*abstract-operation*) ne fait pas partie de ceux auxquels l'utilisateur du service abstrait MS s'est abonné.

9.4 Erreur de suppression (*Delete-error*)

Un paramètre erreur de suppression **Delete-error** 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) problèmes **Problems** (M): 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) entrée secondaire spécifiée **Child-entry-specified**: une tentative a été faite pour supprimer une entrée secondaire;
 - b) problème de restriction de suppression **Delete-restriction-problem**: on a essayé de transgresser une restriction spécifiée pour l'opération abstraite (*abstract-operation*) suppression (voir le § 8.5).

9.5 Erreur de restriction d'extraction (*Fetch-restriction-error*)

Un paramètre erreur de restriction d'extraction **Fetch-restriction-error** indique une tentative de transgression d'une restriction liée à l'opération abstraite (*abstract-operation*) extraction.

FetchRestrictionError ::= ABSTRACT-ERROR

PARAMETER SET {

problems	[0]	SET SIZE (1 .. ub-default-registrations) OF SET {
problem	[3]	FetchRestrictionProblem,
restriction	CHOICE {	
content-type	[0]	OBJECT IDENTIFIER,
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)

Ce paramètre a la signification suivante:

- 1) problèmes **Problems** (M): problèmes particuliers rencontrés. Un nombre quelconque de problèmes particuliers peut être signalé. Avec chaque problème, on indique le type de contenu, le type d'information codée ou la longueur du contenu à l'origine du problème:
 - a) problème de type de contenu **Content-type-problem** (C): le type de contenu du message extrait est interdit par les restrictions d'extraction **fetch-restrictions** en vigueur;
 - b) problème d'EIT **EIT-problem** (C): les types de codages requis dans l'opération abstraite (*abstract-operation*) d'extraction sont interdits par les restrictions d'extraction **fetch-restrictions** en vigueur;
 - c) problème de longueur de contenu **Content-length-problem** (C): la longueur de contenu du message extrait est supérieure à celle qui est autorisée par les restrictions d'extraction **fetch-restrictions** en vigueur.

9.6 Erreur de paramètres non valides (*Invalid-parameters-error*)

Un paramètre erreur de paramètres non valides **Invalid-parameters-error** indique un problème sémantique dans l'ensemble des paramètres reçus. Cette erreur sera 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 d'étendue (*Range-error*)

Un paramètre erreur d'étendue **Range-error** indique un problème lié à l'étendue spécifiée dans un sélecteur comme argument d'une opération abstraite (*abstract-operation*).

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

Ce paramètre a la signification suivante:

- 1) problèmes **Problems** (M): problèmes particuliers rencontrés:
 - a) sens inverse **Reversed** (C): la limite supérieure a indiqué un numéro d'ordre ou une date-heure de création antérieur(e) à ce qui a été indiqué par la limite inférieure.

9.8 Erreur de sécurité (*Security-error*)

Le paramètre erreur de sécurité **Security-error** indique que l'opération abstraite (*abstract-operation*) demandée ne peut être assurée car elle transgresserait les mesures de sécurité en vigueur. Cette erreur est définie dans la Rec. X.411 du CCITT | ISO/CEI 10021-4.

