



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

H.501

(03/2002)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Procédures de mobilité et de collaboration – Aperçu
général de la mobilité et de la collaboration, définitions,
protocoles et procédures

Protocole pour la gestion de la mobilité et les communications intra et interdomainiales dans les systèmes multimédias

Recommandation UIT-T H.501

RECOMMANDATIONS UIT-T DE LA SÉRIE H
SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

CARACTÉRISTIQUES DES SYSTÈMES VISIOPHONIQUES	H.100–H.199
INFRASTRUCTURE DES SERVICES AUDIOVISUELS	
Généralités	H.200–H.219
Multiplexage et synchronisation en transmission	H.220–H.229
Aspects système	H.230–H.239
Procédures de communication	H.240–H.259
Codage des images vidéo animées	H.260–H.279
Aspects liés aux systèmes	H.280–H.299
SYSTÈMES ET ÉQUIPEMENTS TERMINAUX POUR LES SERVICES AUDIOVISUELS	H.300–H.399
SERVICES COMPLÉMENTAIRES EN MULTIMÉDIA	H.450–H.499
PROCÉDURES DE MOBILITÉ ET DE COLLABORATION	
Aperçu général de la mobilité et de la collaboration, définitions, protocoles et procédures	H.500–H.509
Mobilité pour les systèmes et services multimédias de la série H	H.510–H.519
Applications et services de collaboration multimédia mobile	H.520–H.529
Sécurité pour les systèmes et services multimédias mobiles	H.530–H.539
Sécurité pour les applications et services de collaboration multimédia mobile	H.540–H.549
Procédures d'interfonctionnement de la mobilité	H.550–H.559
Procédures d'interfonctionnement de collaboration multimédia mobile	H.560–H.569

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

Recommandation UIT-T H.501

Protocole pour la gestion de la mobilité et les communications intra et interdomainiales dans les systèmes multimédias

Résumé

La présente Recommandation a pour objet de définir des messages et des procédures pour la gestion de la mobilité et pour la communication au sein des domaines et entre domaines dans les environnements mobiles et non mobiles pour la résolution d'adresses, l'authentification d'utilisateur, l'échange de données de service, l'autorisation d'accès, la validation d'appel et le journal d'utilisation.

Source

La Recommandation H.501 de l'UIT-T, élaborée par la Commission d'études 16 (2001-2004) de l'UIT-T, a été approuvée le 29 mars 2002 selon la procédure définie dans la Résolution 1 de l'AMNT.

Mots clés

Annexe G/H.225.0, authentification, autorisation, domaines administratifs, gestion de la mobilité, rapport d'utilisation, résolution d'adresse, systèmes multimédias.

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

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

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

Dans certains secteurs des technologies de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

NOTE

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

DROITS DE PROPRIÉTÉ INTELLECTUELLE

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

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

© UIT 2002

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

TABLE DES MATIÈRES

	Page
1	Domaine d'application 1
2	Références normatives..... 1
3	Définitions 2
4	Symboles et abréviations 2
5	Prescriptions 3
5.1	Prescriptions relatives au transport..... 3
5.2	Considérations relatives à la sécurité 3
5.3	Conventions d'adressage..... 4
5.4	Canevas et descripteurs d'adresse..... 4
6	Définition des messages 5
6.1	Descripteur (<i>descriptor</i>) 6
6.2	Informations de descripteur (<i>descriptorInfo</i>) 6
6.3	Canevas d'adresse (<i>address template</i>) 6
6.3.1	Modèle (<i>pattern</i>)..... 7
6.3.2	Informations de routage (<i>routeInformation</i>) 7
6.3.3	Informations de tarification (<i>pricingInformation</i>)..... 8
6.3.4	Informations de contact (<i>ContactInformation</i>)..... 9
6.4	Structures communes..... 10
6.4.1	Autre élément homologue (<i>AlternatePE</i>)..... 10
6.4.2	Informations sur un participant 11
6.4.3	Informations sur l'appel 11
6.4.4	Informations d'utilisateur..... 11
6.4.5	Spécification d'utilisation 11
6.4.6	Mode sécurité 13
6.5	Relation de service 13
6.5.1	Demande de service (<i>ServiceRequest</i>)..... 13
6.5.2	Confirmation de service (<i>ServiceConfirmation</i>) 14
6.5.3	Rejet de service (<i>ServiceRejection</i>) 15
6.5.4	Libération du service (<i>ServiceRelease</i>) 15
6.6	Répartition des descripteurs 16
6.6.1	Demande de descripteur (<i>DescriptorRequest</i>)..... 16
6.6.2	Confirmation de descripteur (<i>DescriptorConfirmation</i>) 16
6.6.3	Rejet de descripteur (<i>DescriptorRejection</i>) 16
6.6.4	Demande d'identificateur de descripteur (<i>DescriptorIDRequest</i>) 17
6.6.5	Confirmation d'identificateur de descripteur (<i>DescriptorIDConfirmation</i>) ... 17
6.6.6	Rejet d'identificateur de descripteur (<i>DescriptorIDRejection</i>)..... 18

	Page
6.6.7 Mise à jour de descripteur (<i>DescriptorUpdate</i>).....	18
6.6.8 Accusé de réception de mise à jour de descripteur (<i>DescriptorUpdateAcknowledgement</i>)	19
6.7 Résolution d'adresses.....	19
6.7.1 Demande d'accès (<i>AccessRequest</i>)	19
6.7.2 Confirmation d'accès (<i>AccessConfirmation</i>)	19
6.7.3 Rejet d'accès (<i>AccessRejection</i>).....	20
6.8 Demande en cours (<i>RequestInProgress</i>)	21
6.9 Messages non normalisés	21
6.9.1 Demande non normalisée (<i>NonStandardRequest</i>).....	21
6.9.2 Confirmation non normalisée (<i>NonStandardConfirmation</i>)	21
6.9.3 Rejet non normalisé (<i>NonStandardRejection</i>).....	21
6.10 Réponse à un message non reconnu (<i>UnknownMessageResponse</i>)	22
6.11 Rapports d'utilisation.....	22
6.11.1 Demande d'utilisation	22
6.11.2 Confirmation d'utilisation.....	22
6.11.3 Rejet d'utilisation.....	23
6.11.4 Indication d'utilisation	23
6.11.5 Confirmation d'indication d'utilisation	24
6.11.6 Rejet d'indication d'utilisation	24
6.12 Validation	25
6.12.1 Demande de validation.....	25
6.12.2 Confirmation de validation.....	25
6.12.3 Rejet de validation.....	26
6.13 Authentification.....	26
6.13.1 Demande d'authentification.....	26
6.13.2 Confirmation d'authentification.....	26
6.13.3 Rejet de l'authentification.....	27
Annexe A – Syntaxe des messages.....	28

Recommandation UIT-T H.501

Protocole pour la gestion de la mobilité et les communications intra et interdomainiales dans les systèmes multimédias

1 Domaine d'application

La présente Recommandation propose un protocole pour la communication entre éléments logiques d'un réseau multimédia à mode paquet afin de permettre les communications vers et depuis des utilisateurs gérés par de tels éléments logiques. Ce protocole peut être utilisé dans les environnements mobiles et non mobiles pour la résolution d'adresses, l'authentification d'utilisateur, l'échange de données de service, l'autorisation d'accès, la validation d'appel et le journal d'utilisation. Ces capacités permettent d'utiliser le protocole pour la gestion de la mobilité dans les environnements mobiles.

La procédure générale consiste, pour les éléments logiques, à échanger des informations relatives à l'emplacement des utilisateurs ou des points d'extrémité sous la forme d'adresses que chaque domaine administratif peut résoudre. Les adresses peuvent être spécifiées de manière générale ou de manière de plus en plus spécifique. Des informations additionnelles permettent aux éléments dans un domaine administratif de déterminer le domaine administratif le plus approprié pour servir de destination de l'appel. Les éléments logiques peuvent gérer l'accès à leurs adresses exposées et exigent des comptes rendus d'utilisation pendant les appels à ces adresses.

D'autres Recommandations spécifieront la manière dont le protocole défini dans la présente Recommandation est utilisé dans des applications particulières. Une application ne doit pas nécessairement implémenter l'ensemble du protocole; elle peut sélectionner parmi les messages et procédures ceux qui répondent à ses besoins.

2 Références normatives

La présente Recommandation se réfère à certaines dispositions des Recommandations UIT-T et textes suivants qui, de ce fait, en sont partie intégrante. Les versions indiquées étaient en vigueur au moment de la publication de la présente Recommandation. Toute Recommandation ou tout texte étant sujet à révision, les utilisateurs de la présente Recommandation sont invités à se reporter, si possible, aux versions les plus récentes des références normatives suivantes. La liste des Recommandations de l'UIT-T en vigueur est régulièrement publiée.

- Recommandation UIT-T E.164 (1997), *Plan de numérotage des télécommunications publiques internationales*.
- Recommandation UIT-T H.225.0 Version 4 (2000), *Protocoles de signalisation d'appel et mise en paquets des trains multimédias dans les systèmes de communication multimédia en mode paquet*.
- Recommandation UIT-T H.235 Version 2 (2000), *Sécurité et cryptage des terminaux multimédias de la série H (terminaux H.323 et autres terminaux de type H.245)*.
- Recommandation UIT-T H.323 Version 4 (2000), *Systèmes de communication multimédia en mode paquet*.
- Recommandation UIT-T X.680 (1997), *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base*.
- Recommandation UIT-T X.691 (1997), *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage compact*.
- IETF RFC 2401 (1998), *Security Architecture for the Internet Protocol*.

- IETF RFC 2402 (1998), *IP Authentication Header*.
- IETF RFC 2406 (1998), *IP Encapsulating Security Payload (ESP)*.
- ISO 4217:2001, *Codes pour la représentation des monnaies et types de fonds*.

3 Définitions

La présente Recommandation définit les termes suivants:

3.1 domaine administratif: ensemble d'entités logiques et physiques gérées par une entité administrative. Un domaine administratif peut être constitué d'une ou de plusieurs zones.

3.2 portier: élément logique qui offre des services spécifiques (traduction de numéros, commande d'accès, etc.) à d'autres entités au sein d'un domaine administratif (voir également la Rec. UIT-T H.323).

3.3 élément logique: entité ayant une fonction définie dans un réseau. Un élément logique n'impose aucune restriction à son utilisation; sa fonction peut être implémentée de la manière qui convient dans le matériel et dans le logiciel.

3.4 élément homologue: élément logique de début ou de fin de message de signalisation, défini dans la présente Recommandation. Des exemples sont les éléments frontières de l'Annexe G/H.225.0 et les portiers H.323.

3.5 zone: le sous-ensemble d'entités d'un domaine administratif se trouvant sous le contrôle d'un seul portier.

4 Symboles et abréviations

La présente Recommandation utilise les abréviations suivantes:

AD	domaine administratif (<i>administrative domain</i>)
DH	protocole d'échange de clés de Diffie-Hellman (Rec. UIT-T X.509) (<i>Diddie-Hellman key agreement protocol</i>)
DNS	système de dénomination de domaine (<i>domain name system</i>)
IP	protocole Internet (<i>Internet protocol</i>)
OID	identificateur d'objet (<i>object identifier</i>)
PDU	unité de données protocolaire (<i>protocol data unit</i>)
RAS	protocole d'enregistrement, admission et statut (<i>registration, admission and status protocol</i>)
RCC	réseau à commutation de circuits
TCP	protocole de commande de transmission (<i>transmission control protocol</i>)
TPKT	paquet de transport (<i>transport packet</i>)
UDP	protocole datagramme d'utilisateur (<i>user datagram protocol</i>)
URL	localisateur uniforme de ressource (<i>uniform resource locator</i>)
UTC	temps universel coordonné (<i>universal time coordinated</i>)

5 Prescriptions

5.1 Prescriptions relatives au transport

Les messages peuvent être envoyés en utilisant un service de transport non fiable (tel que le protocole UDP) ou un service fiable (tel que le protocole TCP) vers une adresse bien connue. Sur les réseaux IP, il convient d'utiliser le port bien connu (2099) tant pour les protocoles TCP que UDP, sauf si un autre port a été indiqué à l'expéditeur. Des éléments seront à l'écoute sur les ports TCP et UDP.

Si les messages sont émis au moyen d'un service de transport fiable, des messages complets peuvent être envoyés dans les limites définies par l'unité de données protocolaire (PDU, *protocol data unit*) de transport fiable (dans des implémentations IP, comme celle présentée dans l'Appendice IV/H.225.0, cette unité PDU est définie par un paquet TPKT; voir la Figure 1. Chaque unité PDU H.501 contient un seul message défini dans la présente Recommandation).