9.9 Erreur de numéro d'ordre (*Sequence-number-error*)

Un paramètre erreur de numéro d'ordre **Sequence-number-error** indique un problème lié au numéro d'ordre spécifié dans un argument d'une opération abstraite (*abstract-operation*).

```
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) problèmes **Problems** (M): 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:
nulle entrée semblable **No-such-entry**: le numéro d'ordre fourni ne correspond à aucune entrée de la base d'information.

9.10 *Erreur de service (Service-error)*

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

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 **general-attribute-types** spécifiés pour le MS. Le § 12 définit les types d'actions automatiques générales **general-auto-action-types** spécifiés pour le MS.

11 Types d'attributs généraux (General-attribute-types)

Les types d'attributs généraux **general-attribute-types** sont valables pour tous les types de contenu de message. D'autres types d'attributs (*attribute-types*), 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 de la Rec. X.420 du CCITT | ISO/CEI 10021-7.

11.1 Présentation générale des types d'attributs généraux

Les attributs généraux **general-attributes** qui peuvent apparaître dans une entrée de base d'information (*information-base*) 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 Rec. X.411 du CCITT | ISO/CEI 10021-4. Ces attributs sont désignés en conséquence. Certains attributs généraux **general-attributes** sont créés, certains également mis à jour par la MS.

Le tableau 1/X.413 définit les divers attributs généraux **general-attributes** 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 géré ou non par le 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 une entrée de, respectivement, message remis, rapport remis ou contenu renvoyé;
- si le type d'attribut peut ou non être renvoyé dans une opération abstraite listage ou d'avertissement;
- si le type d'attribut peut être utilisé ou non dans une opération abstraite de 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.

Les types d'attribut définis comme étant présents dans une entrée de message remis peuvent ne pas être toujours présents dans une entrée secondaire. Les règles régissant la présence de types d'attribut dans des entrées secondaires pourront faire l'objet de compléments dans la Recommandation définissant le type de contenu de l'entrée secondaire.

Tous les attributs gérés sont, sous réserve d'abonnement, soumis à l'opération abstraite d'extraction.

11.2 Description des types d'attributs généraux

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

A noter que certains attributs généraux **general-attributes** 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 **general-attributes** conviennent pour les résumés.

TABLEAU 1/X.413

Types d'attributs généraux pour la base d'information de remise

Nom du type d'attribut	A valeur unique/à plusieurs valeurs	Niveau de fourniture par le MS et accès-UA	Présence dans l'entrée de message remis	Présence dans l'entrée de rapport remis	Présence dans l'entrée de contenu renvoyé	Disponible pour listage et avertissement	Disponible pour résumé
Child-sequence-numbers	M	M	C	C	C	Y	N
Content	S	M	P	—	P	N	N
Content-confidentiality-algorithm-identifier	S	O	C	—	—	Y	N
Content-correlator	S	O	—	C	—	Y	N
Content-identifier	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-identifier	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	—	Y	N
Security-classification	S	O	C	C	—	Y	Y
Sequence-number	S	M	P	P	P	Y	N
Subject-submission-identifier	S	M	—	P	—	Y	N
This-recipient-name	S	O	P	—	—	Y	N

11.2.1 *Numéros d'ordre secondaire (Child-sequence-numbers)*

Cet attribut général, à plusieurs valeurs, contient un ou plusieurs «pointeurs» indiquant, le cas échéant, le niveau suivant des entrées secondaires. Il est créé par le MS. Cet attribut est présent dans une entrée maîtresse 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-numbers ATTRIBUTE
WITH ATTRIBUTE-SYNTAX SequenceNumber
MULTI VALUE
::= id-att-child-sequence-numbers

11.2.2 *Contenu (Content)*

Cet attribut général contient le contenu **content** complet d'un message tel qu'il est remis à l'aide de l'opération abstraite de remise de messages ou sous forme de contenu renvoyé **returned-content** à l'aide de l'opération abstraite de remise de rapports. Dans ce dernier cas, l'attribut général de contenu est créé dans l'entrée secondaire, de contenu renvoyé et non pas dans l'entrée de rapport remis proprement dite. On trouvera de plus amples détails aux § 8.2.1.1.1.37 et 8.3.1.2.1.14 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.3 *Identificateur d'algorithme de confidentialité du contenu (Content-confidentiality-algorithm-identifier)*

Cet attribut général contient l'identificateur d'algorithme **algorithm-identifiant** 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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.4 *Corrélateur de contenu (Content-correlator)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.5 *Identificateur de contenu (Content-identifiant)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.6 *Vérification d'intégrité du contenu (Content-integrity-check)*

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 § 8.2.1.1.1.28 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.7 *Longueur du contenu (Content-length)*

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 de remise de messages, ou d'un contenu renvoyé (le cas échéant) notifié par l'opération abstraite de remise de rapports. Lorsqu'il n'y a pas de contenu renvoyé, cet attribut est absent. Il est créé par le MS.

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

11.2.8 *Contenu renvoyé (Content-returned)*

Cet attribut général indique si le contenu d'un message a été renvoyé à l'aide de l'opération abstraite de 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 (Content-type)*

Cet attribut général est créé à partir du type de contenu à l'aide de l'opération abstraite de remise de messages ou de remise de rapports. Se reporter également au § 8.2.1.1.1.34 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Conversion-with-loss-prohibited)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4 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 d'information codée convertis (Converted EITs)*

Cet attribut général, qui est à plusieurs valeurs, identifie les types d'information codée du contenu après conversion indiqués par l'opération abstraite de remise de messages. Il est créé d'après les types d'information codée convertis issus de l'opération abstraite de remise de messages. 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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Creation-time)*

Cet attribut général indique la date et l'heure auxquelles l'entrée a été créée dans le MS. Il est produit par le MS.

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 d'information codée remis (Delivered EITs)*

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

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

11.2.14 *Indicateurs de remise (Delivery-flags)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (DL-expansion-history)*

Cet attribut général, à plusieurs valeurs, est utilisé pour décrire la chronologie de l'allongement de la liste de distribution. S'il est présent dans un message remis, il contient un ou plusieurs noms de liste de distribution utilisés pendant le processus d'allongement. Si toutefois il est présent dans un rapport remis, il contient le nom de l'expéditeur et un ou plusieurs noms de liste de distribution utilisés pendant le processus d'allongement. Il est absent si le message en question, sur la base duquel la remise de rapport est fondée, n'a nécessité aucun allongement de la liste de distribution. On trouvera de plus amples détails au § 8.3.1.1.1.7 et au 8.3.1.2.1.3 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.16 *Etat de l'entrée (Entry-status)*

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 (Entry-type)*

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 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)

11.2.18 *Nom du destinataire prévu (Intended-recipient-name)*

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 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Message-delivery-envelope)*

Cet attribut général contient l'enveloppe de remise de message **message-delivery-envelope** complet 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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.20 *Identificateur de remise de message (Message-delivery-identifiant)*

Cet attribut général contient l'identificateur de remise de message **message-delivery-identifiant** provenant de l'opération abstraite de remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.21 *Date et heure de remise de message (Message-delivery-time)*

Cet attribut général contient la date et l'heure de remise de message **message-delivery-time** provenant de l'opération abstraite de remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

Remarque — Il n'y a pas d'attribut général correspondant au paramètre date heure de remise de l'opération abstraite de remise de rapports car, pour être utile, ce paramètre doit être mis en correspondance avec le nom du destinataire auquel le message a été remis. Ces informations sont incluses dans l'attribut champs de remise de rapports par destinataire général (*per-recipient-report-delivery-fields*).

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 (Message-origin-authentication-check)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Message-security-label)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Message-submission-time)*