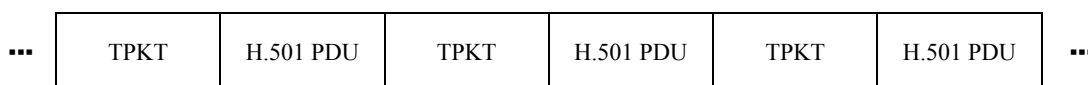


Figure 1/H.501 – Transport par protocole TCP

En cas d'utilisation d'un service de transport non fiable, des messages de demande peuvent être retransmis. La valeur par défaut du temporisateur de retransmission doit être déterminée par une méthode adaptative sensible au délai (comme celle qu'utilise le protocole TCP). Des temps d'attente exponentiels seront utilisés pour les retransmissions subséquentes. Le nombre de retransmissions ne dépassera pas cinq. Les réponses ne seront pas retransmises.

Dans les implémentations IP à protocole UDP, les messages comporteront un préfixe constitué d'en-têtes de paquet TPKT, permettant l'émission de plusieurs messages par paquet (voir la Figure 2). Chaque unité PDU H.501 contient un seul message défini dans la présente Recommandation. Le champ de longueur du paquet UDP contiendra la longueur totale de la charge utile, y compris tous les messages et leurs en-têtes TPKT.

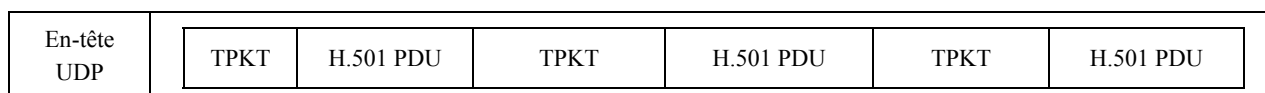


Figure 2/H.501 – Format de datagramme UDP

5.2 Considérations relatives à la sécurité

Le fonctionnement de sécurité du protocole IP, tel qu'il est décrit dans la Norme IETF RFC 2401, complété de la Norme IETF RFC 2402 ou IETF RFC 2406, ou des deux, sera mis en œuvre lorsque l'authentification, l'intégrité et le cryptage des messages échangés entre éléments homologues sont souhaités au niveau de la couche Transport.

Lorsque la protection est requise au niveau de la couche Application, les procédures et expressions de la Rec. UIT-T H.235 seront utilisées pour la prise en charge de la sécurité au niveau application. Ceci concerne plus précisément les formats de jeton et les échanges d'authentification. Les jetons et les jetons (*token*) cryptés reçus dans les messages de réponse seront utilisés dans une demande connexe ultérieure.

5.3 Conventions d'adressage

Il est important, pour l'interfonctionnement entre domaines, que les formats d'adressage utilisés dans des messages de signalisation émis soient compris par le système récepteur. Un élément homologue doit prendre en charge tous les types de *AliasAddress* qu'une application est globalement susceptible d'utiliser.

Le minimum à prendre en charge sont les formats *email-id* et *partyNumber* (utilisant *PublicNumber* avec *PublicTypeOfNumber* de *internationalNumber*). Pour les communications avec d'autres éléments homologues, il convient d'utiliser uniquement les types *email-id* et *partyNumber* de *AliasAddress*, à moins d'un accord préalable sur l'utilisation d'autres formats entre les domaines administratifs concernés. Si un groupe de domaines administratifs a, par exemple, convenu de l'interprétation d'un plan de numérotage privé, ces numéros peuvent être utilisés pour l'échange de messages entre eux.

Dans un environnement mobile, on peut utiliser le format *mobileUIM* comme un module d'identification d'utilisateur global après accord entre les domaines administratifs concernés. Ce format est décrit en détail dans la Rec. UIT-T H.225.0.

5.4 Canevas et descripteurs d'adresse

Un canevas d'adresse (en abrégé un "canevas") définit un ensemble d'identificateurs d'adresse de pseudonyme, des informations de tarification pour l'établissement d'appels à destination de ces adresses et le protocole devant être utilisé pour atteindre les adresses de cet ensemble. Un domaine administratif indique les appels qu'il est en mesure de résoudre en publiant des canevas. Les canevas sont regroupés par un identificateur appelé "descripteur". Une fois qu'un canevas a été associé à un descripteur, toute modification de ce canevas implique une modification du descripteur "groupe". Les informations de canevas peuvent permettre la mise en commun d'informations d'adressage si le schéma d'adressage est organisé de manière hiérarchique ou pouvant faire l'objet d'un acheminement. Une zone donnée peut, par exemple, traiter le numéro 1303538* qui équivaut à tous les numéros débutant par les chiffres 1303538.

NOTE – Comme le caractère "*" est significatif et que le canevas contient dans ce cas un champ explicite indiquant si l'adresse est spécifique ou s'il s'agit d'un "joker". Les exemples qui suivent utilisent le caractère "*" pour indiquer un "joker", mais la représentation effective dans le canevas se fait au moyen du champ explicite). Le caractère "*" n'est pas envoyé sur la ligne s'il représente un joker.

Le canevas suivant est donné à titre d'exemple:

"pour le numéro 1 555 123 4567	émettre un message AccessRequest à destination de l'élément homologue A"
"pour les chiffres 1 555 987*	émettre un message AccessRequest à destination de l'élément homologue B"
"pour le numéro 1 555 987 6543	émettre un message Setup à destination de la passerelle X"
"pour *@example.org	émettre un message AccessRequest à destination de l'élément homologue A"
"pour les chiffres 1*	émettre un message de demande d'accès à destination de l'élément homologue B"
"pour les chiffres privés 31*	émettre un message de demande d'accès à destination de l'élément homologue C"
"pour les chiffres 44 171 112*	n'existe pas"

6 Définition des messages

Le présent paragraphe spécifie les messages et les éléments du protocole définis dans la présente Recommandation. L'Annexe A contient la définition ASN.1 correspondante conformément à la Rec. UIT-T X.680. Les messages seront codés au moyen de la variante alignée de base des règles de codage compact, conformément à la Rec. UIT-T X.691.

Chaque message contient un ensemble de champs communs en plus des informations propres au message. Les champs communs sont les suivants:

Champ	Description
Numéro de séquence (<i>sequenceNumber</i>)	Tout message de demande ou de mise à jour contient un numéro de séquence unique. Le message émis en réponse à un message de demande (message de confirmation ou de rejet) utilise le numéro de séquence du message de demande. Les messages retransmis auront le même numéro de séquence.
Adresse de réponse (<i>replyAddress</i>)	Adresse à laquelle il faut envoyer la réponse à un message de demande. Tout message de demande contiendra une adresse de réponse, sauf si l'adresse peut être obtenue de la couche Transport. Si, dans un réseau IP, l'expéditeur du message de demande est à l'écoute du port par défaut (2099), il n'y a pas lieu d'inclure le message de réponse. Dans un tel cas, le destinataire obtient l'adresse de transport de l'expéditeur en joignant le port par défaut (2099) à l'adresse IP de l'expéditeur telle qu'elle est reçue dans l'en-tête IP du paquet de demande ¹ .
Version	Version de protocole utilisée par l'expéditeur du message.
Version Annexe G (<i>annexGversion</i>)	Version de protocole de l'Annexe G/H.225.0 (présente seulement pour la compatibilité arrière; elle indiquera "Annexe G V2").
Compteur de bonds (<i>hopCount</i>)	Définition du nombre d'éléments homologues à travers lesquels ce message peut se propager. Lorsqu'un élément homologue reçoit ce message et décide de le réacheminer à un autre élément homologue, il commence par décrémenter le compteur de bonds. Si la nouvelle valeur du compteur est supérieure à zéro, l'élément homologue insère cette valeur dans le message à retransmettre. L'élément homologue ne retransmet pas le message si le comptage a atteint zéro. Si le message est une demande, l'élément homologue répondra alors au moyen d'un message de confirmation contenant toute information pertinente. Il répondra au moyen d'un message de rejet si aucune information n'est disponible.
Valeur de vérification d'intégrité (<i>integrityCheckValue</i>)	Permet d'améliorer l'intégrité/authentification du message. La valeur de vérification d'intégrité fondée sur un mécanisme cryptographique est calculée par l'expéditeur, qui applique un algorithme d'intégrité négocié et la clé secrète à la totalité du message. Avant le calcul de la valeur de vérification d'intégrité, chaque octet de ce champ doit être mis à zéro. Après le calcul, l'expéditeur place la valeur de vérification d'intégrité calculée dans le champ <i>integrityCheckValue</i> et transmet le message.
Jetons (<i>tokens</i>)	Il s'agit de données qui peuvent être nécessaires pour l'opération. Si elles sont disponibles, elles seront introduites dans le message.

¹ Comme on suppose qu'il n'y a pas d'éléments homologues cachés derrière les dispositifs de traduction d'adresse de réseau (NAT, *network address translation*), il n'est pas nécessaire de préférer l'adresse de transport à l'adresse de réponse, comme c'est le cas pour les messages RAS.

Jetons (tokens) cryptés (<i>cryptoTokens</i>)	Jetons cryptés.
NonStandard	Informations non normalisées.
Identificateur de service (<i>serviceID</i>)	Il identifie une session de relations de service particulière entre deux éléments homologues. Quand un élément homologue reçoit un message <i>ServiceRequest</i> demandant l'établissement d'une nouvelle relation de service (qui est indiquée par l'absence du champ d'identificateur du service dans le message <i>ServiceRequest</i>), il attribue un identificateur de service mondialement unique et le renvoie à l'expéditeur du message <i>serviceRequest</i> dans le message <i>ServiceConfirmation</i> . Dès qu'une relation est établie, l'identificateur de service est inclus dans tous les messages subséquents échangés avec l'élément homologue (tels que <i>UsageIndication</i> , <i>DescriptorIDRequest</i> , <i>DescriptorRequest</i> , <i>AccessRequest</i>). Il est utilisé par l'élément homologue destinataire pour vérifier s'il a une relation de service avec l'expéditeur du message.
Données génériques (<i>genericData</i>)	Achemine toute donnée générique associée au message. <i>GenericData</i> est expliqué dans les Recs. UIT-T H.323 et H.225.0.
Ensemble de caractéristiques (<i>featureSet</i>)	Sert à négocier des ensembles de caractéristiques génériques. Le système de négociation est le même que celui expliqué pour la signalisation RAS dans les Recs. UIT-T H.323 et H.225.0.

6.1 Descripteur (*descriptor*)

Le descripteur n'est pas un message, mais un élément de message utilisé comme étiquette pour un ensemble de canevas.

Le descripteur contient les informations suivantes:

Champ	Description
Informations de descripteur (<i>descriptorInfo</i>)	Identificateur exclusif du descripteur et de l'instant de sa dernière modification (se référer aux informations de descripteur ci-dessous).
Canevas (<i>templates</i>)	Ensemble de canevas qui définit les adresses pouvant être résolues par ce descripteur.
Identificateur de portier (<i>gatekeeperID</i>)	Identificateur de type texte indiquant le propriétaire du descripteur (c'est-à-dire, le portier qui a créé ce message).

6.2 Informations de descripteur (*descriptorInfo*)

Les informations de descripteur identifient de manière exclusive le descripteur et indiquent l'instant de sa dernière modification.

Champ	Description
Identificateur de descripteur (<i>descriptorID</i>)	Identificateur mondialement exclusif utilisé pour distinguer ce descripteur parmi un grand nombre d'autres descripteurs possibles.
Dernière modification (<i>lastChanged</i>)	Date et heure UTC de la dernière modification de ce descripteur.

6.3 Canevas d'adresse (*address template*)

Le canevas d'adresse décrit un ensemble d'une ou plusieurs adresses de pseudonyme. Le canevas n'est pas un message, mais un élément utilisé par d'autres éléments comme bloc de construction. Le canevas se compose d'autres structures qui sont décrites dans les sous-paragraphe qui suivent.

Champ	Description
Modèle (<i>pattern</i>)	Liste de modèles (voir modèle ci-dessous).
Informations de routage (<i>routeInfo</i>)	Liste d'informations de routage pour ce canevas (se référer aux informations de routage ci-dessous).
Durée de vie (<i>timeToLive</i>)	Durée de validité de ce canevas, exprimée en secondes.
Protocoles pris en charge (<i>supportedProtocols</i>)	Identifie les types de protocole pris en charge par ce canevas (par exemple voix ou télécopie)
Ensemble de caractéristiques (<i>featureSet</i>)	Spécifie les ensembles de caractéristiques génériques que ce canevas d'adresse prend en charge et les caractéristiques génériques qu'il souhaite dans un point d'extrémité distant. Les informations sur les caractéristiques génériques spécifiées à ce niveau s'appliquent à toutes les informations de routage applicables à ce canevas d'adresse. Le cadre générique d'extensibilité est décrit dans les Recs. UIT-T H323 et H.225.0.

6.3.1 Modèle (*pattern*)

Une structure de modèle apparaît dans le canevas d'adresse. Le modèle permet de spécifier une adresse de pseudonyme, une adresse de pseudonyme joker ou une série d'adresses de pseudonyme: les adresses joker supposent un espace d'adresse hiérarchiquement structuré, les gammes d'adresses n'ayant de sens que pour les types d'adresse qui représentent une structure ordonnée.

Champ	Description
Spécifique (<i>specific</i>)	Désigne une adresse de pseudonyme spécifique.
Joker (<i>wildcard</i>)	Forme de définition hiérarchique qui représente des extensions possibles de la chaîne. Cette extension est possible à la fin d'un numéro E.164 ou en début de chaîne dans le cas d'une adresse de messagerie électronique. Par exemple si ce champ <i>wildcard</i> est "+1 303", le modèle peut représenter tout numéro dans la zone de code de Denver, Colorado, indicatif interurbain des Etats-Unis.
Série (<i>range</i>)	Il s'agit d'une série d'adresses, y compris la première et la dernière de la série.

6.3.2 Informations de routage (*routeInformation*)

La structure "informations de routage" figurant dans l'élément "canevas" (*template*) (le champ *routeInfo*) contient les champs suivants:

Champ	Description
Type de message (<i>messageType</i>)	Type de message à émettre lors d'une tentative de résolution d'une adresse spécifique utilisant ce canevas. Les types possibles sont les suivants "émission de demande d'accès" (<i>sendAccessRequest</i>), "émission d'établissement" (<i>sendSetup</i>) et "inexistant" (<i>nonExistent</i>) (indique que l'adresse n'existe pas).
Spécificité d'appel (<i>callSpecific</i>)	Si ce champ est mis à la valeur TRUE (Vrai), chaque appel vers cette route doit obligatoirement faire l'objet d'une autorisation, ce qui implique la nécessité pour le message de demande d'accès de contenir les informations propres à l'appel. Ce champ booléen a une signification uniquement quand le champ <i>messageType</i> prend la valeur <i>sendAccessRequest</i> ; sinon <i>callSpecific</i> sera mis à la valeur FALSE (Faux).

Spécification d'utilisation (<i>usageSpec</i>)	Lorsqu'il est présent, ce champ spécifie les messages "d'indication d'utilisation" qui seront envoyés en ce qui concerne les appels vers cette route.
Informations de tarification (<i>priceInfo</i>)	Liste d'informations de tarification pour cet acheminement particulier (se référer aux informations de tarification ci-dessous). Il convient de noter que des structures de tarification différentes sont décrites dans des structures "informations de routage" multiples.
Contacts (<i>contacts</i>)	Informations de contact concernant l'élément qui acceptera le message tel qu'il est spécifié dans le champ "type de message" <i>messageType</i> des "informations de routage" <i>routeInfo</i> . Les informations de contact peuvent être fournies sous la forme d'une liste de contacts possibles (se référer à la description des informations de contact ci-dessous).
Type (<i>type</i>)	Type de point de terminaison pouvant recevoir l'appel. En cas de routage par portier, ce champ indique les points de terminaison desservis par le portier plutôt que le portier lui-même.
Ensemble de caractéristiques (<i>featureSet</i>)	Spécifie les ensembles de caractéristiques génériques que cette route prend en charge et les caractéristiques génériques qu'elle souhaite dans un point d'extrémité distant. Les informations sur les caractéristiques génériques spécifiées à ce niveau s'appliquent à toutes les informations de contact applicables à cet élément d'information de routage. Le cadre générique d'extensibilité est décrit dans les Recs. H.323 et H.225.0.
Identificateur de circuit (<i>circuitID</i>)	S'il est présent, contient les informations de circuit RCC qui s'appliquent à un appel spécifique. Les informations sur le circuit spécifiées à ce niveau s'appliquent à toutes les informations de contact applicables à cet élément d'information de routage.
Circuits pris en charge (<i>supportedCircuits</i>)	S'il est présent, contient les valeurs d'identificateur de circuit pour les circuits RCC pris en charge dans un domaine ou une zone pour le modèle de destination. Cela permet à un élément homologue d'annoncer la prise en charge des informations de circuit de destination aux éléments homologues distants. Les informations sur le circuit spécifiées à ce niveau s'appliquent à toutes les informations de contact applicables à cet élément d'information de routage.

6.3.3 Informations de tarification (*pricingInformation*)

Les informations de tarification constituent un élément figurant dans la structure "informations de routage" (champ *priceInfo*). Les informations de tarification sont définies par les structures "spécification d'informations de tarification" (*PriceInfoSpec*) et "élément de prix" (*PriceElement*).

La structure "spécification d'informations de tarification" contient les champs suivants:

Champ	Description
Devise (<i>currency</i>)	Désignation de devise conformément à l'ISO 4217.
Echelle de devise (<i>currencyScale</i>)	Position de la virgule implicite, par rapport à la gauche d'un champ de prix. Si, par exemple, la devise est spécifiée en dollars EU, un champ <i>currencyScale</i> égal à 2 indique que le montant indiqué dans le champ <i>priceElement</i> "élément de prix" est exprimé en cents EU.
Valable à partir de (<i>validFrom</i>)	Date et heure UTC de début de validité de ces informations.
Valable jusqu'à (<i>validUntil</i>)	Date et heure UTC de fin de validité de ces informations.

Heures depuis (<i>hoursFrom</i>)	Heure dans la journée à partir de laquelle ce tarif s'applique.
Heures jusqu'à (<i>hoursUntil</i>)	Heure dans la journée à partir de laquelle ce tarif ne s'applique plus. Ce champ peut être inférieur au champ "heures depuis", ce qui indique un tarif d'étendue horaire nulle.
Elément de prix (<i>priceElement</i>)	Liste optionnelle "d'éléments de prix" additifs constituant le tarif.
Formule de prix (<i>priceFormula</i>)	Chaîne optionnelle contenant une formule de tarification utilisée en variante à la structure "élément de prix".

La structure "élément de prix" (*PriceElement*) contient les champs suivants:

Champ	Description
Montant (<i>amount</i>)	Incrément de comptage. Le compteur est incrémenté d'une unité pour chaque <i>quantum</i> ou fraction de <i>quantum</i> .
Quantum (<i>quantum</i>)	Nombre d'unités auxquelles s'applique le "montant". Une valeur de 60, avec des "unités" <i>units</i> exprimées en secondes indique, par exemple, que l'appel est taxé toutes les minutes ou fractions de minute. Si le champ unités est mis à l'une des valeurs " <i>initial</i> ", " <i>minimum</i> " ou " <i>maximum</i> ", le champ "unités" est sans intérêt, et sa valeur ne doit pas être prise en compte par le destinataire.
Unités (<i>units</i>)	Unité utilisée pour exprimer le quantum: • <i>seconds</i> – secondes de durée de l'appel; • <i>packets</i> – nombre de paquets émis ou reçus; • <i>bytes</i> – nombre d'octets émis ou reçus; • <i>initial</i> – taxation initiale au moment de la connexion; • <i>minimum</i> – taxation minimale d'un appel; • <i>maximum</i> – taxation maximale d'un appel.

6.3.4 Informations de contact (*ContactInformation*)

La structure "informations de contact" est un élément de la structure "informations de routage" (champ "*contacts*").

Champ	Description
Adresse de transport (<i>transportAddress</i>)	Adresse de pseudonyme (par exemple, une adresse de transport ou un identificateur URL) à destination de laquelle doit être émis le message spécifié dans le champ "type de message" (<i>messageType</i>) de la structure "informations de routage". Dans tous les cas où cela est possible, l'utilisation d'une adresse de transport est obligatoire.
Priorité (<i>priority</i>)	Lorsque plusieurs contacts figurent dans la liste, le champ "priorité" (<i>priority</i>) spécifie l'ordre des tentatives de contact multiple. Les contacts de la liste peuvent partager une même priorité, par exemple lorsqu'il n'existe pas de préférence pour l'ordre des tentatives de contact. Une priorité égale à 0 indique la priorité la plus élevée (premier choix).
Qualité de service du transport (<i>transportQoS</i>)	Indique qui a la responsabilité de la réservation des ressources pour l'appel établi par ce contact.
Sécurité (<i>security</i>)	Mécanisme de sécurité indiquant l'ordre de préférence à observer pour communiquer avec le contact.

Jetons d'accès (<i>accessTokens</i>)	Ensemble de jetons qui seront transmis dans le message adressé à ce contact ("Etablissement" ou "Demande d'accès"). Ces jetons seront également envoyés dans les messages "Indication d'utilisation" relatifs aux appels utilisant ce canevas.
Appels multiples (<i>multipleCalls</i>)	Ce champ n'a de signification que si la valeur du champ "type de message" (<i>messageType</i>) de la structure "informations de routage" (<i>RouteInformation</i>) est "émission d'établissement" (<i>sendSetup</i>). Si "appels multiples" (<i>multipleCalls</i>) est à "Vrai" (<i>TRUE</i>), le contact a la capacité d'appels multiples sur une connexion de signalisation d'appel unique. A "Faux" (<i>FALSE</i>), le contact n'a pas cette capacité.
Ensemble de caractéristiques (<i>featureSet</i>)	Spécifie les ensembles de caractéristiques génériques que prend en charge l'entité associée à cet élément "informations de contact" (<i>ContactInformation</i>) et les caractéristiques génériques dont elle a besoin et qu'elle souhaite à un point d'extrémité distant. Le cadre générique d'extensibilité est présenté dans les Recs. UIT-T H.323 et H.225.0.
Identificateur de circuit (<i>circuitID</i>)	S'il est présent, contient les informations de circuit RCC qui s'appliquent à un appel spécifique. Les informations sur le circuit spécifiées à ce niveau s'appliquent au contact correspondant à cet élément d'information de contact (<i>ContactInformation</i>).
Circuits pris en charge (<i>supportedCircuits</i>)	S'il est présent, contient les valeurs d'identificateur de circuit pour les circuits RCC pris en charge dans un domaine ou une zone pour le modèle de destination. Cela permet à un élément homologue d'annoncer la prise en charge des informations de circuit de destination aux éléments homologues distants. Les informations sur le circuit spécifiées à ce niveau s'appliquent au contact correspondant à cet élément d'information de contact (<i>ContactInformation</i>).

6.4 Structures communes

Les structures définies dans le présent paragraphe apparaissent dans un grand nombre de messages.

6.4.1 Autre élément homologue (*AlternatePE*)

Champ	Description
Adresse de contact (<i>contactAddress</i>)	Adresse de transport d'un autre élément homologue (l'adresse à destination de laquelle doivent être envoyés les messages de ce protocole).
Priorité (<i>priority</i>)	Lorsque des variantes multiples figurent dans la liste, le champ "priorité" (<i>priority</i>) spécifie l'ordre dans lequel ces variantes doivent être essayées. Les variantes dans la liste peuvent partager une même priorité, par exemple lorsqu'il n'existe pas de préférence pour le choix des variantes. Une priorité égale à 0 indique la priorité la plus élevée (premier choix).
Identificateur d'élément (<i>elementIdentifier</i>)	Chaîne de caractères Unicode utilisée comme identifiant par cet autre élément homologue.

6.4.2 Informations sur un participant

Cette structure contient des informations relatives à un participant (à l'origine ou à destination) de l'appel.

Champ	Description
Adresse logique (<i>logicalAddress</i>)	Adresses de format type adresse de messagerie électronique ou numéro E.164 permettant d'identifier le participant.
Identificateur de domaine (<i>domainIdentifier</i>)	Adresse de pseudonyme identifiant le domaine administratif à l'origine ou à destination de l'appel. Lorsque plusieurs domaines sont impliqués dans l'établissement d'un appel, il faut indiquer le domaine qui a fait office d'origine ou de destination du point de vue de l'expéditeur.
Adresse de transport (<i>transportAddress</i>)	Adresse de transport du point d'extrémité.
Type de point d'extrémité (<i>endpointType</i>)	Indications détaillées concernant le type de point d'extrémité et les capacités correspondantes.
Information d'utilisateur (<i>userInfo</i>)	Information concernant l'utilisateur à l'origine de l'appel. Voir "Information d'utilisateur" ci-après.
Fuseau horaire (<i>timeZone</i>)	Indication du fuseau horaire du participant, tel qu'il s'applique aux fins de tarification. Si l'appel provient d'une passerelle, le fuseau horaire de la passerelle doit être transmis. Exprimé en secondes par rapport au temps universel coordonné.