Cet attribut général contient la date et l'heure de dépôt du message **message-submission-time** provenant de l'opération abstraite remise de messages. On trouvera de plus amples détails au § 8.2.1.1.2.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.25 *Jeton de message (Message-token)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.26 *Types d'information codée d'origine (Original-EITs)*

Cet attribut général, à plusieurs valeurs, identifie les types d'information codée **encoded-information-types** contenus dans le message déposé. Il est créé d'après les types d'information codée d'origine **original-encoded-information-types** issus de l'opération abstraite de remise de messages ou de remise de rapports. On trouvera de plus amples détails au § 8.2.1.1.1.33 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Originator-certificate)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.28 *Nom de l'expéditeur (Originator-name)*

Cet attribut général contient le nom O/R de l'expéditeur provenant de l'opération abstraite de remise de messages. On trouvera de plus amples détails au § 8.2.1.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.29 *Noms des autres destinataires (other-recipient-names)*

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 de remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.6 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Parent-sequence-number)*

Cet attribut général est associé à une entrée maîtresse. 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 (Per-recipient-report-delivery-fields)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.32 *Priorité (Priority)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.33 *Demande de preuve de remise (Proof-of-delivery-request)*

Cet attribut général indique si l'expéditeur du message exige une preuve de remise **proof-of-delivery** 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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.34 *Chronologie du réacheminement (Redirection-history)*

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. On trouvera de plus amples détails au § 8.3.1.1.1.5 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

mt-redirection-history ATTRIBUTE
WITH ATTRIBUTE-SYNTAX Redirection
MULTI VALUE
::= id-att-redirection-history.

11.2.35 *Enveloppe de remise de rapport (Report-delivery-envelope)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Reporting-DL-name)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.37 *Indication de certificat MTA (Reporting-MTA-certificate)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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 (Report-origin-authentication-check)*

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 Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.39 *Classification de sécurité (Security-classification)*

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 de résumé. On trouvera de plus amples détails au § 8.5.9 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.40 *Numéro d'ordre (Sequence-number)*

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 (Subject-submission-identifier)*

Cet attribut général contient l'identificateur de dépôt du message ou de l'identificateur de dépôt d'essai au sujet du rapport. On trouvera de plus amples détails au § 8.3.1.2.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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

11.2.42 *Nom de ce destinataire (This-recipient-name)*

Cet attribut général contient le nom O/R de ce destinataire (MS) de l'opération abstraite de remise de messages. On trouvera de plus amples détails au § 8.3.1.1.1.3 de la Rec. X.411 du CCITT | ISO/CEI 10021-4.

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.

11.4 *Abonnement aux types d'attributs*

L'abonnement aux types d'attributs relève des autorités locales. 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 relève des autorités locales. 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.

TABLEAU 2/X.413

Création des types d'attributs généraux

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Paramètre source	Paramètre source créé par	Règles de création
Child-sequence-numbers	M	—	MS	Une valeur est créée pour chaque entrée secondaire correspondant à une entrée maîtresse
Content	S	content returned-content	MD RD	La valeur d'attribut dans l'entrée de message remis est celle du paramètre source. La valeur d'attribut est dans l'entrée secondaire de contenu de l'entrée de rapport remis renvoyé est la valeur du paramètre source
Content-confidentiality-algorithm-identifier	S	content-confidentiality-algorithm-identifier	MD	La valeur de l'attribut est celle du paramètre source
Content-correlator	S	content-correlator	RD	La valeur de l'attribut est celle du paramètre source
Content-identifier	S	content-identifier	MD RD	La valeur de l'attribut est celle du paramètre source
Content-integrity-check	S	content-integrity-check	MD	La valeur de l'attribut est celle du paramètre source
Content-length	S	—	MS	La taille (approximative) en octets du contenu mis en mémoire, d'après le contenu remis ou le contenu renvoyé
Content-returned	S	—	MS	La valeur est mise sur vrai true si le paramètre contenu renvoyé est présent dans l'opération remise de rapports et sur faux false si ce paramètre n'est pas présent
Content-type	S	content-type	MD RD	La valeur du paramètre, si elle est représentée par OBJECT IDENTIFIER. Si elle est représentée par INTEGER, convertir à l'OBJECT IDENTIFIER correspondant
Conversion-with-loss-prohibited	S	conversion-with-loss-prohibited	MD	La valeur de l'attribut est celle du paramètre source
Converted-EITs	M	converted-encoded-information-types	MD	Une valeur correspondante est créée pour chaque bit mis à 1 dans le paramètre built-in-encoded-information-types (types d'information codée intégrés) et à partir de chaque ExternalEncoded InformationType (type d'information codée externe) présent dans le paramètre external-encoded-information-type (type d'information codée externe)
Creation-time	S	—	MS	La date et l'heure de création de l'entrée
Delivered-EITs	M	Converted-EITs and Original-EITs general attributes, and converted-encoded-information-types	MS	Types d'information codée convertis si présents, sinon types d'information codée d'origine
Delivery-flags	S	delivery-flags	MD	La valeur de l'attribut est celle du paramètre source. Si aucun indicateur de remise delivery-flag ne figure dans le MD, il est créé une valeur par défaut avec aucun ensemble d'indicateurs
DL-expansion-history	M	DL-expansion-history originator-and-DL-expansion-history	MD RD	Une valeur correspondante est créée à partir de chaque composant de la SEQUENCE Une valeur correspondante est créée à partir de chaque composant de la SEQUENCE

TABLEAU 2/X.413 (suite)

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Paramètre source	Paramètre source créé par	Règles de création
Entry-status	S	—	MS	Créé lorsque l'entrée est créée avec la valeur nouveau «new»
Entry-type	S	MessageDelivery ARGUMENT ReportDelivery ARGUMENT	MS MS	La valeur est mise à remise de messages «delivery-message» La valeur est mise à remise de rapport «delivery-report». 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 «returned-content»
Intended-recipient-name	S	intended-recipient-name	MD	La valeur de l'attribut est celle du paramètre source
Message-delivery-envelope	S	Message Delivery Envelope	MD	La valeur de l'attribut est celle du paramètre source
Message-delivery-identifiant	S	message-delivery-identifiant	MD	La valeur de l'attribut est celle du paramètre source
Message-delivery-time	S	message-delivery-time	MD	La valeur de l'attribut est celle du paramètre source
Message-origin-authentication-check	S	message-origin-authentication-check	MD	La valeur de l'attribut est celle du paramètre source
Message-security-label	S	message-security-label	MD RD	La valeur de l'attribut est celle du paramètre source
Message-submission-time	S	message-submission-time	MD	La valeur de l'attribut est celle du paramètre source
Message-token	S	message-token	MD	La valeur de l'attribut est celle du paramètre source
Original-EITs	M	original-encoded-information-types	MD RD	Une valeur correspondante est créée pour chaque bit mis à 1 dans le paramètre built-in-encoded-information-types (types d'information codée intégrés) et à partir de chaque ExternalEncoded InformationType (types d'information codée externes) présent dans le paramètre ExternalEncoded-information-types (types d'information codée externes)
Originator-certificate	S	originator-certificate	MD	La valeur de l'attribut est celle du paramètre source
Originator-name	S	originator-name	MD	La valeur de l'attribut est celle d'attribut source
Other-recipient-names	M	other-recipient-names	MD	Une valeur correspondante est créée pour chaque composant de la SEQUENCE
Parent-sequence-number	S	—	MS	Lorsque l'on crée une entrée secondaire, cet attribut est créé avec le numéro d'ordre correspondant de l'entrée maîtresse comme valeur
Per-recipient-report-delivery-fields	M	per-recipient-fields	RD	Une valeur correspondante est créée à partir de chaque composant de la SEQUENCE
Priority	S	priority	MD	La valeur de l'attribut est celle du paramètre source
Proof-of-delivery-request	S	proof-of-delivery-request	MD	La valeur de l'attribut est celle du paramètre source
Redirection-history	M	redirection-history	MD	Une valeur correspondante est créée à partir de chaque composant de la SEQUENCE
Report-delivery-envelope	S	Report Delivery Envelope	RD	La valeur de l'attribut est celle du paramètre source

TABLEAU 2/X.413 (fin)

Nom du type d'attribut	A valeur unique/ à plusieurs valeurs	Paramètre source	Paramètre source créé par	Règles de création