6.4.3 Informations sur l'appel

Cette structure contient des informations permettant d'identifier un appel particulier.

Champ	Description
Identificateur d'appel (<i>callIdentifier</i>)	Permet d'identifier l'appel de façon univoque. Constituera l'identificateur d'appel associé au même appel que dans les messages RAS et de signalisation d'appel.
Identificateur de conférence (<i>conferenceID</i>)	Permet d'identifier de façon univoque la conférence à laquelle participe l'appel. Constituera l'identificateur de conférence associé au même appel que dans les messages RAS et de signalisation d'appel.
Identificateur de circuit (<i>circuitID</i>)	S'il est présent, contient les informations de circuit RCC qui s'appliquent à l'appel.

6.4.4 Informations d'utilisateur

Cette structure contient des informations permettant d'identifier l'utilisateur représenté par un participant à l'appel.

Champ	Description
Identificateur d'utilisateur (<i>userIdentifier</i>)	Adresse de pseudonyme qui identifie l'utilisateur de manière univoque.
Authentificateur d'utilisateur (<i>userAuthenticator</i>)	Jetons cryptés permettant une authentification sécurisée.

6.4.5 Spécification d'utilisation

Cet élément décrit les paramètres dont la notification est nécessaire dans les messages "indication d'utilisation". Les appels auxquels cette spécification est applicable sont déterminés par le contexte du message qui contient l'élément "spécification d'utilisation" (*UsageSpecification*).

Champ	Description
Envoyer à (<i>sendTo</i>)	Élément homologue vers lequel les messages d'indication d'utilisation doivent être envoyés. Si l'expéditeur a des relations de service avec cet élément homologue, il s'agit de l'identificateur d'élément renvoyé dans le message de confirmation de service.
Quand (<i>when</i>)	Spécifie les étapes de l'appel pour lesquelles les indications en question doivent être envoyées, ainsi que la fréquence de ces envois: • jamais – arrêter d'envoyer des messages; • commencer – au début de l'appel; • cesser – à la fin de l'appel, ou après; • période – envoyer périodiquement les messages en question pendant la durée de l'appel. Période indiquée en secondes; • échec – notifier les tentatives d'appel ayant échoué.
Champs obligatoires (<i>required</i>)	Liste d'identificateurs des champs qui <i>doivent</i> obligatoirement être présents dans les messages UsageIndication. L'expéditeur du message information sur l'utilisation des ressources doit rejeter ou ignorer le message qui contient ce message, s'il ne peut fournir les informations relatives à ces champs.
Champs théoriquement présents (<i>preferred</i>)	Liste d'identificateurs des champs qui <i>devraient</i> être présents dans les messages UsageIndication.
Envoyer à l'adresse d'élément homologue (<i>sendToPEAddress</i>)	Adresse qui, résolue, spécifie l'adresse d'un élément homologue vers lequel les messages UsageIndication seront envoyés. Si la résolution de ce champ produit plusieurs adresses (par exemple lorsqu'une requête de DNS renvoie une liste d'adresses), l'élément homologue n'enverra les messages UsageIndication qu'à un seul élément homologue de la liste. Si l'élément homologue ne réussit pas l'envoi à une seule adresse, il peut choisir une autre adresse de la liste et tenter d'envoyer les messages UsageIndication à cette nouvelle adresse. L'élément homologue peut poursuivre les tentatives avec toutes les adresses de la liste jusqu'à ce qu'il reçoive un message UsageIndicationConfirmation ou UsageIndicationRejection, ou jusqu'à ce qu'il n'y a plus d'adresses dans la liste. A noter que le champ <i>sendToPEAddress</i> est différent du champ <i>sendTo</i> de <i>UsageSpecification</i>. Le champ <i>sendTo</i> est un identificateur. Il peut être l'identificateur élément homologue spécifique (tel que "élément frontière 1"), ou un identificateur qui représente logiquement un ensemble d'éléments homologues (tel que "éléments frontières de mon entreprise"). Le champ <i>sendToPEAddress</i> résout vers une ou plusieurs adresses.

6.4.6 Mode sécurité

Cet élément décrit un profil de sécurité spécifique à utiliser dans les communications entre homologues.

Champ	Description
Authentification (<i>authentication</i>)	Champ indiquant le mécanisme d'authentification devant être utilisé pour cette relation de service. Le mécanisme d'authentification doit être choisi dans l'ensemble fourni dans le message "demande de service".
Intégrité (<i>integrity</i>)	Indique le mécanisme d'intégrité devant être utilisé. Si ce champ est présent, tous les messages ultérieurs renseigneront le champ "valeur de vérification d'intégrité" <i>integrityCheckValue</i> ; le champ "mode d'authentification" <i>AuthenticationMode</i> décrit alors le mode de génération des clés secrètes (par échange de clés Diffie-Hellman ou a priori).
Identificateurs d'algorithme (<i>algorithmOID</i>)	Indique l'algorithme de cryptage pour le mécanisme de sécurité.

6.5 Relation de service

6.5.1 Demande de service (*ServiceRequest*)

Un élément homologue peut établir une relation de service par l'émission d'un message "demande de service" à destination d'un autre élément homologue. La relation définit les mécanismes de sécurité devant être utilisés entre les éléments homologues et permet l'identification d'éléments homologues de remplacement ou de secours. Il convient de noter que cette relation est unidirectionnelle. La sécurité entre les deux éléments est utilisée pour des demandes émises par l'élément qui est à l'origine de la demande de service et pour des réponses émises par le récepteur de la demande de service. Des clés de session peuvent être créées pendant le processus d'établissement de la relation de service. Ces clés resteront valides pendant la durée de vie de la relation de service. Il est possible d'utiliser des jetons à cet effet, tel qu'indiqué dans la Rec. UIT-T H.235.

Le destinataire d'une demande de service peut indiquer d'autres éléments homologues pouvant être utilisés par le récepteur de la demande pour une demande de service de secours. L'établissement d'une relation de service est une procédure facultative, mais il se peut que la stratégie d'un élément homologue impose une telle relation.

Un élément homologue peut émettre un message "demande de service" à destination d'un élément homologue avec lequel il a déjà établi une relation dans le but de remplacer les caractéristiques de la relation existante par de nouvelles caractéristiques. Les relations de service peuvent avoir une durée de vie limitée. Un élément homologue peut alors régénérer la relation en envoyant une nouvelle demande de service.

Champ	Description
Identificateur d'élément (<i>elementIdentifier</i>)	Chaîne identifiant l'élément homologue qui envoie la demande.
Identificateur de domaine (<i>domainIdentifier</i>)	Indique le domaine administratif qui demande la relation de service.
Capacité de sécurité (<i>securityCapability</i>)	Ensemble de mécanismes de sécurité que cet élément homologue peut prendre en charge.
Durée de vie (<i>timeToLive</i>)	Durée de vie proposée, en secondes, pour la relation de service. En l'absence de cette indication, on suppose une durée de vie infinie.
Spécification d'utilisation (<i>usageSpec</i>)	Spécifie les informations sur l'utilisation que l'élément homologue de départ demande à l'élément homologue destinataire d'envoyer pour tous les appels entre les éléments homologues de départ et de destination.

6.5.2 Confirmation de service (*ServiceConfirmation*)

Lorsqu'il reçoit un message "demande de service", un élément homologue répond au moyen d'un message "confirmation de service" pour indiquer qu'il accepte l'établissement d'une relation de service. Chaque nouvelle relation de service est identifiée par un identificateur de service. Chaque fois qu'un élément homologue reçoit un message "demande de service" sans identificateur de service, il attribue un identificateur de service exclusif et renvoie celui-ci à l'expéditeur du message de demande dans le message "confirmation de service". Si l'élément homologue possède déjà une relation de service avec l'élément homologue qui a émis le message "demande de service", l'émission du message "confirmation de service" indique que les caractéristiques de la relation initiale ne sont plus valides et ont été remplacées par les nouvelles caractéristiques. Le message "confirmation de service" contiendra le même identificateur de service que celui envoyé dans le message "demande de service". Un élément homologue qui reçoit un message "demande de service" contenant un identificateur de service qu'il ne reconnaît pas répondra au moyen d'un message "rejet de service".

Champ	Description
Identificateur d'élément (<i>elementIdentifier</i>)	Chaîne identifiant l'élément homologue.
Identificateur de domaine (<i>domainIdentifier</i>)	Désigne le domaine administratif qui répond à la demande.
Variante (<i>alternates</i>)	Liste d'autres éléments homologues pouvant être contactés en cas d'absence de réponse de cet élément homologue.
Mode sécurité (<i>securityMode</i>)	Indique le mécanisme de sécurité à utiliser pour cette relation de service. Le mécanisme de sécurité doit être choisi dans l'ensemble de mécanismes mentionnés dans le message de demande de service.
Durée de vie (<i>timeToLive</i>)	Durée de vie exprimée en secondes de la relation de service déterminée par l'élément homologue serveur.
Spécification d'utilisation (<i>usageSpec</i>)	Spécifie les informations d'utilisation que l'élément homologue peut prendre en charge pour tous les appels entre les éléments homologues de départ et de destination.

6.5.3 Rejet de service (*ServiceRejection*)

Lorsqu'il reçoit un message "demande de service", un élément homologue répond au moyen d'un message "rejet de service" pour indiquer qu'il refuse l'établissement d'une relation de service. Si l'élément homologue possède déjà une relation de service avec l'élément homologue qui a émis le message "demande de service", l'émission du message "rejet de service" indique que les nouvelles caractéristiques ont été rejetées, mais la relation initiale reste valide.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande de service par l'élément homologue. Les valeurs possibles sont les suivantes: • service indisponible (<i>serviceUnavailable</i>) – Cet élément homologue n'est pas disponible actuellement pour le service; • renvoi du service (<i>serviceRedirected</i>) – La tentative doit être renouvelée en utilisant la liste des autres éléments homologues; • sécurité (<i>security</i>) – Cet élément homologue ne peut pas prendre en charge les mécanismes de sécurité proposés dans le message "demande de service"; • continuation (<i>continue</i>) – Indique le message suivant de demande de service à envoyer, afin de poursuivre la procédure à plusieurs étapes d'échange de clés; • non défini (<i>undefined</i>) – Le motif du rejet de la demande de service ne correspond à aucun des choix précédents; • identificateur de service inconnu (<i>unknownServiceID</i>) – Le champ "identificateur de service" du message "demande de service" n'est pas reconnu par l'élément homologue; • spécification d'utilisation ne pouvant être prises en charge (<i>cannotSupportUsageSpec</i>) – L'élément homologue ne peut se conformer à la spécification d'utilisation proposée; • caractéristique requise (<i>neededFeature</i>) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ <i>neededFeatures</i> du champ <i>MessageCommonInfo</i> de la réponse; • motif de données génériques (<i>genericDataReason</i>) – Indique que le code de motif réel est contenu dans le champ <i>genericData</i> envoyé avec ce message. A noter que <i>genericData</i> peut contenir plusieurs codes de motif; • utilisation indisponible (<i>usageUnavailable</i>) – L'élément homologue ne prend pas en charge le compte rendu d'utilisation; • destination inconnue (<i>unknownUsageSendTo</i>) – L'adresse <i>sendTo</i> ou <i>sendToPEAddress</i> indiquée dans les "spécifications d'utilisation" n'a pas pu être résolue.
Variantes (<i>alternates</i>)	Liste d'autres éléments homologues qui sont éventuellement en mesure d'établir une relation de service. Au moins une liste doit être fournie si le motif est "renvoi du service".

6.5.4 Libération du service (*ServiceRelease*)

L'un ou l'autre des éléments homologues engagés dans une relation de service peut y mettre fin par l'émission d'un message "libération du service".

Champ	Description
Motif (<i>reason</i>)	Motif pour lequel cet élément homologue a mis fin à la relation de service. Les valeurs possibles sont les suivantes: • hors service (<i>outOfService</i>) – L'élément homologue passe dans l'état "hors service"; • maintenance – L'élément homologue est mis hors service pour des raisons de maintenance; • terminé (<i>terminated</i>) – L'élément homologue a décidé de mettre fin à la relation; • expiration (<i>expired</i>) – La durée de vie de la relation de service s'est écoulée.
Variantes (<i>alternates</i>)	Liste d'autres éléments homologues qui sont éventuellement en mesure d'établir une relation de service.

6.6 Répartition des descripteurs

6.6.1 Demande de descripteur (*DescriptorRequest*)

Le message "demande de descripteur" permet à une entité de demander à un élément homologue la fourniture de descripteurs spécifiques.

Champ	Description
Identificateur de descripteur (<i>descriptorID</i>)	Identificateur d'un ou plusieurs descripteurs particuliers demandés par l'émetteur de ce message.

6.6.2 Confirmation de descripteur (*DescriptorConfirmation*)

Le message "confirmation de descripteur" constitue la réponse positive d'un élément homologue à une demande de descripteur, lorsque cet élément peut interpréter la demande et que les règles d'implémentation permettent l'échange d'informations.

Champ	Description
Descripteur (<i>descriptor</i>)	Champ(s) "descripteur" décrit(s) ci-dessus.

6.6.3 Rejet de descripteur (*DescriptorRejection*)

Un élément homologue peut rejeter une demande de descripteur pour divers motifs.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande de descripteur. Les valeurs possibles sont les suivantes: • taille de paquet dépassée (<i>packetSizeExceeded</i>) – La taille de la réponse est supérieure à la taille maximale de paquet, de sorte que le demandeur devrait émettre la demande en utilisant un autre mécanisme de transport (il doit, par exemple, utiliser le protocole TCP à la place du protocole UDP); • identificateur non valide (<i>illegalID</i>) – Le destinataire de la demande de descripteur ne possède aucun enregistrement au sujet du descripteur concerné; • sécurité (<i>security</i>) – La demande de descripteur ne répond pas aux critères de sécurité du destinataire;

- dépassement du compteur de bonds (*hopCountExceeded*) – Le compteur de bonds a atteint la valeur nulle et aucune information n'est disponible;
- non disponible (*unavailable*) – Le destinataire ne peut fournir les descripteurs. Il convient alors d'utiliser une méthode de fourniture statique ou hors-bande;
- pas de relation de service (*noServiceRelationship*) – Le destinataire échangera cette information seulement après établissement d'une relation de service;
- non défini (*undefined*) – Le motif du rejet de la demande de descripteur ne correspond à aucun des choix précédents;
- caractéristique requise (*neededFeature*) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ *neededFeatures* du champ *MessageCommonInfo* de la réponse;
- motif de données génériques (*genericDataReason*) – Indique que le code de motif réel est contenu dans le champ *genericData* envoyé avec ce message. A noter que *genericData* peut contenir plusieurs codes de motif;
- identificateur de service inconnu (*unknownServiceID*) – Le champ "identificateur de service" du message *DescriptorIDRequest* n'est pas reconnu par l'élément homologue.

Identificateur de descripteur (<i>descriptorID</i>)	Identificateur du descripteur particulier fourni dans cette réponse.
--	--

6.6.4 Demande d'identificateur de descripteur (*DescriptorIDRequest*)

La demande d'identificateur de descripteur permet à une entité d'interroger un élément homologue au sujet de la liste des identificateurs de descripteur de son domaine administratif.

6.6.5 Confirmation d'identificateur de descripteur (*DescriptorIDConfirmation*)

Un message "confirmation d'identificateur de descripteur" constitue la réponse positive d'un élément homologue à un message "demande d'identificateur de descripteur". Un élément homologue peut, une fois qu'il a reçu le message "demande d'identificateur de descripteur", émettre le message "demande de descripteur" pour demander la transmission des descripteurs.

Champ	Description
Informations de descripteur (<i>descriptorInfo</i>)	Liste d'informations de descripteur dont chaque élément identifie sans ambiguïté le descripteur et l'instant de sa dernière modification.

6.6.6 Rejet d'identificateur de descripteur (*DescriptorIDRejection*)

Un élément homologue peut rejeter une demande d'identificateur de descripteur pour divers motifs.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande d'identificateur de descripteur. Les valeurs possibles sont les suivantes: • pas de descripteurs (<i>noDescriptors</i>) – L'élément homologue ne dispose pas de descripteurs dont il peut rendre compte; • sécurité (<i>security</i>) – La demande d'identificateur de descripteur ne répond pas aux critères de sécurité du destinataire; • dépassement du compteur de bonds (<i>hopCountExceeded</i>) – Le compteur de bonds a atteint la valeur nulle et aucune information n'est disponible; • non disponible (<i>unavailable</i>) – Le destinataire ne peut fournir les descripteurs. Il convient alors d'utiliser une méthode de fourniture statique ou hors-bande; • pas de relation de service (<i>noServiceRelationship</i>) – Le destinataire échangera cette information seulement après établissement d'une relation de service; • non défini (<i>undefined</i>) – Le motif du rejet de la demande d'identificateur de descripteur ne correspond à aucun des choix précédents; • caractéristique requise (<i>neededFeature</i>) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ <i>neededFeatures</i> du champ <i>MessageCommonInfo</i> de la réponse; • motif de données génériques (<i>genericDataReason</i>) – Indique que le code de motif réel est contenu dans le champ <i>genericData</i> envoyé avec ce message. A noter que <i>genericData</i> peut contenir plusieurs codes de motif; • identificateur de service inconnu (<i>unknownServiceID</i>) – Le champ "identificateur de service" du message <i>DescriptorIDRequest</i> n'est pas reconnu par l'élément homologue.

6.6.7 Mise à jour de descripteur (*DescriptorUpdate*)

Le message "mise à jour de descripteur" constitue la notification d'un élément homologue indiquant que les informations d'adresse ont été modifiées. Un élément homologue peut également émettre ce message pendant l'initialisation. Un élément homologue qui reçoit le message "mise à jour de descripteur" peut demander des informations à l'élément identifié dans ce message.

Champ	Description
Emetteur (<i>sender</i>)	Un élément qui reçoit la mise à jour de descripteur peut émettre une demande à destination de cette adresse (par exemple, une adresse de transport ou un identificateur URL).
Informations de mise à jour (<i>updateInfo</i>)	Liste de mises à jour dont chaque élément fournit le descripteur ou l'identificateur de descripteur mis à jour. L'élément de la liste indique également si le descripteur a été modifié, ajouté ou supprimé.

6.6.8 Accusé de réception de mise à jour de descripteur (*DescriptorUpdateAcknowledgement*)

Un élément homologue doit accuser réception d'un message "mise à jour de descripteur" en émettant le message "accusé de réception de mise à jour de descripteur". Le numéro de séquence utilisé dans l'accusé de réception doit être identique à celui reçu dans le message "mise à jour de descripteur". Un élément homologue ne doit pas fournir d'accusé de réception d'un message "mise à jour de descripteur" transmis par multidiffusion.

6.7 Résolution d'adresses

6.7.1 Demande d'accès (*AccessRequest*)

Un élément homologue peut émettre un message "demande d'accès" à destination d'un autre élément homologue pour demander la résolution d'une adresse de pseudonyme spécifique.

Champ	Description
Informations de destination (<i>destinationInfo</i>)	Adresse devant être résolue.
Informations d'origine (<i>sourceInfo</i>)	Informations sur le participant qui a déclenché l'appel faisant l'objet d'une demande d'accès.
Informations propres à l'appel (<i>callInfo</i>)	Champ permettant d'identifier l'appel particulier faisant l'objet d'une demande d'autorisation d'accès. En l'absence de cette indication, la demande porte sur des appels non définis vers les destinations spécifiées.
Spécification d'utilisation (<i>usageSpec</i>)	Champ indiquant les messages d'utilisation que le participant à l'origine de l'appel demande au participant qui répond et qui concernent l'appel faisant l'objet d'une demande d'accès dans ce message. Utilisé uniquement en présence du champ <i>CallInfo</i> .
Protocoles souhaités (<i>desiredProtocols</i>)	Ce champ identifie, par ordre de préférence, les types de protocole recherchés par l'extrémité pour son appel (par exemple voix ou télécopie). Une entité de résolution peut utiliser ce champ pour localiser une extrémité qui prend également en charge le protocole, compte tenu de l'ordre de préférence. L'entité appelée ignorera le champ <i>supportedPrefixes</i> de cette structure.

6.7.2 Confirmation d'accès (*AccessConfirmation*)

Un élément homologue renvoie dans le message "confirmation d'accès" les informations demandées dans le message "demande d'accès".

Champ	Description
Canevas (<i>templates</i>)	Liste des canevas correspondant aux attributs de la demande d'accès.
Réponse partielle (<i>partialResponse</i>)	Ce message contient, en cas de réponse positive, certaines parties des informations disponibles. La totalité des informations n'a pas été émise pour respecter la taille maximale de paquet. La totalité des informations doit être récupérée en utilisant un autre type de transport (par exemple, le protocole TCP).
Protocoles pris en charge (<i>supportedProtocols</i>)	Ce champ identifie les types de protocole pris en charge (tels que voix ou télécopie).
Commande de service (<i>serviceControl</i>)	Tel que défini dans la Rec. UIT-T H.225.0 afin d'être utilisé comme cela est décrit dans l'Annexe K/H.323.

6.7.3 Rejet d'accès (*AccessRejection*)

Un élément homologue peut rejeter une demande de descripteur pour divers motifs.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande de descripteur. Les valeurs possibles sont les suivantes: • pas de correspondance (<i>noMatch</i>) – La destination spécifiée dans la demande d'accès ne peut pas être résolue; • taille de paquet dépassée (<i>packetSizeExceeded</i>) – La taille de la réponse dépasse la taille maximale de paquet, de sorte que le demandeur doit émettre la demande en utilisant un autre mécanisme de transport (il doit, par exemple, utiliser le protocole TCP à la place du protocole PDU); • sécurité (<i>security</i>) – La demande de descripteur ne répond pas aux critères de sécurité du destinataire; • dépassement du compteur de bonds (<i>hopCountExceeded</i>) – Le compteur de bonds a atteint la valeur nulle et aucune information n'est disponible; • pas de relation de service (<i>noServiceRelationship</i>) – Le destinataire échangera cette information seulement après établissement d'une relation de service; • besoin d'informations d'appel (<i>needCallInformation</i>) – La demande ne contenait pas d'informations spécifiques propres à l'appel; • non défini (<i>undefined</i>) – Le motif du rejet de la demande de descripteur ne correspond à aucun des choix précédents; • caractéristique requise (<i>neededFeature</i>) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ <i>neededFeatures</i> du champ <i>MessageCommonInfo</i> de la réponse; • motif de données génériques (<i>genericDataReason</i>) – Indique que le code de motif réel est contenu dans le champ <i>genericData</i> envoyé avec ce message. A noter que <i>genericData</i> peut contenir plusieurs codes de motif; • destination pas disponible (<i>destinationUnavailable</i>) – La destination a été résolue mais n'est pas disponible; • pseudonymes incohérents (<i>aliasesInconsistent</i>) – Plusieurs pseudonymes identifient des destinations distinctes; • ressource pas disponible (<i>resourceUnavailable</i>) – Une ou plusieurs ressources ne sont pas disponibles; • adresse incomplète (<i>incompleteAddress</i>) – La destination ne peut pas être clairement identifiée; • identificateur de service inconnu (<i>unknownServiceID</i>) – Le champ "identificateur de service" du message "demande de service" n'est pas reconnu par l'élément homologue; • utilisation indisponible (<i>usageUnavailable</i>) – L'élément homologue ne prend pas en charge la notification d'utilisation;

- spécification d'utilisation ne pouvant être prises en charge (*cannotSupportUsageSpec*) – L'élément homologue ne peut se conformer à la spécification d'utilisation proposée;
- destination inconnue (*unknownUsageSendTo*) – L'adresse *sendTo* ou *sendToPEAddress* indiquée dans les "spécifications d'utilisation" n'a pas pu être résolue.

Commande de service (<i>serviceControl</i>)	Tel que défini dans la Rec. UIT-T H.225.0 afin d'être utilisé comme cela est décrit dans l'Annexe K/H.323.
--	--

6.8 Demande en cours (*RequestInProgress*)

Un élément homologue peut renvoyer le message "demande en cours" dans le but d'indiquer que la durée nécessaire pour répondre à une demande peut dépasser la durée prévue normalement pour une réponse. Le numéro de séquence doit être identique à celui figurant dans la demande concernée par l'envoi de ce message.

Champ	Description
Retard (<i>delay</i>)	Laps de temps, exprimé en millisecondes, prévu pour la réponse de l'élément homologue à la demande initiale.
Commande de service (<i>serviceControl</i>)	Tel que défini dans la Rec. UIT-T H.225.0 afin d'être utilisé comme cela est décrit dans l'Annexe K/H.323.

6.9 Messages non normalisés

6.9.1 Demande non normalisée (*NonStandardRequest*)

Le message "demande non normalisée" peut être émis par un élément homologue dans le cas d'un message de demande qui n'est pas défini dans la présente Recommandation. Les informations non normalisées sont véhiculées dans l'élément "non normalisé" défini dans le champ "informations communes" (*MessageCommonInfo*).

6.9.2 Confirmation non normalisée (*NonStandardConfirmation*)

Le message "confirmation non normalisée" peut être émis par un élément homologue en réponse à un message "demande non normalisée". Les informations non normalisées sont véhiculées dans l'élément "non normalisé" défini dans le champ "informations communes" (*MessageCommonInfo*).