Reporting-DL-name	S	reporting-DL-name	RD	La valeur de l'attribut est celle du paramètre source
Reporting-MTA-certificate	S	reporting-MTA-certificate	RD	La valeur de l'attribut est celle du paramètre source
Report-origin-authentication-check	S	report-origin-authentication-check	RD	La valeur de l'attribut est celle du paramètre source
Security-classification	S	security-classification	MD RD	La valeur de l'attribut est celle du paramètre source
Sequence-number	S	—	MS	Lorsque l'on crée une entrée, le MS attribue une valeur unique à cet attribut, par ordre croissant
Subject-submission-identifier	S	subject-submission-identifier	RD	La valeur de l'attribut est celle du paramètre source
This-recipient-name	S	this-recipient-name	MD	La valeur de l'attribut est celle du paramètre source

Remarque — Lorsqu'une entrée de remise de message est créée, il n'y a pas d'attributs généraux distincts créés pour les arguments remise physique (Physical Delivery) et méthode de remise (Delivery Method) 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 message (message-delivery-envelope).

12 Types d'actions automatiques générales (General-auto-action-types)

Les types d'actions automatiques générales **general-auto-action-types** 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 leurs Recommandations particulières; à titre d'exemple, la procédure spécifique au service de messagerie de personne à personne (IPMS) pour le type d'action générale retransmission automatique **auto-forward general-auto-action-type** est décrite au § 19.4 de la Rec. X.420 du CCITT | ISO/CEI 10021-7. D'autres types d'actions automatiques **auto-action-types** spécifiques du contenu peuvent être définis dans leurs Recommandations respectives.

Les actions automatiques **auto-actions**, présentés au § 6.5, sont enregistrés et désenregistrés à l'aide de l'opération abstraite d'enregistrement-MS décrite au § 8.6. En variante, les informations d'enregistrement peuvent être transférées au MS, par option d'abonnement.

Les types d'actions automatique générale **general-auto-action-types** suivants sont définis:

- a) retransmission automatique (*auto-forward*);
- b) avertissement automatique (*auto-alert*).

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

Les paragraphes suivants donnent une brève description de chaque type d'action automatique générale **general-auto-action-type** ainsi que de sa syntaxe abstraite, à l'aide de la macro **AUTO-ACTION** définie au § 6.5.

12.1 *Retransmission automatique (Auto-forward)*

L'action automatique retransmission **auto-forward auto-action** permet au fournisseur du service abstrait MS d'effectuer une retransmission **forward** automatique de 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.

Remarque — Dans la prochaine version de la présente Recommandation, l'action automatique retransmission automatique **auto-forward** sera classifiée comme spécifique des IPM et donc transférée dans la Rec. X.420 du CCITT | ISO/CEI 10021-7.

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

Le paramètre d'enregistrement de retransmission automatique **auto-forward-registration-parameter** spécifie si l'entrée principale (et toute entrée secondaire qui lui est associée) correspondant au message doit être supprimée après retransmission automatique **auto-forwarding**. Si l'un quelconque des paramètres manipulés 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 OPTIONAL,
 per-message-indicators PerMessageIndicators OPTIONAL
 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 composants du paramètre d'enregistrement de retransmission automatique **auto-forward-registration-parameter** ont la signification suivante:

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

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

- 2) **arguments de retransmission automatique Auto-forward-arguments (M)**: il s'agit d'un ensemble d'arguments enregistrés à utiliser pour chaque opération abstraite de dépôt de messages (voir le § 8.2.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4). Tout argument qui n'est ni enregistré ni obligatoire, ni mentionné spécifiquement ci-dessous, sera absent de chaque dépôt de messages.

Si le paramètre conversion avec perte interdite **conversion-with-loss-prohibited** est enregistré avec la valeur conversion avec perte autorisée **conversion-with-loss-allowed**, soit après enregistrement explicite de la valeur ou sans enregistrement mais en prenant cette valeur par défaut, la valeur utilisée pour chaque opération abstraite de dépôt de message doit être la valeur de l'argument correspondant de remise de message. S'il est enregistré avec la valeur conversion avec perte interdite **conversion-with-loss-prohibited**, cette valeur doit être utilisée pour chaque opération abstraite de dépôt de message.

Si le paramètre conversion implicite interdite **implicit-conversion-prohibited** est enregistré avec la valeur «zéro», indiquant que la conversion implicite est autorisée, ou si aucune valeur n'est enregistrée, la valeur utilisée pour chaque opération abstraite de dépôt de message doit être la valeur de l'argument

correspondant de remise de message. S'il est enregistré avec la valeur «one», indiquant que la conversion implicite est interdite, cette valeur doit être utilisée pour chaque opération abstraite de dépôt de message.

Si les arguments suivants ne sont pas enregistrés, leur présence en tant qu'arguments de dépôt de message dépend de la présence des arguments correspondants de remise de message, dont les valeurs seront transformées selon le cas, comme suit: jeton de message **message-token**, identificateur d'algorithme de confidentialité du contenu **content-confidentiality-algorithm-identifiant**, vérification d'intégrité du contenu **content-integrity-check**, vérification d'authentification d'origine de message **message-origin-authentication-check**, étiquette de sécurité de message **message-security-label** et priorité **priority**.

Certains arguments de dépôt de messages ne doivent pas être enregistrés, à savoir: demande de preuve de dépôt **proof-of-submission-request**, types d'information codée d'origine **original-encoded-information-types**, type de contenu **content-type** et contenu **content**.

- 3) **supprimer après retransmission automatique Delete-after-auto-forwarding** (O): 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) **autres paramètres Other-parameters** (O): 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 **auto-forwarding**.

Remarque — Ainsi, par exemple, dans le cas du service de messagerie de personne à personne (IPM), ce paramètre peut contenir l'observation de retransmission automatique **auto-forward-comment** renvoyée dans la notification 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 de retransmission automatique **auto-forward-comment** est décrite au § 19.4 de la Rec. X.420 du CCITT | ISO/CEI 10021-7.

La gestion de l'action automatique de retransmission automatique **auto-forward auto-action** par un MS, ou par un UA ayant accès au MS, suppose la gestion de l'enregistrement du paramètre d'enregistrement de retransmission automatique **auto-forward-registration-parameter** via l'opération abstraite d'enregistrement-MS.

12.2 Avertissement automatique (Auto-alert)

L'action automatique avertissement automatique **auto-alert auto-action** permet au fournisseur 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.

Le type d'action automatique avertissement automatique **auto-alert auto-action-type** permet d'enregistrer avec le MS un ou plusieurs ensembles de paramètres avertissement automatique **auto-alert**, chacun étant identifié par son identificateur d'enregistrement avertissement automatique **auto-alert-registration-identifiant**. Chaque paramètre d'enregistrement avertissement automatique **auto-alert-registration-parameter** spécifie des critères permettant de déterminer s'il s'applique à tel ou tel message au rapport 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 **alert abstract-operation** ne sera invoquée 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 avertissement automatique **auto-alert-registration parameter** ont la signification suivante:

- 1) **filtre Filter (O)**: il s'agit d'un ensemble de critères auxquels une nouvelle entrée représentant un rapport ou un message remis doit se conformer pour que le fournisseur du service abstrait MS retransmette automatiquement avertissement automatique **auto-alert** cette nouvelle entrée à l'aide de cet ensemble de paramètres.

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

- 2) **adresses d'avertissement Alert-addresses (O)**: cet argument identifie les types de services avertissement **alert** à invoquer, ainsi que toute information nécessaire pour accéder à des branches particulières de ces services avertissement **alert**, et toute autre information qui doit être acheminée pendant ces avertissements **alerts**.

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'usager du service abstrait MS et le prestataire du service abstrait MS), soit en signalant à l'aide d'un indicateur, dans l'opération abstraite de rattachement, la prochaine fois que l'utilisateur du service abstrait MS établira une association abstraite (voir le § 7). Si le paramètre des attributs requis est présent, l'utilisateur du service abstrait MS (UA) sera considéré comme faisant partie des adresses à avertir.

Certains types d'avertissements **alerts** 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 **alert-qualifier** contient toute autre information devant être acheminée pendant l'avertissement automatique **auto-alert**. 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) **attributs requis Requested-attributes (O)**: ce composant 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 **alert-registration-identifiant** sera présent dans l'argument d'avertissement **alert-argument**.