6.9.3 Rejet non normalisé (*NonStandardRejection*)

Le message "rejet non normalisé" peut être émis par un élément homologue en réponse à un message "demande non normalisée". Les informations non normalisées sont véhiculées dans l'élément "non normalisé" défini dans le champ "informations communes" (*MessageCommonInfo*).

Champ	Description
Motif (<i>reason</i>)	Motifs du rejet de la demande. Les valeurs possibles sont les suivantes: • non prise en charge (<i>notSupported</i>) – Le destinataire sait qu'il s'agit d'une demande non normalisée, mais ne comprend pas ou ne prend pas en charge les données non normalisées; • pas de relation de service (<i>noServiceRelationship</i>) – Le destinataire échangera cette information seulement après établissement d'une relation de service; • non défini (<i>undefined</i>) – Le motif du rejet de la demande non normalisée ne correspond à aucun des choix précédents;

- caractéristique requise (*neededFeature*) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ *neededFeatures* du champ *MessageCommonInfo* de la réponse;
- motif de données génériques (*genericDataReason*) – Indique que le code de motif réel est contenu dans le champ *genericData* envoyé avec ce message. A noter que *genericData* peut contenir plusieurs codes de motif;
- identificateur de service inconnu (*unknownServiceID*) – Le champ "identificateur de service" du message "demande non normalisée" n'est pas reconnu par l'élément homologue.

6.10 Réponse à un message non reconnu (*UnknownMessageResponse*)

Un élément homologue qui ne comprend pas un message reçu doit répondre à l'émetteur au moyen d'un message "réponse à un message non reconnu". L'élément homologue ne doit pas utiliser ce message si un autre message fournit une réponse adéquate (un message "rejet de descripteur" constitue, par exemple, une réponse adéquate à une demande de descripteur contenant un identificateur de descripteur non valide).

Champ	Description
Message non reconnu (<i>unknownMessage</i>)	Contenu du message non reconnu.
Motif (<i>reason</i>)	Motif d'utilisation du message "réponse à un message non reconnu". Les valeurs possibles sont les suivantes: • non compris (<i>notUnderstood</i>) – Le message n'a pas été compris; • non défini (<i>undefined</i>) – Le motif de l'émission du message "réponse à un message non reconnu" ne correspond à aucun des choix précédents.

6.11 Rapports d'utilisation

6.11.1 Demande d'utilisation

Ce message permet de demander au destinataire d'envoyer des messages d'indication d'utilisation concernant un appel particulier.

Champ	Description
Informations propres à l'appel (<i>callInfo</i>)	Appel pour lequel la demande d'indications est formulée.
Spécification d'utilisation (<i>usageSpec</i>)	Spécifie quand les indications doivent parvenir et quel doit être leur contenu.

6.11.2 Confirmation d'utilisation

Le message de confirmation d'utilisation est envoyé en réponse à un message de demande d'utilisation, afin d'indiquer que le destinataire a accepté la demande et enverra des indications d'utilisation.

6.11.3 Rejet d'utilisation

Le message de rejet d'utilisation est envoyé en réponse à un message de demande d'utilisation, afin d'indiquer que le destinataire a rejeté la demande et n'enverra pas subséquemment d'indications d'utilisation.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande d'utilisation par l'élément homologue. Les valeurs possibles sont les suivantes: • appel non valide (<i>invalidCall</i>) – L'appel spécifié dans la demande d'utilisation n'est pas un appel reconnu; • sécurité (<i>security</i>) – La demande d'utilisation n'était pas conforme aux exigences de sécurité du destinataire; • non disponible (<i>unavailable</i>) – Le destinataire n'a pas d'information d'utilisation pour l'appel demandé; • pas de relation de service (<i>noServiceRelationship</i>) – Le destinataire n'échangera cette information qu'après l'établissement d'une relation de service; • non défini (<i>undefined</i>) – Le motif du rejet de la demande d'utilisation ne correspond à aucun autre choix; • caractéristique requise (<i>neededFeature</i>) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ <i>neededFeatures</i> du champ <i>MessageCommonInfo</i> de la réponse; • motif de données génériques (<i>genericDataReason</i>) – Indique que le code de motif réel est contenu dans le champ <i>genericData</i> envoyé avec ce message. A noter que <i>genericData</i> peut contenir plusieurs codes de motif; • identificateur de service inconnu (<i>unknownServiceID</i>) – Le champ "identificateur de service" du message <i>usageRequest</i> n'est pas reconnu par l'élément homologue.

6.11.4 Indication d'utilisation

Ce message notifie des indications détaillées sur l'appel ainsi que des informations concernant l'utilisation des ressources. Il est envoyé en rapport avec le dernier élément de spécification d'utilisation (*UsageSpecification*) reçu par l'élément homologue et concernant l'appel.

Champ	Description
Informations propres à l'appel (<i>callInfo</i>)	Appel auquel l'indication se rapporte.
Jetons d'accès (<i>accessTokens</i>)	Jetons d'accès relatifs à l'appel. Ces jetons ont été reçus dans le canevas d'adresse utilisé pour l'appel et communiqué dans le message de demande d'accès/d'établissement relatif au même appel.
Rôle de l'expéditeur (<i>senderRole</i>)	Rôle de l'expéditeur de l'indication: • origine – participant à l'origine de l'appel; • destination – participant destinataire; • non normalisé – autre.

Etat d'appel d'utilisation (<i>usageCallStatus</i>)	Etat actuel de l'appel: • préconnexion; • appel en cours; • appel achevé; • enregistrement perdu.
Information de la source (<i>srcInfo</i>)	Numéro E.164 ou adresse de messagerie électronique de l'appelant.
Adresse de la destination (<i>destAddress</i>)	Numéro E.164 ou adresse de messagerie électronique de l'appelé.
Instant de début (<i>startTime</i>)	Instant du début de l'appel exprimé en temps universel coordonné (UTC, <i>coordinated universal time</i>). Concerne uniquement les appels ayant franchi l'étape de l'établissement. Si plusieurs types de média sont utilisés pendant l'appel, chacun doit annoncer un instant de début différent, soit celui où ce flux de média a commencé. Pour les messages périodiques, l'instant de début doit correspondre à l'instant de fin du message précédent.
Instant de fin (<i>endTime</i>)	Instant de fin de l'appel exprimé en temps universel coordonné (UTC). Concerne uniquement les appels terminés. Si plusieurs types de média sont utilisés pendant l'appel, chacun doit annoncer un instant de fin différent, soit celui où ce flux de média s'est terminé. Pour les messages périodiques, l'instant de fin est celui auquel une période de notification se termine.
Cause de fin (<i>terminationCause</i>)	Motif de la fin de l'appel. Concerne uniquement les appels terminés.
Champs d'utilisation (<i>usageFields</i>)	Ensemble de champs d'information. Chaque champ est représenté par un "champ d'utilisation" susceptible d'être normalisé ou non. Les champs d'utilisation normalisés doivent faire l'objet d'un complément d'étude.

6.11.5 Confirmation d'indication d'utilisation

Le message de confirmation d'indication d'utilisation est envoyé en réponse à un message de d'indication d'utilisation, afin d'indiquer que le destinataire a accepté l'indication notifiée.

6.11.6 Rejet d'indication d'utilisation

Le message de rejet d'indication d'utilisation est envoyé en réponse à un message de d'indication d'utilisation, afin d'indiquer que le destinataire a rejeté l'indication notifiée et n'en tiendra pas compte.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande d'utilisation par l'élément homologue. Les valeurs possibles sont les suivantes: • appel inconnu (<i>unknownCall</i>) – L'appel spécifié dans l'indication d'utilisation n'est pas un appel reconnu; • incomplet (<i>incomplete</i>) – L'indication d'utilisation ne contenait pas toute l'information nécessaire pour la spécification d'utilisation qui s'applique à cette indication d'utilisation; • sécurité (<i>security</i>) – L'indication d'utilisation ne répondait pas aux critères de sécurité du destinataire;

- pas de relation de service (*noServiceRelationship*) – Le destinataire n'échangera cette information qu'après l'établissement d'une relation de service;
- non défini (*undefined*) – Le motif du rejet de l'indication d'utilisation ne correspond à aucun autre choix;
- caractéristique requise (*neededFeature*) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ *neededFeatures* du champ *MessageCommonInfo* de la réponse;
- motif de données générique (*genericDataReason*) – Indique que le code de motif réel est contenu dans le champ *genericData* envoyé avec ce message. A noter que *genericData* peut contenir plusieurs codes de motif;
- identificateur de service inconnu (*unknownServiceID*) – Le champ "identificateur de service" du message *UsageIndication* n'est pas reconnu par l'élément homologue.

6.12 Validation

6.12.1 Demande de validation

Un élément homologue qui termine un appel peut envoyer un message de demande de validation à un autre élément homologue pour vérifier la validité de l'origine de l'appel.

Champ	Description
Jetons d'accès (<i>accessToken</i>)	Jetons reçus du participant à l'origine de l'appel attestant l'autorisation d'accès relative à l'appel considéré.
Information de destination (<i>destinationInfo</i>)	Informations détaillées concernant la destination de l'appel.
Informations d'origine (<i>sourceInfo</i>)	Informations concernant le point d'extrémité à l'origine de l'appel.
Informations propres à l'appel (<i>callInfo</i>)	Ce champ identifie l'appel particulier qui fait l'objet d'une demande d'autorisation d'accès.
Spécification d'utilisation (<i>usageSpec</i>)	Lorsqu'il est présent ce champ indique à l'élément homologue adressant les demandes de message qu'il doit recevoir un message d'indication d'utilisation concernant l'appel validé.

6.12.2 Confirmation de validation

Message indiquant que l'appel est validé. L'élément homologue demandeur peut terminer l'appel. L'élément homologue qui effectue la validation peut indiquer des pseudonymes afin de terminer l'appel.

Champ	Description
Information de destination (<i>destinationInfo</i>)	Autres paramètres relatifs à la destination à utiliser par l'élément homologue destinataire.
Spécification d'utilisation (<i>usageSpec</i>)	Lorsqu'il est présent ce champ indique à l'élément homologue adressant les demandes de confirmation qu'il doit recevoir un message d'indication d'utilisation concernant l'appel validé.

6.12.3 Rejet de validation

Message indiquant que l'appel n'est pas valide. L'élément homologue demandeur peut ne pas établir l'appel.

Champ	Description
Motif (<i>reason</i>)	Motif du rejet de la demande. Les valeurs possibles sont les suivantes: • jeton non valide (<i>tokenNotValid</i>) – Les jetons d'accès fournis ne sont pas valides pour l'appel en question; • sécurité (<i>security</i>) – La demande de validation ne répond pas aux critères de sécurité du destinataire; • dépassement du compteur de bonds (<i>hopCountExceeded</i>) – Le compteur de bonds a atteint la valeur nulle et aucune information n'est disponible; • information d'origine manquante (<i>missingSourceInfo</i>) – L'information d'origine fournie ne suffit pas pour valider l'appel; • information de destination manquante (<i>missingDestInfo</i>) – L'information de destination fournie ne suffit pas pour valider l'appel; • pas de relation de service (<i>noServiceRelationship</i>) – Le destinataire échangera cette information seulement après établissement d'une relation de service; • non défini (<i>undefined</i>) – Le motif du rejet de la demande de descripteur ne correspond à aucun des choix précédents; • caractéristique requise (<i>neededFeature</i>) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ <i>neededFeatures</i> du champ <i>MessageCommonInfo</i> de la réponse; • motif de données génériques (<i>genericDataReason</i>) – Indique que le code de motif réel est contenu dans le champ <i>genericData</i> envoyé avec ce message. A noter que <i>genericData</i> peut contenir plusieurs codes de motif; • identificateur de service inconnu (<i>unknownServiceID</i>) – Le champ "identificateur de service" du message <i>ValidationRequest</i> n'est pas reconnu par l'élément homologue.

6.13 Authentification

6.13.1 Demande d'authentification

Un élément homologue peut envoyer un message de demande d'authentification à un autre élément homologue pour authentifier un utilisateur.

Champ	Description
Message d'application (<i>applicationMessage</i>)	Contient un message de protocole d'application qui doit être authentifié.

6.13.2 Confirmation d'authentification

Indique que l'authentification a abouti. Il n'y a pas de champ spécifique défini pour ce message; les informations importantes (jetons, jetons cryptés) sont contenues dans la partie commune du message.

6.13.3 Rejet de l'authentification

Indique que l'authentification a échoué.