SECTION 4 — PROCÉDURES APPLICABLES À L'ENREGISTREMENT DE MESSAGES ET À LA MISE EN ŒUVRE DES ACCÈS

13 Présentation générale

La présente section 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

Le présent paragraphe indique comment un MS utilisera le service abstrait MTS défini au § 8 de la Rec. X.411 du CCITT | ISO/CEI 10021-4. 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 de remise de messages et de remise de rapports, ainsi que de l'invocation de l'appel de l'opération abstraite de 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). Ces 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 de remise de messages (MessageDelivery abstract-operation)*

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

- 1) Il crée une nouvelle entrée (et éventuellement une ou plusieurs entrées secondaires) et met à jour l'état d'entrée.
- 2) Il renvoie un résultat de remise de messages au MTA pour indiquer que la remise du message a eu lieu. Ce résultat de 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 Rec. X.411 du CCITT | ISO/CEI 10021-4.
- 3) 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. L'exécution d'une action automatique peut donner lieu à des changements d'état d'entrée, 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 par le MS au MTA à l'aide de l'opération abstraite de 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'une notification de non-réception) comme c'est le cas pour le service de messagerie de personne à personne (IPMS) décrit au § 19.4 de la Rec. X.420 du CCITT | ISO/CEI 10021-7. Selon les valeurs des arguments de l'opération abstraite d'enregistrement-MS pour la retransmission automatique, une copie du message remis peut être conservée dans le MS. Si le dépôt d'un message de retransmission automatique échoue, une copie est toujours conservée.

Remarque — Le traitement d'un résultat ou d'une erreur provenant du dépôt d'un tel message relève des autorités locales.

- b) Si des enregistrements d'avertissements automatiques ont été effectués à l'aide de l'opération abstraite d'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 «concordance», on essaie d'invoquer une opération abstraite d'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 en a pas, 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 d'avertissement a été menée à bien et il n'est plus procédé à de nouveaux enregistrements d'avertissement. Si aucun chemin 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 celui-ci 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.

- 4) 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 de 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 1) n'est réutilisé (pour ne pas contrevenir à de futures extensions d'enregistrement éventuels).

14.1.2 *Fonctionnement de l'opération abstraite de remise de rapports (ReportDelivery abstract-operation)*

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

- 1) il crée une nouvelle entrée (et éventuellement une ou plusieurs entrées secondaires) et met à jour l'état d'entrée;
- 2) il renvoie un résultat de remise de rapports au MTA pour indiquer que la remise a eu lieu. Le résultat de remise de rapports n'a pas de paramètres. On trouvera de plus amples détails au § 8.3.1.2.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 3) ensuite, toute action automatique ou autre procédure interne activée sera effectuée. Ces opérations sont soit générales ou variables selon le contenu. Dans le deuxième cas elles sont décrites dans les Recommandations pertinentes.

14.1.3 *Invocation de l'opération abstraite de commande de remise (DeliveryControl abstract-operation)*

Si le MS souhaite empêcher temporairement le MTA soit de transmettre des messages et des rapports ou 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 invoque une opération abstraite de commande de remise contenant les paramètres à modifier. On trouvera de plus amples détails au § 8.3.1.3 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 3) lorsque le MS est en mesure d'accepter de nouveau tout message et/ou tout rapport en attente, il doit invoquer une nouvelle opération abstraite de commande de remise pour annuler les restrictions. Les effets d'une opération abstraite de commande de remise sont annulés lorsqu'une nouvelle opération abstraite de commande de remise modifie les restrictions ou lorsque l'association abstraite est libérée.

14.1.4 *Règles de création applicables aux attributs généraux*

Les attributs doivent être créés lorsqu'un rapport est remis ou peuvent être créés lorsqu'une opération abstraite ou une action automatique 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 font l'objet d'un abonnement de la part de l'utilisateur. Les attributs créés constituent une nouvelle entrée (dans certains cas une entrée maîtresse et des entrées secondaires, voir le § 6.3.4) 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 rapport. 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.

On trouvera au tableau 2/X.413 et au § 11.3 les règles de création des attributs généraux. Les règles de création des attributs spécifiques de contenu sont décrites dans les Recommandations spécifiques de contenu (par ex.: les attributs spécifiques de IPMS sont décrits dans l'annexe C de la Rec. X.420 du CCITT | ISO/CEI 10021-7).

14.2 *Utilisation des services abstraits d'accès pour dépôt*

Ce paragraphe traite de l'invocation des opérations abstraites de dépôt de messages, de dépôt d'essai et d'annulation de remise différée ainsi que de l'utilisation de l'opération abstraite de commande de dépôt. L'utilisation par le service abstrait MS des services abstraits d'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 *Invocation de l'opération abstraite de dépôt de messages (MessageSubmission abstract-operation)*

Le déclenchement d'une association abstraite de 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 de dépôt de messages. Pour déposer le message auprès du MTA, le MS effectue les opérations suivantes:

- 1) si l'argument de dépôt de messages ne contient pas l'extension de demande de retransmission (voir le § 6.6), il invoque une opération abstraite de 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 Rec. X.411 du CCITT | ISO/CEI 10021-4. Par ailleurs, il faut vérifier que l'entrée est un message remis et qu'elle comprend des informations provenant d'une entrée de message remis de la base d'information de messages enregistrés, puis appeler l'opération abstraite de dépôt de messages avec le nouveau contenu. La retransmission d'entrées qui ne sont pas des messages remis pourra faire l'objet d'une future normalisation.

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 de message remis en question doit pouvoir être inclus dans l'argument de contenu;

- 2) il obtient un résultat de dépôt de messages en retour lorsque le MTA a accepté le dépôt. Le résultat de 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 Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 3) si l'opération abstraite de dépôt de messages a été déclenchée par une opération abstraite correspondante dépôt de messages de l'UA à destination du MS, le résultat de cette opération abstraite est transmis en retour à l'UA sous forme d'un résultat de dépôt de messages émis par le MS. Cette procédure garantit que le message a bien été accepté par le MTA avant que le résultat 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 non approprié, 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 ou par des restrictions de sécurité temporaires, une erreur de sécurité doit être signalée.

14.2.2 *Invocation de l'opération abstraite de dépôt d'essai (ProbeSubmission abstract-operation)*

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

- 1) il invoque une opération abstraite de 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 Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 2) il obtient un résultat de dépôt d'essai en retour lorsque le MTA a accepté le dépôt. Ce 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 Rec. X.411 du CCITT | ISO/CEI 10021-4;

- 3) le résultat de l'opération abstraite est transmis en retour à l'UA sous forme d'un résultat de dépôt d'essai émis par le MS. Cette procédure garantit que l'essai a bien été accepté par le MTA avant que le résultat 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 ou par des restrictions de sécurité temporaires, une erreur de dépôt d'essai est créée.

14.2.3 *Invocation de l'opération abstraite d'annulation de remise différée (CancelDeferredDelivery abstract-operation)*

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

- 1) il invoque une opération abstraite d'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 Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 d'annulation de remise différée soumis par le MS.

14.2.4 *Fonctionnement de l'opération abstraite de commande de dépôt (SubmissionControl abstract-operation)*

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 invoque une opération abstraite de commande de dépôt. On trouvera de plus amples détails au § 8.2.1.4.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4. Le MS vérifie qu'une association abstraite existe déjà entre le MS et l'UA. Sinon, le MTA est informé par une erreur de rattachement distant qu'un changement de commande de dépôt ne peut avoir lieu actuellement. Si une association existe, le MS réagit en effectuant les opérations suivantes:

- 1) il invoque une opération abstraite correspondante de commande de dépôt du MS ou l'UA;