Champ	Description
Motif (<i>reason</i>)	C'est le motif du rejet de la demande. Les valeurs possibles sont: • sécurité (<i>security</i>) – La demande de validation ne répond pas aux critères de sécurité du destinataire; • dépassement du compteur de bonds (<i>hopCountExceeded</i>) – Le compteur de bonds a atteint la valeur nulle et aucune information n'est disponible; • pas de relation de service (<i>noServiceRelationship</i>) – Le destinataire échangera cette information seulement après établissement d'une relation de service; • non défini (<i>undefined</i>) – Le motif du rejet de la demande de descripteur ne correspond à aucun des choix précédents; • caractéristique requise (<i>neededFeature</i>) – La demande a échoué parce que l'entité qui fait la demande ne prend pas en charge une ou plusieurs caractéristiques génériques. Les caractéristiques nécessaires sont indiquées dans le champ <i>neededFeatures</i> du champ <i>MessageCommonInfo</i> de la réponse; • motif de données génériques (<i>genericDataReason</i>) – Indique que le code de motif réel est contenu dans le champ <i>genericData</i> envoyé avec ce message. A noter que <i>genericData</i> peut contenir plusieurs codes de motif; • identificateur de service inconnu (<i>unknownServiceID</i>) – Le champ "identificateur de service" du message <i>AuthenticationRequest</i> n'est pas reconnu par l'élément homologue; • sécurité, erreur d'instant de synchronisation (<i>securityWrongSyncTime</i>) – Le demandeur a constaté un problème de sécurité dû à une erreur d'horodateurs. Il pourrait résulter d'un problème de serveur temporel, de perte de synchronisation ou de retard excessif dans le réseau; • sécurité, répétition (<i>securityReplay</i>) – Une agression par répétition a eu lieu. C'est le cas lorsque le même numéro de séquence se produit plus d'une fois pour un horodateur donné; • sécurité, erreur d'identificateur général (<i>securityWrongGeneralID</i>) – Indique une discordance de l'identificateur général dans le message. Peut être causé par une erreur d'adressage; • sécurité, erreur d'identificateur de demandeur (<i>securityWrongSendersID</i>) – Indique une discordance de l'identificateur général dans le message. Peut être causé par une erreur d'entrée d'un utilisateur; • sécurité, échec d'intégrité du message (<i>securityMessageIntegrityFailed</i>) – Le contrôle d'intégrité/signature a échoué. Peut être causé par un mot de passe erroné ou mal orthographié pendant la demande initiale ou l'application d'une clé publique/privée incorrecte, ou par une agression active;

- sécurité, identificateur d'objet erroné (*securityWrongOID*) – Indique une discordance dans les identificateurs d'objet de jeton (normal ou cryptographique) ou dans les identificateurs d'objet d'algorithme cryptographique. Des algorithmes ou profils de sécurité différents sont implémentés.
-

Annexe A

Syntaxe des messages

```
H501-MESSAGES DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
```

```
IMPORTS
AuthenticationMechanism,
TimeStamp,
ClearToken
    FROM H235-SECURITY-MESSAGES
```

```
AliasAddress,
TransportAddress,
ReleaseCompleteReason,
ConferenceIdentifier,
CallIdentifier,
CryptoH323Token,
CryptoToken,
EndpointType,
GatekeeperIdentifier,
GloballyUniqueID,
NonStandardParameter,
NumberDigits,
PartyNumber,
SupportedProtocols,
TransportQOS,
VendorIdentifier,
IntegrityMechanism,
ICV,
FeatureSet,
GenericData,
EnumeratedParameter,
ServiceControlSession,
CircuitInfo,
CircuitIdentifier
    FROM H323-MESSAGES;
```

```
Message ::= SEQUENCE
{
    body      MessageBody,
    common    MessageCommonInfo,
    ...
}
```

```
MessageBody ::= CHOICE
{
    serviceRequest      ServiceRequest,
    serviceConfirmation ServiceConfirmation,
    serviceRejection     ServiceRejection,
    serviceRelease       ServiceRelease,
    descriptorRequest     DescriptorRequest,
```

```

descriptorConfirmation      DescriptorConfirmation,
descriptorRejection         DescriptorRejection,
descriptorIDRequest         DescriptorIDRequest,
descriptorIDConfirmation    DescriptorIDConfirmation,
descriptorIDRejection       DescriptorIDRejection,
descriptorUpdate            DescriptorUpdate,
descriptorUpdateAck         DescriptorUpdateAck,
accessRequest               AccessRequest,
accessConfirmation          AccessConfirmation,
accessRejection             AccessRejection,
requestInProgress           RequestInProgress,
nonStandardRequest          NonStandardRequest,
nonStandardConfirmation     NonStandardConfirmation,
nonStandardRejection        NonStandardRejection,
unknownMessageResponse      UnknownMessageResponse,
usageRequest                UsageRequest,
usageConfirmation           UsageConfirmation,
usageIndication             UsageIndication,
usageIndicationConfirmation UsageIndicationConfirmation,
usageIndicationRejection    UsageIndicationRejection,
usageRejection              UsageRejection,
validationRequest           ValidationRequest,
validationConfirmation       ValidationConfirmation,
validationRejection         ValidationRejection,
...,
authenticationRequest       AuthenticationRequest,
authenticationConfirmation  AuthenticationConfirmation,
authenticationRejection     AuthenticationRejection
}

MessageCommonInfo ::= SEQUENCE
{
    sequenceNumber            INTEGER (0..65535),
    annexGversion             ProtocolVersion, -- set to "H.225.0 Annex G V2"
    hopCount                  INTEGER (1..255),
    replyAddress               SEQUENCE OF TransportAddress OPTIONAL,
                             -- Must be present in request
    integrityCheckValue       ICV OPTIONAL,
    tokens                     SEQUENCE OF ClearToken OPTIONAL,
    cryptoTokens               SEQUENCE OF CryptoH323Token OPTIONAL,
    nonStandard                SEQUENCE OF NonStandardParameter OPTIONAL,
    ...,
    serviceID                  ServiceID OPTIONAL,
    genericData                SEQUENCE OF GenericData OPTIONAL,
    featureSet                 FeatureSet OPTIONAL,
    version                    ProtocolVersion -- current H.501 protocol version
}

ServiceID ::= GloballyUniqueID

--
-- H.501 messages
--

ServiceRequest ::= SEQUENCE
{
    elementIdentifier          ElementIdentifier OPTIONAL,
    domainIdentifier           AliasAddress OPTIONAL,
    securityMode                SEQUENCE OF SecurityMode OPTIONAL,
    timeToLive                  INTEGER (1..4294967295) OPTIONAL,
    ...,
    usageSpec                   UsageSpecification OPTIONAL
}

```

```

SecurityMode ::= SEQUENCE
{
    authentication      AuthenticationMechanism OPTIONAL,
    integrity            IntegrityMechanism OPTIONAL,
    algorithmOIDs       SEQUENCE OF OBJECT IDENTIFIER OPTIONAL,
    ...
}

ServiceConfirmation ::= SEQUENCE
{
    elementIdentifier    ElementIdentifier,
    domainIdentifier     AliasAddress,
    alternates           AlternatePEInfo OPTIONAL,
    securityMode         SecurityMode OPTIONAL,
    timeToLive           INTEGER (1..4294967295) OPTIONAL,
    ...,
    usageSpec           UsageSpecification OPTIONAL
}

ServiceRejection ::= SEQUENCE
{
    reason              ServiceRejectionReason,
    alternates          AlternatePEInfo OPTIONAL,
    ...
}

ServiceRejectionReason ::= CHOICE
{
    serviceUnavailable  NULL,
    serviceRedirected   NULL,
    security            NULL,
    continue            NULL,
    undefined           NULL,
    ...,
    unknownServiceID    NULL,
    cannotSupportUsageSpec NULL, -- Cannot comply with proposed spec
    neededFeature       NULL,
    genericDataReason   NULL,
    usageUnavailable    NULL, -- Usage reporting not supported
    unknownUsageSendTo  NULL  -- Usage sendTo could not be resolved
}

ServiceRelease ::= SEQUENCE
{
    reason              ServiceReleaseReason,
    alternates          AlternatePEInfo OPTIONAL,
    ...
}

ServiceReleaseReason ::= CHOICE
{
    outOfService        NULL,
    maintenance         NULL,
    terminated          NULL,
    expired             NULL,
    ...
}

```

```

DescriptorRequest ::= SEQUENCE
{
    descriptorID          SEQUENCE OF DescriptorID,
    ...
}

DescriptorConfirmation ::= SEQUENCE
{
    descriptor            SEQUENCE OF Descriptor,
    ...
}

DescriptorRejection ::= SEQUENCE
{
    reason                DescriptorRejectionReason,
    descriptorID          DescriptorID OPTIONAL,
    ...
}

DescriptorRejectionReason ::= CHOICE
{
    packetSizeExceeded    NULL,    -- use other transport type
    illegalID             NULL,    -- no descriptor for provided descriptorID
    security              NULL,    -- request did not meet security requirements
    hopCountExceeded      NULL,
    noServiceRelationship NULL,
    undefined             NULL,
    ...,
    neededFeature         NULL,
    genericDataReason     NULL,
    unknownServiceID      NULL    -- The serviceID is not recognized by
                                -- the peer element
}

DescriptorIDRequest ::= SEQUENCE
{
    ...
}

DescriptorIDConfirmation ::= SEQUENCE
{
    descriptorInfo        SEQUENCE OF DescriptorInfo,
    ...
}

DescriptorIDRejection ::= SEQUENCE
{
    reason                DescriptorIDRejectionReason,
    ...
}

DescriptorIDRejectionReason ::= CHOICE
{
    noDescriptors         NULL,    -- no descriptors to report
    security              NULL,    -- request did not meet security requirements
    hopCountExceeded      NULL,
    noServiceRelationship NULL,
    undefined             NULL,

```

```

    ...,
    neededFeature          NULL,
    genericDataReason      NULL,
    unknownServiceID       NULL    -- The serviceID is not recognized by
                                   -- the peer element
}

```

DescriptorUpdate ::= SEQUENCE

```

{
    sender                AliasAddress,
    updateInfo            SEQUENCE OF UpdateInformation,
    ...
}

```

UpdateInformation ::= SEQUENCE

```

{
    descriptorInfo        CHOICE
    {
        descriptorID      DescriptorID,
        descriptor         Descriptor,
        ...
    },
    updateType            CHOICE
    {
        added              NULL,
        deleted            NULL,
        changed            NULL,
        ...
    },
    ...
}

```

DescriptorUpdateAck ::= SEQUENCE

```

{
    ...
}

```

AccessRequest ::= SEQUENCE

```

{
    destinationInfo       PartyInformation,
    sourceInfo             PartyInformation OPTIONAL,
    callInfo              CallInformation OPTIONAL,
    usageSpec             UsageSpecification OPTIONAL,
    ...,
    desiredProtocols       SEQUENCE OF SupportedProtocols OPTIONAL
}

```

AccessConfirmation ::= SEQUENCE

```

{
    templates              SEQUENCE OF AddressTemplate,
    partialResponse        BOOLEAN,
    ...,
    supportedProtocols     SEQUENCE OF SupportedProtocols OPTIONAL,
    serviceControl         SEQUENCE OF ServiceControlSession OPTIONAL
}

```

AccessRejection ::= SEQUENCE

```

{

```

```

        reason          AccessRejectionReason,
        ...,
        serviceControl   SEQUENCE OF ServiceControlSession OPTIONAL
    }

AccessRejectionReason ::= CHOICE
{
    noMatch              NULL,    -- no template matched the destinationInfo
    packetSizeExceeded   NULL,    -- use other transport type
    security              NULL,    -- request did not meet security requirements
    hopCountExceeded     NULL,
    needCallInformation   NULL,    -- Call Information must be specified
    noServiceRelationship NULL,
    undefined             NULL,
    ...,
    neededFeature         NULL,
    genericDataReason     NULL,
    destinationUnavailable NULL,    -- Destination was resolved but is
                                   -- unavailable
    aliasesInconsistent   NULL,    -- Multiple aliases identify distinct
                                   -- destinations
    resourceUnavailable    NULL,    -- One or more required resources are
                                   -- unavailable
    incompleteAddress     NULL,    -- Destination cannot be distinctly
                                   -- identified
    unknownServiceID      NULL,    -- The serviceID is not recognized by
                                   -- the peer element
    usageUnavailable      NULL,    -- Usage reporting not supported
    cannotSupportUsageSpec NULL,    -- Cannot comply with proposed spec
    unknownUsageSendTo    NULL    -- Usage sendTo could not be resolved
}

```

```

UsageRequest ::= SEQUENCE
{
    callInfo      CallInformation,
    usageSpec     UsageSpecification,
    ...
}