- 2) il attend que l'UA renvoie un résultat de commande de dépôt contenant des informations permettant de savoir si des messages ou des essais sont en attente dans l'UA en raison des restrictions en vigueur. On trouvera de plus amples détails au § 8.2.1.4.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 3) le MS envoie en retour au MTA un résultat de 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 invoquer une nouvelle opération abstraite de commande de dépôt pour annuler les restrictions. Les effets d'une opération abstraite de commande de dépôt sont annulés lorsqu'une nouvelle opération abstraite de commande de dépôt modifie les restrictions ou lorsque l'association abstraite est libérée. Le MS invoque alors une opération abstraite correspondante de commande de dépôt vers l'UA et attend le résultat de commande de dépôt.

14.3 *Utilisation des services abstraits d'accès d'administration*

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 *Invocation de l'opération abstraite d'enregistrement (Register abstract-operation)*

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

- 1) il invoque une opération abstraite d'enregistrement contenant les nouvelles données à enregistrer. On trouvera de plus amples détails au § 8.4.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;

- 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 *Invocation de l'opération abstraite de modification des pouvoirs (ChangeCredentials abstract-operation)*

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

- 1) il invoque une opération abstraite de modification des pouvoirs auprès du MTA, contenant les nouveaux pouvoirs à enregistrer. On trouvera de plus amples détails au § 8.4.1.2.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 2) il obtient en retour un résultat de modification des pouvoirs lorsque le MTA a accepté la modification. Si l'opération a réussi, le MS enregistre les nouveaux pouvoirs. Le résultat de 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 de modification des pouvoirs (ChangeCredentials abstract-operation)*

Lorsque le MS reçoit du MTA une opération abstraite de 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 de modification des pouvoirs. On trouvera de plus amples détails au § 8.4.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 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 de modification des pouvoirs;
- 4) si l'UA envoie en retour un résultat de modification des pouvoirs vide, indiquant que l'opération a réussi, le MS envoie en retour au MTA un résultat de modification des pouvoirs correspondant indiquant un succès et enregistre les pouvoirs. Si l'UA renvoie une erreur, celle-ci est transmise au MTA pour indication. 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 d'enregistrement de messages**

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

15.1 *Fourniture des services abstraits d'accès de recherche*

Ce paragraphe traite de la fourniture des opérations abstraites de résumé, de listage, d'extraction, de suppression, d'enregistrement-MS et d'avertissement. La fourniture par le service abstrait MS des services abstraits d'accès de recherche suppose qu'une association abstraite existe entre le fournisseur de l'accès de recherche (le MS) et l'utilisateur de l'accès de recherche (l'UA). Ces 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 de résumé (Summarize abstract-operation)*

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

- 1) il établit à quelle base d'information s'adresse l'opération abstraite de résumé;
- 2) il vérifie s'il y a des entrées dans la base d'information. Si cette base d'information est vide, un résultat de 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 de 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 à quelques attributs spécifiques de contenu reconnus par le MS;
- 5) il renvoie le résultat de 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 de résumé. Si une opération abstraite de 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 de listage (List abstract-operation)*

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

- 1) il établit à quelle base d'information s'adresse l'opération abstraite de 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 de 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 a été spécifié 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 d'enregistrement-MS, sont renvoyées si elles sont présentes. On trouvera de plus amples détails au § 8.3.2. La valeur du paramètre d'état de l'entrée pour chaque message sélectionné est mise sur liste «listed»;
- 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 de listage. Si l'opération abstraite de 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 (Fetch abstract-operation)*

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

- 1) il établit à quelle base d'information s'adresse l'opération abstraite d'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 d'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 maîtresse sont exclues, à moins qu'elles ne soient explicitement retenues dans l'argument;
- 4) les restrictions d'extraction établies par le rattachement abstrait seront appliquées (si elles n'ont pas été désactivées) afin de déterminer si l'entrée peut être renvoyée ou si une erreur en résulte. Voir le point 4) du § 7.1.1;
- 5) si un ensemble d'attributs requis ont été spécifiés comme arguments de l'opération abstraite, ces attributs, s'ils sont présents, sont renvoyés à l'UA pour la première entrée sélectionnée. Si aucun attribut requis n'est spécifié, les valeurs par défaut de l'opération abstraite d'extraction, telles qu'elles ont été précédemment spécifiées à l'aide de l'opération abstraite d'enregistrement-MS, sont renvoyées si elles sont présentes. Les types d'information codée autorisés dans les restrictions d'extraction au sujet du rattachement abstrait peuvent limiter les informations renvoyées. On trouvera de plus amples détails au § 7.1.1. 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;
- 6) 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 d'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 de suppression (Delete abstract-operation)*

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

- 1) il établit à quelle base d'information s'adresse l'opération abstraite de suppression;
- 2) il vérifie que les arguments fournis sont valables pour une opération abstraite de 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 de suppression vide est renvoyé à l'UA comme indication de succès.

15.1.5 *Fonctionnement de l'opération abstraite d'enregistrement-MS (Register-MS abstract-operation)*

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

- 1) il vérifie que les arguments fournis sont valables pour une opération abstraite d'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 d'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 d'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 *Invocation de l'opération abstraite d'avertissement (Alert abstract-operation)*

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

Si l'action automatique d'avertissement est déclenchée par l'UA au moyen d'une opération abstraite d'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 d'association abstraite et il est alors impossible d'invoquer une opération abstraite d'avertissement;
- 2) s'il existe une association abstraite, le MS invoque une opération abstraite contenant les informations d'argument pertinentes (on trouvera de plus amples détails au § 8.7.1) et attend que l'UA renvoie un résultat d'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 terminal approprié, associé à l'utilisateur. Cette dernière méthode peut également être utilisée si l'opération abstraite d'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 seront pas transgressées pendant l'opération d'avertissement. Si l'opération abstraite d'avertissement est interdite, soit par les mesures de sécurité ou par des restrictions ou 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, de dépôt de messages, de dépôt d'essai et d'annulation de remise différée ainsi que de l'invocation de l'opération abstraite de commande de dépôt. La fourniture, par le service abstrait MS, des services abstraits d'accès de 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). Ces 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 de dépôt de messages (MessageSubmission abstract-operation)*

Lorsque le MS reçoit de l'UA une opération abstraite de 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 de dépôt de messages. On trouvera de plus amples détails au § 8.2.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 d'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'entrée autres que des entrées de message remis doit faire l'objet d'une future normalisation;
- 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 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 de dépôt de messages, après toute modification indiquée au point 2) ci-dessus;
- 5) si le MTA envoie en retour un résultat de dépôt de messages (on trouvera de plus amples détails au § 8.2.1.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4) indiquant un succès, le MS envoie en retour à l'UA un résultat correspondant de dépôt de messages, indiquant le succès. A noter que le MS ne renvoie jamais d'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 revient;
- 6) le MS peut choisir de mettre fin à l'association abstraite avec le MTA, soit après une certaine période d'activité ou lorsque l'UA met fin à son association abstraite correspondante avec le MS.

15.2.2 *Fonctionnement de l'opération abstraite de dépôt d'essai (ProbeSubmission abstract-operation)*

Lorsque le MS reçoit de l'UA une opération abstraite de 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 de dépôt d'essai. On trouvera de plus amples détails au § 8.2.1.2.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 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 de dépôt d'essai;
- 4) si le MTA envoie en retour un résultat de dépôt d'essai (on trouvera de plus amples détails au § 8.2.1.2.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4) indiquant un succès, le MS envoie en retour à l'UA un résultat correspondant de dépôt d'essai indiquant un succès. A noter que le MS ne renvoie jamais d'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 ou lorsque l'UA met fin à son association abstraite correspondante avec le MS.

15.2.3 *Fonctionnement de l'opération abstraite d'annulation de remise différée (CancelDeferredDelivery abstract-operation)*

Lorsque le MS reçoit de l'UA une opération abstraite d'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 d'annulation de remise différée. On trouvera de plus amples détails au § 8.2.1.3.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;

- 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 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 d'annulation de remise différée;
- 4) si le MTA envoie en retour un résultat d'annulation de remise différée (on trouvera de plus amples détails au § 8.2.1.3.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4) indiquant un succès, le MS envoie en retour à l'UA un résultat correspondant d'annulation de remise différée indiquant un succès. A noter que le MS ne renvoie jamais d'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 ou lorsque l'UA met fin à son association abstraite correspondante avec le MS.

15.2.4 *Invocation de l'opération abstraite de commande de dépôt (SubmissionControl abstract-operation)*

Si le MS reçoit du MTA une opération abstraite de 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 invoque une opération abstraite de commande de dépôt vers l'UA. On trouvera de plus amples détails au § 8.2.1.4.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 2) il attend que l'UA lui renvoie un résultat de commande de dépôt (on trouvera de plus amples détails au § 8.2.1.4.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4) confirmant l'acceptation de l'opération abstraite de commande de dépôt;
- 3) si l'opération abstraite de commande de dépôt a été déclenchée par une opération abstraite correspondante demandée au MTA par le MS, le résultat de commande de dépôt provenant de l'UA est communiqué du MS au MTA et le MS.

15.3 *Fourniture des services abstraits d'accès d'administration*

Ce paragraphe traite du fonctionnement des opérations abstraites d'enregistrement et de modification des pouvoirs. La fourniture par le service abstrait MS des services abstraits d'accès d'administration 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). Ces 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 d'enregistrement (Register abstract-operation)*

Lorsque le MS reçoit de l'UA une opération abstraite d'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 d'enregistrement. On trouvera de plus amples détails au § 8.4.1.1.1 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 aucune autre action n'est effectuée;
- 3) si l'association abstraite entre le MS et le MTA existe, le MS invoque une opération abstraite d'enregistrement vers le MTA;
- 4) si le MTA envoie en retour un résultat d'enregistrement (on trouvera de plus amples détails au § 8.4.1.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4) indiquant un succès, le MS renvoie à l'UA un résultat d'enregistrement correspondant, indiquant un succès. A noter que le MS ne renvoie jamais d'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 ou 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 de modification des pouvoirs (ChangeCredentials abstract-operation)*

Une opération abstraite de modification des pouvoirs est déclenchée parce que le MTA a invoqué une opération abstraite de modification des pouvoirs vers le MS. 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 de modification des pouvoirs. On trouvera de plus amples détails au § 8.4.1.2 de la Rec. X.411 du CCITT | CCITT ISO/CEI 10021-4. 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 invoque vers l'UA une opération abstraite de modification des pouvoirs contenant les nouveaux pouvoirs à enregistrer. On trouvera de plus amples détails au § 8.4.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 3) il obtient en retour un résultat de modification des pouvoirs lorsque l'UA a accepté la modification et il met en mémoire les nouveaux pouvoirs. Le résultat de modification des pouvoirs ou l'erreur résultante provenant de l'UA est transmis au MTA.

15.3.3 *Fonctionnement de l'opération abstraite de modification des pouvoirs (ChangeCredentials abstract-operation)*

Lorsque le MS reçoit de l'UA une opération abstraite de 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 de modification des pouvoirs. On trouvera de plus amples détails au § 8.4.1.2 de la Rec. X.411 du CCITT | ISO/CEI 10021-4;
- 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 l'on ne procède à aucune autre opération;
- 3) si l'association abstraite entre le MS et le MTA existe, le MS invoque vers le MTA une opération abstraite de modification des pouvoirs;
- 4) si le MTA envoie en retour un résultat de modification des pouvoirs vide, indiquant un succès, le MS envoie en retour un résultat correspondant de 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 d'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, ou 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 de recherche, de dépôt et d'administration du service abstrait MS sont fournis. On trouvera au § 8 de la Rec. X.411 du CCITT | ISO/CEI 10021-4 une description de la fourniture par le service abstrait MTS des accès de recherche, de dépôt et d'administration.

16.1 *Accès de recherche*

Les services abstraits d'accès de recherche sont fournis un à un entre les opérations abstraites et les opérations réelles de l'élément de service recherche de messages (MRSE) dont il est question dans la Rec. X.419 du CCITT | ISO/CEI 10021-6.

16.2 Accès de dépôt indirect

Les services abstraits d'accès de dépôt indirect sont fournis un à un entre les opérations abstraites et les opérations réelles de l'élément de service dépôt de message (MSSE) dont il est question dans la Rec. X.419 du CCITT | ISO/CEI 10021-6.

16.3 Accès d'administration

Les services abstraits d'accès d'administration 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 Rec. X.419 du CCITT | ISO/CEI 10021-6.

ANNEXE A

(à la Recommandation X.413)

Attribution formelle des identificateurs d'objets

(La présente annexe fait partie intégrante de la présente 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 ASN.1 des annexes ultérieures.

La présente annexe est définitive pour toutes les valeurs, à l'exception de celles qui sont 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 clauses 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 MXObjectIdentifiers { joint-iso-ccitt mhs-motis(6) arch(5) modules(0) object-identifiers(0) };

-- Catégories

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

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

id-pt *-- types d'accès* **ID ::= { id-ms 2 }**

id-att *-- types d'attribut* **ID ::= { id-ms 3 }**

id-act *-- types d'actions
automatique* **ID ::= { id-ms 4 }**

-- Modules

id-mod-object-identifiers **ID ::= { id-mod 0 }** *-- attribution non définitive*

id-mod-abstract-service **ID ::= { id-mod 1 }** *-- attribution non définitive*

id-mod-attribute-types **ID ::= { id-mod 2 }** *-- attribution non définitive*

id-mod-action-types **ID ::= { id-mod 3 }** *-- attribution non définitive*

id-mod-upper-bounds **ID ::= { id-mod 4 }** *-- attribution non définitive*

-- 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-redirectation-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 }

-- Types d'actions automatiques

id-act-auto-forward	ID ::= { id-act 0 }
id-act-auto-alert	ID ::= { id-act 1 }

END -- fin des identificateurs d'objets MS

Définition formelle du service abstrait d'enregistrement de messages

(La présente annexe fait partie intégrante de la présente Recommandation)

La présente annexe, qui complète la section 2, définit formellement le service abstrait d'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 Rec. X.407 du CCITT | ISO/CEI 10021-3.

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 soient dans chaque cas invoquées et signalées au-delà de la limite entre systèmes ouverts. Toutefois, cela sera fait fréquemment. La Rec. X.419 du CCITT | ISO/CEI 10021-6 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, 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,

valuesSEQUENCE SIZE (1..ub-attribute-values) OF Attribute Value }

AttributeType ::= OBJECT IDENTIFIER

Attribute Value ::= ANY

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) }**

-- Rattachement abstrait

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 -- aucune restriction par défaut --,**
 ms-configuration-request **[5] BOOLEAN DEFAULT FALSE }**

Restrictions ::= SET {
 allowed-content-types **[0] SET SIZE (1..ub-content-types) OF OBJECT IDENTIFIER OPTIONAL**
 -- aucune restriction par défaut --,
 allowed-EITs **[1] MS-EITs OPTIONAL -- aucune restriction par défaut --,**
 maximum-content-length **[2] ContentLength OPTIONAL -- aucune restriction par défaut -- }**

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 -- l'omission de ce composant indique qu'il n'y a pas de limite inférieure --,**
 to **[1] SequenceNumber OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas de limite supérieure -- }**

```

TimeRange ::= SEQUENCE {
    from      [0] CreationTime OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas de limite
            inférieure --,
    to        [1] CreationTime OPTIONAL -- l'omission de ce composant indique qu'il n'y a pas de limite
            supérieure -- }

Filter ::= CHOICE {
    item      [0] FilterItem,
    and       [1] SET OF Filter,
    or        [2] SET OF Filter,
    not       [3] Filter }

FilterItem ::= CHOICE {
    equality          [0] AttributeValueAssertion,
    substrings       [1] SEQUENCE {
        type          AttributeType,
        strings       SEQUENCE OF CHOICE {
            initial    [0] AttributeValue,
            any         [1] AttributeValue,
            final      [2] AttributeValue } },
    greater-or-equal [2] AttributeValueAssertion,
    less-or-equal   [3] AttributeValueAssertion,
    present         [4] AttributeType,
    approximate-match [5] AttributeValueAssertion }

AttributeValueAssertion ::= SEQUENCE {
    type      AttributeType,
    value     AttributeValue }

Selector ::= SET {
    child-entries [0] BOOLEAN DEFAULT FALSE,
    range         [1] Range OPTIONAL -- pas de limites par défaut --,
    filter        [2] Filter OPTIONAL -- toutes les entrées sont censées être dans l'étendue spécifiée --,
    limit         [3] INTEGER (1..ub-messages) OPTIONAL,
    override     [4] OverrideRestrictions OPTIONAL -- toutes restrictions d'extraction éventuellement applicables
            sont censées s'appliquer par défaut -- }

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 -- à utiliser si le type est plusieurs valeurs --,
    count     [1] INTEGER (1..ub-attribute-values) OPTIONAL -- à utiliser si le type est plusieurs valeurs -- }

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 Attribute Type OPTIONAL
 -- absent si aucun résumé n'est demandé -- }

SummarizeResult ::= SET {
 next [0] SequenceNumber OPTIONAL,
 count [1] INTEGER (0..ub-messages) -- décompte des entrées sélectionnées --,
 span [2] Span OPTIONAL -- décompte des entrées sélectionnées, supprimé si nul --,
 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 -- décompte des entrées pour lesquelles l'attribut est absent --,
 present [1] SET SIZE (1..ub-attribute-values) OF -- un résumé pour chaque valeur d'attribut --
 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 -- composant absent si aucune entrée n'est trouvée -- }

--

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 -- si une entrée a été sélectionnée --,
    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] InformationBaseDEFAULT 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] Credentials,
        new-credentials [1] Credentials } OPTIONAL
    -- même CHOIX CHOICE que pour les anciens pouvoirs --,
    user-security-labels[5] SET SIZE (1..ub-labels-and-redirections) OF SecurityLabel OPTIONAL }

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

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] OBJECT IDENTIFIER,**
 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 présente 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° 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

-- Identificateurs de types d'attributs généraux

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-att-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-per-recipient-report-delivery-fields, 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-security-classification, 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 d'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 d'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, DLExpansion, MessageDeliveryEnvelope,
    MessageDeliveryIdentifier, MessageDeliveryTime, MessageOriginAuthenticationCheck,
    MessageSecurityLabel, MessageSubmissionTime, MessageToken, OriginatorCertificate, ORName,
    PerRecipientReportDeliveryFields, Priority, ProofOfDeliveryRequest, RedirectionHistory,
    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 DLExpansion
 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

Définition formelle des types d'actions automatiques générales

(La présente annexe fait partie intégrante de la présente 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 générales
auto-forward, auto-alert;

IMPORTS

-- Identificateurs d'objets des types d'actions automatiques générales

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 données du service abstrait MS

Filter, EntryInformationSelection

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

-- Types de données 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) }

-- Limites supérieures du MTS

ub-recipients

--

FROM MTSUpperBounds { joint-iso-ccitt mhs-motis(6) mts(3) modules(0) upper-bounds(3) }

-- Limite supérieure 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 OPTIONAL,
    per-message-indicators PerMessageIndicators OPTIONAL,
    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 MS

(La pr sente annexe ne fait pas partie int grante de la pr sente 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) le plus grand entier repr sentable sur 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) le plus grand entier repr sentable sur 32 bits --
ub-nested-filters       INTEGER ::= 32
ub-per-auto-action      INTEGER ::= 32767      -- (215 - 1) le plus grand entier repr sentable sur 16 bits --
ub-per-entry            INTEGER ::= 1024
ub-summaries            INTEGER ::= 16

```

```

END -- des limites sup rieures du MS

```

ANNEXE F

(à la Recommandation X.413)

Exemple d'opération abstraite de résumé

(La présente annexe ne fait pas partie intégrante de la présente Recommandation)

La présente annexe donne un exemple d'utilisation de l'opération abstraite de 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. Un (—) indique que l'attribut est absent de 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 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 de priorité. Le résultat demandé est la liste des décomptes 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 de listage

Priorité	Décompte
—	1 (15)
urgent	1 (20)
normal	4 (15,18,22,23)
faible	0

Les composants de l'argument de résumé devraient avoir la valeur suivante:

sélecteur:

 filtre: état de l'entrée = nouveau

demandes de résumé: type d'attribut = Priorité

Les composants du résultat de résumé pourraient être les suivants:

décompte: 5

portée:

 limite inférieure: 15

 limite supérieure: 23

résumés:

 { absent: 1

 présent: { valeur = normale, décompte = 3 }

 { valeur = urgent, décompte = 1 } }

ANNEXE G

(à la Recommandation X.413)

Différences entre la Recommandation X.413 (1992) et l'ISO/CEI 10021-5:1990

(La présente annexe ne fait pas partie intégrante de la présente Recommandation)

Il n'y a que deux différences connues entre la présente Recommandation et l'ISO/CEI 10021-5:1990.

- 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 l'ISO/CEI 10021-5:1990, mais elles font partie intégrante de la présente Recommandation.

A l'ISO/CEI, 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.

Imprimé en Suisse

Genève, 1993