```

```

UsageConfirmation ::= SEQUENCE
{
    ...
}

```

```

UsageRejection ::= SEQUENCE
{
    reason      UsageRejectReason,
    ...
}

```

```

UsageIndication ::= SEQUENCE
{
    callInfo          CallInformation,
    accessTokens      SEQUENCE OF AccessToken OPTIONAL,
    senderRole        Role,
    usageCallStatus    UsageCallStatus,
    srcInfo            PartyInformation OPTIONAL,
    destAddress        PartyInformation,
    startTime          TimeStamp OPTIONAL,
    endTime            TimeStamp OPTIONAL,
}

```

```

        terminationCause  TerminationCause OPTIONAL,
        usageFields       SEQUENCE OF UsageField,
        ...
    }

UsageField ::= SEQUENCE
{
    id          OBJECT IDENTIFIER,
    value       OCTET STRING,
    ...
}

UsageRejectReason ::= CHOICE
{
    invalidCall          NULL,
    unavailable          NULL,
    security             NULL,
    noServiceRelationship NULL,
    undefined            NULL,
    ...,
    neededFeature        NULL,
    genericDataReason     NULL,
    unknownServiceID     NULL      -- The serviceID is not recognized by
                                   -- the peer element
}

UsageIndicationConfirmation ::= SEQUENCE
{
    ...
}

UsageIndicationRejection ::= SEQUENCE
{
    reason      UsageIndicationRejectionReason,
    ...
}

UsageIndicationRejectionReason ::= CHOICE
{
    unknownCall          NULL,
    incomplete           NULL,
    security             NULL,
    noServiceRelationship NULL,
    undefined            NULL,
    ...,
    neededFeature        NULL,
    genericDataReason     NULL,
    unknownServiceID     NULL      -- The serviceID is not recognized by
                                   -- the peer element
}

ValidationRequest ::= SEQUENCE
{
    accessToken          SEQUENCE OF AccessToken OPTIONAL,
    destinationInfo      PartyInformation OPTIONAL,
    sourceInfo           PartyInformation OPTIONAL,
    callInfo             CallInformation,
    usageSpec            UsageSpecification OPTIONAL,
    ...
}

```



```

ValidationConfirmation ::= SEQUENCE
{
    destinationInfo      PartyInformation OPTIONAL,
    usageSpec            UsageSpecification OPTIONAL,
    ...
}

ValidationRejection ::= SEQUENCE
{
    reason      ValidationRejectionReason,
    ...
}

ValidationRejectionReason ::= CHOICE
{
    tokenNotValid      NULL,
    security            NULL, -- request did not meet security requirements
    hopCountExceeded   NULL,
    missingSourceInfo   NULL,
    missingDestInfo     NULL,
    noServiceRelationship NULL,
    undefined           NULL,
    ...,
    neededFeature       NULL,
    genericDataReason   NULL,
    unknownServiceID    NULL -- The serviceID is not recognized by
                           -- the peer element
}

RequestInProgress ::= SEQUENCE
{
    delay              INTEGER (1..65535),
    ...,
    serviceControl     SEQUENCE OF ServiceControlSession OPTIONAL
}

NonStandardRequest ::= SEQUENCE
{
    ...
}

NonStandardConfirmation ::= SEQUENCE
{
    ...
}

NonStandardRejection ::= SEQUENCE
{
    reason      NonStandardRejectionReason,
    ...
}

NonStandardRejectionReason ::= CHOICE
{
    notSupported      NULL,
    noServiceRelationship NULL,
    undefined         NULL,
    ...,
    neededFeature      NULL,
    genericDataReason  NULL,

```

```

        unknownServiceID      NULL -- The serviceID is not recognized by
                                -- the peer element
    }
    UnknownMessageResponse ::= SEQUENCE
    {
        unknownMessage      OCTET STRING,
        reason               UnknownMessageReason,
        ...
    }

    UnknownMessageReason ::= CHOICE
    {
        notUnderstood      NULL,
        undefined           NULL,
        ...
    }

    AuthenticationRequest ::= SEQUENCE
    {
        applicationMessage  ApplicationMessage, -- e.g. RAS message in
                                                -- ITU-T Rec. H.323
        ...
    }

    ApplicationMessage ::= OCTET STRING

    AuthenticationConfirmation ::= SEQUENCE
    {
        ...
    }

    AuthenticationRejection ::= SEQUENCE
    {
        reason              AuthenticationRejectionReason,
        ...
    }
    AuthenticationRejectionReason ::= CHOICE
    {
        security                NULL,
        hopCountExceeded        NULL,
        noServiceRelationship    NULL,
        undefined               NULL,
        neededFeature            NULL,
        genericDataReason        NULL,
        unknownServiceID        NULL,
        securityWrongSyncTime    NULL, -- time server problem or network delay
        securityReplay           NULL, -- replay attack encountered
        securityWrongGeneralID   NULL, -- wrong general ID
        securityWrongSendersID   NULL, -- wrong senders ID
        securityIntegrityFailed  NULL, -- integrity check failed
        securityWrongOID         NULL, -- wrong token OIDs or crypto alg OIDs
        ...
    }

--
-- structures common to multiple messages
--

```

```

AddressTemplate ::= SEQUENCE
{
    pattern                SEQUENCE OF Pattern,
    routeInfo              SEQUENCE OF RouteInformation,
    timeToLive             INTEGER (1..4294967295),
    ...,
    supportedProtocols     SEQUENCE OF SupportedProtocols OPTIONAL,
    featureSet             FeatureSet OPTIONAL
}

Pattern ::= CHOICE
{
    specific      AliasAddress,
    wildcard      AliasAddress,
    range         SEQUENCE
        {
            startOfRange    PartyNumber,
            endOfRange      PartyNumber
        },
    ...
}

RouteInformation ::= SEQUENCE
{
    messageType          CHOICE
        {
            sendAccessRequest    NULL,
            sendSetup            NULL,
            nonExistent          NULL,
            ...
        },
    callSpecific          BOOLEAN,
    usageSpec             UsageSpecification OPTIONAL,
    priceInfo             SEQUENCE OF PriceInfoSpec OPTIONAL,
    contacts              SEQUENCE OF ContactInformation,
    type                  EndpointType OPTIONAL,
                        -- must be present if messageType = sendSetup
    ...,
    featureSet            FeatureSet OPTIONAL,
    circuitID             CircuitInfo OPTIONAL,
    supportedCircuits     SEQUENCE OF CircuitIdentifier OPTIONAL
}

ContactInformation ::= SEQUENCE
{
    transportAddress      AliasAddress,
    priority              INTEGER (0..127),
    transportQoS          TransportQOS OPTIONAL,
    security              SEQUENCE OF SecurityMode OPTIONAL,
    accessTokens          SEQUENCE OF AccessToken OPTIONAL,
    ...,
    multipleCalls         BOOLEAN OPTIONAL,
    featureSet            FeatureSet OPTIONAL,
    circuitID             CircuitInfo OPTIONAL,
    supportedCircuits     SEQUENCE OF CircuitIdentifier OPTIONAL
}

```

```

PriceInfoSpec ::= SEQUENCE
{
    currency          IA5String (SIZE(3)),    -- e.g. "USD"
    currencyScale     INTEGER(-127..127),
    validFrom         GlobalTimeStamp OPTIONAL,
    validUntil        GlobalTimeStamp OPTIONAL,
    hoursFrom         IA5String (SIZE(6)) OPTIONAL, -- "HHMMSS" UTC
    hoursUntil        IA5String (SIZE(6)) OPTIONAL, -- "HHMMSS" UTC
    priceElement      SEQUENCE OF PriceElement OPTIONAL,
    priceFormula      IA5String (SIZE(1..2048)) OPTIONAL,
    ...
}

PriceElement ::= SEQUENCE
{
    amount            INTEGER(0..4294967295), -- meter increment
    quantum           INTEGER(0..4294967295), -- each or part thereof
    units             CHOICE
    {
        seconds       NULL,
        packets       NULL,
        bytes         NULL,
        initial       NULL,
        minimum       NULL,
        maximum       NULL,
        ...
    },
    ...
}

Descriptor ::= SEQUENCE
{
    descriptorInfo    DescriptorInfo,
    templates         SEQUENCE OF AddressTemplate,
    gatekeeperID      GatekeeperIdentifier OPTIONAL,
    ...
}

DescriptorInfo ::= SEQUENCE
{
    descriptorID      DescriptorID,
    lastChanged       GlobalTimeStamp,
    ...
}

AlternatePEInfo ::= SEQUENCE
{
    alternatePE       SEQUENCE OF AlternatePE,
    alternateIsPermanent BOOLEAN,
    ...
}

AlternatePE ::= SEQUENCE
{
    contactAddress    AliasAddress,
    priority          INTEGER (1..127),
    elementIdentifier ElementIdentifier OPTIONAL,
    ...
}

```

```

AccessToken ::= CHOICE
{
    token          ClearToken,
    cryptoToken    CryptoH323Token,
    ...,
    genericData    GenericData
}

CallInformation ::= SEQUENCE
{
    callIdentifier    CallIdentifier,
    conferenceID      ConferenceIdentifier,
    ...,
    circuitID        CircuitInfo OPTIONAL
}

UsageCallStatus ::= CHOICE
{
    preConnect        NULL,          -- Call has not started
    callInProgress    NULL,          -- Call is in progress
    callEnded         NULL,          -- Call ended
    ...,
    registrationLost  NULL          -- Uncertain if call ended or not
}

UserInformation ::= SEQUENCE
{
    userIdentifier    AliasAddress,
    userAuthenticator SEQUENCE OF CryptoH323Token OPTIONAL,
    ...
}

UsageSpecification ::= SEQUENCE
{
    sendTo            ElementIdentifier,
    when              SEQUENCE
    {
        never         NULL OPTIONAL,
        start         NULL OPTIONAL,
        end           NULL OPTIONAL,
        period        INTEGER(1..65535) OPTIONAL,    -- in seconds
        failures      NULL OPTIONAL,
        ...
    },
    required          SEQUENCE OF OBJECT IDENTIFIER,
    preferred         SEQUENCE OF OBJECT IDENTIFIER,
    ...,
    sendToPEAddress   AliasAddress OPTIONAL
}

PartyInformation ::= SEQUENCE
{
    logicalAddresses  SEQUENCE OF AliasAddress,
    domainIdentifier  AliasAddress OPTIONAL,
    transportAddress  AliasAddress OPTIONAL,
    endpointType      EndpointType OPTIONAL,
    userInfo          UserInformation OPTIONAL,
    timeZone          TimeZone OPTIONAL,
    ...
}

```

```

Role ::= CHOICE
{
    originator          NULL,
    destination         NULL,
    nonStandardData     NonStandardParameter,
    ...
}

TimeZone ::= INTEGER (-43200..43200)  -- number of seconds relative to UTC
                                         -- including DST if appropriate

TerminationCause ::= SEQUENCE
{
    releaseCompleteReason  ReleaseCompleteReason,
    causeIE                INTEGER (1..65535) OPTIONAL,
    nonStandardData        NonStandardParameter OPTIONAL,
    ...
}

ProtocolVersion ::= OBJECT IDENTIFIER
    -- shall be set to
    -- {itu-t(0) recommendation(0) h(8) h-225-0(2250) annex(1) g(7)
    --   version(0) 2} in field annexGversion;
    -- {itu-t(0) recommendation(0) h(8) 501 version(0) 1}
    -- in field version

DescriptorID      ::= GloballyUniqueID

ElementIdentifier ::= BMPString (SIZE(1..128))

GlobalTimeStamp   ::= IA5String (SIZE(14))
    -- UTC, in the form YYYYMMDDHHmmSS
    -- where YYYY = year, MM = month, DD = day,
    -- HH = hour, mm = minute, SS = second
    -- (for example, 19981219120000 for noon
    -- 19 December 1998)

--
-- REPOSITORY FOR APPLICATION SPECIFIC DATA --
--
-- H.225.0 Annex-G profile data
--
idAnnexGProfiles  INTEGER ::= 0
idAnnexGProfileA  INTEGER ::= 1

annexGProfileA EnumeratedParameter ::=
{
    id      standard:idAnnexGProfileA
}

END  -- of H501-MESSAGES

```


SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série B	Moyens d'expression: définitions, symboles, classification
Série C	Statistiques générales des télécommunications
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	Systèmes audiovisuels et multimédias
Série I	Réseau numérique à intégration de services
Série J	Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	RGT et maintenance des réseaux: systèmes de transmission, circuits téléphoniques, télégraphie, télécopie et circuits loués internationaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	Réseaux de données et communication entre systèmes ouverts
Série Y	Infrastructure mondiale de l'information et protocole Internet
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication