

Union internationale des télécommunications

**UIT-T**

SECTEUR DE LA NORMALISATION  
DES TÉLÉCOMMUNICATIONS  
DE L'UIT

**H.248.1**

(09/2005)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET  
MULTIMÉDIAS

Infrastructure des services audiovisuels – Procédures de  
communication

---

**Protocole de commande de passerelle:  
version 3**

Recommandation UIT-T H.248.1

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
<b>Procédures de communication</b>	<b>H.240–H.259</b>
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.349
Architecture des services d'annuaire pour les services audiovisuels et multimédias	H.350–H.359
Architecture de la qualité de service pour les services audiovisuels et multimédias	H.360–H.369
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
SERVICES À LARGE BANDE ET MULTIMÉDIAS TRI-SERVICES	
Services multimédias à large bande sur VDSL	H.610–H.619

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

## Recommandation UIT-T H.248.1

### Protocole de commande de passerelle: version 3

#### Résumé

Aux fins d'une meilleure évolutivité, la présente Recommandation décompose en sous-éléments fonctionnels la fonction de passerelle H.323 définie dans la Rec. UIT-T H.246. Elle spécifie les protocoles utilisés par ces éléments pour communiquer, ce qui permet aux implémentations de passerelles H.323 d'avoir une évolutivité élevée et qui incite à la multiplication de capacités, telles que les commutateurs SS7, de réseau à commutation de circuits (RCC). Ces sous-éléments permettent également aux passerelles H.323 d'être constituées d'éléments issus de multiples vendeurs répartis sur de multiples plates-formes physiques. L'objet de la présente Recommandation est d'ajouter les capacités actuellement définies pour les systèmes H.323, en vue d'offrir de nouveaux moyens d'effectuer les opérations déjà assurées dans les systèmes H.323.

La présente Recommandation apporte plusieurs améliorations à la version 2 de la Rec. UIT-T H.248.1:

- capacité de définir des propriétés de contexte au moyen de paquetages;
- propriété de contexte de plan IEPS;
- fanion indiquant que la passerelle MG a des terminaisons OutOfService à signaler lors de l'enregistrement;
- nouveau paquetage de segmentation de message et procédures pour transports non segmenteurs;
- affinement des exigences relatives à la définition de paquetages et nouveau gabarit de paquetage;
- affinement des exigences relatives à la définition du profil et nouveau gabarit de profil;
- adjonction de statistiques au niveau du flux;
- adjonction d'un identificateur de demande de signal afin de différencier des signaux similaires à l'intérieur d'une liste de signaux (SignalList);
- adjonction d'un paramètre de signal de base afin d'indiquer le sens de reproduction du signal;
- adjonction de deux nouveaux types topologiques (Topology);
- adjonction d'un temporisateur d'intervalle entre signaux applicables aux signaux contenus dans une liste de signaux (SignalList);
- adjonction d'une nouvelle structure d'identificateur de contexte (ContextID) pour les commandes et les réponses;
- adjonction d'une structure de liste d'identificateurs de terminaison (TerminationIDList) pour les commandes et les réponses;
- affinement des procédures de changement de service (ServiceChange);
- adjonction d'une capacité permettant au contrôleur MGC de régler le débit auquel il reçoit les notifications;
- adjonction de la possibilité d'ajouter des conditions de filtrage afin de vérifier les requêtes.

#### Source

La Recommandation UIT-T H.248.1 a été approuvée le 13 septembre 2005 par la Commission d'études 16 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8.

## AVANT-PROPOS

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

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

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

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

## NOTE

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

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

## DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou 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 avait 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 développeurs de consulter la base de données des brevets du TSB sous <http://www.itu.int/ITU-T/dbase/patent/index.html>.

© UIT 2006

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

## TABLE DES MATIÈRES

		Page
1	Domaine d'application .....	1
2	Références.....	1
	2.1 Références normatives.....	1
	2.2 Références informatives .....	3
3	Définitions .....	4
4	Abréviations.....	5
5	Conventions .....	6
6	Modèle de connexion.....	6
	6.1 Contextes .....	8
	6.2 Terminaisons .....	9
	6.3 Principes de remplacement par des structures génériques .....	15
7	Commandes .....	18
	7.1 Descripteurs.....	19
	7.2 Interface de programmation d'application pour les commandes .....	43
8	Transactions.....	60
	8.1 Paramètres communs .....	61
	8.2 Interface de programmation d'application pour les transactions .....	62
	8.3 Messages.....	64
9	Transport.....	65
	9.1 Ordonnancement des commandes .....	65
	9.2 Protection contre l'avalanche de redémarrage .....	66
	9.3 Protection contre l'avalanche de notifications .....	67
10	Considérations relatives à la sécurité.....	67
	10.1 Protection des connexions de protocole .....	67
	10.2 Système provisoire d'en-tête AH.....	68
	10.3 Protection des connexions de médias .....	69
11	Interface de commande MG-MGC .....	69
	11.1 Multiples passerelles MG virtuelles .....	69
	11.2 Démarrage à froid.....	70
	11.3 Négociation de version de protocole .....	71
	11.4 Défaillance d'une passerelle MG .....	71
	11.5 Défaillance d'un contrôleur MGC .....	72
	11.6 Surveillance de l'association de commande MGC-MG.....	73
12	Définition des paquetages.....	73
	12.1 Directives pour la définition des paquetages.....	73
	12.2 Directives de définition des paramètres relatifs aux événements et signaux.....	77

	<b>Page</b>
12.3	Identificateurs ..... 77
12.4	Enregistrement de paquetage..... 78
13	Définition des profils ..... 78
14	Considérations relatives à l'autorité IANA ..... 78
14.1	Paquetages ..... 78
14.2	Codes d'erreur..... 79
14.3	Raisons de changement de service ..... 79
14.4	Profils ..... 79
Annexe A – Codage binaire du protocole ..... 80	
A.1	Codage des structures génériques..... 80
A.2	Spécification de syntaxe en notation ASN.1 ..... 81
A.3	Noms des scripts de numérotation et des conduits..... 99
Annexe B – Codage alphanumérique du protocole ..... 100	
B.1	Codage des structures génériques..... 100
B.2	Spécification en formalisme ABNF ..... 100
B.3	Codage hexadécimal des octets ..... 115
B.4	Séquence hexadécimale pour les octets..... 116
Annexe C – Etiquettes des propriétés de flux média ..... 116	
C.1	Attributs généraux des médias..... 117
C.2	Propriétés relatives au multiplexage..... 118
C.3	Propriétés générales de support ..... 118
C.4	Propriétés générales du mode ATM ..... 118
C.5	Relais de trames..... 121
C.6	IP..... 121
C.7	ATM AAL 2 ..... 121
C.8	ATM AAL 1 ..... 123
C.9	Capacités du circuit support ..... 124
C.10	Propriétés de couche AAL 5..... 132
C.11	Equivalents SDP ..... 132
C.12	H.245 ..... 133
Annexe D – Transport sur protocole IP ..... 134	
D.1	Transport sur protocoles IP/UDP au moyen du verrouillage de niveau d'application..... 134
D.2	Utilisation du protocole TCP..... 138
Annexe E – Paquetages de base..... 139	
E.1	Paquetage générique ..... 139
E.2	Paquetage racine de base ..... 141
E.3	Paquetage de générateur de tonalités..... 144
E.4	Paquetage de détection de tonalité ..... 146

	<b>Page</b>
E.5	Paquetage de générateur de tonalités DTMF de base..... 149
E.6	Paquetage de détection de tonalités DTMF ..... 150
E.7	Paquetage de générateur de tonalités de progression d'appel..... 153
E.8	Paquetage de détection de tonalités de progression d'appel ..... 154
E.9	Paquetage de supervision de ligne analogique ..... 154
E.10	Paquetage de continuité de base ..... 158
E.11	Paquetage de réseau..... 160
E.12	RTP Package ..... 163
E.12	Paquetage du protocole de transport en temps réel (RTP) ..... 163
E.13	Paquetage de circuit TDM..... 165
E.14	Paquetage de segmentation..... 166
E.15	Comportement de notification ..... 169
Annexe F – Procédures relatives au changement de service ..... 173	
F.1	Introduction ..... 173
F.2	Définition de l'association de commande ..... 175
F.3	Evénements conduisant à des procédures de changement de service ..... 175
F.4	Description de l'élément ServiceChange ..... 180
F.5	Utilisation des paramètres de l'élément ServiceChange ..... 184
F.6	Changement de service par rapport à état de terminaison ..... 186
Appendice I – Exemples de flux de communication ..... 187	
I.1	Communication de passerelle résidentielle à passerelle résidentielle ..... 188
Appendice II – Gabarit de paquetage H.248 ..... 197	
Appendice III – Gabarit de définition de profil H.248 ..... 200	



# Recommandation UIT-T H.248.1

## Protocole de commande de passerelle: version 3

### 1 Domaine d'application

La présente Recommandation définit les protocoles utilisés entre les éléments de décomposition physique d'une passerelle multimédia. Il n'y a pas de différences fonctionnelles, en termes de système, entre une passerelle décomposée (comportant des sous-éléments théoriquement répartis sur plusieurs unités physiques) et une passerelle monolithique comme celle qui est décrite dans la Rec. UIT-T H.246. La présente Recommandation ne définit pas la façon dont fonctionnent les passerelles, les ponts de conférence ou les répondeurs vocaux interactifs (IVR, *interactive voice response*) mais elle crée un cadre général qui convient à ces applications.

Les interfaces avec les réseaux en mode paquet peuvent être de type IP, ATM, etc. Ces interfaces prendront en charge divers systèmes de signalisation de réseau à commutation de circuits (RCC) dont la signalisation par fréquences vocales, le RNIS, le sous-système utilisateur du RNIS, la signalisation à l'interface Q et le système GSM. Les variantes nationales de ces systèmes de signalisation seront éventuellement prises en charge.

Les produits pour lesquels il est indiqué qu'ils sont conformes à la version 1 de la Rec. UIT-T H.248.1 doivent être conformes à toutes les spécifications obligatoires de la Rec. UIT-T H.248-1 approuvée initialement en juin 2000 et rééditée en mars 2002.

Les produits pour lesquels il est indiqué qu'ils sont conformes à la Version 2 de la Rec. UIT-T H.248.1 doivent être conformes à toutes les spécifications obligatoires de la Rec. UIT-T H.248.1 approuvée initialement en mai 2002 et mise à jour en mars 2004.

Les produits pour lesquels il est indiqué qu'ils sont conformes à la présente Recommandation doivent être conformes à toutes les spécifications obligatoires de la Rec. UIT-T H.248.1 approuvée en septembre 2005.

Les produits doivent porter la mention de la version du protocole employé: la valeur du paramètre ServiceChangeVersion sera "1" pour indiquer la Rec. UIT-T H.248.1 (mars 2002), "2" pour indiquer le Corrigendum 1 (mars 2004) de la Rec. UIT-T H.248.1 et "3" pour indiquer la Rec. UIT-T H.248.1 (septembre 2005).

### 2 Références

#### 2.1 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. La référence à un document figurant dans la présente Recommandation ne donne pas à ce document en tant que tel le statut d'une Recommandation.

- Recommandation UIT-T E.106 (2003), *Plan international de priorité en période de crise destiné aux opérations de secours en cas de catastrophe.*
- Recommandation UIT-T H.225.0 (2003), *Protocoles de signalisation d'appel et paquets des flux monomédias dans les systèmes de communication multimédias en mode paquet.*

- Recommandation UIT-T H.235.0 (2005), *Cadre de sécurité H.323: cadre de sécurité pour les systèmes multimédias de la série H (systèmes H.323 et autres systèmes de type H.245)*.
- Recommandation UIT-T H.245 (2005), *Protocole de commande pour communications multimédias*.
- Recommandation UIT-T H.246 (1998), *Interfonctionnement des terminaux multimédias de la série H avec d'autres terminaux multimédias de la série H et des terminaux vocaux ou en bande vocale sur le RTGC et le RNIS*.
- Recommandation UIT-T H.248.4 (2000), *Protocole de commande de passerelle: transport par le protocole de transmission de commande de flux (SCTP)*, plus Corrigendum 1 (2004).
- Recommandation UIT-T H.248.5 (2000), *Protocole de commande de passerelle: transport sur réseau ATM*.
- Recommandation UIT-T H.248.8 (2005), *Protocole de commande de passerelle: description des codes d'erreur et des motifs de changement de service*.
- Recommandation UIT-T H.248.14 (2002), *Protocole de commande de passerelle: paquetage de temporisateur d'inactivité*.
- Recommandation UIT-T H.323 (2003), *Systèmes de communication multimédia en mode paquet*.
- Recommandation UIT-T I.363.1 (1996), *Spécification de la couche d'adaptation ATM du RNIS-LB: AAL de type 1*.
- Recommandation UIT-T I.363.2 (2000), *Spécification de la couche d'adaptation ATM du RNIS-LB: AAL de type 2*.
- Recommandation UIT-T I.363.5 (1996), *Spécification de la couche d'adaptation ATM du RNIS LB: AAL de type 5*.
- Recommandation UIT-T I.366.1 (1998), *Sous-couche de convergence propre au service de segmentation et de réassemblage pour la couche d'adaptation ATM de type 2*.
- Recommandation UIT-T I.366.2 (2000), Corr. 1 (2002), *Sous-couche de convergence propre au service de la couche AAL de type 2 pour les services à bande étroite*.
- Recommandation UIT-T I.371 (2004), *Gestion du trafic et des encombrements dans le RNIS-LB*.
- Recommandation UIT-T Q.763 (1999), *Système de signalisation n° 7 – Formats et codes du sous-système utilisateur du RNIS*, plus Amendement 3 (2004).
- Recommandation UIT-T Q.765.5 (2004), *Système de signalisation n° 7 – Mécanisme de transport d'application: commande d'appel indépendante du support*.
- Recommandation UIT-T Q.931 (1998), *Spécification de la couche 3 de l'interface utilisateur-réseau RNIS pour la commande de l'appel de base*, plus Amendement 1 (2002): *Extensions pour la prise en charge des équipements de multiplexage numérique*.
- Recommandation UIT-T Q.2630.1 (1999), *Protocole de signalisation de couche AAL de type 2 – Ensemble de capacités 1*.
- Recommandation UIT-T Q.2931 (1995), *Système de signalisation d'abonné numérique n° 2 – Spécification de la couche 3 de l'interface utilisateur-réseau pour la commande de connexion/appel de base*, plus Amendement 4 (1999).
- Recommandation UIT-T Q.2941.1 (1997), *Système de signalisation d'abonné numérique n° 2 – Transport des identificateurs génériques*.

- Recommandation UIT-T Q.2961.1 (1995), *Système de signalisation d'abonné numérique n° 2 – Paramètres de trafic supplémentaires: Capacités de signalisation supplémentaires pour la prise en charge des paramètres de trafic relatifs à l'option d'étiquetage et au jeu de paramètres de débit cellulaire permanent acceptable.*
- Recommandation UIT-T Q.2961.2 (1997), *Système de signalisation d'abonné numérique n° 2 – Paramètres de trafic supplémentaires: prise en charge de la capacité de transfert ATM dans l'élément d'information de capacité de support à large bande, plus Corrigendum 1 (1999).*
- Recommandation UIT-T Q.2965.1 (1999), *Système de signalisation d'abonné numérique n° 2 – Prise en charge des classes de qualité de service, plus Amendement 1 (2000).*
- Recommandation UIT-T Q.2965.2 (1999), *Système de signalisation d'abonné numérique n° 2 – Signalisation des paramètres de qualité de service individuelle.*
- Recommandation UIT-T V.76 (1996), *Multiplexeur générique utilisant les procédures basées LAPM de la Recommandation V.42, plus Corrigendum 1 (2005).*
- Recommandation UIT-T X.213 (2001) | ISO/CEI 8348:2002, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service réseau.*
- Recommandation UIT-T X.680 (2002) | ISO/CEI 8824-1:2004, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- Recommandation UIT-T X.690 (2002) | ISO/CEI 8825-1:2002, *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives, plus Amendement 1 (2003): Prise en charge des règles de codage XML étendues (EXTENDED-XER).*
- ISO/CEI 10646 (2003), *Technologies de l'information – Jeu universel de caractères codés sur plusieurs octets (JUC).*
- ATM Forum (1996), *ATM User-Network Interface (UNI) Signalling Specification (Spécification de signalisation à l'interface usager-réseau (UNI) en mode ATM) – Version 4.0.*
- IETF RFC 1006 (1987), *ISO Transport Service on top of the TCP, Version 3 (Service de transport ISO au-dessus du protocole TCP, version 3).*
- IETF RFC 2234 (1997), *Augmented BNF for Syntax Specifications: ABNF (Développement du formalisme BNF pour les spécifications en notation syntaxique: le formalisme ABNF).*
- IETF RFC 2327 (1998), *SDP: Session Description Protocol (Protocole de description de session: SDP).*
- IETF RFC 2402 (1998), *IP Authentication Header (En-tête d'authentification IP).*
- IETF RFC 2406 (1998), *IP Encapsulating Security Payload (ESP) (Charge utile de sécurité par encapsulation IP (ESP)).*

## **2.2 Références informatives**

- Recommandation UIT-T E.180/Q.35 (1998), *Caractéristiques techniques des tonalités du service téléphonique.*
- Recommandation UIT-T G.711 (1988), *Modulation par impulsions et codage (MIC) des fréquences vocales.*

- Recommandation UIT-T H.221 (2004), *Structure de trame pour un canal d'un débit de 64 à 1920 kbit/s pour les téléservices audiovisuels*.
- Recommandation UIT-T H.223 (2001), *Protocole de multiplexage pour communications multimédias à faible débit*.
- Recommandation UIT-T H.226 (1998), *Protocole d'agrégation de canaux pour l'exploitation en mode multilaision sur les réseaux à commutation de circuits*.
- Recommandation UIT-T Q.724 (1998), *Procédures de signalisation du sous-système utilisateur de téléphonie*, plus Amendement 1 (1993).
- Recommandation UIT-T Q.764 (1999), *Système de signalisation n° 7 – Procédures de signalisation du sous-système utilisateur du RNIS*, plus Amendement 3 (2004).
- Recommandation UIT-T Q.1904.2 (2001), *Protocole de commande d'appel indépendante du support (ensemble de capacités 2): procédures d'appel de base*, plus Amendement 2 (2004).
- IETF RFC 768 (1980), *User Datagram Protocol*.
- IETF RFC 791 (1981), *Internet protocol*.
- IETF RFC 793 (1981), *Transmission control protocol*.
- IETF RFC 1661 (1994), *The Point-to-Point Protocol (PPP)* (le protocole point à point).
- IETF RFC 2401 (1998), *Security Architecture for the Internet Protocol* (Architecture de sécurité pour le protocole Internet).
- IETF RFC 2460 (1998), *Internet Protocol, Version 6 (IPv6) Specification*.
- IETF RFC 2805 (2000), *Media Gateway Control Protocol Architecture and Requirements* (Architectures et prescriptions pour le protocole de commande de passerelle de média).
- IETF RFC 3261 (2002), *SIP: Session Initiation Protocol* (protocole d'ouverture de session).
- IETF RFC 3550 (2003), *RTP: A Transport Protocol for Real-Time Applications* (un protocole de transport pour applications en temps réel).
- IETF RFC 3551 (2003), *RTP Profile for Audio and Video Conferences with Minimal Control* (Profil du protocole RTP pour conférences audiovisuelles avec commande minimale).

### 3 Définitions

La présente Recommandation définit les termes suivants:

**3.1 passerelle d'accès:** type de passerelle offrant une interface de type utilisateur-réseau (UNI), comme pour le RNIS.

**3.2 descripteur:** élément syntaxique du protocole qui regroupe des propriétés associées. Par exemple, le contrôleur de passerelle média (MGC) peut régler les propriétés d'un flux média en insérant le descripteur approprié dans une commande.

**3.3 passerelle média (MG, media gateway):** passerelle qui convertit le format du média fourni par un type de réseau de façon qu'il soit accepté par un autre type de réseau. Par exemple, une passerelle média peut fermer des voies supports issues d'un réseau à commutation de circuits (des signaux DS0 par exemple) ainsi que des flux média issus d'un réseau en mode paquet (par exemple des flux RTP dans un réseau IP). Cette passerelle peut traiter des signaux audio, vidéo et T.120 isolés ou combinés d'une manière quelconque. Elle pourra également assurer des conversions de

média en duplex, restituer des messages audio/vidéo, remplir d'autres fonctions de répondeur IVR ou assurer des conférences multimédias.

**3.4 contrôleur de passerelle média (MGC, *media gateway controller*):** entité qui commande les parties de l'état d'appel qui correspondent à la commande de connexion pour les voies média d'une passerelle MG.

**3.5 pont de conférence; unité de commande multipoint (MCU, *multipoint control unit*):** entité qui commande l'établissement et la coordination d'une conférence entre plusieurs utilisateurs comportant normalement le traitement audio, vidéo et données.

**3.6 passerelle résidentielle:** passerelle qui assure l'interfonctionnement d'une ligne analogique avec un réseau en mode paquet. Normalement, une passerelle résidentielle contient une ou deux lignes analogiques et est située dans les locaux d'abonné.

**3.7 passerelle de signalisation de réseau RCC en mode service par service:** fonction contenant l'interface de signalisation de réseau RCC à laquelle aboutissent des canaux sémaphores SS7, RNIS ou d'autres canaux dans lesquels la voie de commande d'appel et les voies supports sont compositionnées dans le même arc physique.

**3.8 passerelle de signalisation de réseau RCC en mode autre que service par service:** fonction contenant l'interface de signalisation de réseau RCC à laquelle aboutissent des canaux sémaphores SS7 ou d'autres canaux dans lesquels les voies de commande d'appel sont séparées des voies supports.

**3.9 flux:** média bilatéral ou débit de commande reçu/envoyé par une passerelle média dans le cadre d'une communication ou d'une conférence.

**3.10 jonction:** canal de communication entre deux systèmes de commutation comme un signal DS0 sur une ligne à débit T1 ou E1.

**3.11 passerelle de jonction:** passerelle entre un réseau RCC et un réseau en mode paquet à laquelle aboutissent normalement un grand nombre de circuits numériques.

## 4 Abréviations

La présente Recommandation utilise les abréviations suivantes:

AAD	temps d'acquittement moyen ( <i>average acknowledgement delay</i> )
AAL	couche d'adaptation au mode ATM ( <i>ATM adaptation layer</i> )
ADEV	écart type d'Allan moyen ( <i>average deviation</i> )
ALF	verrouillage au niveau application ( <i>application level framing</i> )
ATM	mode de transfert asynchrone ( <i>asynchronous transfer mode</i> )
C	contexte
CAS	signalisation en mode voie par voie ( <i>channel associated signalling</i> )
DNS	système de dénomination de domaine ( <i>domain name system</i> )
DTMF	multifréquence bitonalité ( <i>dual tone multi-frequency</i> )
FAS	signalisation en mode service par service ( <i>facility associated signalling</i> )
GSM	système mondial de communications mobiles ( <i>global system for mobile communications</i> )
GW	passerelle ( <i>gateway</i> )

IANA	autorité chargée de l'assignation des numéros Internet ( <i>Internet assigned numbers authority</i> ) remplacée par l'ICANN
ICANN	corporation Internet pour les noms et numéros assignés ( <i>Internet corporation for assigned names and numbers</i> )
IEPS	plan international de priorité en période de crise ( <i>International emergency preference scheme</i> )
IP	protocole Internet ( <i>Internet protocol</i> )
IS	en service ( <i>in-service</i> )
ISUP	sous-système utilisateur du RNIS ( <i>ISDN user part</i> )
IVR	réponse vocale interactive ( <i>interactive voice response</i> )
MG	passerelle média ( <i>media gateway</i> )
MGC	contrôleur de passerelle média ( <i>media gateway controller</i> )
MWD	temps d'attente maximal ( <i>maximum waiting delay</i> )
NFAS	signalisation autre qu'en mode service par service ( <i>non-facility associated signalling</i> )
OoS	hors service ( <i>out-of-service</i> )
PRI	interface à débit primaire ( <i>primary rate interface</i> )
QS	qualité de service
RCC	réseau à commutation de circuits
RTP	protocole de transport en temps réel ( <i>real-time transport protocol</i> )
RTPC	réseau téléphonique public commuté
SC	changement de service ( <i>ServiceChange</i> )
SG	passerelle de signalisation ( <i>signalling gateway</i> )
SS7	système de signalisation n° 7
T, Term	terminaison

## 5 Conventions

Dans la présente Recommandation, la forme verbale "doit" correspond à une exigence obligatoire, tandis que la forme "devrait" ou "il convient" correspond à la suggestion d'un élément de service ou d'une procédure, mais à titre facultatif. La forme "peut" correspond à une action possible sans qu'une préférence soit exprimée.

## 6 Modèle de connexion

Le modèle de connexion pour le protocole décrit les entités logiques – ou objets contenus dans la passerelle média, que le contrôleur de passerelle média peut commander. Les principales abstractions utilisées dans ce modèle de connexion sont les terminaisons et les contextes.

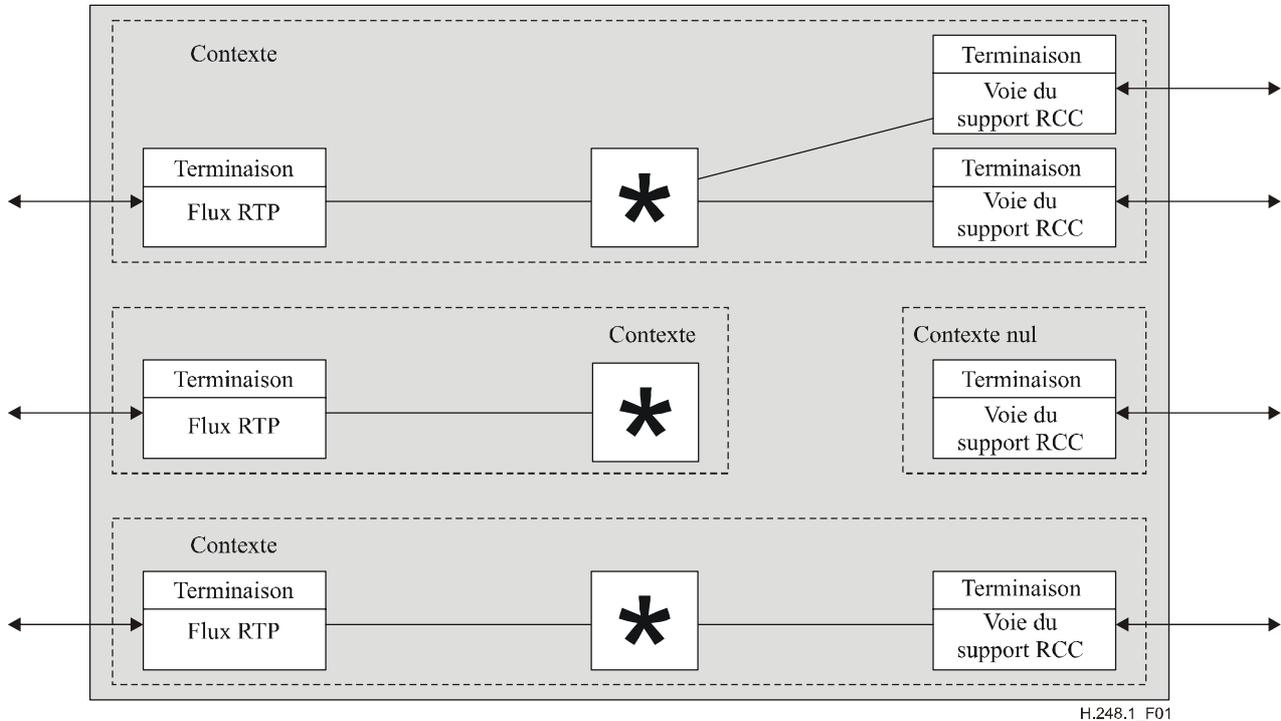
Une *terminaison* envoie et/ou collecte un ou plusieurs flux. Dans une conférence multimédia, une terminaison peut être de type multimédia et peut envoyer ou collecter de multiples flux médias. Les paramètres des flux médias, ainsi que les paramètres de support sont encapsulés dans la terminaison.

Un *contexte* est un paquetage de terminaisons associées. Il existe un type spécial de contexte, le contexte *NULL* (*néant*), qui contient toutes les terminaisons non présentes dans un autre contexte et

donc non associées à une autre terminaison. Par exemple, dans une passerelle d'accès décomposée, toutes les lignes au repos sont représentées par des terminaisons du contexte NULL.

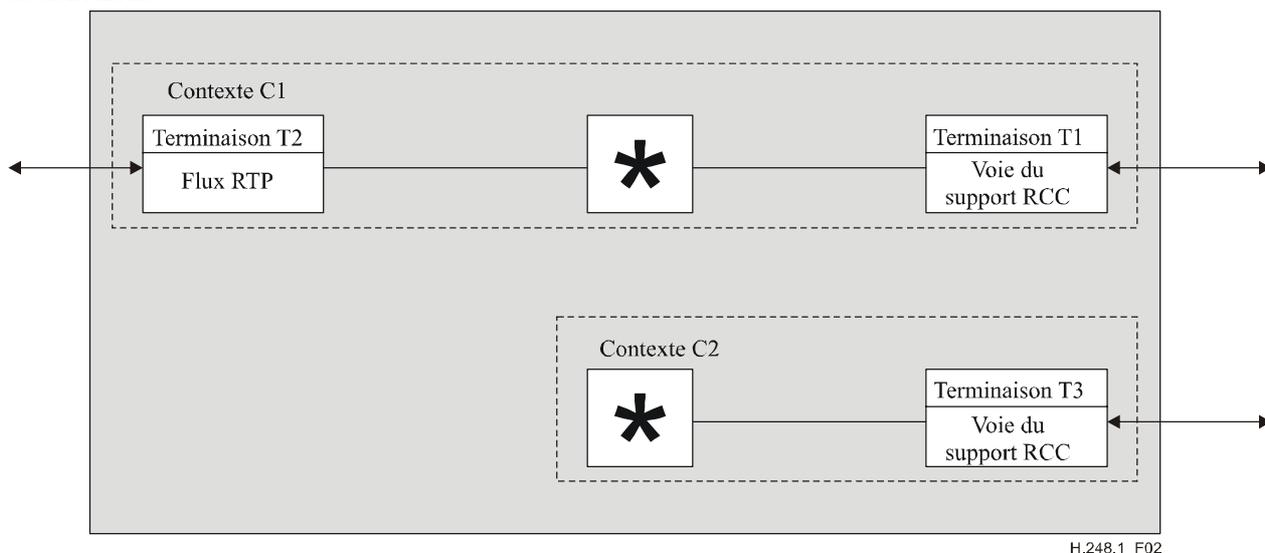
Une description graphique de ces concepts est donnée dans le schéma de la Figure 1, qui donne plusieurs exemples et qui ne vise pas à illustrer tous les cas. L'astérisque encadré de chaque contexte représente l'association logique des terminaisons appartenant au contexte.

Passerelle média

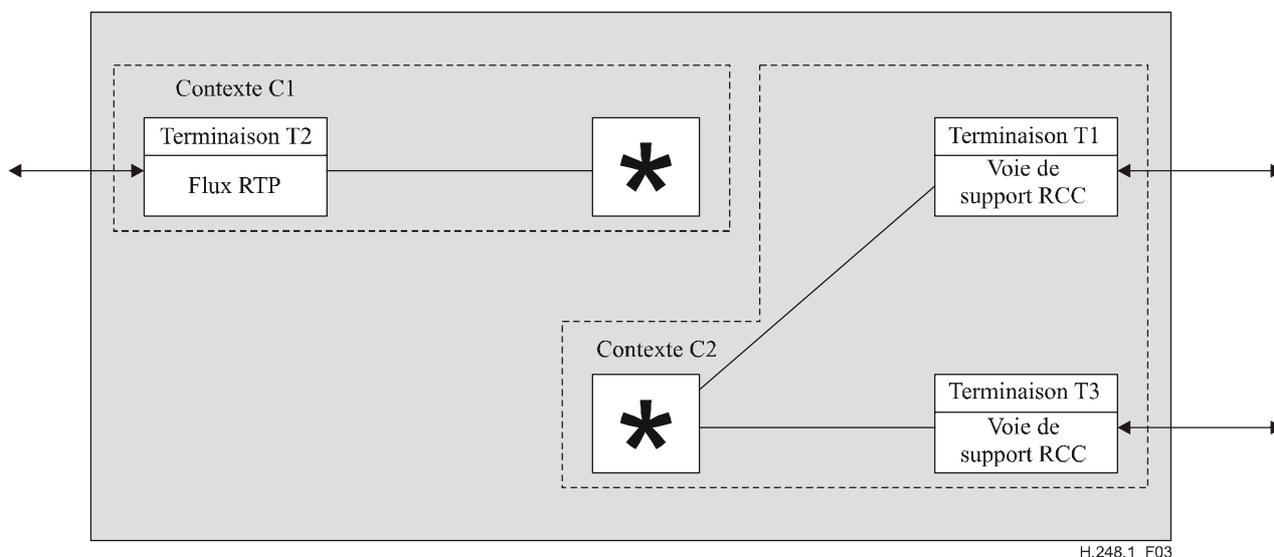


**Figure 1/H.248.1 – Exemple de modèle de connexion H.248.1**

L'exemple de la Figure 2 montre un mode de réalisation d'un scénario de signal d'appel dans une passerelle d'accès décomposée, illustrant la relocalisation d'une terminaison entre plusieurs contextes. Les terminaisons T1 et T2 appartiennent au contexte C1 dans une communication audio bilatérale. Une deuxième communication audio est en instance pour la terminaison T1 en provenance de la terminaison T3, qui est seule dans le contexte C2. T1 accepte l'appel de T3 et met T2 en instance. Cette action se traduit par le passage de T1 dans le contexte C2, comme représenté dans la Figure 3.



**Figure 2/H.248.1 – Exemple de scénario de signal d'appel/sonnerie appliqué à la terminaison T1**



**Figure 3/H.248.1 – Exemple de scénario de signal d'appel/réponse appliqué à la terminaison T1**

### 6.1 Contextes

Un contexte est une association entre un certain nombre de terminaisons. Le contexte décrit la topologie (qui entend/voit qui) et les paramètres de mélange et/ou de commutation de média si plusieurs terminaisons font partie de l'association.

Il existe un contexte spécial appelé contexte *NULL* qui contient les terminaisons qui ne sont associées à aucune autre terminaison. Dans le contexte *NULL*, les terminaisons peuvent avoir leurs paramètres examinés ou modifiés et des événements peuvent être détectés à leur sujet.

En général, une commande d'addition (Add) permet d'ajouter des terminaisons à un contexte. Si le contrôleur MGC ne spécifie pas un contexte existant auquel la terminaison doit être ajoutée, la passerelle MG crée un nouveau contexte. Une terminaison peut être retirée d'un contexte par une commande de soustraction (Subtract) et une terminaison peut être déplacée d'un contexte à un autre par une commande de déplacement (Move). Une terminaison ne doit exister que dans un seul contexte à la fois.

Le nombre maximal de terminaisons contenues dans un contexte est une propriété de passerelle MG. Les passerelles MG qui n'offrent qu'une connexité point à point peuvent admettre au plus deux terminaisons par contexte. Les passerelles MG qui prennent en charge les conférences multipoint peuvent admettre au moins trois terminaisons par contexte.

### 6.1.1 Attributs et descripteurs de contexte

Les attributs des contextes sont les suivants:

- l'identificateur ContextID;
- le descripteur "Topology" (qui entend/voit qui);  
la topologie d'un contexte décrit le flux média entre les terminaisons contenues dans un contexte. En revanche, la propriété de mode d'une terminaison ("SendOnly"/"RecvOnly"/...) décrit le flux média à la sortie/l'entrée de la passerelle média;
- la priorité accordée à un contexte afin de fournir à la passerelle MG des informations sur un certain traitement de préséance pour un contexte. Le contrôleur MGC peut également utiliser la priorité pour commander de manière autonome et souple la préséance de trafic dans la passerelle MG lors de certaines situations (comme un redémarrage) où un grand nombre de contextes doivent être traités simultanément. La priorité 0 est la plus basse tandis que la priorité 15 est la plus haute;
- un indicateur d'appel d'urgence fourni également pour permettre un traitement préférentiel dans la passerelle MG;
- un indicateur de communication en plan IEPS est fourni afin de permettre de réaliser les éléments de service et les techniques de la Rec. UIT-T E.106;
- un descripteur "ContextAttribute" qui permet de définir des attributs de contexte au moyen du mécanisme d'extension de paquetage (voir le § 7.1.19).

### 6.1.2 Création, suppression et modification de contextes

Le protocole peut être utilisé pour (implicitement) créer des contextes et modifier les valeurs paramétriques des contextes existants. Ce protocole contient des commandes permettant d'ajouter des terminaisons à des contextes, de retirer des terminaisons de contextes et de déplacer des terminaisons entre contextes. Lorsque la dernière terminaison restante est soustraite ou déplacée, son contexte est implicitement supprimé.

## 6.2 Terminaisons

Une terminaison est une entité logique d'une passerelle MG qui envoie et collecte des flux de média et de commande. Une terminaison est décrite par un certain nombre de propriétés de caractérisation qui sont groupées dans un paquetage de descripteurs utilisés dans les commandes. Les terminaisons ont des identificateurs uniques (TerminationID) qui sont attribués par la passerelle MG au moment de la création de ces terminaisons.

Les terminaisons qui représentent des entités physiques ont une existence semi-permanente. Par exemple, une terminaison représentant une voie MRT peut exister aussi longtemps qu'elle est fournie dans la passerelle. Les terminaisons représentant des flux informationnels éphémères, comme les flux de protocole RTP, n'existeront généralement que pendant leur durée d'utilisation.

Les terminaisons éphémères sont créées au moyen d'une commande Add. Elles sont détruites au moyen d'une commande Subtract. En revanche, lorsqu'une terminaison physique est ajoutée à un contexte ou en est soustraite, elle est respectivement extraite du contexte NULL ou placée dans celui-ci.

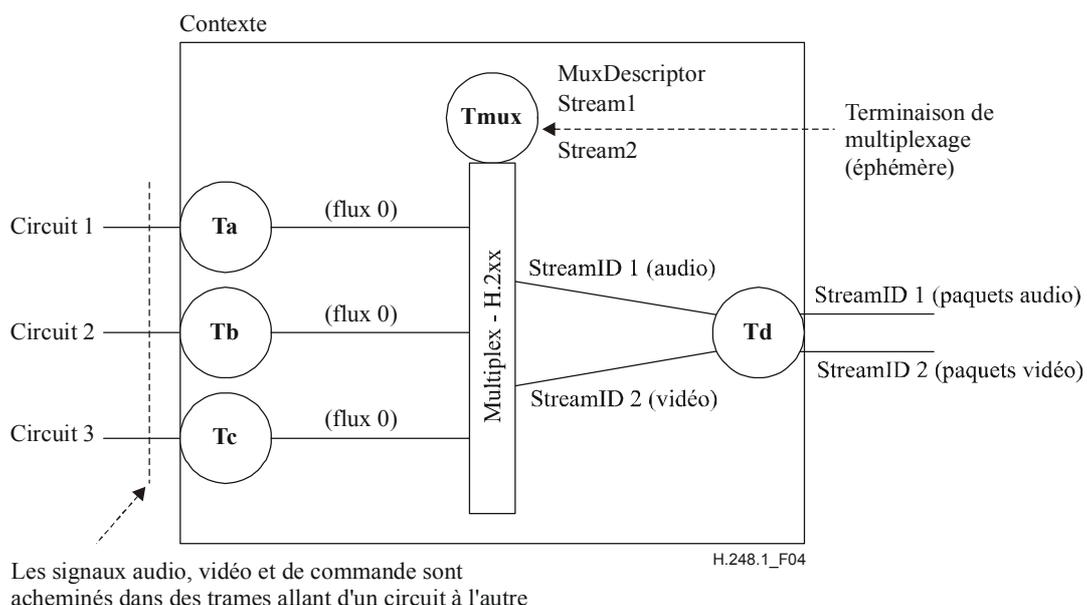
Les terminaisons peuvent subir l'application de signaux (voir le § 7.1.11). Les terminaisons peuvent être programmées de façon à détecter des événements dont l'apparition peut déclencher l'envoi de messages de notification vers le contrôleur MGC ou déclencher une action de la passerelle MG. Des statistiques peuvent être cumulées au sujet d'une terminaison. Ces statistiques sont signalées au contrôleur MGC sur demande (au moyen d'une commande AuditValue, voir le § 7.2.5) et lorsque la terminaison cesse d'exister ou est renvoyée au contexte NULL en raison de l'application d'une commande Subtract.

Les passerelles MG peuvent traiter des flux médias multiplexés. Par exemple, la Rec. UIT-T H.221 décrit une structure de trame pour le multiplexage de plusieurs flux médias dans un certain nombre de voies numériques à 64 kbit/s. Un tel cas est pris en charge comme suit dans le modèle de connexion. Il existe une "terminaison support" physique ou éphémère pour chaque canal support qui achemine une partie des flux multiplexés. Les terminaisons supports qui envoient/collectent ces voies numériques sont connectées à une terminaison distincte appelée "terminaison de multiplexage". La terminaison de multiplexage est une terminaison éphémère représentant une session orientée trame. Le descripteur "Multiplex" MultiplexDescriptor de cette terminaison décrit le multiplex utilisé (par exemple H.221 pour une session H.320) et indique l'ordre dans lequel les voies numériques contenues sont assemblées en une trame.

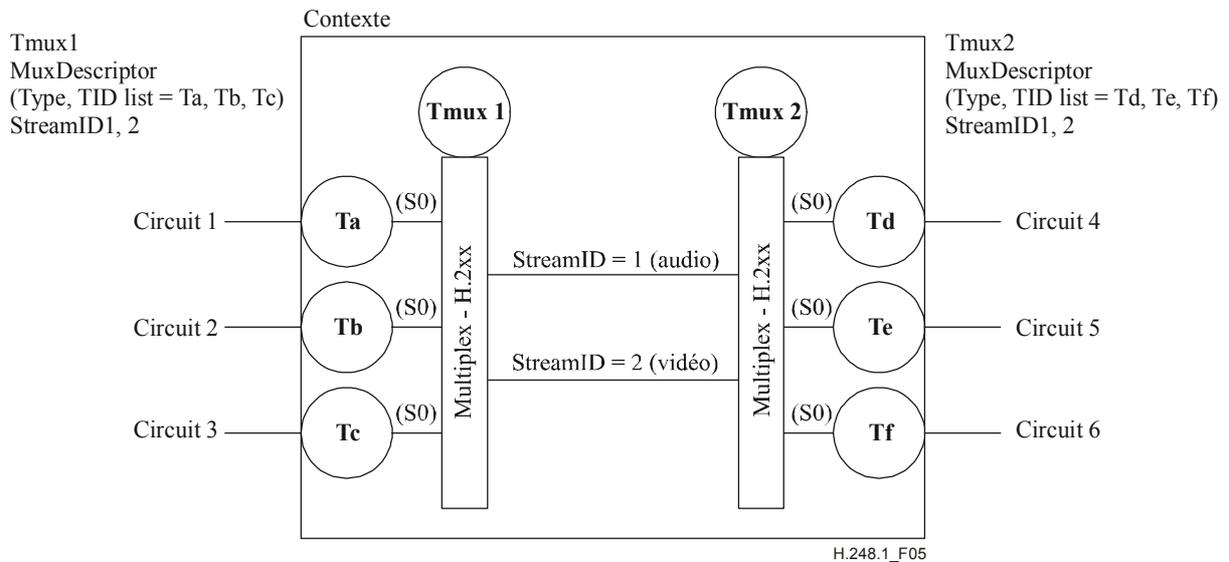
Les terminaisons de multiplexage peuvent être placées en cascade (par exemple un multiplex H.226 de voies numériques alimentant un multiplex H.223 qui prend en charge une session H.324).

Les flux médias individuels acheminés au cours de la session sont décrits par les descripteurs de flux StreamDescriptors au niveau de la terminaison de multiplexage. Ces flux médias peuvent être associés à des flux envoyés/collectés par des terminaisons du contexte autres que les terminaisons support prenant en charge la terminaison de multiplexage. Toutes les terminaisons supports prennent en charge un seul flux de données. Ces flux de données n'apparaissent pas explicitement comme flux au niveau de la terminaison de multiplexage et ils sont cachés du reste du contexte.

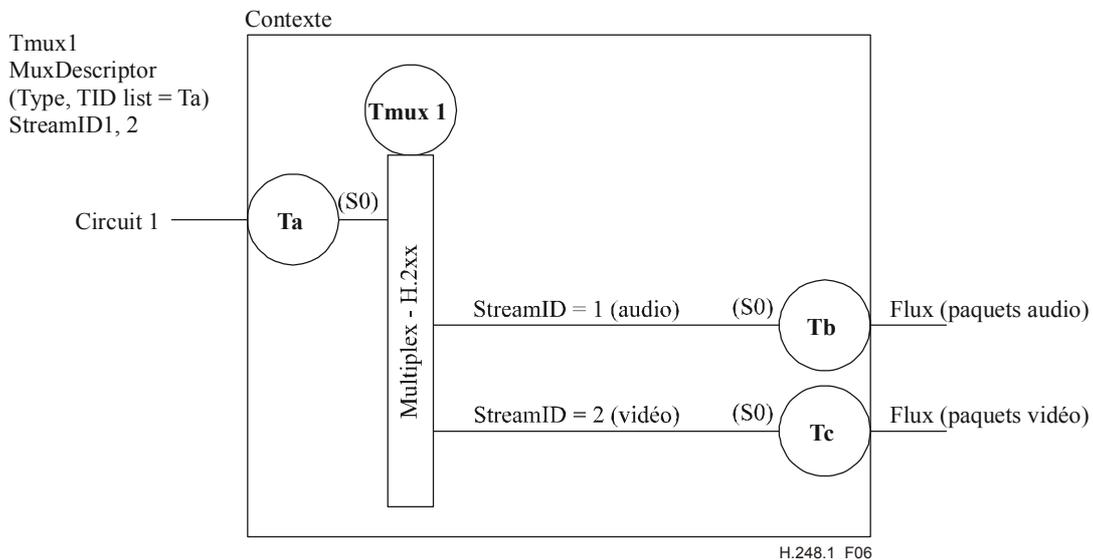
Les Figures 4, 5 et 6 représentent des applications typiques de la terminaison de multiplexage et du descripteur "Multiplex".



**Figure 4/H.248.1 – Scénario de terminaison multiplexée – De circuits à paquets**



**Figure 5/H.248.1 – Scénario de terminaison multiplexée – De circuit à circuit**



**Figure 6/H.248.1 – Scénario de terminaison multiplexée – De terminaison simple à terminaisons multiples**

Contrairement aux terminaisons de multiplexage décrites dans l'alinéa précédent, les terminaisons supports multiplexées correspondant aux supports multiplexés tels que les supports en mode ATM AAL de type 2, n'acheminent pas de flux média. Elles ne servent qu'à modéliser la création et la destruction du support proprement dit. Lorsqu'un circuit multiplexé doit être créé, une terminaison éphémère est créée dans un contexte établi à cet effet. Lorsque la terminaison est soustraite, le circuit support multiplexé est détruit.

### 6.2.1 Dynamique de terminaison

Le protocole peut être utilisé pour créer de nouvelles terminaisons et pour modifier les valeurs des propriétés de terminaisons existantes. Ces modifications comprennent la possibilité d'ajouter ou de supprimer des événements et/ou des signaux. Les propriétés des terminaisons ainsi que les événements et les signaux sont décrits dans les paragraphes qui suivent. Un contrôleur MGC ne

peut libérer/modifier que les terminaisons (et les ressources qu'elles représentent) qui sont dans le contexte NULL ou qui ont été préalablement saisies au moyen, par exemple, de la commande Add.

### **6.2.2 TerminationID (identificateurs de terminaison)**

Les terminaisons sont désignées par un identificateur de terminaison qui est une séquence arbitraire, choisie par la passerelle MG.

Les identificateurs de terminaisons physiques sont fournis dans la passerelle MG. Ils peuvent être choisis de façon à posséder une structure. Par exemple, un identificateur de terminaison peut se composer d'un faisceau de circuits et d'une jonction locale dans ce faisceau.

Un mécanisme de remplacement par des structures génériques, utilisant deux types de structure générique, peut être utilisé avec les identificateurs de terminaison. Ces deux structures sont ALL et CHOOSE. La première sert à désigner simultanément plusieurs terminaisons tandis que la seconde sert à indiquer à une passerelle MG qu'elle doit sélectionner une terminaison correspondant à l'identificateur de terminaison partiellement spécifié. Cela permet à un contrôleur MGC de demander, par exemple, à une passerelle MG de choisir un circuit dans un faisceau de circuits.

Les identificateurs de terminaison peuvent également être spécifiés dans une liste dont l'usage est recommandé lorsqu'une structure hiérarchique d'identificateurs de terminaison n'est pas possible et lorsque l'on ne souhaite pas envoyer de commandes individuelles pour chaque identificateur de terminaison.

### **6.2.3 Paquetages**

Différents types de passerelle peuvent implémenter des terminaisons possédant des caractéristiques très différentes. Le protocole tient compte des variantes de terminaison en permettant que leurs implémentations par des passerelles MG aient des propriétés, des événements, des signaux et des statistiques de type facultatif.

Afin de réaliser l'interopérabilité MG/MGC, de telles options sont groupées en paquetages. Une terminaison réalise en règle générale un surensemble de tels paquetages. Pour plus d'informations sur la définition des paquetages, voir le § 12. Un contrôleur MGC peut examiner une terminaison afin de déterminer les paquetages qu'elle réalise.

Les propriétés, événements, signaux et statistiques définis dans les paquetages, ainsi que des paramètres les concernant, sont référencés par des identificateurs (ID). Les identificateurs sont détectés dans un domaine de visibilité. Pour chaque paquetage, les identificateurs PropertyID, EventID, SignalID, StatisticsID et ParameterID ont des espaces nominatifs uniques et le même identificateur peut être utilisé dans chacun d'eux. Deux identificateurs PropertyID présents dans des paquetages différents peuvent également avoir le même identificateur, etc.

Afin de prendre en charge un paquetage particulier, la passerelle MG doit reconnaître l'ensemble des propriétés, signaux, événements et statistiques définis dans un paquetage. Elle doit aussi prendre en charge tous les paramètres de signal et d'événement. Si la fonctionnalité offerte par ces propriétés, signaux, événements et statistiques n'est pas implémentée, la passerelle MG ne doit renvoyer ni une erreur de syntaxe ni une erreur d'identificateur inconnu pour l'un quelconque de ces éléments mais devrait renvoyer l'erreur 501: "Fonctionnalité non implémentée".

La passerelle MG peut prendre en charge un sous-ensemble des valeurs énumérées dans un paquetage pour une propriété particulière ou pour un paramètre donné. Si une valeur non prise en charge est spécifiée par le contrôleur MGC, la passerelle MG doit renvoyer l'erreur 501: "Fonctionnalité non implémentée".

Lorsque les paquetages sont étendus, il est possible de faire référence aux propriétés, événements, signaux et statistiques définis dans le paquetage de base au moyen soit du nom de paquetage étendu, soit du nom de paquetage de base. Par exemple, si le paquetage A définit un événement e1 et que le paquetage B étend le paquetage A, alors B/e1 est un événement pour une terminaison implémentant

le paquetage B. Par définition, la passerelle MG doit aussi implémenter le paquetage de base, mais elle peut choisir de publier ou non le paquetage de base en tant qu'interface admise. Si elle publie le paquetage A, ce paquetage tout comme le paquetage B sera indiqué dans le descripteur de paquetage dans la valeur AuditValue, et l'événement A/e1 serait disponible sur une terminaison. Si la passerelle MG ne publie pas le paquetage A, seul l'événement B/e1 serait disponible. Si la publication se fait au moyen de la valeur AuditValue, les événements A/e1 et B/e1 sont le même événement.

Pour améliorer l'interfonctionnement et la compatibilité vers l'arrière, une passerelle MG peut publier tous les paquetages pris en charge par ses terminaisons, y compris les paquetages de base à partir desquels les paquetages étendus sont définis. Ceci peut se faire sauf dans les cas où les paquetages de base sont expressément définis comme étant "Conçus uniquement pour être étendus".

#### 6.2.4 Propriétés et descripteurs de terminaison

Les terminaisons possèdent des propriétés qui ont des identificateurs PropertyID uniques. La plupart des propriétés ont des valeurs par défaut, qui sont définies de manière explicite dans la présente spécification de protocole ou dans un paquetage (voir le § 12) ou définies par la fourniture. Si elles ne sont pas fournies autrement, les propriétés dans tous les descripteurs, à l'exception de TerminationState et LocalControl, possèdent par défaut la valeur vide/"pas de valeur" lorsqu'une terminaison est créée pour la première fois ou est renvoyée au contexte NULL. Lorsqu'une terminaison est, pour la première fois, créée ou renvoyée au contexte NULL, cet état représente une ligne, jonction ou autre entité "au repos". Les contenus par défaut de ces deux exceptions sont décrits aux § 7.1.5 et 7.1.7.

La fourniture d'une valeur de propriété dans la passerelle MG va permettre à toute valeur par défaut d'être supplantée, que celle-ci soit fournie dans la spécification de protocole ou dans un paquetage. Donc, s'il est essentiel pour le contrôleur MGC de contrôler entièrement les valeurs des propriétés d'une terminaison, il devrait fournir des valeurs explicites lorsqu'il ajoute la terminaison à un contexte. En variante, pour une terminaison physique, il peut déterminer les valeurs des propriétés fournies en examinant la terminaison tandis que celle-ci est dans le contexte NULL.

Il existe un certain nombre de propriétés communes aux terminaisons et des propriétés spécifiques aux flux médias. Ces propriétés communes sont également appelées *propriétés d'état de terminaison*. Pour chaque flux média, il existe des propriétés locales et des propriétés propres aux flux reçus et émis.

Les propriétés non incluses dans le protocole de base sont définies dans des paquetages. Elles sont désignées par un nom formé du nom de paquetage et de l'identificateur de propriété (PropertyID). La plupart des propriétés ont des valeurs par défaut, qui figurent dans la description de paquetage. Les propriétés peuvent être de type à lecture seulement ou à lecture-écriture. Les valeurs possibles d'une propriété peuvent être examinées, ainsi que leurs valeurs actuelles. Dans le cas des propriétés qui sont accessibles en lecture/écriture, le contrôleur MGC peut en régler la valeur. On peut déclarer "globale" une propriété qui a une seule valeur partagée par toutes les terminaisons réalisant le paquetage. Par commodité, les propriétés associées sont groupées en descripteurs.

Lorsqu'une terminaison est ajoutée à un contexte, la valeur de ses propriétés en lecture/écriture peut être réglée par inclusion dans la commande Add des descripteurs appropriés sous forme de paramètres. De même, une propriété de terminaison contenue dans un contexte peut subir une modification de valeur par la commande Modify. Les valeurs des propriétés peuvent être également modifiées lorsqu'une terminaison est déplacée d'un contexte à un autre à la suite d'une commande Move. Dans certains cas, des descripteurs sont fournis en tant que données de sortie d'une commande.

Le fait de régler dans le même contexte les propriétés de différentes terminaisons donne à la passerelle MG l'instruction implicite d'exécuter certaines fonctions. Par exemple, si un codec G.711 est activé à la terminaison A et qu'un codec G.729 soit activé à la terminaison B, alors la passerelle

MG activera une fonction de transcodage dès qu'un flux de média passera entre les deux terminaisons (c'est-à-dire en réglant la propriété de mode à un état autre que "Inactif" à chaque terminaison).

NOTE – Afin d'éviter une activation inutile des ressources de la passerelle MG, il convient que le contrôleur MGC règle la propriété de mode à la valeur "Inactif" pour une terminaison donnée et un flux donné, jusqu'à ce que les propriétés à utiliser pour ce flux aient été déterminées.

En général, lorsqu'un descripteur ne figure pas du tout dans l'une des commandes susmentionnées, les propriétés dans ce descripteur conservent leurs valeurs antérieures pour la ou les terminaisons sur lesquelles agit la commande. D'autre part, si certaines propriétés de lecture/écriture sont omises d'un descripteur dans une commande (c'est-à-dire que le descripteur n'est que partiellement spécifié), ces propriétés reprendront leurs valeurs par défaut pour la ou les terminaisons sur lesquelles agit la commande, à moins que le paquetage n'indique un autre comportement. Pour plus de détails, voir le § 7.1 qui traite des différents descripteurs.

Le comportement ci-dessus s'applique également aux signaux et aux événements ainsi qu'à leurs paramètres respectifs. La fourniture de descripteurs "Events" comportera l'identificateur RequestID à utiliser ainsi que tous les éventuels paramètres des événements visés. Il convient que toute fourniture relative à des descripteurs "Events" dans la passerelle MG soit dupliquée dans le contrôleur MGC afin d'éviter des réponses d'erreur à des commandes de notification (Notify) issues de la passerelle MG.

Le tableau ci-dessous énumère tous les descripteurs possibles ainsi que leur usage. Tous les descripteurs ne sont pas autorisés en tant que paramètres d'entrée ou de sortie d'une commande donnée.

<b>Nom du descripteur</b>	<b>Description</b>
Modem	Désigne, le cas échéant, le type et les propriétés d'un modem (Note).
Mux	Décrit le type de multiplex pour des terminaisons multimédias (par exemple de type H.221, H.223, H.225.0) et pour des terminaisons formant le multiplex d'entrée.
Media	Liste de spécifications de flux média (voir le § 7.1.4).
TerminationState	Propriétés d'une terminaison qui peuvent être définies dans des paquetages et qui ne sont pas spécifiques d'un flux.
Stream	Liste de descripteurs de commande distante ou locale (de type Remote/Local//LocalControl) pour un flux isolé.
Local	Contient des propriétés qui spécifient les flux médias que la passerelle MG reçoit de l'entité distante.
Remote	Contient des propriétés qui spécifient les flux média que la passerelle MG envoie à l'entité distante.
LocalControl	Contient des propriétés qui peuvent être définies dans des paquetages et qui sont utiles entre la passerelle MG et le contrôleur MGC.
Events	Décrit des événements qui doivent être détectés par la passerelle MG et la suite qui doit être donnée lors d'une telle détection.
EventBuffer	Décrit des événements qui doivent être détectés par la passerelle MG lorsqu'une mise en mémoire tampon d'événement est active.
Signals	Décrit des signaux (voir le § 7.1.11) appliqués aux terminaisons.
Audit	Désigne, dans des commandes de type Audit, les informations à rechercher.
Packages	Renvoie, dans une commande AuditValue, une liste de paquetages réalisés par une terminaison.

<b>Nom du descripteur</b>	<b>Description</b>
DigitMap	Définit des structures avec lesquelles des séquences d'un ensemble spécifié d'événements doivent être mises en correspondance de telle sorte qu'elles puissent être rapportées en tant que groupe et non individuellement.
ServiceChange	Indique, dans la commande ServiceChange, la nature, la raison, etc., d'un éventuel changement de service.
ObservedEvents	Signale les événements observés dans une commande de AuditValue ou Notify.
Statistics	Signale, dans une commande Subtract ou Audit, des statistiques conservées au sujet d'une terminaison.
Topology	Spécifie les sens des flux entre les terminaisons dans un contexte.
ContextAttribute	Contient des propriétés (qui peuvent être définies dans des paquetages) affectant le contexte dans son ensemble.
Error	Contient un code d'erreur et éventuellement un texte d'erreur; il peut figurer dans les réponses aux commandes et dans les demandes de notification.
NOTE – Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).	

### 6.2.5 Terminaison racine

Occasionnellement, une commande doit faire référence à la passerelle en tant qu'entité en soi plutôt qu'à une des terminaisons qu'elle contient. Un identificateur de terminaison spécial, "Root", est réservé à cette fin. Des paquetages peuvent être définis au sujet d'une terminaison racine, qui peut donc avoir des propriétés, des événements, des signaux et des statistiques. En conséquence, l'identificateur de terminaison racine peut apparaître dans les messages suivants:

- une commande Modify – afin de modifier une propriété, d'envoyer un signal ou de fixer un événement;
- une commande Notify – afin de signaler un événement;
- une commande AuditValue – afin d'examiner les valeurs des propriétés et des statistiques implémentées dans la terminaison racine;
- une commande AuditCapabilities – afin de déterminer les propriétés implémentées dans la terminaison racine;
- une commande ServiceChange – afin de déclarer l'ensemble de la passerelle en service ou hors service.

Tout autre usage de l'identificateur de terminaison racine est une erreur. Le code d'erreur 410 ("Identificateur incorrect") doit être renvoyé dans ces cas.

### 6.3 Principes de remplacement par des structures génériques

Le présent paragraphe spécifie le comportement relatif au remplacement, par des structures génériques (ou métacaractères), des identificateurs de contexte et/ou de terminaison qui doivent être appliqués à toutes les commandes. Lorsque l'on traite ces commandes, deux types de remplacement doivent être considérés:

- 1) remplacement dans un identificateur de contexte;
- 2) remplacement dans un identificateur de terminaison.

Dans le cadre des procédures de remplacement par des métacaractères, une liste d'identificateurs de terminaison (TerminationIDList), composée de plusieurs identificateurs de terminaison, est considérée comme étant un identificateur de terminaison remplacé par des structures génériques.

Lorsque l'on effectue une transaction qui contient des contextes remplacés par des métacaractères et éventuellement des terminaisons ainsi remplacées, toutes les commandes contenues dans la transaction sont exécutées séquentiellement pour une instance particulière de l'identificateur de contexte avant de passer à l'instance suivante de cet identificateur. Lorsqu'une transaction comprend plusieurs commandes, il faut que l'identificateur de terminaison (générique ou spécifique) spécifié dans la première commande corresponde à une instance spécifique d'un identificateur de contexte pour que les commandes suivantes puissent être exécutées. Si un identificateur de terminaison (générique ou spécifique) spécifié dans la ou les commandes suivantes de cette transaction ne correspond pas à cette instance spécifique de l'identificateur de contexte, le code d'erreur 431 ("Aucun identificateur de terminaison n'a été mis en correspondance avec une structure générique") est renvoyé et le traitement des instances suivantes de l'identificateur de contexte générique est arrêté, sauf si la commande ayant produit l'erreur est signalée comme étant facultative.

L'exécution de combinaisons particulières d'identificateurs comprenant des structures génériques est étudiée ci-dessous.

### 6.3.1 Identificateur de contexte spécifique avec identificateur de terminaison générique

Lorsque l'identificateur de contexte est spécifique et que la structure générique ALL est utilisée dans l'identificateur de terminaison d'une commande, l'effet est identique à une répétition de la commande pour chacun des identificateurs de terminaison qui correspondent. L'utilisation de la structure ALL ne concerne pas la terminaison racine. Etant donné que chacune de ces commandes peut produire une réponse, la taille de la réponse complète peut être importante. Ainsi, si la structure générique concorde avec plusieurs identificateurs de terminaison dans le contexte considéré, toutes les correspondances possibles sont essayées et les résultats sont signalés à chaque fois. Si aucune des terminaisons spécifiées par l'identificateur de terminaison générique ne se trouve dans le contexte spécifique considéré, le code d'erreur 431 ("Aucun identificateur de terminaison n'a été mis en correspondance avec une structure générique") est renvoyé. Aucune erreur n'est renvoyée pour les terminaisons individuelles spécifiées par l'identificateur de terminaison générique qui ne se trouvent pas dans le contexte spécifique considéré.

Supposons par exemple qu'une passerelle comprenne quatre terminaisons: t1/1, t1/2, t2/1 et t2/2. On suppose que t1/1 et t2/1 appartiennent au contexte 1 et que t1/2 et t2/2 appartiennent au contexte 2.

La commande:

```
Context=1 {Command=t1/* {Descriptor/s}}
```

renvoie:

```
Context=1 {Command=t1/1 {Descriptor/s}}
```

### 6.3.2 Identificateur de contexte générique (ALL) avec identificateur de terminaison spécifique

Lorsque l'identificateur de contexte est remplacé par une structure générique (c'est-à-dire que ContextID = ALL) et que l'identificateur de terminaison est complètement spécifié, l'effet est identique à celui qui est produit par une commande spécifiant un contexte autre que NULL qui contient la terminaison spécifiée. Une recherche doit donc être menée pour trouver le contexte approprié et une seule instance de la commande est exécutée. Aucune erreur n'est signalée pour les contextes ne comprenant pas la terminaison spécifiée. Si la terminaison ne figure dans aucun contexte (autre que NULL), le code d'erreur 431 ("Aucun identificateur de terminaison n'a été mis en correspondance avec un métacaractère") est renvoyé. S'il n'existe pas d'autre contexte que NULL, le code d'erreur 411 ("la transaction fait référence à un ContextID inconnu") est renvoyé. L'emploi de ce type d'action plutôt que la spécification de l'identificateur de contexte est déconseillé mais peut être utile, par exemple pour remédier à un conflit d'états entre la passerelle média et le contrôleur de passerelle média.

Par exemple, en reprenant la configuration de passerelle décrite plus haut, la commande:

```
Context=* {Command=t1/1 {Descriptor/s}}
```

renvoie:

```
Context=1 {Command=t1/1 {Descriptor/s}}
```

### 6.3.3 Identificateur de contexte générique (ALL) avec identificateur de terminaison générique

Lorsque l'identificateur de contexte est remplacé par une structure générique (c'est-à-dire que ContextID = ALL) et que l'identificateur de terminaison est remplacé par une structure générique, l'effet est identique à celui qui est produit par la répétition de la commande, chacun des identificateurs de terminaison spécifiés par l'identificateur de terminaison correspondant à la structure générique pour chaque contexte autre que NULL qui contient un ou plusieurs de ces identificateurs de terminaison. Ainsi, si la structure générique concorde avec plusieurs identificateurs de terminaison d'une instance spécifique de l'identificateur de contexte remplacé par une structure générique, toutes les correspondances possibles sont essayées et les résultats sont signalés à chaque fois. Aucune erreur n'est renvoyée pour les contextes ne contenant pas de terminaison correspondant à l'identificateur de terminaison générique. Aucun code d'erreur n'est renvoyé pour les terminaisons individuellement spécifiées par l'identificateur de terminaison générique qui ne se trouvent pas dans une instance spécifique de l'identificateur de contexte générique. S'il n'existe pas de correspondance entre l'identificateur de contexte générique et l'identificateur de terminaison générique, le code d'erreur 431 ("Aucun identificateur de terminaison n'a été mis en correspondance avec un métacaractère") est renvoyé.

Par exemple, en reprenant la configuration de passerelle décrite plus haut, la commande:

```
Context=* {Command=t1/* {Descriptor/s}}
```

renvoie:

```
Context=1 {Command=t1/1 {Descriptor/s}}
```

```
Context=2 {Command=t1/2 {Descriptor/s}}
```

Lorsque plusieurs commandes figurent dans une demande spécifiant un identificateur de terminaison générique et/ou un identificateur de contexte générique, alors, si la première commande ne correspond pas à la première instance d'un identificateur de contexte et d'un identificateur de terminaison, la commande suivante de la demande ne sera pas exécutée pour cette instance.

### 6.3.4 Réponses génériques

Si des réponses individuelles ne sont pas requises, une réponse générique peut être demandée. Dans un tel cas, une seule réponse est produite, qui contient la REUNION logique de toutes les réponses individuelles qui autrement auraient été produites, les répétitions de valeur étant supprimées. Par exemple, pour une terminaison Ta ayant les propriétés p1=a, p2=b et pour une terminaison Tb ayant les propriétés p2=c, p3=d, une réponse par REUNION contiendrait un identificateur de terminaison générique ainsi que la séquence de propriétés p1=a, p2=b,c et p3=d. Faire appel à une réponse générique peut être particulièrement utile dans le cas d'une commande "Audit". Si l'on utilise une réponse générique par REUNION associée à un contexte remplacé par une structure générique, une seule réponse est envoyée avec la REUNION de toutes les terminaisons individuellement spécifiées par l'identificateur de terminaison. Une telle réponse contiendrait le paramètre Context = ALL, un identificateur de terminaison générique et la séquence des propriétés.

Si une erreur se produit au cours de l'exécution d'une requête générique spécifiant une réponse générique, un traitement particulier est nécessaire afin de fournir des informations utiles concernant les éventuelles erreurs tout en conservant une réponse de taille modeste. Lorsqu'une réponse générique est requise, toutes les instances (telles que spécifiées plus haut) de la commande

considérée doivent être exécutées même si une ou plusieurs d'entre elles conduisent à des erreurs; toutefois, les commandes suivantes de la transaction ne seront pas exécutées (sauf spécification du caractère facultatif de la commande). Plusieurs réponses de commande doivent être renvoyées pour la commande ayant produit l'erreur. La première réponse de commande doit être la réponse générique normale comprenant la REUNION des réponses aux commandes dont l'exécution a été réussie. Si aucune commande n'a pu être exécutée correctement, la REUNION doit être vide. D'autres réponses de commandes relatives aux différents identificateurs de transaction ayant échoué doivent être renvoyées avec le descripteur d'erreur approprié.

Par exemple, la commande:

```
Context=*{Command=t1/*{Descriptor/s}}
```

renvoie en cas d'erreur la réponse suivante:

```
Context=*{Command=t1/*{Union response descriptors},  
Command=t1/3{Error=errorcode}}
```

Le codage du mécanisme de remplacement par des structures génériques est détaillé dans les Annexes A et B.

## 7 Commandes

Le protocole fournit des commandes pour manipuler les entités logiques du modèle de connexion du protocole, les contextes et les terminaisons. Ces commandes permettent un réglage au plus fin degré de granularité pris en charge par le protocole. Par exemple, des commandes permettent d'ajouter des terminaisons à un contexte, de modifier des terminaisons, de soustraire des terminaisons d'un contexte et d'examiner les propriétés de contextes ou de terminaisons. Les commandes permettent un contrôle complet des propriétés des contextes et des terminaisons, dont la spécification des événements qu'une terminaison doit signaler, la spécification des signaux/actions qui doivent être appliqués à une terminaison et la spécification de la topologie d'un contexte (qui entend/voit qui).

La plupart des commandes sont réservées à l'usage particulier du contrôleur MGC en tant qu'émetteur de commandes pour contrôler des passerelles MG en tant que récepteurs de commandes. Les exceptions sont les commandes Notify et ServiceChange, la première étant envoyée par une passerelle MG à un contrôleur MGC et la seconde pouvant être envoyée par une des deux entités. On trouvera ci-dessous une vue du paquetage des commandes, qui sont expliquées plus en détail dans le § 7.2.

- 1) **Add**: cette commande ajoute une terminaison à un contexte. Appliquée à la première terminaison d'un contexte, elle sert à créer un contexte.
- 2) **Modify**: cette commande modifie les propriétés, les événements et les signaux d'une terminaison.
- 3) **Subtract**: cette commande déconnecte une terminaison de son contexte et renvoie des statistiques sur la participation de cette terminaison à ce contexte. Appliquée à la dernière terminaison d'un contexte, elle sert à supprimer ce contexte.
- 4) **Move**: cette commande déplace atomiquement une terminaison vers un autre contexte.
- 5) **AuditValue**: cette commande renvoie les états actuels des propriétés, des événements, des signaux et des statistiques associés aux terminaisons.
- 6) **AuditCapabilities**: cette commande renvoie toutes les valeurs possibles des propriétés, des événements et des signaux de terminaisons, autorisées par la passerelle MG.
- 7) **Notify**: cette commande permet à la passerelle MG d'informer le contrôleur MGC de l'apparition d'événements dans cette passerelle.

- 8) **ServiceChange**: cette commande permet à la passerelle MG de signaler au contrôleur MGC qu'une terminaison ou un groupe de terminaisons est sur le point d'être mis hors service ou vient d'être remis en service. Cette commande est également utilisée par la passerelle MG pour annoncer sa disponibilité à un contrôleur MGC (enregistrement) et pour signaler au contrôleur MGC le redémarrage imminent ou récent de cette passerelle MG. Le contrôleur MGC peut annoncer un transfert à la passerelle MG en lui envoyant une commande ServiceChange. Le contrôleur MGC peut également utiliser cette commande pour demander à la passerelle MG de mettre en ou hors service une terminaison ou un groupe de terminaisons.

Ces commandes sont détaillées aux § 7.2.1 à 7.2.8.

## 7.1 Descripteurs

Les paramètres relatifs à une commande s'appellent des "descripteurs". Un descripteur se compose d'un nom et d'une liste d'éléments dont certains peuvent avoir des valeurs. De nombreuses commandes partagent des descripteurs communs. Le présent paragraphe énumère ces descripteurs. Les descripteurs peuvent être fournis en tant que données de sortie d'une commande. Dans tout renvoi de contenu de descripteur de la sorte, un descripteur vide est représenté par son nom sans être accompagné d'aucune liste. Les paramètres et leur utilisation spécifique pour un type de commande donné sont décrits dans le paragraphe relatif à cette commande.

### 7.1.1 Spécification des paramètres

Les paramètres de commande sont structurés en un certain nombre de descripteurs. En général, le format alphanumérique des descripteurs est le suivant: DescriptorName=<someID>{parm=value, parm=value...}.

Les paramètres peuvent être entièrement spécifiés, surspécifiés ou sous-spécifiés:

- 1) les paramètres entièrement spécifiés ont une seule valeur non ambiguë, que l'émetteur de commande demande au récepteur de la commande d'utiliser pour le paramètre spécifié;
- 2) les paramètres sous-spécifiés utilisent la valeur "CHOOSE" afin de permettre au récepteur de la commande de choisir toute valeur qu'il peut prendre en charge;
- 3) les paramètres surspécifiés possèdent une liste de valeurs possibles dont l'ordre dans la liste correspond à l'ordre de sélection préféré par l'émetteur de la commande. Le récepteur de la commande choisit une seule valeur dans la liste offerte puis renvoie cette valeur à l'émetteur de la commande.

Si un descripteur obligatoire autre que le descripteur "Audit" est non spécifié (c'est-à-dire totalement absent) pour une commande, l'éventuel paquetage de valeurs antérieur de ce descripteur pour cette terminaison est conservé. Dans des commandes autres que Substract, un descripteur "Audit" manquant équivaut à un descripteur "Audit" vide. Le comportement de la passerelle MG au regard de paramètres non spécifiés dans un descripteur varie en fonction du descripteur considéré, comme indiqué dans les paragraphes qui suivent. Chaque fois qu'un paramètre est sous-spécifié ou surspécifié, le descripteur qui contient la valeur choisie par le récepteur est inclus en tant que donnée de sortie de la commande.

Chaque commande spécifie le paramètre TerminationID sur lequel elle opère. Cet identificateur TerminationID peut être "remplacé par une structure générique". Si tel est le cas, l'effet doit être le même que si la commande était répétée avec chacun des identificateurs de terminaisons trouvées concordantes.

### 7.1.2 Descripteur Modem

Le descripteur "Modem" spécifie le type et les paramètres d'un modem, éventuellement nécessaires pour utilisation lors d'une conversation en mode H.324 et en mode texte. Ce descripteur désigne les

types de modem suivants: V.18, V.22, V.22 *bis*, V.32, V.32 *bis*, V.34, V.90, V.91, RNIS synchrone, avec possibilité d'extensions. Par défaut, aucun descripteur "Modem" n'est présent dans une terminaison.

L'emploi du descripteur "Modem" est indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002) et dans les versions ultérieures. Cela veut dire que ce descripteur ne fera pas partie du contenu transmis et, s'il est reçu, qu'il sera soit ignoré soit traité en fonction de l'implémentation. Le type de modem est à spécifier comme attribut des flux de données dans les descripteurs "Local" et "Remote".

### 7.1.3 Descripteur Multiplex

Dans les communications multimédias, un certain nombre de flux médias sont acheminés sur un certain nombre (éventuellement différent du précédent) de supports. Le descripteur "Multiplex" associe les médias et les supports. Il désigne les types de multiplex suivants:

- H.221;
- H.223;
- H.226;
- V.76;
- $N \times 64K$ ;
- extensions possibles,

ainsi qu'un paquetage d'identificateurs de terminaison représentant les supports multiplexés, dans l'ordre. Par exemple:

$$\text{Mux} = \text{H.221}\{\text{MyT3}/1/2, \text{MyT3}/2/13, \text{MyT3}/3/6, \text{MyT3}/21/22\}$$

Le type de multiplex  $N \times 64K$  permet d'implémenter un service à  $N \times 64K$  (par exemple, tel que défini par le débit de transfert des informations de la Rec. UIT-T Q.931 ou par la caractéristique du support de transmission de la Rec. UIT-T Q.763). Du côté du contexte, il prend en charge un seul flux de données à large bande. Lorsqu'une terminaison support est implicitement ajoutée à un contexte à la suite de la création d'une terminaison de multiplexage à  $N \times 64K$ , le descripteur "Stream" de la terminaison support prendra les mêmes valeurs que le descripteur "Stream" défini pour la terminaison multiplexée, mais la largeur de bande de la terminaison support sera de 64 kbit/s.

### 7.1.4 Descripteur Media

Le descripteur "Media" spécifie les paramètres de tous les flux médias. Ces paramètres sont structurés en deux descripteurs: un descripteur "TerminationState" qui spécifie les propriétés d'une terminaison qui ne dépendent pas du flux et un ou plusieurs descripteurs "Stream" dont chacun décrit un seul flux média.

Un flux est identifié par un identificateur (StreamID). L'identificateur de flux doit être compris entre 1 et 65535. Il sert à associer ce flux au contexte auquel il appartient. Plusieurs flux sortant d'une terminaison doivent être synchronisés les uns avec les autres. A l'intérieur du descripteur "Stream", il existe jusqu'à quatre sous-descripteurs, désignés par "LocalControl", "Local", "Remote" et "Statistics". La relation entre ces descripteurs est donc la suivante:

Descripteur "Media"

```
TerminationState Descriptor
Stream
Descriptor
  LocalControl Descriptor
  Local Descriptor
```

Pour plus de commodité, il est possible d'inclure les sous-descripteurs "LocalControl", "Local", "Remote" ou "Statistics" dans le descripteur "Media" sans inclure de descripteur "Stream". Dans ce cas, l'on part du principe que l'identificateur de flux est égal à l'unité.

### 7.1.5 Descripteur TerminationState

Le descripteur "TerminationState" contient la propriété "ServiceStates", la propriété "EventBufferControl" et les propriétés d'une terminaison (définies dans des paquetages) qui ne sont pas spécifiques du flux.

La propriété ServiceStates décrit l'état global de la terminaison (non spécifique du flux). Une terminaison peut être dans l'un des états suivants: "Test" (essai), "OutOfService" (hors service), ou "InService" (en service). L'état "Test" indique que la terminaison est soumise à l'essai. L'état "OutOfService" indique que la terminaison ne peut pas être utilisée pour le trafic. L'état "InService" indique qu'une terminaison peut être utilisée ou est en cours d'utilisation pour le trafic normal. L'état par défaut est "InService".

Les valeurs attribuées aux propriétés peuvent être des valeurs simples (nombre entier/chaîne/énumération) ou être des valeurs sous-spécifiées. Dans ce dernier cas, plusieurs valeurs sont fournies et la passerelle MG peut effectuer un choix entre les possibilités suivantes:

- valeurs à choisir: plusieurs valeurs dans une liste, dont l'une doit être choisie;
- étendues: valeurs minimale et maximale; une valeur comprise entre min et max doit être choisie, valeurs des bornes comprises;
- supérieure/inférieure: la valeur doit être supérieure/inférieure à la valeur spécifiée;
- structure générique CHOOSE: la passerelle MG choisit parmi les valeurs autorisées pour la propriété.

La propriété "EventBufferControl" spécifie si les événements sont mis en mémoire tampon après la détection d'un événement dans le descripteur "Events" ou s'ils sont traités immédiatement. Pour les détails, voir le § 7.1.9.

#### 7.1.5.1 Propriétés relatives à l'état de la terminaison

##### 7.1.5.1.1 Etats de service

**Nom de la propriété:** ServiceStates

**Description:** la valeur de cette propriété indique l'état de service actuel de la terminaison.

**Type:** énumération

**Valeurs possibles:**

InService: la terminaison est en service et fonctionne normalement

OutOfService: la terminaison est hors service et indisponible pour le trafic

Test: la terminaison est en cours d'essais

**Valeur par défaut:** InService

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

### 7.1.5.1.2 Commande de tampon d'événement

**Nom de la propriété:** EventBufferControl

**Description:** spécifie si les événements sont mis en mémoire tampon après détection d'un événement dans le descripteur d'événements, ou s'ils sont traités immédiatement. Voir le § 7.1.9.

**Type:** énumération

**Valeurs possibles:**

LockStep: les événements sont mis en mémoire tampon et traités conformément au § 7.1.9

OFF: les événements sont traités immédiatement

**Valeur par défaut:** OFF

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

### 7.1.6 Descripteur Stream

Un descripteur "Stream" spécifie les paramètres d'un flux bilatéral unique. Ces paramètres sont structurés en trois descripteurs: un qui contient les propriétés spécifiques à un flux et un pour les flux local et distant. Le descripteur "Stream" comporte un identificateur de flux qui désigne le flux. On crée les flux en spécifiant un nouvel identificateur StreamID sur une des terminaisons dans un contexte. On supprime un flux en réglant des descripteurs "Local" et "Remote" vides pour ce flux, les paramètres ReserveGroup et ReserveValue dans LocalControl étant réglés à "False" sur toutes les terminaisons contenues dans le contexte qui prenait précédemment en charge ce flux.

Les identificateurs StreamID ont une signification locale entre le contrôleur MGC et la passerelle MG; ils sont attribués par le contrôleur MGC. Dans un contexte, l'identificateur StreamID sert à indiquer les flux médias qui sont interconnectés: les flux de même identificateur StreamID sont connectés.

Si une terminaison est déplacée d'un contexte à un autre, l'effet sur le contexte d'arrivée de la terminaison déplacée est le même que si l'on ajoutait une nouvelle terminaison ayant les mêmes identificateurs StreamID que la terminaison déplacée.

### 7.1.7 Descripteur LocalControl

Le descripteur "LocalControl" contient la propriété "Mode", les propriétés "ReserveGroup" et "ReserveValue" ainsi que les propriétés d'une terminaison (définies dans des paquetages) qui sont spécifiques au flux et qui présentent un intérêt entre la passerelle MG et le contrôleur MGC. Les valeurs des propriétés peuvent être spécifiées comme indiqué au § 7.1.1.

Les valeurs autorisées pour la propriété "Mode" sont les suivantes: "SendOnly" (émission seulement), "RecvOnly" (réception seulement), "Send/Recv" (émission/réception), "Inactive" (inactive) et "Loop-back" (bouclage). Les valeurs "SendOnly", "RecvOnly" et "LoopBack" se rapportent à l'extérieur du contexte si bien que, par exemple, un flux réglé sur "mode = SendOnly" ne transmettra pas au contexte le média reçu. Lorsqu'un flux est réglé à la valeur "LoopBack" sur une terminaison, le média reçu (descripteur "Local") sur cette terminaison sera bouclé vers le côté émetteur (descripteur "Remote") de la terminaison et aucun média ne sera transmis entre cette terminaison et les autres terminaisons contenues dans ce contexte. Le média bouclé doit être émis conformément au descripteur "Remote". La valeur par défaut de la propriété de mode est: "Inactive". Les signaux et les événements ne sont pas affectés par le paramètre de mode. La

propriété de mode "LocalControl" a priorité sur tout autre mode spécifié dans les descripteurs "Local" et "Remote".

Les propriétés "Reserve" à valeurs booléennes d'une terminaison: "ReserveValue" et "ReserveGroup", indiquent ce que la passerelle MG doit faire lorsqu'elle reçoit un descripteur "Local" et/ou "Remote".

Si la valeur d'une propriété "Reserve" est mise à "True" (Vrai), la passerelle MG doit réserver des ressources pour toutes les variantes spécifiées dans les descripteurs "Local" et/ou "Remote" pour lesquelles elle a actuellement des ressources disponibles. Elle doit répondre avec les variantes pour lesquelles elle réserve des ressources. Si elle ne peut prendre en charge une quelconque de ces variantes, elle doit envoyer au contrôleur MGC une réponse qui contient des descripteurs "Local" et/ou "Remote" vides. Si les médias commencent à s'écouler tandis que plus d'une variante est réservée, les paquets médias peuvent être envoyés ou reçus par l'une quelconque des variantes et doivent être traités, même si à un moment donné une seule variante à la fois peut être active.

Si la valeur d'une propriété "Reserve" est mise à "False" (Faux), la passerelle MG doit choisir une des variantes spécifiées dans le descripteur "Local" (s'il est présent) et une des variantes spécifiées dans le descripteur "Remote" (s'il est présent). Si la passerelle MG n'a pas encore réservé de ressources pour prendre en charge la variante choisie, elle doit le faire. Si, d'autre part, elle a déjà réservé des ressources pour la terminaison concernée (du fait d'un échange précédent où les propriétés ReserveValue et/ou ReserveGroup ont la valeur "True"), elle doit libérer les éventuelles ressources en excès qu'elle avait précédemment réservées. Enfin, la passerelle MG doit envoyer au contrôleur MGC une réponse contenant les variantes pour le descripteur "Local" et/ou "Remote" qu'elle a choisi. Si la passerelle n'a pas les ressources suffisantes pour prendre en charge une quelconque des variantes spécifiées, elle doit répondre avec un code d'erreur 510 ("Ressources insuffisantes").

La valeur par défaut des propriétés "ReserveValue" et "ReserveGroup" est "False" (Faux). Des renseignements complémentaires relatifs à l'utilisation des deux propriétés "Reserve" se trouvent au § 7.1.8.

Un nouveau réglage du descripteur "LocalControl" remplace entièrement le réglage précédent de ce descripteur dans la passerelle MG. Ainsi, pour conserver des informations provenant du réglage précédent, le contrôleur MGC doit inclure ces informations dans le nouveau réglage. Si le contrôleur MGC souhaite supprimer une quelconque information du descripteur existant, il renvoie simplement ce descripteur (dans une commande Modify), l'information indésirable en étant éliminée.

NOTE – La propriété "Mode" est également désignée par le terme de "StreamMode" dans les définitions de codage présentées dans les Annexes A et B. Ces termes sont interchangeables dans la Rec. UIT-T H.248.

### **7.1.7.1 Propriétés de commande locale**

#### **7.1.7.1.1 Mode**

**Nom de la propriété:** StreamMode

**Description:** la valeur de cette propriété indique l'état de service actuel de la terminaison.

**Type:** énumération

**Valeurs possibles:**

Inactive: la terminaison ne transmet aucun média pour le flux

SendOnly: la terminaison transmet le média pour le flux de l'intérieur à l'extérieur du contexte

RecvOnly: la terminaison transmet le média pour le flux de l'extérieur à l'intérieur du contexte  
SendRecv: la terminaison transmet le média pour le flux aussi bien à l'intérieur qu'à l'extérieur du contexte  
LoopBack: la terminaison reboucle vers l'émetteur le média reçu pour le flux

**Valeur par défaut:** Inactive  
**Propriété définie dans l'état:** LocalControl  
**Caractéristiques:** lecture/écriture

#### 7.1.7.1.2 ReserveGroup

**Nom de la propriété:** ReserveGroup

**Description:** spécifie si la passerelle MG doit réserver les ressources afin de prendre en charge un seul groupe de médias ou autant de groupes de médias qu'elle le peut, tels qu'ils sont définis dans les descripteurs "Local" et "Remote". Voir le § 7.1.8.

NOTE – Le terme "groupe de médias" désigne le contenu de la production ASN.1 "PropertyGroup" (voir l'Annexe A) en codage binaire, ou une description en codage textuel d'une session individuelle de protocole SDP.

**Type:** valeur booléenne

**Valeurs possibles:**

True: la passerelle MG va réserver tous les groupes de médias possibles qui sont indiqués dans les descripteurs "Local" et "Remote".  
False: la passerelle MG va réserver un seul groupe de médias à partir de chacun des descripteurs "Local" et "Remote" conformément aux politiques décrites dans le § 7.1.8.

**Valeur par défaut:** False  
**Propriété définie dans l'état:** LocalControl  
**Caractéristiques:** lecture/écriture

#### 7.1.7.1.3 ReserveValue

**Nom de la propriété:** ReserveValue

**Description:** spécifie si la passerelle MG doit réserver des ressources afin de prendre en charge un seul ensemble de valeurs de propriété (par exemple un seul codec et ses attributs associés) ou autant de tels ensembles qu'elle le peut, tels qu'ils sont définis dans les descripteurs "Local" et "Remote". Voir le § 7.1.8.

**Type:** valeur booléenne

## Valeurs possibles:

True: la passerelle MG va réserver des ressources afin de desservir autant que possible des ensembles de valeurs de propriété qui sont indiqués dans le groupe de médias sélectionné (si ReserveGroup a la valeur "False") ou qui sont indiqués dans chaque groupe de médias (si ReserveGroup a la valeur "True") contenu dans les descripteurs "Local" et "Remote".

False: la passerelle MG va réserver un seul ensemble de valeurs de propriété à partir de ceux qui sont indiqués dans le groupe de médias sélectionné (si ReserveGroup a la valeur "False") ou dans chaque groupe de médias (si ReserveGroup a la valeur "True") contenu dans les descripteurs "Local" et "Remote".

**Valeur par défaut:** False

**Propriété définie dans l'état:** LocalControl

**Caractéristiques:** lecture/écriture

### 7.1.8 Descripteurs Local et Remote

Le contrôleur MGC utilise les descripteurs "Local" et "Remote" pour réserver et engager des ressources de la passerelle MG en vue du codage et décodage de média pour le ou les flux et la terminaison auxquels elles s'appliquent. La passerelle inclut ces descripteurs dans sa réponse afin d'indiquer ce qu'elle est préparée à prendre effectivement en charge. La passerelle MG doit inclure des propriétés supplémentaires et leurs valeurs dans sa réponse si ces propriétés sont obligatoires bien que ne figurant pas dans les demandes faites par le contrôleur MGC (par exemple en spécifiant des paramètres de codage vidéo détaillés alors que le contrôleur MGC n'a spécifié que le type de charge utile).

Afin d'éviter toute ambiguïté lors de la demande de réservation et de validation de ressources par la passerelle MG, il y a lieu que le contrôleur MGC fournisse autant d'informations que nécessaire lors d'une application de sous-spécification (c'est-à-dire lors de l'utilisation de la structure CHOOSE), de façon que la passerelle MG puisse faire une sélection non ambiguë. Par exemple, lors de l'utilisation de la structure CHOOSE sans spécifier le type d'application requis (par exemple le "nom du média" en cas de codage de protocole SDP), des informations complémentaires peuvent être requises (par exemple des lignes d'attribut en cas de codage de protocole SDP).

Le descripteur "Local" se rapporte aux médias reçus par la passerelle MG et le descripteur "Remote" à ceux qu'elle envoie.

Lors du codage en mode texte du protocole, ces descripteurs se composent de descriptions de session telles que définies dans le protocole SDP (RFC 2327). Dans les descriptions de session envoyées par le contrôleur MGC à la passerelle MG, les exceptions suivantes à la syntaxe du Document RFC 2327 sont autorisées:

- les lignes "s = ", "t = " et "o = " sont facultatives;
- l'utilisation de CHOOSE est permise à la place d'une seule valeur de paramètre;
- l'utilisation de variantes est autorisée à la place d'une seule valeur de paramètre.

Un descripteur "Stream" spécifie un seul flux média bilatéral et une unique description de session ne doit donc pas contenir plus d'une description de média (ligne "m = "). Un descripteur de session peut contenir des descriptions de session supplémentaires en tant que solutions de remplacement. Chaque flux média d'une terminaison doit figurer dans un descripteur "Stream" distinct. Lorsque plusieurs descriptions de session sont fournies dans un même descripteur, les lignes "v = " sont

requis en tant que délimiteurs; sinon, elles sont facultatives dans les descriptions de session envoyées à la passerelle MG. Les implémentations doivent accepter des descriptions de session entièrement conformes au document RFC 2327 conformément aux restrictions ci-dessus. Lors du codage en mode binaire du protocole, le descripteur se compose de groupes de paramètres (paires de valeurs d'étiquette) comme spécifié dans l'Annexe C. Chacun de ces groupes peut contenir les paramètres d'une description de session.

La sémantique des descripteurs "Local" et "Remote" est spécifiée en détails ci-après. La spécification comporte deux parties. La première partie spécifie l'interprétation du contenu du descripteur. La seconde partie spécifie les actions que la passerelle MG doit entreprendre à la réception des descripteurs "Local" et "Remote". Ces actions dépendent des valeurs des propriétés ReserveValue et ReserveGroup du descripteur "LocalControl".

Chacun des descripteurs "Local" et "Remote" (ou tous les deux) peut être:

- non spécifié (c'est-à-dire absent);
- vide;
- sous-spécifié par le biais de l'utilisation de la structure CHOOSE dans une valeur de propriété;
- complètement spécifié;
- surspécifié par le biais de la présentation de plusieurs groupes de propriétés et de plusieurs valeurs de propriétés possibles dans un ou plusieurs de ces groupes.

Lorsqu'ils sont transmis du contrôleur MGC à la passerelle MG, les descripteurs sont interprétés conformément aux règles indiquées dans le § 7.1.1, avec les commentaires supplémentaires suivants vers une fin de clarification:

- a) un descripteur "Local" ou "Remote" non spécifié est considéré comme étant un paramètre obligatoire qui manque. Il exige que la passerelle MG utilise la dernière valeur spécifiée pour ce descripteur. Il est possible qu'il n'y ait pas de valeur précédemment spécifiée, auquel cas le descripteur concerné est ignoré dans le traitement ultérieur de la commande;
- b) un descripteur "Local" ("Remote") vide contenu dans un message provenant du contrôleur MGC signifie une demande de libérer les éventuelles ressources réservées pour le flux média qui ont été reçues (envoyées);
- c) si plusieurs groupes de propriétés sont présents dans un descripteur "Local" ou "Remote" ou s'il y a plusieurs valeurs dans un groupe, l'ordre de préférence est décroissant;
- d) les propriétés sous-spécifiées ou surspécifiées contenues dans un groupe de propriétés envoyées par le contrôleur MGC constituent des demandes à la passerelle MG de choisir une ou plusieurs valeurs qu'elle peut prendre en charge pour chacune de ces propriétés. Dans le cas d'une propriété surspécifiée, la liste des valeurs est dans l'ordre décroissant de préférence.

Sous réserve des règles précédentes, les actions subséquentes dépendent des valeurs des propriétés ReserveValue et ReserveGroup du descripteur "LocalControl".

Si la valeur de ReserveGroup est "True" (Vrai), la passerelle MG réserve les ressources requises pour prendre en charge un aussi grand nombre que possible des variantes de groupes de propriétés qu'elle peut prendre actuellement en charge. Si la valeur de ReserveValue est "True" (Vrai), la passerelle MG réserve les ressources requises pour prendre en charge un aussi grand nombre que possible des variantes de valeurs de propriétés qu'elle peut prendre actuellement en charge.

NOTE – Si un descripteur "Local" ou "Remote" contient plusieurs groupes de propriétés et si la valeur de ReserveGroup est "True" (Vrai), il est demandé à la passerelle MG de réserver des ressources de manière qu'elle puisse décoder ou coder le flux média conformément à l'une quelconque des variantes. Par exemple, si le descripteur "Local" contient deux groupes de propriétés, l'un spécifiant l'audio de la loi A de la Rec. UIT-T G.711 en mode paquet et l'autre l'audio de la Rec. UIT-T G.723.1, la passerelle MG réserve des

ressources de manière à pouvoir décoder un flux audio codé dans le format loi A G.711 ou dans le format G.723.1. La passerelle n'a pas à réserver de ressources pour décoder simultanément deux flux audio, l'un codé selon le format loi A de G.711 et l'autre dans le format G.723.1. L'utilisation de ReserveValue est analogue.

Si la valeur de ReserveGroup est "True" ou si celle de ReserveValue est "True", les règles ci-après s'appliquent:

- si la passerelle MG n'a pas de ressources suffisantes pour prendre en charge toutes les variantes demandées par le contrôleur MGC et si celui-ci a demandé des ressources dans chacun des descripteurs "Local" et "Remote", il convient que la passerelle MG réserve des ressources pour prendre en charge au moins une variante dans chaque descripteur "Local" et "Remote";
- si la passerelle MG n'a pas de ressources suffisantes pour prendre en charge au moins une variante dans un descripteur "Local" ("Remote") reçu du contrôleur MGC, elle doit renvoyer un descripteur "Local" ("Remote") vide comme réponse;
- dans sa réponse au contrôleur MGC, lorsque celui-ci avait inclus des descripteurs "Local" et "Remote", la passerelle MG doit inclure des descripteurs "Local" et "Remote" pour tous les groupes de propriétés et toutes les valeurs de propriétés pour lesquels elle avait réservé des ressources. Si la passerelle est incapable de prendre en charge au moins l'une des options contenues dans le descripteur "Local" ("Remote") reçu du contrôleur MGC, elle doit renvoyer un descripteur "Local" ("Remote") vide;
- si la propriété Mode du descripteur "LocalControl" est "RecvOnly", "SendRecv" ou "LoopBack", la passerelle MG doit être préparée pour recevoir des médias codés selon l'une quelconque des variantes incluses dans sa réponse au contrôleur MGC.

Si la valeur de ReserveGroup est "False" (Faux) et si celle de ReserveValue est "False" (Faux), il convient que la passerelle MG applique les règles ci-après pour ramener chaque descripteur "Local" et "Remote" à une seule variante:

- la passerelle MG choisit la première variante dans le descripteur "Local" pour laquelle elle est à même de prendre en charge au moins une variante dans le descripteur "Remote";
- si la passerelle MG est incapable de prendre en charge au moins une variante contenue dans "Local" et une contenue dans "Remote", elle renvoie le code d'erreur 510 ("Ressources insuffisantes");
- la passerelle renvoie la variante qu'elle a choisie dans chacun des descripteurs "Local" et "Remote".

Un nouveau réglage d'un descripteur "Local" ou "Remote" remplace entièrement le réglage précédent de ce descripteur dans la passerelle MG. Ainsi, pour conserver des informations provenant du réglage précédent, le contrôleur MGC doit inclure ces informations dans le nouveau réglage. Si le contrôleur MGC souhaite supprimer une quelconque information du descripteur existant, il renvoie simplement le descripteur (dans une commande Modify), l'information indésirable en étant éliminée.

### **7.1.9 Descripteur Events**

Le descripteur "Events" contient un identificateur de demande et une liste d'événements que la passerelle MG est appelée à détecter et à signaler. L'identificateur de demande (RequestIdentifier) est utilisé pour associer la demande aux notifications que celle-ci peut déclencher. Les événements demandés sont, par exemple, des tonalités de télécopie, des résultats d'essai de continuité et des transitions de raccrochage-décrochage. Le paramètre RequestIdentifier est omis si le descripteur "Events" est vide (c'est-à-dire aucun événement n'est spécifié).

Chaque événement indiqué dans le descripteur "Events" contient le nom de l'événement, un identificateur de flux facultatif, un fanion KeepActive (maintenir en activité) facultatif, un fanion

facultatif NotifyBehaviour, un fanion facultatif ResetEventsDescriptor et des paramètres facultatifs. Le nom d'événement se compose d'un nom de paquetage (dans lequel l'événement est défini) et d'un identificateur d'événement. La structure générique ALL peut être utilisée pour l'identificateur d'événement afin d'indiquer que tous les événements issus du paquetage spécifié doivent être détectés. L'identificateur de flux par défaut est 0, indiquant que l'événement à détecter n'est pas associé à un flux média particulier. Les événements peuvent avoir des paramètres, ce qui permet à une même description d'événement d'avoir une certaine variation de sens, ce qui évite d'avoir à créer de grands nombres d'événements individuels. Les autres paramètres d'événement sont définis dans le paquetage.

Lorsqu'un événement de fin de script de numérotation (DigitMap) est présent dans l'identificateur EventsDescriptor, de manière explicite ou implicite, le paramètre EventDM est utilisé pour acheminer soit le nom soit la valeur du script de numérotation DigitMap associé. Voir le § 7.1.14 pour plus de détails.

Lorsqu'un événement est traité en fonction du contenu d'un descripteur "Events" actif et qu'il s'avère être présent dans ce descripteur ("reconnu"), l'action par défaut de la passerelle MG consiste à envoyer une commande Notify au contrôleur MGC. La notification peut être différée si l'événement est absorbé dans la chaîne de numérotation courante d'un script de numérotation actif (voir le § 7.1.14). L'envoi d'une commande Notify peut être influencé par le fanion NotifyBehaviour. De plus, la reconnaissance d'événement peut provoquer l'arrêt de signaux actifs ou le remplacement du descripteur actuel Events et/ou Signals, conformément à la description donnée à la fin du présent paragraphe. A moins que le descripteur "Events" ne soit remplacé par un autre descripteur, il reste actif après qu'un événement a été reconnu.

Si la valeur contenue dans la propriété EventBufferControl est égale à "LockStep", à la suite de la détection d'un tel événement, le traitement normal des événements est suspendu. Tout événement détecté subséquemment et présent dans le descripteur "EventBuffer" (mémoire tampon d'événements) est ajouté à la fin de la mémoire tampon d'événements (une file d'attente FIFO – premier arrivé, premier servi) ainsi que l'heure à laquelle il a été détecté. La passerelle doit attendre le chargement d'un nouveau descripteur "Events". Un nouveau descripteur "Events" peut être chargé soit en tant que résultat de la réception d'une commande avec un nouveau EventsDescriptor soit en activant le descripteur "Events" imbriqué.

Si EventBufferControl est égal à "Off" (inactif), la passerelle MG poursuit le traitement en s'appuyant sur le descripteur "Events" actif.

Dans le cas d'activation d'un descripteur "Events" imbriqué, la passerelle MG poursuit le traitement des événements en s'appuyant sur le descripteur "Events" nouvellement activé.

NOTE 1 – Pour des besoins de traitement de mémoire tampon d'événements, l'activation d'un descripteur "Events" imbriqué équivaut à la réception d'un nouveau descripteur "Events".

Lorsque la passerelle MG reçoit une commande avec un nouveau descripteur "Events" (EventsDescriptor), sa mémoire tampon d'événements peut déjà contenir un ou plusieurs événements. La valeur de EventBufferControl détermine alors la manière dont la passerelle MG traite ces événements placés en mémoire tampon.

### *Cas 1*

Si EventBufferControl est égal à "LockStep" et si la passerelle MG reçoit un nouveau descripteur "Events", elle vérifie la mémoire tampon d'événements FIFO et exécute les actions suivantes:

- 1) si la mémoire tampon d'événements est vide, la passerelle MG attend la détection d'événements fondés sur le nouveau descripteur "Events";

- 2) si la mémoire tampon d'événements n'est pas vide, la passerelle MG traite la file d'attente FIFO en commençant par le premier événement:
  - a) si l'événement contenu dans la file d'attente figure parmi les événements énumérés dans le nouveau descripteur "Events", la passerelle MG agit sur cet événement en le retirant de la mémoire tampon d'événements. L'horodatage de la commande Notify doit être l'heure de la détection effective de l'événement. La passerelle attend ensuite un nouveau descripteur "Events". Pendant cette attente, tous les événements détectés qui apparaissent dans le descripteur "EventsBuffer" sont placés dans la mémoire tampon d'événements. Lorsqu'un nouveau descripteur "Events" est reçu, le traitement des événements se répète à partir de l'étape 1);
  - b) si l'événement n'est pas contenu dans le nouveau descripteur "Events", la passerelle MG doit l'ignorer et recommencer depuis l'étape 1).

## Cas 2

Si EventBufferControl est égal à "Off" (inactif) et si la passerelle MG reçoit un nouveau descripteur "Events", elle traite les nouveaux événements contenus dans le nouveau descripteur "Events".

Si la passerelle MG reçoit une commande lui demandant de fixer la valeur de EventBufferControl à "Off" (inactif), tous les événements contenus dans la mémoire tampon d'événements doivent être rejetés.

La passerelle MG peut rapporter plusieurs événements dans une seule transaction tant que cela ne retarde pas inutilement le rapport des événements individuels.

Pour les procédures relatives à la transmission de la commande Notify, se reporter à l'annexe ou à la Recommandation de la sous-série H.248.x appropriée concernant des considérations de transport spécifiques.

La valeur par défaut de la propriété EventBufferControl est "Off" (inactif).

NOTE 2 – Etant donné que la propriété EventBufferControl est contenue dans le descripteur "TerminationState" (descripteur d'état de terminaison), la passerelle pourrait recevoir une commande qui modifie la propriété EventBufferControl et n'inclut pas un descripteur "Events".

Normalement, la reconnaissance d'un événement arrête tous les signaux actifs éventuels. Lorsque KeepActive est spécifié dans l'événement, la passerelle MG ne doit pas interrompre d'éventuels signaux qui étaient actifs dans la terminaison où l'événement a été détecté.

Le fanion NotifyBehaviour peut être utilisé afin d'indiquer que la commande Notify est:

- envoyée immédiatement (la valeur par défaut est: "NotifyImmediate");
- jamais envoyée ("NeverNotify");
- ou régulée (c'est-à-dire envoyée ou supprimée) en fonction de la charge du contrôleur MGC ("RegulatedNotify").

Voir dans l'Annexe E.15 ci-dessous de plus amples détails sur l'utilisation du fanion NotifyBehaviour. Lorsqu'il est utilisé avec des scripts de numérotation, le comportement de notification apparaît dès que le script de numérotation actif est réalisé. La notification réglée peut être associée à un autre descripteur "Events" ou "Signals" imbriqué et réglé. Si une notification est réglée (c'est-à-dire supprimée), alors cet autre descripteur imbriqué et réglé doit être activé. Si la notification n'est pas réglée, alors le descripteur original et imbriqué est déclenché. Si la valeur "NotifyImmediate" ou "NeverNotify" est choisie, alors le descripteur original et imbriqué est déclenché dès la détection de l'événement.

Les résultats de l'apparition du fanion "ResetEventDescriptor" qui a été activé au sujet d'un événement varient selon que le descripteur "Events" contenant l'événement est ou non imbriqué. Si c'est le cas, le fanion ResetEventsDescriptor provoque, dans la terminaison considérée, la

réinitialisation du descripteur "Events" à son état antérieur à l'activation du descripteur imbriqué (c'est-à-dire du dernier descripteur "Events" explicitement activé par une commande Modify ou du descripteur "Events" fourni par la passerelle MG s'il n'y a pas eu de commande Modify depuis la dernière réinitialisation de la terminaison par sa mise dans le contexte NULL). Dans le cas d'un descripteur "Events" non imbriqué, le fanion ResetEventsDescriptor provoque la réinitialisation du descripteur "Events" actif dans la terminaison considérée, par réactivation de tout événement de réalisation de script de numérotation qui aurait été mis en correspondance et désactivé.

Le fanion ResetEventsDescriptor ne doit être activé que sur des terminaisons physiques. Il doit l'être lorsque la terminaison est dans le contexte NULL. Si le fanion ResetEventsDescriptor apparaît lorsque la terminaison est hors du contexte NULL, alors il ne doit avoir aucun effet.

Un événement peut comporter un descripteur "Signals" imbriqué et un descripteur "Events" imbriqué qui, s'ils sont présents lors de la reconnaissance de l'événement, remplacent le descripteur "Signals" ou "Events" actuel. Il est possible, par exemple, de spécifier que le signal de tonalité d'invitation à numéroter sera produit lors de la reconnaissance d'un événement de décrochage ou que le signal de tonalité d'invitation à numéroter soit interrompu si un chiffre est reconnu. Un contrôleur MGC ne doit pas envoyer de descripteurs "Events" contenant un événement qui, à la fois, est marqué et contient un descripteur "Signals" imbriqué.

Un seul niveau d'imbrication est autorisé. Un descripteur "Events" imbriqué ne doit pas contenir un autre descripteur "Events" imbriqué mais il peut contenir un descripteur "Signals" imbriqué.

Un descripteur "Events" reçu par une passerelle MG remplace tout descripteur "Events" antérieur. La notification d'événement en cours doit se terminer et les événements détectés après l'exécution de la commande contenant le nouveau descripteur "Events" doivent être traités conformément au nouveau descripteur "Events".

Un descripteur "Events" vide désactive toute reconnaissance ou tout rapport d'événement. Il vide la mémoire tampon EventBuffer et désactive toute accumulation d'événements en mode "LockStep": les seuls événements rapportés seront ceux qui se produisent tandis que le descripteur "Events" est actif. Si le descripteur "Events" est activé pendant que la terminaison fonctionne en mode LockStep, la mémoire tampon d'événements est immédiatement vidée.

#### **7.1.10 Descripteur EventBuffer (mémoire tampon d'événements)**

Le descripteur "EventBuffer" contient une liste d'événements, avec leurs paramètres le cas échéant, que la passerelle MG est appelée à détecter et mémoriser en tampon lorsque la propriété EventBufferControl est égale à "LockStep" (voir le § 7.1.9).

#### **7.1.11 Descripteur Signals**

Les signaux sont des médias produits par une passerelle MG, tels que des tonalités et des annonces, ainsi que des signaux liés au support tels que ceux qui sont destinés au crochet commutateur. Des signaux plus complexes peuvent comporter une suite de tels signaux simples, entrecoupée de signaux médias ou liés au support, et conditionnée par la réception et l'examen de ceux-ci. Des exemples en sont le renvoi en écho de données reçues comme dans le paquetage des essais de continuité. Les signaux peuvent aussi demander que le contenu média soit préparé pour des signaux ultérieurs.

Un descripteur "Signals" est un paramètre qui contient le paquetage des signaux que la passerelle MG est appelée à appliquer à une terminaison. Un descripteur "Signals" contient un certain nombre de signaux et/ou de listes séquentielles de signaux. Un descripteur "Signals" peut contenir un nombre égal à zéro de signaux et de listes séquentielles de signaux. La prise en charge de listes séquentielles de signaux est facultative.

Les signaux sont définis dans des paquetages. Le nom d'un signal doit être celui du paquetage dans lequel il est défini, plus un identificateur de signal. Aucune structure générique ne doit être utilisée

comme identificateur de signal. Les signaux qui sont contenus dans un descripteur "Signals" possèdent un paramètre facultatif d'identificateur de flux (dont la valeur par défaut est 0, ce qui indique que le signal n'est associé à aucun flux média particulier), un type facultatif de signal (voir ci-dessous), une durée facultative et éventuellement des paramètres définis dans le paquetage qui définit le signal. Cela permet à un même signal d'avoir une certaine variation de sens ce qui évite d'avoir à créer de grands nombres de signaux individuels.

Le paramètre facultatif "NotifyCompletion" permet à un contrôleur MGC d'indiquer son souhait d'être avisé lorsque le signal achève son exécution. Les cas possibles sont les suivants: expiration de la temporisation du signal (ou sinon achèvement de lui-même), interruption du signal par un événement, fin d'un cycle ou d'une itération du signal, arrêt du signal par le remplacement d'un descripteur "Signals" ou arrêt ou non-lancement du signal pour d'autres raisons. Si le paramètre NotifyCompletion ne figure pas dans un descripteur "Signals", une notification n'est produite que si le signal a été arrêté ou s'il n'a jamais été lancé pour d'autres raisons. Pour qu'il soit rapporté, l'événement de fin de signal (voir le § E.1.2) doit être activé dans le descripteur "Events" actuellement actif. Le paramètre facultatif "RequestID" peut être associé à une instance particulière d'un identificateur SignalID lorsque de multiples signaux ayant le même identificateur sont demandés. Cela permet au contrôleur MGC de différencier différents événements observés de réalisation d'un signal pour l'instance considérée de ce signal. Le paramètre RequestID ne doit pas être inclus si le paramètre NotifyCompletion n'est pas présent.

La durée est une valeur d'entier exprimée en centièmes de seconde.

Il existe trois types de signaux:

- signal commuté (OO, *OnOff*): le signal dure jusqu'à ce qu'il soit interrompu;
- signal temporisé (TO, *TimeOut*): le signal dure jusqu'à ce qu'il soit interrompu ou jusqu'à ce qu'un intervalle de temps spécifique se soit écoulé;
- signal bref (BR, *brief*): le signal s'interrompt de lui-même à moins qu'un nouveau descripteur "Signals" ne soit appliqué pour l'interrompre plus tôt; aucune valeur de temporisation n'est nécessaire.

Si le type par défaut d'un signal autre que le type TO est supplanté par celui-ci dans le descripteur "Signals", le paramètre de durée doit être présent.

Si le type du signal est spécifié dans le descripteur "Signals", il supplante le type de signal par défaut (voir le § 12.1.4). Si une durée est spécifiée pour un signal commuté, elle doit être ignorée.

Une liste séquentielle de signaux se compose d'un identificateur de liste de signaux (SignalListID) et d'une séquence de signaux à exécuter consécutivement. Seul l'élément final de la séquence de signaux, dans une telle liste, peut être un signal commuté. La durée d'une liste séquentielle de signaux est la somme des durées des signaux qu'elle contient, plus la somme des durées d'intervalle de retard entre signaux qui sont spécifiées sous forme de paramètres de ces signaux.

Si un retard entre signaux est spécifié pour un signal qui n'est pas contenu dans une liste séquentielle de signaux ou qui est le dernier élément d'une liste séquentielle de signaux, ce retard doit être ignoré et le signal doit être considéré comme ayant été exécuté à la fin de ce signal, avant l'application du rythme de retard entre signaux. La durée d'un signal, figurant dans une liste de signaux avec retard entre signaux, inclut le rythme de retard entre signaux.

De multiples signaux et listes séquentielles de signaux doivent être exécutés simultanément dans le même descripteur "Signals".

Les signaux sont définis comme se propageant de la terminaison vers l'extérieur du contexte, sauf spécification contraire. Si le sens du signal est spécifié dans un descripteur "Signals", ce sens supplante le sens par défaut du signal. Dans le cas des signaux qui possèdent un paramètre de sens, le paramètre de sens du protocole de base a priorité sur tout autre paramètre de sens défini dans un

paquetage lorsque les deux paramètres sont spécifiés. Si le contrôleur MGC spécifie un sens pour un signal avec lequel la passerelle MG ne peut pas être conforme, celle-ci doit renvoyer un code d'erreur 501 ("Fonctionnalité non implémentée"). Lorsque le même signal est appliqué à de multiples terminaisons à l'intérieur d'une même transaction, il convient que la passerelle MG envisage d'utiliser la même ressource pour produire ces signaux.

La production d'un signal dans une terminaison est arrêtée par l'application d'un nouveau descripteur "Signals" ou par la détection d'un événement dans cette terminaison (voir le § 7.1.9).

Un nouveau descripteur "Signals" remplace tout descripteur "Signals" existant. Tous les signaux appliqués à la terminaison qui ne sont pas indiqués dans le descripteur de remplacement doivent être arrêtés et de nouveaux signaux doivent être appliqués, sauf dans les conditions suivantes. Les signaux présents dans le descripteur de remplacement et contenant le fanion KeepActive doivent se poursuivre s'ils sont en cours d'exécution et n'ont pas encore fini. Si un signal qui n'est pas en cours d'exécution et qui contient le fanion KeepActive est contenu dans un descripteur "Signals" de remplacement, ce signal doit être ignoré. Si le descripteur de remplacement contient une liste séquentielle de signaux contenant le même identificateur que le descripteur existant,

- le type de signal et la séquence de signaux dans la liste séquentielle figurant dans le descripteur de remplacement doivent être ignorés;
- l'exécution des signaux contenus dans la liste séquentielle du descripteur existant ne doit pas être interrompue.

### 7.1.12 Descripteur Audit

Le descripteur "Audit" spécifie les informations à examiner. Il spécifie la liste de descripteurs et/ou les propriétés distinctes à retourner. Il peut être utilisé dans n'importe quelle commande afin de forcer le renvoi d'un descripteur quelconque contenant les valeurs courantes de ses propriétés, événements, signaux et statistiques même si ce descripteur n'était pas présent dans la commande ou ne possède aucun paramètre sous-spécifié. Les éléments pouvant être contenus dans le descripteur "Audit" sont les suivants:

Modem (à éviter, voir le § 7.1.2)	
Mux	
Events	
Media	
Signals	
ObservedEvents	
DigitMap	
Statistics	
Packages	
EventBuffer	
Eléments analysés distincts:	
Propriétés des médias Événements Tampon d'événements Signaux, listes de signaux	Scripts de numérotation Statistiques Paquetages Attribut de contexte

La commande Audit peut être vide: dans ce cas, aucun descripteur n'est renvoyé. Cela est utile dans la commande Subtract, afin d'empêcher le renvoi de statistiques, surtout en cas d'utilisation d'une structure générique.

### 7.1.13 Descripteur ServiceChange (changement de service)

Le descripteur "ServiceChange" contient les paramètres suivants:

- ServiceChangeMethod;
- ServiceChangeReason;
- ServiceChangeAddress;
- ServiceChangeDelay;
- ServiceChangeProfile;
- ServiceChangeVersion;
- ServiceChangeMGCID;
- TimeStamp;
- Extension;
- ServiceChangeInfo;
- ServiceChangeIncompleteFlag.

Voir le § 7.2.8.

### 7.1.14 Descripteur DigitMap (script de numérotation)

#### 7.1.14.1 Définition, création, modification et suppression d'un script de numérotation

Un script de numérotation est un plan de numérotation résidant dans la passerelle MG, qui sert à détecter et à signaler les événements de numérotation reçus au sujet d'une terminaison. Le descripteur "DigitMap" contient le nom d'un script d'instructions et le script à attribuer. Un script peut être préenregistré dans la passerelle MG par une action de gestion et être répertorié par son nom dans un descripteur "Events"; il peut également être défini dynamiquement puis être répertorié par son nom ou bien le script de numérotation lui-même peut être spécifié dans le descripteur "Events". Il est admissible qu'un événement de fin de script de numérotation contenu dans un descripteur "Events" se rapporte par nom à un script de numérotation qui est défini par un descripteur de scripts de numérotation contenu dans la même commande, indépendamment de l'ordre dans lequel ces descripteurs respectifs ont été transmis.

Les scripts de numérotation définis dans un descripteur "DigitMap" peuvent apparaître dans une quelconque des commandes normales de manipulation de terminaison du protocole. Une fois définie, un script de numérotation peut être utilisé dans toutes les terminaisons spécifiées par l'identificateur de terminaison (éventuellement remplacé par une structure générique) contenu dans une telle commande. Les descripteurs DigitMap définis dans une terminaison racine sont globaux et on peut les utiliser dans toutes les terminaisons contenues dans la passerelle MG, à la condition qu'un descripteur "DigitMap" portant le même nom n'ait pas été défini dans la terminaison considérée. Lorsqu'un script de numérotation est défini dynamiquement dans un descripteur "DigitMap":

- un nouveau script de numérotation est créé par la spécification d'un nom qui n'est pas encore défini mais dont la valeur doit être présente;
- une valeur de script de numérotation est mise à jour par fourniture d'une nouvelle valeur pour un nom déjà défini; les terminaisons utilisant actuellement le script de numérotation doivent continuer à utiliser l'ancienne définition; les descripteurs Events subséquents qui utilisent le nom, y compris tout descripteur "Events" dans la commande contenant le descripteur "DigitMap", doivent utiliser le nouveau nom;
- un script de numérotation est supprimé par fourniture d'une valeur vide pour un nom déjà défini. Les terminaisons utilisant actuellement le script de numérotation doivent continuer à utiliser l'ancienne définition.

### 7.1.14.2 Temporisateurs de script de numérotation ("DigitMap")

L'ensemble des chiffres conformes à un script de numérotation peut être protégé par trois temporisateurs, à savoir un temporisateur de départ (T), un temporisateur court (S) et un temporisateur long (L), comme suit:

- 1) le temporisateur de départ (T) est utilisé avant que des chiffres soient disponibles pour traitement en fonction du script de numérotation. Si la valeur du temporisateur de départ est supplantée par une valeur fixée à zéro ( $T = 0$ ), le temporisateur de départ doit être désactivé. Cela implique que la passerelle MG attendra indéfiniment les chiffres.
- 2) si la passerelle MG peut déterminer qu'au moins un chiffre de plus est nécessaire dans une chaîne de chiffres pour que celle-ci corresponde à l'une des structures autorisées dans le script de numérotation, la valeur du temporisateur entre chiffres doit être réglée sur une longue (L) durée (par exemple 16 s);
- 3) si la chaîne de chiffres correspond à un des motifs contenus dans un script de numérotation mais qu'il soit possible que d'autres chiffres soient reçus pour aboutir à une correspondance avec un motif différent, la passerelle doit appliquer un temporisateur court (S) et attendre d'autres chiffres, au lieu de rapporter immédiatement la correspondance.

Les temporisateurs sont des paramètres configurables en fonction d'un script de numérotation. Les valeurs par défaut de ces temporisateurs devraient être fournies au niveau de la passerelle MG, mais elles peuvent être supplantées par des valeurs spécifiées dans le script de numérotation.

### 7.1.14.3 Syntaxe de script de numérotation

La syntaxe formelle du script de numérotation ("DigitMap") est décrite par la règle DigitMap qui se trouve dans la description de syntaxe formelle du protocole (voir les Annexes A et B). Conformément à cette syntaxe, un script de numérotation est défini soit par une chaîne soit par une liste de chaînes. Chaque chaîne de la liste est une séquence d'événements en variante, spécifiée soit sous forme d'une séquence de symboles de script de numérotation, soit sous forme d'une expression normale de symboles de script de numérotation. Les symboles de script de numérotation, chiffres de "0" à "9" et lettres de "A" à une valeur maximale qui dépend du système de signalisation concerné mais qui ne dépasse pas "K", correspondent aux événements spécifiés dans un paquetage qui a été désigné dans le descripteur "Events" sur la terminaison à laquelle le script est actuellement appliqué. (Le mappage entre événements et symboles de script de numérotation est défini dans la documentation des paquetages associés aux systèmes de signalisation en mode voie par voie tels que DTMF, MF ou R2. Les chiffres de "0" à "9" doivent être mappés avec les événements de chiffres correspondants, qui sont contenus dans le système de signalisation concerné. Il convient que les lettres soient attribuées de façon logique, afin de faciliter l'utilisation de la notation d'étendue pour d'autres événements.)

La lettre "x" est utilisée comme structure générique, désignant n'importe quel événement correspondant aux symboles dans l'étendue de "0" à "9". La chaîne peut également contenir des étendues explicites et, d'une manière plus générale, des jeux de symboles explicites, désignant d'autres événements dont chacun satisfait à cette position dans le script de numérotation. Enfin le symbole point "." représente zéro ou davantage de répétitions du sélecteur d'événements (événement, étendue d'événements, ensemble d'événements en variante ou structure générique) qui le précède. Comme conséquence de la troisième règle de temporisation ci-dessus, la temporisation interévénements, tout en correspondant à un symbole de point terminal, utilise par défaut le temporisateur court.

En plus de ces symboles d'événement, la chaîne peut contenir les spécificateurs de temporisation interévénements "S" et "L" ainsi que le modificateur de durée "Z". "S" et "L" indiquent respectivement qu'il convient que la passerelle MG utilise le temporisateur court (S) ou le temporisateur long (L) pour des événements subséquents, annulant ainsi les règles décrites précédemment. Si un spécificateur de temporisation explicite est en vigueur dans une séquence

d'événements en variante, mais qu'aucun autre ne soit fourni dans les autres variantes candidates, on doit utiliser la valeur de temporisation fixée par le spécificateur explicite. Si toutes les séquences comportant des contrôles de temporisation explicites sont supprimées de l'ensemble candidat qui a été choisi, la temporisation reprend les règles par défaut fournies précédemment. S'ils sont utilisés à l'intérieur d'une notation d'étendue, les spécificateurs S et L doivent être ignorés. Enfin, le temporisateur long sera employé si des spécificateurs de temporisation sont en conflit et en vigueur dans des séquences en variante différentes.

La lettre "Z" désigne un événement de longue durée: placée en face de symbole(s) désignant un ou des événements qui satisfont à la position de chiffre donnée, elle indique que la position n'est satisfaite que si la durée de l'événement ne dépasse pas le seuil de longue durée. La valeur de ce seuil est supposée fournie dans la passerelle MG, mais elle peut, à l'instar des temporisateurs T, L et S, être supplantée par une autre valeur spécifiée dans le script de numérotation. Si le spécificateur Z n'est pas suivi d'un élément de chiffre (de 0 à 9 ou de A à K), alors la passerelle MG doit rejeter le script de numérotation en tant que procédure invalide. Lorsqu'il est utilisé en notation d'étendue, le spécificateur Z ne s'applique qu'à l'élément de chiffre immédiatement suivant. Lorsqu'il est utilisé immédiatement avant une étendue, le modificateur Z s'applique à tous les éléments de chiffre contenus dans cette étendue (ce qui nécessite que la correspondance avec les éléments de l'étendue soit à long terme).

#### **7.1.14.4 Événement d'achèvement de script de numérotation**

Un script de numérotation est actif tant que le descripteur "Events" qui l'a invoqué est actif et qu'il n'a pas été achevé. Un script de numérotation s'achève

- lorsqu'une temporisation a expiré;
- lorsqu'il y a eu correspondance avec une séquence d'événements en variante et qu'il ne peut pas y avoir de correspondance avec une autre séquence dans le script de numérotation par la détection d'un événement supplémentaire (correspondance non ambiguë);
- lorsqu'un événement a été détecté de telle manière qu'une correspondance avec une séquence complète d'événements en variante contenue dans le script de numérotation soit impossible, quels que soient les événements supplémentaires reçus.

Après l'achèvement, il doit être créé un événement d'achèvement de script de numérotation tel que défini dans le paquetage qui fournit les événements en cours de mappage dans le script. A ce moment-là, le script de numérotation est désactivé. Les événements subséquents du paquetage sont traités comme suit:

- si la propriété EventBufferControl est de durée indéfinie ("LockStep"), les événements de numérotation suivants sont traités de la même manière que les autres événements;
- si la propriété EventBufferControl est désactivée ("OFF"), que le descripteur "Events" n'ait pas été modifié et que les événements de numérotation distincts ne soient pas activés dans ce descripteur, la mise en mémoire tampon de la numérotation sera entamée. Cette mise en mémoire se poursuivra jusqu'à ce que le temps imparti, spécifié dans l'événement d'achèvement du script de numérotation initial, soit expiré ou jusqu'à ce que le descripteur "Events" actif soit remplacé.

La mémoire tampon de numérotation prendra la forme logique d'une chaîne de numérotation qui comporte les caractères numériques figurant dans le script de numérotation, éventuellement précédés de la lettre "Z". Le seuil de la durée de la tonalité employée pour identifier les événements de longue durée devra être le même que celui qui est utilisé pour le script de numérotation achevé en dernier lieu.

Le temps de la mise en mémoire tampon est ramené à zéro par défaut (pas de mise en mémoire), à moins qu'une autre valeur ne soit explicitement indiquée dans l'événement d'achèvement du script de numérotation. Si la mise en mémoire s'arrête en raison de

l'expiration du temporisateur de cette mise en mémoire, la mémoire tampon de numérotation est vidée de son contenu.

Si la mise en mémoire tampon a été arrêtée par un nouveau descripteur "Events", alors, si ce descripteur contient un nouvel événement d'achèvement du script de numérotation provenant du même paquetage que le précédent, les chiffres mis en mémoire sont traités conformément au script de numérotation comme décrit ci-après. Les chiffres mis en mémoire non utilisés par le nouveau script de numérotation sont traités comme s'ils avaient été observés après l'achèvement du script.

Si, au lieu de cela, le nouveau descripteur "Events" autorise le rapport des événements de numérotation distincts, l'ensemble entier de chiffres mis en mémoire tampon sera immédiatement traité, les événements applicables seront signalés, et la mémoire tampon sera vidée.

Finalement, si le nouveau descripteur "Events" n'active dans le paquetage concerné ni un événement d'achèvement du script de numérotation ni le rapport des événements de numérotation distincts, la mémoire tampon est vidée de son contenu et la mise en mémoire tampon s'achève.

#### **7.1.14.5 Procédures de script de numérotation**

En attendant l'achèvement, les événements subséquents doivent être traités selon les règles ci-après:

- 1) la chaîne "current dial string" (chaîne de numérotation actuelle), variable interne, est initialement vide. Le paquetage de séquences candidates d'événements en variante comprend toutes les variantes spécifiées dans le script de numérotation;
- 2) à chaque étape, si les chiffres mis en mémoire tampon sont disponibles, le plus ancien chiffre (accompagné éventuellement d'un qualificateur (Z) de longue durée) est écarté de la mémoire tampon et l'étape suivante du traitement est abordée, comme si l'événement de numérotation venait d'être observé. Sinon un temporisateur est réglé pour attendre l'événement suivant, en s'appuyant soit sur les règles de temporisation par défaut fournies ci-dessus, soit sur la temporisation explicite spécifiée dans une ou plusieurs séquences d'événements en variante. Si la temporisation expire et si un membre de l'ensemble candidat de variantes est pleinement satisfait, un achèvement de temporisation avec correspondance totale est rapporté. Si la temporisation expire et si une partie d'une variante candidate est satisfaite ou si aucune ne l'est, un achèvement de temporisation avec correspondance partielle est rapporté. Dans les deux cas, si l'événement d'achèvement du script de numérotation autorise le rapport détaillé de l'expiration, la chaîne de numérotation signalée se terminera par les lettres "L", "S" ou "T", suivant le cas;
- 3) si un événement est détecté avant l'expiration de la temporisation, il est mis en correspondance avec un symbole de chaîne de chiffres et ajouté provisoirement à la fin de la chaîne de numérotation en cours. La durée de l'événement (longue ou non) est notée si et seulement si c'est pertinent dans la position de symbole actuel (car au moins une des séquences candidates d'événement en variante contient le modificateur "Z" à cette position dans la séquence);
- 4) la chaîne de numérotation actuelle est comparée aux séquences candidates d'événements en variante. Si et seulement s'il y a correspondance avec une séquence attendant un événement de longue durée en cette position (c'est-à-dire que l'événement avait une longue durée et a satisfait à la spécification relative à cette position), toutes les séquences d'événements en variante qui ne spécifient pas un événement de longue durée en cette position sont ignorées et la chaîne de numérotation actuelle est modifiée par insertion de la lettre "Z" en face de l'événement représentant le dernier événement. Toute séquence attendant un événement de longue durée en cette position mais ne correspondant pas à l'événement observé est ignorée de l'ensemble candidat. Si, après application des règles ci-dessus, il reste dans l'ensemble

candidat des séquences d'événements en variante qui ne spécifient pas un événement de longue durée dans la position donnée, la durée observée des événements est traitée comme n'étant pas pertinente dans l'évaluation de leurs correspondances;

- 5) s'il reste exactement un candidat et qu'il présente une correspondance complète, un événement d'achèvement est produit, qui indique une correspondance non ambiguë. S'il ne reste aucun candidat, le dernier événement est retiré de la chaîne de numérotation actuelle et un événement d'achèvement est produit, qui indique une correspondance totale si un candidat provenant de l'étape précédente avait été complètement satisfait avant la détection du dernier événement; sinon, il indique une correspondance partielle. L'événement qui a été retiré de la chaîne de numérotation actuelle est ensuite signalé comme un événement distinct, mis en mémoire tampon, ou écarté suivant les règles décrites dans le précédent paragraphe. Cette déclaration est libellée comme suit:
  - a) l'événement d'achèvement du script de numérotation peut indiquer que l'événement écarté doit être signalé comme paramètre de l'événement d'achèvement. Cela se fait indépendamment du traitement ultérieur de l'événement de numérotation;
  - b) l'événement d'achèvement du script de numérotation peut indiquer que le chiffre supplémentaire doit être écarté. Dans ce cas, cela se fait immédiatement. Toute mise en mémoire tampon ou tout autre traitement ne s'applique qu'aux événements suivants;
- 6) si aucun événement d'achèvement n'est rapporté de l'étape 5), le traitement retourne à l'étape 2).

#### **7.1.14.6 Activation d'un script de numérotation**

Un script de numérotation est activé toutes les fois qu'un nouveau descripteur "Events" est appliqué à la terminaison ou qu'un descripteur "Events" imbriqué est activé et que ce descripteur "Events" contient un événement d'achèvement de script de numérotation. L'événement d'achèvement de script de numérotation contient un champ "EventDM" dans le champ des actions demandées. Chaque nouvelle activation d'un script de numérotation commence à l'étape 1 de la procédure précédente avec une chaîne de numérotation actuelle vide. Le contenu éventuel de la chaîne de numérotation actuelle provenant d'une activation précédente est perdu.

Un événement d'achèvement de script de numérotation qui ne contient pas de champ EventDM dans son champ d'actions demandées est considéré comme erroné. Lors de la réception d'un tel événement dans le descripteur "Events", une passerelle MG renverra une réponse d'erreur, notamment le code d'erreur 457 ("Paramètre manquant dans un signal ou dans un événement").

#### **7.1.14.7 Interaction de script de numérotation et traitement d'événement**

Pendant que le script de numérotation est activé, la détection est activée pour tous les événements définis dans le paquetage qui contient l'événement d'achèvement de script de numérotation considéré. Le comportement normal des événements (par exemple l'arrêt des signaux à moins que l'événement d'achèvement de numérotation n'ait le fanion KeepActive activé) continue à s'appliquer pour un tel événement détecté éventuel, sauf que:

- les événements du paquetage qui contient l'événement d'achèvement de script de numérotation considéré, autre que l'événement d'achèvement lui-même, ne sont pas rapportés individuellement et n'ont pas d'effets secondaires à moins d'être activés séparément;
- un événement qui déclenche un événement d'achèvement de correspondance partielle n'est pas reconnu et, par conséquent, n'a pas d'effets secondaires tant qu'il ne subit pas un retraitement à la suite de la reconnaissance d'un événement d'achèvement de script de numérotation. De même, les événements de numérotation mis en mémoire tampon ne sont pas reconnus et n'ont pas d'effets secondaires jusqu'à leur traitement.

### 7.1.14.8 Structures génériques

Il convient de noter que, si un paquetage contient un événement d'achèvement de script de numérotation, une spécification d'événement constituée de l'identificateur du paquetage (PackageID) avec un identificateur d'événement EventID remplacé par une structure générique activera un script de numérotation; à ces fins, la spécification d'événement doit inclure un champ EventDM conformément au § 7.1.14.6. Si ce paquetage contient également les événements de chiffre eux-mêmes, cette forme de spécification d'événement provoque le rapport des événements individuels au contrôleur MGC au fur et à mesure de leur détection.

### 7.1.14.9 Exemple

A titre d'exemple, considérons le plan de numérotation suivant:

0	Opérateur local
00	Opérateur à grande distance
xxxx	Numéro de poste supplémentaire local (commence par 1 à 7)
8xxxxxxx	Numéro local
#xxxxxxx	Prolongement hors site
*xx	Services confort
91xxxxxxxxxx	Numéro à grande distance
9011 + jusqu'à 15 chiffres	Numéro international

Si le paquetage de détection DTMF décrit dans le § E.6 est utilisé pour recueillir les chiffres composés, le plan de numérotation illustré précédemment fournit le script de numérotation suivant:

```
(0 | 00 | [1-7]xxx | 8xxxxxxx | Fxxxxxxx | Exx | 91xxxxxxxxxx | 9011x.)
```

### 7.1.15 Descripteur Statistics

Le descripteur "Statistics" donne des informations décrivant le statut et l'usage d'une terminaison pendant son (éphémère) existence ou pendant qu'elle se trouve à l'extérieur du contexte (physique) NULL. Les statistiques peuvent se situer au niveau d'une terminaison ou être associées à un descripteur de flux particulier. Par défaut, les statistiques apparaissent au niveau d'une terminaison. Si un flux n'est pas en mesure de prendre en charge une statistique, le code d'erreur 460 ("Impossible de relever une statistique sur ce flux") doit être renvoyé. Un ensemble de statistiques de base est conservé pour chaque terminaison si cela est approprié (par exemple: nombre d'octets émis et reçus). Par défaut, les statistiques particulières qui sont signalées pour une terminaison donnée sont déterminées par les paquetages réalisés par cette terminaison. Il est également possible, au moyen du descripteur, d'indiquer les statistiques qui doivent être recueillies. L'activation d'un descripteur "Statistics" supprime un tel descripteur déjà activé. Ainsi, afin de conserver des statistiques déjà activées, celles-ci doivent être incluses dans le nouveau descripteur "Statistics" et les valeurs statistiques ne doivent pas être réinitialisées. Les statistiques qui ont été supprimées d'un descripteur "Statistics" doivent conserver leurs valeurs jusqu'à ce que la terminaison soit retirée. Cependant, si la statistique particulière est réactivée par un descripteur "Statistics" subséquent, sa valeur doit être réinitialisée. Un contrôleur MGC peut réactiver toutes les statistiques d'une terminaison ou d'un flux en émettant un descripteur "Statistics" contenant une seule statistique et remplaçant les identificateurs PackageID et StatisticID par la structure générique "ALL". Afin de réactiver toutes les statistiques d'un paquetage donné dans une terminaison ou dans un flux, un descripteur "Statistics" est émis avec une seule statistique, l'identificateur PackageID étant spécifié et l'identificateur StatisticID étant remplacé par la structure générique "ALL". La réception d'un descripteur vide signifie qu'aucune statistique ne doit être recueillie pour la terminaison spécifiée.

Par défaut, si un descripteur "Statistics" vide a déjà été utilisé afin d'indiquer qu'aucune statistique n'est à recueillir, les statistiques relatives à la terminaison et au niveau de flux sont signalées lorsque la terminaison cesse d'exister ou est remise au contexte NULL en raison d'une commande Subtract. Ce comportement par défaut peut également être supplanté par l'inclusion d'un descripteur "Audit" vide dans la commande Subtract. Si un flux est supprimé conformément au § 7.1.6, les statistiques de niveau de flux ne sont pas signalées par défaut. Afin de recueillir les statistiques, un examen du descripteur "Statistics" contenu dans le flux doit toujours être effectué avant que ce flux soit supprimé. Des statistiques peuvent aussi être renvoyées à partir de la commande AuditValue ou à partir de toute commande Add/Move/Modify utilisant le descripteur "Audit".

Les statistiques sont cumulatives. Le fait de les signaler ne les réinitialise pas. La valeur d'une statistique au niveau d'une terminaison est le résultat d'une fonction supérieure significative (comme une somme ou une moyenne) appliquée aux valeurs comme si elle avait été placée dans tous les flux de la terminaison. Une telle fonction supérieure dépend du type particulier de statistique. Sauf spécification contraire dans le packaging qui définit une statistique particulière, le comportement par défaut est une somme des valeurs. Les statistiques sont réinitialisées lorsqu'une terminaison cesse d'exister ou est remise au contexte NULL en raison d'une commande Subtract.

#### **7.1.16 Descripteur Packages (paquetages)**

Ce descripteur n'est utilisé qu'avec la commande AuditValue. Il renvoie une liste de paquetages réalisés par la terminaison.

#### **7.1.17 Descripteur ObservedEvents (éléments observés)**

Ce descripteur est fourni avec la commande Notify pour informer le contrôleur MGC de l'événement ou des événements détectés. Utilisé avec la commande AuditValue, le descripteur "ObservedEvents" renvoie les événements contenus dans la mémoire d'événements qui n'ont pas été notifiés. Le descripteur "ObservedEvents" contient l'identificateur RequestID du descripteur "Events" qui a déclenché la notification, l'événement ou les événements détectés, en option le ou les temps de détection et tous paramètres de l'événement observé. Les temps de détection sont consignés avec une précision de centièmes de seconde.

#### **7.1.18 Descripteur Topology**

Ce descripteur est utilisé pour spécifier les sens des flux entre les terminaisons d'un contexte. Contrairement aux descripteurs des paragraphes précédents, le descripteur "Topology" s'applique à un contexte au lieu d'une terminaison. La topologie par défaut d'un contexte est telle que l'émission d'une terminaison soit reçue par toutes les autres terminaisons. L'implémentation du descripteur "Topology" est facultative. Une passerelle MG qui ne prend pas en charge les descripteurs Topology, mais reçoit une commande qui en contient, renvoie le code d'erreur 444 ("Descripteur non pris en charge ou inconnu") et inclut éventuellement dans le texte du descripteur "Error" une chaîne contenant le nom du descripteur non pris en charge ("Topology").

Le descripteur "Topology" apparaît avant les commandes dans un paramètre d'action. Il est possible qu'une action ne contienne qu'un seul descripteur "Topology", à condition que le contexte auquel l'action s'applique existe déjà.

Un descripteur "Topology" se compose d'une suite de terminaisons associées de la forme (*T1*, *T2*, *association*[,*StreamID*]). *T1* et *T2* spécifient des terminaisons dans le contexte, éventuellement au moyen de la structure générique ALL ou CHOOSE. Si le champ facultatif StreamID est employé, l'association ne s'applique qu'au flux particulier entre *T1* et *T2* étiqueté par StreamID. Si le champ StreamID est omis, la topologie s'applique à tous les flux contenus dans la terminaison. L'élément *association* spécifie comme suit la façon dont les médias s'écoulent entre ces deux terminaisons:

- l'expression (*T1*, *T2*, isolate) signifie que les terminaisons correspondant à *T2* ne reçoivent pas de média des terminaisons correspondant à *T1* et inversement;

- l'expression (*T1*, *T2*, oneway) signifie que les terminaisons correspondant à *T2* reçoivent des médias des terminaisons correspondant à *T1* mais pas inversement. Dans ce cas, l'utilisation de la structure générique ALL, de façon qu'il y ait des terminaisons correspondant soit à *T1* soit à *T2* *mais pas aux deux*, est autorisée;
- l'expression (*T1*, *T2*, onewayexternal) signifie que les terminaisons correspondant à *T2* reçoivent des médias émis vers l'extérieur par des terminaisons correspondant à *T1* mais pas inversement. Dans ce cas, l'utilisation de la structure générique ALL pour *T1* n'est pas autorisée;
- l'expression (*T1*, *T2*, onewayboth) signifie que les terminaisons correspondant à *T2* reçoivent des médias émis et reçus extérieurement par des terminaisons correspondant à *T1* mais pas inversement. Dans ce cas, l'utilisation de la structure générique ALL pour *T1* et/ou *T2* n'est pas autorisée;
- l'expression (*T1*, *T2*, bothway) signifie que les terminaisons correspondant à *T2* reçoivent des médias des terminaisons correspondant à *T1* et inversement. Dans ce cas, il est permis d'utiliser des structures génériques de façon qu'il ait des terminaisons correspondant à la fois à *T1* et à *T2*. Cependant, dans ce dernier cas, aucun bouclage n'est introduit.

La structure générique CHOOSE peut également être utilisée dans *T1* et *T2*, aux conditions suivantes:

- l'action (voir le § 8) dont le descripteur "Topology" fait partie contient une commande Add dans laquelle une structure générique CHOOSE est utilisée;
- si une structure générique CHOOSE apparaît dans *T1* ou *T2*, un nom partiel ne doit pas être spécifié.

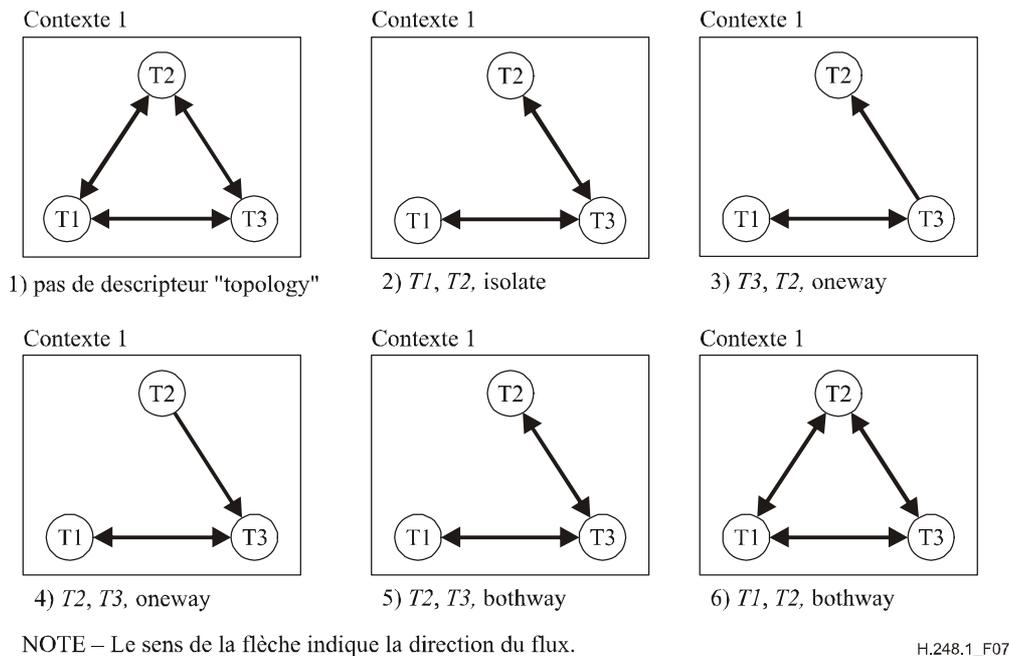
La structure générique CHOOSE contenue dans un descripteur "Topology" correspond à l'identificateur TerminationID que la passerelle MG attribue à la première commande Add qui utilise une structure générique CHOOSE dans la même action. Une terminaison existante qui correspond à *T1* ou à *T2* dans le contexte auquel une terminaison a été ajoutée est connectée à la terminaison nouvellement ajoutée comme spécifié par le descripteur "Topology". Si une terminaison n'est pas mentionnée dans un descripteur "Topology", toute topologie qui lui est associée reste inchangée. Si, toutefois, une nouvelle terminaison est ajoutée dans un contexte, son association avec les autres terminaisons dans le contexte est par défaut bilatérale, à moins qu'un descripteur "Topology" ne soit donné afin de modifier cela (par exemple, si *T3* est ajouté à un contexte avec *T1* et *T2* et la topologie (*T3*, *T1* oneway), sa connexion à *T2* sera bilatérale).

Si la topologie est appliquée à un flux particulier (*T1*, *T2*, association, StreamID), la topologie des autres flux entre les terminaisons reste inchangée.

Un descripteur "Topology" ne doit pas inclure de combinaison d'associations entre deux terminaisons (*Ti*, *Tj*) avec et sans champ facultatif StreamID, afin d'éviter un comportement non défini. Par exemple, (*T1*, *T2*, bothway) et (*T1*, *T2*, isolate, S1) ne doivent pas figurer dans le même descripteur. A la réception d'un tel descripteur "Topology", une passerelle MG répondra par un message d'erreur, notamment le code d'erreur 421 ("Action inconnue ou combinaison d'actions non autorisée").

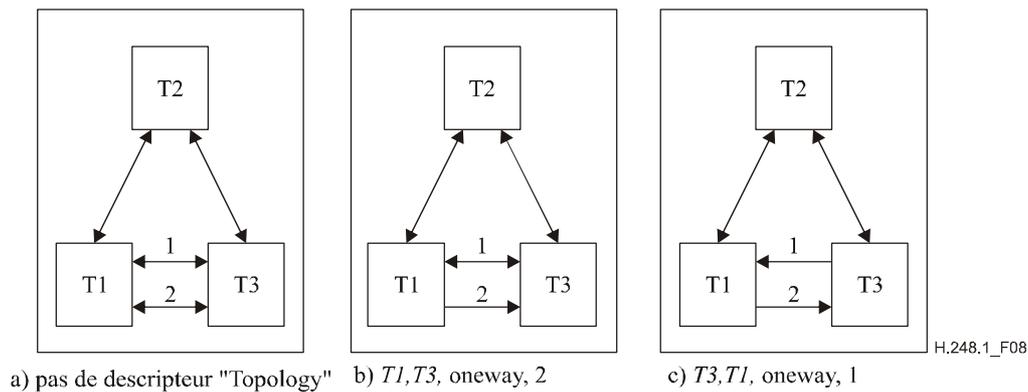
Une connexion unilatérale doit toujours être implémentée de façon que les autres terminaisons contenues dans le contexte ne soient pas informées du changement de topologie.

La Figure 7, le tableau qui lui fait suite et la Figure 8 ci-dessous montrent quelques exemples des conséquences de l'inclusion de descripteurs "Topology" dans des actions. Dans ces exemples, il est implicitement admis que les descripteurs "Topology" sont appliqués en séquence. Les Figures 9 et 10 sont des exemples autonomes montrant les effets spécifiques des réglages topologiques "onewayexternal" et "onewayboth".

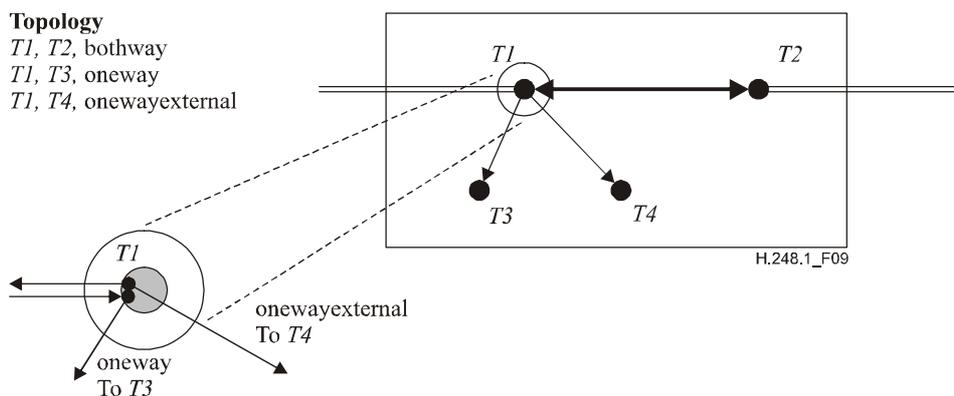


**Figure 7/H.248.1 – Exemples de topologie**

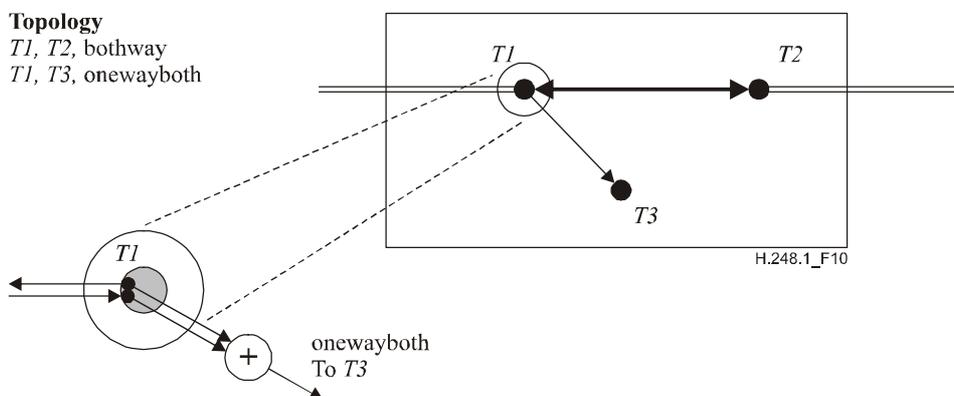
Topologie	Description
1	Pas de descripteur "Topology" Si aucun descripteur "Topology" n'est inclus, toutes les terminaisons ont une connexion bilatérale avec toutes les autres terminaisons.
2	<i>T1, T2</i> Isolate Cette topologie supprime la connexion entre <i>T1</i> et <i>T2</i> . <i>T3</i> possède donc une connexion bilatérale avec <i>T1</i> comme avec <i>T2</i> . <i>T1</i> et <i>T2</i> ont une connexion bilatérale avec <i>T3</i> .
3	<i>T3, T2</i> oneway Une connexion unilatérale de <i>T3</i> à <i>T2</i> (c'est-à-dire que <i>T2</i> reçoit le flux média de <i>T3</i> ). Une connexion bilatérale existe entre <i>T1</i> et <i>T3</i> .
4	<i>T2, T3</i> oneway Une connexion unilatérale existe de <i>T2</i> à <i>T3</i> . <i>T1</i> et <i>T3</i> restent en connexion bilatérale.
5	<i>T2, T3</i> bothway <i>T2</i> est en connexion bilatérale avec <i>T3</i> , ce qui donne le même résultat qu'au chiffre 2 "Topology".
6	<i>T1, T2</i> bothway ( <i>T2, T3</i> bothway et <i>T1, T3</i> bothway peuvent être implicites ou explicites). Chacune des terminaisons est en connexion bilatérale avec toutes les autres terminaisons.



**Figure 8/H.248.1 – Exemple de topologie au niveau du flux**



**Figure 9/H.248.1 – Topologie "onewayexternal" contre topologie "oneway"**



**Figure 10/H.248.1 – Fonctionnement de la topologie "onewayboth"**

### 7.1.19 Descripteur ContextAttribute (attribut de contexte)

Un descripteur "ContextAttribute" sert à spécifier des propriétés (définies dans des paquets) qui s'appliquent au contexte dans son ensemble. Ce descripteur est appliqué à un contexte plutôt qu'à une terminaison. Les propriétés des terminaisons ne sont pas valides dans le descripteur "ContextAttribute". Celui-ci apparaît avant les commandes dans une action. Il est possible d'avoir une action contenant seulement un descripteur "ContextAttribute", à condition que le contexte indiqué existe déjà. Les valeurs de contexte des propriétés peuvent être sous-spécifiées comme dans le § 7.1.1.

Un nouveau réglage du descripteur "ContextAttribute" remplace complètement le réglage précédent de ce descripteur dans la passerelle MG. Ainsi, afin de conserver les informations issues du réglage précédent, le contrôleur MGC doit toujours inclure ces informations dans le nouveau réglage. Si le contrôleur MGC souhaite supprimer certaines informations du descripteur existant, il n'a qu'à réexpédier ce descripteur après en avoir retiré les informations inutiles. L'inclusion d'un descripteur ContextAudit ou ContextAttribute dans une action qui ne contient qu'une commande AuditValue ou AuditCapabilities ne constitue pas un nouveau réglage.

Le paramètre ContextIDList sert à fournir une liste compacte d'identificateurs de contexte (ContextID). Afin de demander l'utilisation du paramètre ContextIDList dans une réponse, il convient d'inclure le paramètre ContextIDList dans la requête correspondante.

### 7.1.20 Descripteur Error

Si un répondeur rencontre une erreur au cours du traitement d'une demande de transaction, il doit inclure un descripteur "Error" dans sa réponse. Une demande de notification peut aussi contenir un descripteur "Error".

Un descripteur "Error" consiste en un code d'erreur enregistré par l'autorité IANA, éventuellement accompagné d'un texte d'erreur. La Rec. UIT-T H.248.8 contient une liste des codes et des descriptions d'erreur valables.

Un descripteur "Error" doit être spécifié au "niveau le plus profond", approprié sur le plan sémantique à l'erreur décrite et possible, étant donné les problèmes d'examen de la demande initiale. Un descripteur "Error" peut se référer à une structure syntaxique ailleurs qu'à l'endroit où il apparaît. Par exemple, le descripteur "Error" 422 ("Erreur de syntaxe dans l'action"), pourrait apparaître dans une commande même lorsqu'il se réfère à une structure plus large – l'action – et non à la commande particulière dans laquelle il apparaît.

## 7.2 Interface de programmation d'application pour les commandes

On trouvera ci-après une interface de programmation d'application (API, *application programming interface*) décrivant les commandes du protocole. Cette interface API vise à illustrer les commandes et leurs paramètres. Elle n'est pas destinée à spécifier une implémentation (par exemple au moyen d'appels de fonction de blocage). Elle décrit les paramètres d'entrée entre parenthèses après le nom de la commande et les valeurs de retour en face de cette commande. Cette interface ne vise que la description. La syntaxe et le codage proprement dits des commandes sont spécifiés dans des paragraphes ultérieurs. L'ordre des paramètres relatifs aux commandes n'est pas fixé. Des descripteurs peuvent servir de paramètres à des commandes dans un ordre quelconque. Les descripteurs doivent être traités dans l'ordre dans lequel ils apparaissent.

Toute réponse à une commande peut contenir un descripteur "Error"; l'interface API ne montre pas cela explicitement.

Tous les paramètres mis entre crochets ([...]) sont considérés comme étant facultatifs.

### 7.2.1 Commande Add

La commande Add ajoute une terminaison à un contexte.

TerminationIDList (liste des identificateurs de terminaison)

[,MediaDescriptor]

[,ModemDescriptor] (\*)

[,MuxDescriptor]

[,EventsDescriptor]

[,SignalsDescriptor]

[,DigitMapDescriptor]

```

[,ObservedEventsDescriptor]
[,EventBufferDescriptor]
[,StatisticsDescriptor]
[,PackagesDescriptor]
    Add( TerminationIDList
        [, MediaDescriptor]
        [, ModemDescriptor] (*)
        [, MuxDescriptor]
        [, EventsDescriptor]
        [, EventBufferDescriptor]
        [, SignalsDescriptor]
        [, DigitMapDescriptor]
        [, AuditDescriptor]
        [, StatisticsDescriptor]
    )

```

(\*) Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).

La liste d'identificateurs de terminaison (TerminationIDList) spécifie une ou plusieurs terminaisons qui doivent être ajoutées au contexte. La ou les terminaisons sont créées ou extraites du contexte NULL. Si une structure générique CHOOSE est employée dans l'identificateur de terminaison, l'identificateur de terminaison choisi sera retourné. Des structures génériques ALL peuvent être utilisées dans une commande Add, mais un tel usage sera rare. Si la structure générique concorde avec plusieurs identificateurs de terminaison, toutes les correspondances possibles seront essayées et les résultats seront signalés à chaque fois. L'ordre des essais en cas de multiples correspondances avec des identificateurs de terminaison n'est pas spécifié.

Le descripteur facultatif "Media" décrit tous les flux médias.

Le descripteur facultatif "Multiplex" spécifie, le cas échéant, un multiplexeur. Par commodité, si un descripteur "Multiplex" est présent dans une commande Add et si ce descripteur énumère des terminaisons qui ne sont pas actuellement dans le contexte, ces terminaisons sont ajoutées au contexte comme si des commandes Add individuelles, énumérant ces terminaisons, avaient été invoquées. Si une erreur survient sur une telle commande Add implicite, une erreur 471 ("Commande Add implicite pour échec de multiplex") doit être renvoyée et un traitement plus poussé de la commande doit s'arrêter.

Le paramètre descripteur "Events" est facultatif. S'il est présent, il fournit la liste des événements qui devraient être détectés dans la terminaison.

Le paramètre descripteur "EventBuffer" est facultatif. S'il est présent, il fournit la liste des événements que la passerelle MG est priée de détecter et de mettre en mémoire tampon lorsque la commande EventBufferControl est égale à LockStep.

Le paramètre descripteur "Signals" est facultatif. S'il est présent, il fournit la liste des événements qui devraient être appliqués à la terminaison.

Le paramètre descripteur "DigitMap" est facultatif. S'il est présent, il donne une définition de script de numérotation qui peut être utilisée dans un descripteur "Events".

Le paramètre descripteur "Audit" est facultatif. S'il est présent, la commande renverra les valeurs d'éventuels descripteurs/propriétés/signaux/événements/statistiques spécifiés dans ce paramètre.

Le descripteur "Statistics" est facultatif. S'il est présent, il fournit les statistiques dont le contrôleur MGC souhaite la collecte pour la terminaison ou pour le flux. Chaque fois que ce descripteur est activé, la valeur statistique est réinitialisée.

Tous les descripteurs modifiables peuvent être renvoyés par la passerelle MG si un paramètre a été sous-spécifié ou surspécifié. Les descripteurs "ObservedEvents", "Statistics", Packages ainsi que "EventBuffer" ne sont retournés que s'ils sont demandés dans le descripteur "Audit".

La commande Add ne doit pas être utilisée sur une terminaison avec un état serviceState de valeur "OutOfService" (hors service).

### 7.2.2 Commande Modify

La commande Modify modifie les propriétés d'une terminaison.

TerminationIDList (liste des identificateurs de terminaison)

[,MediaDescriptor]

[,ModemDescriptor] (\*)

[,MuxDescriptor]

[,EventsDescriptor]

[,SignalsDescriptor]

[,DigitMapDescriptor]

[,ObservedEventsDescriptor]

[,EventBufferDescriptor]

[,StatisticsDescriptor]

[,PackagesDescriptor]

```
Modify( TerminationIDList
    [, MediaDescriptor]
    [, ModemDescriptor] (*)
    [, MuxDescriptor]
    [, EventsDescriptor]
    [, EventBufferDescriptor]
    [, SignalsDescriptor]
    [, DigitMapDescriptor]
    [, AuditDescriptor]
    [, StatisticsDescriptor]
)
```

(\*) Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).

La liste "TerminationIDList" spécifie les terminaisons à modifier. L'identificateur de terminaison peut être spécifique si une seule terminaison doit être modifiée dans le contexte. L'utilisation de structures génériques dans l'identificateur de terminaison peut convenir pour certaines opérations. Si la structure générique concorde avec plusieurs identificateurs de terminaison, toutes les correspondances possibles seront essayées et les résultats seront signalés à chaque fois. L'ordre des essais en cas de multiples correspondances avec des identificateurs de terminaison n'est pas spécifié. L'identificateur de terminaison CHOOSE est une erreur car la commande Modify ne peut être utilisée qu'avec des terminaisons existantes.

Par souci de commodité, lorsqu'un descripteur "Multiplex" est présent dans une commande Modify, alors:

- si le nouveau descripteur "Multiplex" énumère des terminaisons qui ne sont pas effectivement dans le contexte, ces terminaisons doivent y être ajoutées comme si des commandes Add distinctes énumérant les terminaisons avaient été invoquées;
- si des terminaisons énumérées précédemment dans le descripteur "Multiplex" ne sont plus présentes dans le nouveau descripteur "Multiplex", elles seront soustraites du contexte comme si des commandes Subtract distinctes énumérant les terminaisons avaient été invoquées.

Les autres paramètres de la commande Modify sont les mêmes que ceux de la commande Add. Les valeurs de retour possibles sont les mêmes que celles de la commande Add.

### 7.2.3 Commande Subtract

La commande Subtract déconnecte une terminaison de son contexte et renvoie des statistiques sur la participation de cette terminaison dans le contexte.

TerminationIDList (liste des identificateurs de terminaison)

[,MediaDescriptor]

[,ModemDescriptor] (\*)

[,MuxDescriptor]

[,EventsDescriptor]

[,SignalsDescriptor]

[,DigitMapDescriptor]

[,ObservedEventsDescriptor]

[,EventBufferDescriptor]

[,StatisticsDescriptor]

[,PackagesDescriptor]

Subtract(TerminationIDList

[, AuditDescriptor]

)

(\*) Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).

La liste d'identificateurs de terminaison contenue dans les paramètres d'entrée représente la ou les terminaisons qui sont en cours de soustraction. L'identificateur de terminaison peut être spécifique ou être une structure générique indiquant que la totalité (ou un paquetage d'associations) des terminaisons contenues dans le contexte de la commande Subtract doit être soustrait(e). Si la structure générique concorde avec plusieurs identificateurs de terminaison, toutes les correspondances possibles seront essayées et les résultats seront signalés à chaque fois. L'ordre des essais en cas de multiples correspondances avec des identificateurs de terminaison n'est pas spécifié.

L'utilisation de la structure CHOOSE dans l'identificateur de terminaison est une erreur, car la commande Subtract ne peut être utilisée que sur des terminaisons existantes.

La structure ALL peut être utilisée en tant qu'identificateur de contexte ainsi que comme identificateur de terminaison dans une commande Subtract, ce qui a pour effet de supprimer tous les contextes, de supprimer toutes les terminaisons éphémères et de renvoyer toutes les terminaisons physiques au contexte Null. La soustraction d'une terminaison du contexte Null n'est pas autorisée.

Par souci de commodité, si une terminaison de multiplexage est l'objet d'une commande Subtract, alors les terminaisons supports énumérées dans son descripteur "Multiplex" sont soustraites du contexte comme si des commandes Subtract distinctes énumérant les terminaisons avaient été invoquées.

Par défaut et sauf priorité contraire, le descripteur "Statistics", au niveau de la terminaison comme à celui du flux, est renvoyé pour signaler des informations collectées au sujet de la terminaison ou des terminaisons spécifiées dans la commande. Les informations ainsi signalées s'appliquent à l'existence de la ou des terminaisons dans le contexte dont celles-ci sont soustraites.

Le descripteur "Audit" est facultatif. S'il est présent, la commande ne renverra que les éléments spécifiés dans ce descripteur, qui peut être vide. Si le descripteur "Audit" est omis, le descripteur "Statistics" est renvoyé par défaut. Les valeurs de retour possibles sont les mêmes que celles de la commande Add.

Lorsqu'une terminaison fournie est soustraite d'un contexte, ses valeurs de descripteur doivent revenir:

- à la valeur par défaut, si celle-ci est spécifiée pour le descripteur et n'est pas annulée par la fourniture;
- à la valeur fournie, autrement.

#### 7.2.4 Commande Move

La commande Move déplace une terminaison vers un autre contexte à partir de son contexte actuel, au cours d'une seule opération atomique. La commande Move est la seule commande qui se rapporte à une terminaison dans un contexte différent de celui auquel la commande s'applique. La commande Move ne doit pas être utilisée pour déplacer des terminaisons à destination ou en provenance du contexte NULL.

TerminationIDList (liste des identificateurs de terminaison)

[,MediaDescriptor]

[,ModemDescriptor] (\*)

[,MuxDescriptor]

[,EventsDescriptor]

[,SignalsDescriptor]

[,DigitMapDescriptor]

[,ObservedEventsDescriptor]

[,EventBufferDescriptor]

[,StatisticsDescriptor]

[,PackagesDescriptor]

Move( TerminationIDList

[, MediaDescriptor]

[, ModemDescriptor] (\*)

[, MuxDescriptor]

[, EventsDescriptor]

[, EventBufferDescriptor]

[, SignalsDescriptor]

[, DigitMapDescriptor]

[, AuditDescriptor]

[, StatisticsDescriptor]  
)

(\*) Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).

La liste d'identificateurs de terminaison spécifie la ou les terminaisons à déplacer. Les identificateurs de terminaison peuvent être remplacés par une structure générique mais la structure CHOOSE ne doit pas être utilisée. Si la structure générique concorde avec plusieurs identificateurs de terminaison, toutes les correspondances possibles seront essayées et les résultats seront signalés à chaque fois. L'ordre des essais en cas de multiples correspondances avec des identificateurs de terminaison n'est pas spécifié. Le contexte vers lequel la terminaison est déplacée est indiqué par l'identificateur de contexte (ContextID) cible contenu dans l'action. Si la dernière terminaison restante est déplacée du contexte, le contexte est supprimé.

La commande Move n'affecte pas les propriétés de la terminaison sur laquelle elle opère, sauf les propriétés explicitement modifiées par les descripteurs inclus dans la commande Move. Le descripteur "Audit" avec un descripteur "Statistics" renverra par exemple des statistiques au sujet de la terminaison juste avant le déplacement de celle-ci. Les descripteurs possibles retournés à partir de la commande Move sont les mêmes que pour la commande Add.

Par souci de commodité, si une terminaison de multiplexage est l'objet d'une commande Move, alors les terminaisons supports énumérées dans son descripteur "Multiplex" sont aussi déplacées comme si des commandes Move distinctes énumérant les terminaisons avaient été invoquées.

La commande Move ne doit pas être utilisée sur une terminaison avec un état serviceState de "OutOfService" (hors service).

### 7.2.5 Commande AuditValue

La commande AuditValue renvoie les valeurs actuelles des propriétés, des événements, des signaux et des statistiques associés aux terminaisons. Elle peut demander le contenu d'un descripteur ou d'une propriété, d'un événement, d'un signal ou d'une statistique. Elle peut être utile afin de conserver le synchronisme des terminaisons entre le contrôleur MGC et la passerelle MG. Une commande AuditValue peut demander que les valeurs retournées soient filtrées sur la base de critères de sélection spécifiés.

TerminationIDList (liste des identificateurs de terminaison)

[,MediaDescriptor]

[,ModemDescriptor] (\*)

[,MuxDescriptor]

[,EventsDescriptor]

[,SignalsDescriptor]

[,DigitMapDescriptor]

[,ObservedEventsDescriptor]

[,EventBufferDescriptor]

[,StatisticsDescriptor]

[,PackagesDescriptor]

AuditValue(TerminationIDList,

AuditDescriptor

)

(\*) Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).

La liste d'identificateurs de terminaison spécifie une séquence d'identificateurs de terminaison qui peut être spécifique ou générique. Si la structure générique concorde avec plusieurs identificateurs de terminaison, toutes les correspondances possibles seront essayées et les résultats seront signalés à chaque fois. L'ordre des essais en cas de multiples correspondances avec des identificateurs de terminaison n'est pas spécifié. Si une réponse remplacée par une structure générique est demandée, un seul renvoi de commande est produit, qui contient la réunion des valeurs de toutes les terminaisons qui correspondent à la structure générique ou à la liste. Cette convention peut réduire le volume de données nécessaires pour examiner un groupe de terminaisons. L'utilisation de la structure CHOOSE est une erreur.

Les descripteurs ou les propriétés, les signaux, les événements et les statistiques peuvent être examinés individuellement. Les valeurs renvoyées peuvent être filtrées sur la base de critères de sélection spécifiés.

- On peut demander l'examen d'un descripteur en indiquant dans le descripteur "Audit" l'identité du descripteur souhaité, sans autre information.
- Afin d'examiner une propriété distincte dans le descripteur "Media", il faut inclure l'identificateur de flux (facultatif), l'identificateur de groupe (facultatif) et l'identificateur de propriété pertinents. La valeur courante de la propriété est renvoyée. L'identificateur de groupe est employé dans le cas où le fanion "Reserve-Group" du descripteur "LocalControl" est employé. L'identificateur de groupe 1 correspond au premier groupe (description de session) réservé, l'identificateur de groupe 2 correspond au groupe suivant, etc.
- Afin d'examiner un signal, on fournit l'identificateur de la liste de signaux et/ou l'identificateur du signal pertinents. Les identificateurs StreamID et/ou RequestID sont facultatifs. Les valeurs de tous les paramètres du signal ne sont renvoyées que lorsque le signal est activé, notamment l'indication "KeepActive", le type de signal, la durée, l'indication d'achèvement du signal et les propriétés définies au niveau du paquetage.
- Afin d'examiner un événement, on fournit l'identificateur de flux (facultatif), l'identificateur de l'événement et l'identificateur de la demande (facultatif) pertinents. Les valeurs de tous les paramètres de l'événement sont renvoyées, notamment les actions concernant les événements et les paramètres définis au niveau du paquetage.
- Afin d'examiner une statistique au niveau d'une terminaison, on fournit l'identité de cette statistique. Afin d'examiner une statistique au niveau du flux, l'on inclut dans le descripteur "Media" l'identificateur de flux (facultatif), l'identificateur de groupe (facultatif) et l'identificateur de statistique pertinents. La valeur courante de la statistique est renvoyée. Cette statistique n'est pas réinitialisée.
- Afin d'examiner un paquetage, on fournit l'identité et la version du paquetage. Tous les signaux, propriétés, événements et statistiques définis dans ce paquetage particulier sont renvoyés avec leur valeur courante.
- Afin de filtrer les valeurs de retour d'examen, des critères de sélection sont inclus dans la demande d'examen. Il est possible d'inclure de multiples critères de sélection, auquel cas l'on peut également inclure un opérateur logique ET ou OU afin d'indiquer comment les critères de sélection doivent être interprétés. Si aucun opérateur logique n'est inclus, l'on suppose qu'il s'agit d'une opération logique ET.

Il est possible d'examiner plusieurs éléments distincts au moyen d'une même demande.

Si un examen de descripteurs est demandé, les descripteurs appropriés, assortis des valeurs actuelles pour la terminaison, sont renvoyés par la commande AuditValue. Les valeurs apparaissant dans de

multiples instances d'un descripteur sont définies de façon à être des variantes prises en charge, chaque paramètre d'un descripteur étant considéré comme indépendant.

Le descripteur "ObservedEvents" renvoie une liste d'événements contenus dans la mémoire d'événements. Si le descripteur "ObservedEvents" est examiné pendant qu'un script de numérotation est actif, le descripteur "ObservedEvents" renvoyé comprend également un événement d'achèvement de script de numérotation qui indique la chaîne de numérotation courante sans donner de méthode de terminaison.

La mémoire EventBuffer renvoie l'ensemble des événements et les valeurs de paramètre associées qui sont actuellement activées dans le descripteur "EventBuffer". Le descripteur "Packages" renvoie une liste de paquetages réalisés par la terminaison. Le descripteur "DigitMap" renvoie le nom ou la valeur du script de numérotation actuel pour la terminaison. Le descripteur "DigitMap" demandé dans une commande AuditValue avec l'identificateur de terminaison ALL renvoie tous les scripts de numérotation contenus dans la passerelle. Le descripteur "Statistics" renvoie les valeurs actuelles de toutes les statistiques enregistrées au niveau de la terminaison. Un examen du descripteur "Media" renverra d'éventuelles statistiques au niveau du flux. La spécification d'un descripteur "Audit" vide se traduit par le renvoi du seul identificateur de terminaison. Cela peut être utile afin d'obtenir une liste d'identificateurs de terminaison, à utiliser avec une structure générique. Les Annexes A et B contiennent une syntaxe spéciale permettant de présenter une telle liste sous forme condensée, de sorte que l'étiquette de commande AuditValue n'a pas besoin d'être répétée pour chaque identificateur de terminaison.

Les résultats (à savoir "spécifique", "NULL" ou "remplacé par une structure générique") de la commande AuditValue dépendent du contexte. (Il est à noter que l'identificateur de contexte "ALL" n'inclut pas le contexte NULL.) L'identificateur de terminaison peut être spécifique ou remplacé par une structure générique.

Les exemples suivants montrent les renvois qui sont effectués lorsque le contexte ou la terminaison est remplacé par une structure générique et qu'une réponse générique a été spécifiée.

Supposons que la passerelle ait 4 terminaisons: t1/1, t1/2, t2/1 et t2/2. Supposons que les terminaisons t1/\* aient implémenté les paquetages aaa et bbb tandis que les terminaisons t2/\* aient implémenté les paquetages ccc et ddd. Supposons que le contexte 1 contienne t1/1 et t2/1 et que le contexte 2 contienne t1/2 et t2/2.

Commande:

```
Context=1 {AuditValue=t1/1 {Audit {Packages}}}
```

Renvoi:

```
Context=1 {AuditValue=t1/1 {Packages {aaa,bbb}}}
```

Commande:

```
Context=* {AuditValue=t2/* {Audit {Packages}}}
```

Renvoi:

```
Context=1 {AuditValue=t2/1 {Packages {ccc,ddd}}},  
Context=2 {AuditValue=t2/2 {Packages {ccc,ddd}}}
```

Commande:

```
Context=* {W-AuditValue=t1/* {Audit {Packages}}}
```

Renvoi:

```
Context=1 {AuditValue=t1/1 {Audit {Packages}}}
```

NOTE – Une réponse générique peut aussi être employée pour d'autres commandes telles que Substract.

Dans le cas d'un contexte remplacé par une structure générique avec identificateur racine ("Root"), le contrôleur MGC peut indiquer qu'il souhaite une liste compacte d'identificateurs de contexte, plutôt que d'avoir chaque contexte subdivisé en réponses d'action distinctes.

Supposons que la passerelle ait 4 contextes: 1, 2, 3, 4, contenant chacun deux terminaisons (t1-t8). Les exemples ci-après indiquent comment la passerelle MG répondra à la commande d'examen:

La commande:

```
Context=* {AuditValue=Root {Audit {}}}
```

renvoie:

```
Context=1 {AuditValue=Context{*}}, Context=2 {AuditValue=Context{*}},  
Context=3 {AuditValue=Context{*}}, Context=4 {AuditValue=Context{*}}
```

Ou

```
Context=1 {AuditValue=t1 {},AuditValue=t2 {}},Context=2 {AuditValue=t3 {},AuditValue=  
t4 {}},Context=3 {AuditValue=t5 {},AuditValue=t6 {}},Context=4 {AuditValue=t7 {},Audit  
Value=t8 {}}
```

La commande:

```
Context=* {ContextAttr {ContextList={*}},AuditValue=Root {Audit {}}}
```

renvoie:

```
Context=* {ContextAttr {ContextList={1,2,3,4}},AuditValue=Root {}}
```

Le tableau ci-dessous décrit d'autres informations qui peuvent être obtenues par la commande AuditValue:

Identificateur de contexte	Identificateur de terminaison	Informations obtenues
Spécifique	Structure générique	Examen des terminaisons concordantes dans un contexte
Spécifique	Spécifique	Examen d'une terminaison unique dans un contexte
NULL	Racine	Examen de l'état et des événements d'une passerelle MG
NULL	Structure générique	Examen de toutes les terminaisons concordantes contenues dans le contexte NULL
NULL	Spécifique	Examen d'une terminaison unique hors de tout contexte
ALL	Structure générique	Examen de toutes les terminaisons concordantes ne se trouvant pas dans le contexte NULL et examen des contextes auxquels elles sont associées
ALL	Racine	Liste de tous les identificateurs de contexte (cette liste peut être renvoyée soit au moyen de réponses d'action multiples, chacune contenant un identificateur de contexte de la liste, soit au moyen du paramètre ContextIDList. La méthode de réponse est déterminée par la présence du paramètre ContextIDList dans la requête)
ALL	Spécifique	Identificateur de contexte (autre que NULL) dans lequel la terminaison existe actuellement

### 7.2.6 AuditCapabilities

La commande "AuditCapabilities" renvoie les valeurs possibles de propriétés, d'événements, de signaux et de statistiques associés à des terminaisons. Elle peut être demandée pour les contenus d'un descripteur ou d'une propriété, d'un événement, d'un signal ou d'une statistique

TerminationIDList (liste des identificateurs de terminaison)

[,MediaDescriptor]

[,ModemDescriptor](\*)

[,MuxDescriptor]

[,EventsDescriptor]

[,SignalsDescriptor]

[,ObservedEventsDescriptor]

[,EventBufferDescriptor]

[,StatisticsDescriptor]

AuditCapabilities(TerminationIDList,  
AuditDescriptor)

(\*) Le descripteur "Modem" a été indiqué comme étant à éviter dans la Rec. UIT-T H.248.1 version 2 (05/2002).

Les descripteurs ou les propriétés, les signaux, les événements et les statistiques distincts peuvent être analysés.

- On peut demander l'examen d'un descripteur entier en indiquant dans le descripteur "Audit" le descripteur souhaité, sans autre information.
- Afin d'examiner une propriété distincte dans le descripteur "Media", il faut inclure l'identificateur de flux StreamID (facultatif) et l'identificateur de la propriété PropertyID. Une liste des valeurs possibles de la propriété est renvoyée.
- Afin d'examiner un signal, on fournit l'identificateur de la liste de signaux (SignalListID) et/ou l'identificateur du signal (SignalID) pertinent. Les identificateurs StreamID et/ou RequestID sont facultatifs. Une liste des valeurs possibles associées à chacun des paramètres du signal est renvoyée (y compris les propriétés définies par le paquetage). L'indication "KeepActive", le type du signal, la durée et l'indication d'achèvement du signal ne sont pas renvoyés.
- Afin d'examiner un événement, on fournit l'identificateur de flux StreamID (facultatif), l'identificateur de l'événement EventID, et l'identificateur de la demande RequestID (facultatif) pertinents. Une liste des valeurs possibles associées à chacun des paramètres de l'événement est renvoyée (y compris les actions concernant les événements et les paramètres définis au niveau du paquetage).
- Afin d'examiner une statistique au niveau d'une terminaison, on fournit l'identité de cette statistique. Afin d'examiner une statistique au niveau du flux, l'on inclut dans le descripteur "Media" les identificateurs StreamID (facultatif), GroupID (facultatif) et StatisticID pertinents. Les valeurs possibles de la statistique sont renvoyées. Cette statistique n'est pas réinitialisée.

Si un examen de descripteurs est demandé, les descripteurs appropriés, assortis des valeurs possibles pour la terminaison, sont renvoyés par la commande AuditCapabilities. Certains descripteurs peuvent être répétés lorsqu'il existe plusieurs valeurs possibles.

Si une réponse remplacée par une structure générique est demandée, un seul renvoi de commande est produit, qui contient la réunion des valeurs de toutes les terminaisons qui sont contenues dans la liste ou qui correspondent à la structure générique. Cette convention peut réduire le volume de données nécessaires pour examiner un groupe de terminaisons.

Si une propriété, un signal, un événement ou une statistique est analysé, les propriétés, signaux, événements et statistiques appropriés sont renvoyés par la commande AuditCapabilities avec les capacités de la terminaison.

L'interprétation quant aux capacités qui sont requises pour diverses valeurs d'identificateur de contexte et de terminaison est la même que pour la commande AuditValue.

Pour les valeurs de propriété et de paramètre de type chaîne, caractère ou chaîne d'octets, la passerelle MG doit renvoyer une valeur vide. En codage textuel, les chaînes et les caractères renvoient une production syntaxique vide "quotedString", alors que les chaînes d'octets renvoient la valeur "NUL" (0x00). Ce comportement peut être supplanté par la définition du paquetage.

Le descripteur "Events" renvoie la liste des événements possibles concernant la terminaison ainsi que la liste de toutes les valeurs possibles pour les paramètres de descripteur "Events". Le descripteur "EventBuffer" renvoie les mêmes informations que le descripteur "Events", ainsi que la liste de toutes les valeurs possibles pour le descripteur "Events". Le descripteur "Signals" renvoie la liste des signaux dont l'application à la terminaison est possible, ainsi que la liste de toutes les valeurs possibles des paramètres de descripteur "Signals". Le descripteur "Statistics" renvoie le nom des statistiques enregistrées dans la terminaison. Le descripteur "ObservedEvents" renvoie les noms des événements actifs dans la terminaison. Les descripteurs "DigitMap" et "Packages" ne sont pas autorisés dans la commande AuditCapability.

Le tableau suivant montre d'autres informations qui peuvent être obtenues au moyen de la commande AuditCapabilities:

Identificateur de contexte	Identificateur de terminaison	Information obtenue
Spécifique	Structure générique	Examen des terminaisons concordantes dans un contexte
Spécifique	Spécifique	Examen d'une terminaison unique dans un contexte
NULL	Racine	Examen de l'état et des événements dans une passerelle MG
NULL	Structure générique	Examen de toutes les terminaisons concordantes contenues dans le contexte NULL
NULL	Spécifique	Examen d'une terminaison unique hors de tout contexte
ALL	Structure générique	Examen de toutes les terminaisons concordantes ne se trouvant pas dans le contexte NULL et examen du contexte auquel elles sont associées
ALL	Racine	Voir la commande AuditValue
ALL	Spécifique	Voir la commande AuditValue

### 7.2.7 Commande Notify

La commande Notify permet à la passerelle MG de signaler au contrôleur MGC que des événements se produisent dans cette passerelle.

TerminationID (identificateur de terminaison)

```
Notify(TerminationID,
      ObservedEventsDescriptor,
      [ErrorDescriptor])
```

Le paramètre TerminationID spécifie la terminaison qui émet la commande Notify. L'identificateur de terminaison doit être un nom entièrement qualifié.

Le descripteur "ObservedEvents" contient l'identification de demande RequestID et une liste d'événements que la passerelle MG a détectés, dans l'ordre de leur détection. Chaque événement de la liste est assorti des paramètres qui lui sont associés ainsi qu'éventuellement de l'indication du

moment de cette détection. Les procédures pour envoyer des commandes Notify avec l'identificateur RequestID égal à zéro feront l'objet d'un complément d'étude.

Les commandes Notify avec un identificateur RequestID différent de zéro ne doivent se produire qu'en tant que résultat de la détection d'un événement spécifié par un descripteur "Events" actif sur la terminaison concernée.

L'identificateur de demande renvoie le paramètre RequestID du descripteur "Events" qui a activé la commande Notify. Il sert à associer la notification à la demande qui l'a activée. Les événements contenus dans la liste doivent avoir été demandés par l'intermédiaire du descripteur "Events" de déclenchement ou par le biais du descripteur "Events" imbriqué, à moins que l'identificateur RequestID ne soit zéro (ce qui fera l'objet d'un complément d'étude).

Le descripteur "Error" peut être envoyé dans la commande Notify à la suite de l'erreur 518 ("Mémoire tampon d'événements pleine").

### 7.2.8 ServiceChange

La commande ServiceChange (changement de service) permet à la passerelle MG de signaler au contrôleur MGC qu'une terminaison ou un groupe de terminaison est sur le point d'être mis hors service ou vient d'être remis en service. Le contrôleur MGC peut indiquer que la ou les terminaisons doivent être mises hors service ou remises en service. La passerelle MG peut signaler au contrôleur MGC que la capacité d'une terminaison a changé. Cette commande permet également à un contrôleur MGC de transférer le contrôle d'une passerelle MG à un autre contrôleur MGC.

TerminationIDList,

[ServiceChangeDescriptor]

```
    ServiceChange(TerminationIDList,  
                ServiceChangeDescriptor  
                )
```

Le paramètre TerminationIDList spécifie la ou les terminaisons qui sont mises hors service ou remises en service. Le remplacement de noms de terminaison par des structures génériques est autorisé, sauf que le mécanisme CHOOSE ne doit pas être utilisé. L'utilisation de l'identificateur de terminaison "Root" indique qu'un changement de service affecte l'ensemble de la passerelle MG.

NOTE – L'utilisation de la liste TerminationIDList n'est pas valide dans la commande "ServiceChange" figurant dans la version 1 initiale de la Rec. UIT-T H.248.1.

#### 7.2.8.1 Contenu du descripteur ServiceChange

Le descripteur "ServiceChange" contient les paramètres suivants, selon les besoins:

- ServiceChangeMethod;
- ServiceChangeReason;
- ServiceChangeDelay;
- ServiceChangeAddress;
- ServiceChangeProfile;
- ServiceChangeVersion;
- ServiceChangeMgcID;
- TimeStamp;
- ExtensionParameter;
- ServiceChangeInfo;
- ServiceChangeIncompleteFlag.

### 7.2.8.1.1 Méthode "ServiceChangeMethod"

Le paramètre ServiceChangeMethod spécifie le type de changement de service qui va se produire ou qui s'est produit:

- 1) la valeur "Graceful" (changement progressif): indique que les terminaisons spécifiées seront mises hors service après le délai spécifié dans le paramètre ServiceChangeDelay; les connexions établies ne sont pas encore affectées mais le contrôleur MGC doit s'abstenir d'établir de nouvelles connexions et doit tenter de supprimer progressivement les connexions existantes sur la ou les terminaisons affectées par la commande ServiceChange. Il convient que la passerelle règle à "OutOfService" (hors service) le paramètre ServiceState au moment de l'expiration du délai ServiceChangeDelay ou au moment du retrait de la terminaison d'un contexte actif (selon ce qui survient en premier);
- 2) la valeur "Forced" (changement forcé): indique que les terminaisons spécifiées ont été mises brusquement hors service et que toutes connexions établies en association avec ces terminaisons peuvent être perdues. Pour les terminaisons qui ne sont pas de type racine, le contrôleur MGC est chargé de nettoyer le contexte (le cas échéant) auquel la terminaison défailante est associée. Au minimum, la terminaison doit être soustraite du contexte. Il convient que l'état ServiceState de la terminaison soit "OutOfService" (hors service). Pour une terminaison de type racine Root, le contrôleur MGC peut supposer que toutes les connexions sont perdues au niveau de la passerelle MG et peut donc considérer que toutes les terminaisons ont été soustraites;
- 3) la valeur "Restart" (redémarrage): indique que le service sera rétabli sur les terminaisons spécifiées après l'expiration du délai ServiceChangeDelay. Il convient que l'état ServiceState soit réglé à la valeur "InService" (en service) au moment de l'expiration du délai ServiceChangeDelay;
- 4) la valeur "Disconnected" (déconnexion): toujours appliquée à l'identificateur de terminaison racine, indique que la passerelle MG a perdu la communication avec le contrôleur MGC mais que cette communication a été rétablie par la suite (éventuellement après qu'il a essayé auprès d'autres contrôleurs MGC énumérés sur une liste établie au préalable). Comme l'état de la passerelle MG peut avoir changé, le contrôleur MGC peut souhaiter utiliser la commande Audit pour resynchroniser son état avec celui de la passerelle MG;
- 5) la valeur "Handoff" (transfert): envoyée par le contrôleur MGC à la passerelle MG, indique que le contrôleur MGC va être mis hors service et qu'une nouvelle association de contrôleur MGC doit être établie. Envoyée de la passerelle MG au contrôleur MGC, cette valeur indique que la passerelle MG essaie d'établir une nouvelle association en conformité avec la valeur "Handoff" reçue du contrôleur MGC auquel elle était précédemment associée;
- 6) la valeur "Failover" (reprise sur défaillance): envoyée de la passerelle MG au contrôleur MGC, indique que la passerelle MG primaire est hors service et qu'une passerelle MG secondaire prend le relais. La méthode ServiceChange est aussi envoyée de la passerelle MG au contrôleur MGC lorsque la passerelle MG détecte que celui-ci a échoué;
- 7) toute autre valeur dont la signification est mutuellement comprise entre la passerelle MG et le contrôleur MGC.

### 7.2.8.1.2 ServiceChangeReason

Le paramètre ServiceChangeReason spécifie la raison pour laquelle le changement de service s'est produit ou va se produire. Il se compose d'un jeton alphanumérique (enregistré par l'autorité IANA) et, facultativement, d'une chaîne explicative.

### 7.2.8.1.3 ServiceChangeAddress et ServiceChangeMgcID

Le paramètre ServiceChangeAddress spécifie l'adresse (par exemple, le numéro d'accès IP pour réseaux en protocole IP) à utiliser pour les communications subséquentes. Il peut être spécifié dans le descripteur de paramètre d'entrée ou dans le descripteur de résultat renvoyé. Les paramètres ServiceChangeAddress et ServiceChangeMgcID ne doivent pas être présents tous les deux dans le descripteur "ServiceChange" ou dans le descripteur "ServiceChangeResult". Le paramètre ServiceChangeAddress fournit une adresse à utiliser à l'intérieur du contexte d'une association en cours de négociation, alors que le paramètre ServiceChangeMgcID fournit une adresse de remplacement où il convient que la passerelle MG cherche à établir une autre association. Il convient de noter que l'emploi du paramètre ServiceChangeAddress n'est pas encouragé. Les contrôleurs MGC et les passerelles MG doivent être en mesure de prendre en charge le paramètre ServiceChangeAddress, qu'il soit une adresse complète ou simplement un numéro de port dans le cas des transports TCP.

### 7.2.8.1.4 ServiceChangeDelay

Le paramètre facultatif ServiceChangeDelay est exprimé en secondes. Si le délai est absent ou mis à zéro, il convient que sa valeur soit considérée comme étant néant. Dans le cas d'une méthode de changement de service progressive ("Graceful"), un délai néant indique que le contrôleur MGC doit attendre l'extinction naturelle des connexions existantes et qu'il ne doit pas en établir de nouvelles. Pour la valeur "Graceful" uniquement, un délai néant signifie que la passerelle MG ne doit pas régler l'état ServiceState à "OutOfService" (hors service) jusqu'à ce que la terminaison soit dans le contexte NULL.

### 7.2.8.1.5 ServiceChangeProfile

Le paramètre facultatif ServiceChangeProfile spécifie le profil (éventuel) du protocole pris en charge. Ce paramètre indique la version du profil pris en charge. En l'absence de ce paramètre, la valeur "NoProfile" est supposée.

### 7.2.8.1.6 ServiceChangeVersion

Le paramètre facultatif ServiceChangeVersion contient la version du protocole et il est utilisé si la négociation de la version du protocole se produit (voir le § 11.3).

### 7.2.8.1.7 TimeStamp

Le paramètre facultatif TimeStamp (horodatage) spécifie l'heure effective telle qu'elle est conservée par l'expéditeur. En tant que telle, ce n'est pas nécessairement l'heure absolue conforme, par exemple, à un fuseau horaire local: c'est plutôt un temps de début arbitraire par rapport auquel les horodateurs ultérieurs expédiés par un émetteur au cours de cette association seront comparés. Il peut être utilisé par le récepteur afin de déterminer comment sa notion du temps diffère de celle de son correspondant. Le paramètre TimeStamp est envoyé avec une précision de centièmes de seconde.

### 7.2.8.1.8 ExtensionParameter

Le paramètre facultatif "Extension" peut contenir toute valeur dont la signification est réciproquement comprise par la passerelle MG et par le contrôleur MGC. La valeur "X-SC" est réservée pour l'utilisation des paramètres de signalisation de changement de service ServiceChange qui ont été ajoutés dans le cadre de la présente version 3 de la Rec. UIT-T H.248.1 (et des versions subséquentes). Ce paramètre n'est utilisé dans la commande ServiceChange initiale de la version 1 de la présente Recommandation, codée de la passerelle MG au contrôleur MGC, que lorsque la version du changement de service (ServiceChangeVersion) est supérieure ou égale à 3. La structure de la valeur est définie par le formalisme ABNF ci-après:

$$\text{X-SC} = 1 * (\text{NAME EQUAL paramValue [COMMA]})$$

Lors de l'adjonction de nouveaux paramètres de changement de service, il est recommandé qu'un nom/valeur soit assigné si le paramètre doit être envoyé dans une commande ServiceChange initiale. Par exemple:

**Nom:** chaînes comptant jusqu'à 64 caractères, ne contenant aucun espace, commençant par un caractère alphabétique et se composant de caractères alphanumériques et/ou de chiffres, et pouvant inclure le caractère spécial de soulignement ("\_").

**Type:** conformément au § 12.1.2 "Propriétés"

**Valeurs possibles:** conformément au § 12.1.2 "Propriétés"

#### 7.2.8.1.9 ServiceChangeInfo

Le paramètre facultatif "ServiceChangeInfo" peut contenir le paquetage/la propriété/le signal/l'événement/la statistique du motif du changement de service.

#### 7.2.8.1.10 ServiceChangeIncompleteFlag

Ce fanion facultatif indique que des commandes ServiceChange subséquentes seront envoyées de la passerelle MG au contrôleur MGC afin d'indiquer l'état de terminaisons. Il n'est utilisé que pendant l'enregistrement ou le redémarrage de la passerelle MG (changement de service à la racine avec redémarrage de la méthode de changement de service) lorsque la passerelle MG souhaite signaler au contrôleur MGC l'état de la passerelle MG entière et de la terminaison. Dès réception de ce fanion, le contrôleur MGC doit s'abstenir de produire des commandes agissant sur des terminaisons autres que de type racine dans la passerelle MG. Après avoir envoyé le fanion ServiceChangeIncompleteFlag dans la commande initiale d'enregistrement/redémarrage, la passerelle MG doit l'envoyer dans toutes les commandes ServiceChange subséquentes jusqu'à ce que la passerelle MG ait déterminé qu'elle a signalé l'état actuel de la passerelle MG et de ses terminaisons. Le fanion doit ensuite être retiré de la dernière commande ServiceChange afin d'indiquer l'état de la passerelle MG ou de ses terminaisons. Dès réception de cette commande ServiceChange sans le fanion ServiceChangeIncompleteFlag, le contrôleur MGC peut de nouveau produire des commandes à destination de la passerelle MG. Lorsque ce fanion est envoyé dans une commande initiale ServiceChange (selon la version 1 de la Rec. UIT-T H.248.1) au moyen du paramètre "X-SC" d'extension de la commande ServiceChange, ce qui suit est utilisé:

**Nom:** SIC

**Type:** valeur booléenne

**Valeurs possibles:** ON (fanion inclus)

NOTE – La valeur OFF n'est pas requise car les commandes subséquentes seront codées selon la Rec. UIT-T H.248.1.

#### 7.2.8.1.11 Commande ServiceChange et réponse

Une commande ServiceChange spécifiant la racine ("Root") pour l'identificateur de terminaison TerminationID et spécifiant une méthode ServiceChangeMethod de valeur égale à "Restart" est une commande d'enregistrement permettant à une passerelle MG d'annoncer son existence au contrôleur MGC. La passerelle MG peut procéder à l'enregistrement en spécifiant la racine ("Root") pour l'identificateur de terminaison et une méthode ServiceChangeMethod de valeur égale à "Failover" (reprise sur défaillance) lorsqu'elle détecte des défaillances de la part des contrôleurs MGC. Un message contenant une commande ServiceChange Command spécifiant l'identificateur de terminaison de la racine et une méthode ServiceChangeMethod de valeur égale à Restart ou Failover ne doit pas contenir d'autres commandes car celles-ci devraient utiliser la nouvelle adresse ServiceChangeAddress et la nouvelle version de protocole négociée.

La passerelle MG est censée être profilée avec le nom d'un contrôleur MGC primaire et, facultativement, avec celui d'un certain nombre de contrôleurs MGC de remplacement.

L'acquiescement de la commande ServiceChange met fin au processus d'enregistrement, sauf si le contrôleur MGC a renvoyé un paramètre ServiceChangeMgcID de rechange tel que décrit ci-dessous.

La passerelle MG peut spécifier, dans le paramètre ServiceChangeAddress contenu dans le descripteur "ServiceChange", l'adresse de changement de service de transport que le contrôleur MGC utilisera pour envoyer des messages. La passerelle MG peut spécifier une adresse dans le paramètre ServiceChangeAddress contenu dans la demande de changement de service et le contrôleur MGC peut faire de même dans la réponse à la demande de changement de service. Dans un cas comme dans l'autre, le destinataire doit toujours utiliser l'adresse fournie comme destination de toutes les demandes subséquentes de transaction contenues à l'intérieur de l'association. En même temps, comme il est indiqué dans le § 9, les réponses de transactions et les indications en attente doivent être envoyées à l'adresse d'origine des demandes correspondantes. Il doit en être ainsi même si cela implique une messagerie supplémentaire car les commandes et les réponses ne peuvent pas être empaquetées ensemble.

Le paramètre d'horodatage TimeStamp doit être envoyé avec une commande d'enregistrement et avec sa réponse.

Lors de l'enregistrement, la passerelle MG peut envoyer le fanion ServiceChangeIncompleteFlag si elle a déterminé qu'elle se trouve dans l'état "InService"; mais des terminaisons individuelles peuvent être dans l'état "OutOfService". Cela vise à empêcher des tentatives de prise de terminaisons hors service. Des commandes ServiceChange subséquentes sont utilisées afin de signaler les terminaisons à l'état "OutOfService".

Le contrôleur MGC peut renvoyer un paramètre ServiceChangeMgcID décrivant le contrôleur MGC que la passerelle MG doit de préférence approcher pour la suite du service. Dans ce cas, la passerelle MG doit réémettre la commande ServiceChange vers le nouveau contrôleur MGC. Le contrôleur MGC spécifié dans un éventuel paramètre ServiceChangeMgcID doit être approché avant tout autre contrôleur MGC de remplacement. Selon un message de transfert envoyé par le contrôleur MGC à la passerelle MG, le paramètre ServiceChangeMgcID indique le nouveau contrôleur MGC qui prendra la suite du contrôleur MGC actuel.

Le retour provenant de la commande ServiceChange est vide sauf si l'identificateur de terminaison "Root" est utilisé. Dans ce cas, il contient les paramètres suivants, selon les besoins:

- ServiceChangeAddress, si le contrôleur MGC récepteur souhaite spécifier une nouvelle destination pour les messages provenant de la passerelle MG pour le reste de l'association;
- ServiceChangeMgcID, si le contrôleur MGC récepteur ne souhaite pas entretenir d'association avec la passerelle MG;
- ServiceChangeProfile, si le récepteur souhaite négocier le profil à utiliser pour l'association. Le profil (nom et version) n'est renvoyé en réponse que dans le cas où le contrôleur MGC n'est pas en mesure de prendre en charge les profils spécifiés dans la demande ServiceChangeRequest. La réponse renvoyée doit indiquer le profil et la version pris en charge ou mentionner "NoProfile", si aucun profil n'est pris en charge. A la réception d'un profil dans la réponse, la passerelle MG peut maintenir sa relation avec le contrôleur MGC actuel ou prendre contact avec les contrôleurs secondaires et établir une relation avec eux. Si le contrôleur MGC a renvoyé dans la réponse un profil autre que celui qui a été fourni dans la requête, la passerelle MG doit:
  - a) continuer l'association de commande en émettant une nouvelle commande ServiceChange avec un profil agréé afin de confirmer au contrôleur MGC que la passerelle MG a agréé ce profil; ou
  - b) conserver l'association de commande active, de façon que le contrôleur MGC utilise le profil qu'il a envoyé dans la réponse à la commande ServiceChange; ou

- c) établir une association de commande avec un contrôleur MGC différent, au moyen de son profil original.
- ServiceChangeVersion, si le récepteur souhaite négocier la version du protocole à utiliser pour l'association.

Des raisons de changement de service (ServiceChangeReasons) sont définies dans la Rec. UIT-T H.248.8. Cette liste peut être étendue par un enregistrement IANA comme il est souligné dans le § 14.3.

### 7.2.9 Manipulation et examen d'attributs de contexte

Les commandes du protocole exposées dans les paragraphes précédents s'appliquent aux terminaisons. Le présent paragraphe spécifie la façon de manipuler et d'examiner des contextes.

Une action peut contenir des instructions de manipulation et d'examen de contexte (voir le § 8).

Une demande d'action envoyée à une passerelle MG peut contenir une demande d'examen d'attributs d'un contexte. Il existe deux types d'examen.

**Examen de valeur:** le contrôleur MGC peut examiner un contexte particulier afin de déterminer la valeur actuelle de propriétés de contexte distinctes. Il peut déterminer les valeurs actuelles de tous les contextes existants (autres que NULL) en attribuant à l'identificateur de contexte, dans la demande d'examen, la valeur ALL. Si des attributs de contexte sont ajoutés ou ont été modifiés en même temps que la demande d'examen, la ou les valeurs seront renvoyées après que l'action a été effectuée. Des attributs de contexte peuvent être utilisés comme critères de sélection afin de filtrer les valeurs de renvoi d'examen. Il est possible d'inclure de multiples critères de sélection. Lorsque des critères multiples sont inclus, un opérateur logique ET ou OU peut également être inclus afin d'indiquer comment les critères de sélection doivent être interprétés. Si aucun opérateur logique n'est inclus, un opérateur logique ET est supposé.

**Examen de capacités:** le contrôleur MGC peut examiner un contexte spécifique afin de déterminer les valeurs possibles que les propriétés individuelles du contexte peuvent prendre. Le contrôleur MGC peut déterminer les valeurs possibles pour l'ensemble de la passerelle MG en spécifiant l'identificateur de contexte ALL dans la demande d'examen.

Le tableau ci-après illustre les informations pouvant être obtenues dans un contexte d'examen:

ContextID	TerminationID	AuditValue	AuditCapabilities
Spécifique	Non applicable	Valeur de propriété de contexte dans le contexte spécifié	Valeurs de propriété de contexte possibles dans le contexte spécifié
NULL	Non applicable	Non autorisé	Non autorisé
ALL	Non applicable	Valeurs actuelles pour tous les contextes (autres que NULL) existants, en spécifiant l'identificateur de contexte ALL dans la demande d'examen.  Une réponse à l'identificateur de contexte ALL est présentée au moyen d'une réponse actionReply pour chaque contexte.	Une réponse à l'identificateur de contexte ALL est renvoyée avec des valeurs de propriété de contexte possibles dans l'ensemble de la passerelle MG

Une action peut également contenir une demande de modification des attributs d'un contexte.

Les propriétés d'un contexte qui peuvent être contenues dans une réponse d'action sont utilisées pour renvoyer des informations au contrôleur MGC. Il peut s'agir d'informations demandées par un examen d'attributs de contexte ou de détails relatifs à l'effet de la manipulation d'un contexte.

Si une passerelle MG reçoit une action contenant à la fois une demande d'examen d'attributs de contexte et une demande de manipulation de ces attributs, la réponse doit contenir les valeurs des attributs après le traitement de la demande de manipulation.

### 7.2.10 Syntaxe de commande générique

Le protocole peut être codé soit en format binaire soit en format alphanumérique. Il y a lieu que les contrôleurs MGC prennent en charge ces deux formats de codage. Les passerelles MG peuvent prendre en charge les deux formats.

La syntaxe de protocole pour le format binaire du protocole est définie dans l'Annexe A. L'Annexe C spécifie le codage des descripteurs "Local" et "Remote" en vue d'une utilisation avec le format binaire.

L'Annexe B donne en formalisme BNF augmenté (ABNF, *augmented Backus-Naur form*) le codage alphanumérique du protocole selon RFC 2234. Le protocole SDP est utilisé pour le codage des descripteurs "Local" et "Remote" en vue d'une utilisation avec le format alphanumérique tel que modifié au § 7.1.8.

## 8 Transactions

Les commandes acheminées entre le contrôleur MGC et la passerelle MG sont groupées en transactions, dont chacune est désignée par un identificateur de transaction. Des transactions se composent d'une ou de plusieurs actions. Une action se compose d'une série non vide de commandes, de modifications de la propriété de contexte ou d'examen de la propriété de contexte dont le fonctionnement est limité à un seul contexte. Par conséquent, chaque action spécifie normalement un identificateur de contexte. Il y a cependant deux circonstances dans lesquelles un identificateur de contexte spécifique n'est pas associé à une action. L'une est le cas d'une modification d'une terminaison extérieure à un contexte. L'autre concerne l'emplacement de création d'un nouveau contexte demandé par le contrôleur à la passerelle. Les relations entre transactions, actions et commandes sont décrites à la Figure 11.

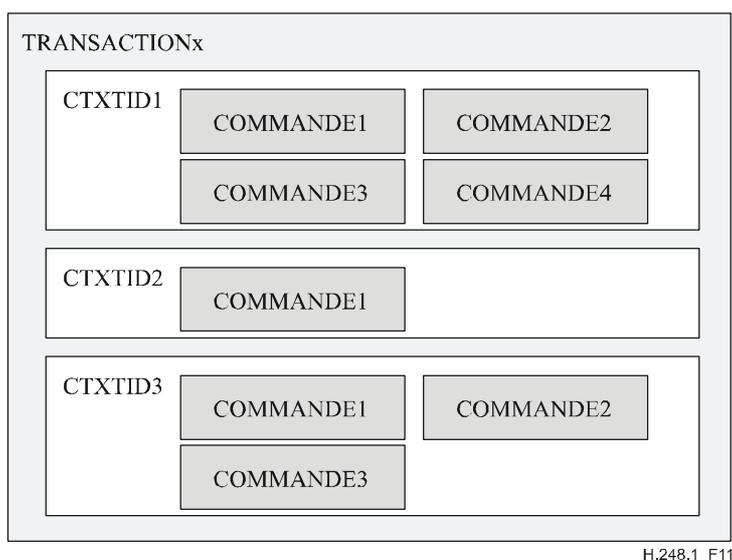


Figure 11/H.248.1 – Transactions, actions et commandes

Les transactions sont présentées sous la forme de demandes de transaction (TransactionRequest). Les réponses correspondantes à une demande de transaction sont reçues dans une seule réponse, éventuellement précédée d'un nombre de messages portant sur les transactions en cours (voir le § 8.2.3).

Les transactions garantissent un traitement ordonné des commandes, c'est-à-dire une exécution séquentielle de celles-ci dans le cadre d'une transaction. L'ordre des transactions n'est pas garanti car les transactions peuvent être exécutées dans un ordre quelconque ou simultanément; cependant, les réponses de transaction devraient être exécutées avant les demandes de transaction lorsque les deux commandes sont contenues dans un message.

Dès la première commande défailante dans une transaction, le traitement des commandes restant à exécuter dans cette transaction est interrompu. Si une commande contient un identificateur de terminaison remplacé par une structure générique, cette commande est exécutée avec chacun des identificateurs de terminaison réels qui correspondent à cette structure. Une réponse contenue dans la commande TransactionReply est incluse pour chaque identificateur de terminaison même si une ou plusieurs instances ont donné lieu à une erreur. Si un identificateur de terminaison correspondant à une structure générique donne lieu à une erreur lors de son exécution, aucune commande n'est tentée après celle qui a été remplacée par cette structure générique.

Les commandes peuvent être marquées comme étant "facultatives", ce qui peut neutraliser ce comportement, c'est-à-dire que, si une commande marquée comme facultative donne lieu à une erreur, les commandes subséquentes de la transaction seront exécutées. Si une commande échoue, la passerelle MG rétablira dans la mesure du possible l'état qui existait avant la tentative d'exécution de la commande avant de poursuivre le traitement des commandes.

Une réponse de transaction effectuée (TransactionReply) contient les résultats pour toutes les commandes contenues dans la demande de transaction correspondante. Cette réponse contient les valeurs de retour pour les commandes qui ont été exécutées normalement ainsi que l'identificateur et le descripteur "Error" de toute commande ayant échoué. Une réponse de transaction en cours (TransactionPending) est utilisée pour signaler périodiquement au récepteur qu'une transaction n'a pas encore été effectuée mais est en cours de traitement actif.

Il convient que les applications implémentent, lors de chaque transaction, un temporisateur de niveau d'application dont l'expiration devrait entraîner la réémission de la demande. La réception d'une réponse de transaction effectuée devrait désarmer le temporisateur. La réception d'une réponse de transaction en cours devrait réarmer le temporisateur.

## **8.1 Paramètres communs**

### **8.1.1 Identificateurs de transaction**

Les transactions sont identifiées par un identificateur de transaction qui est attribué par l'émetteur et qui est unique dans le domaine de celui-ci. Une réponse contenant un descripteur "Error" visant à signaler que l'identificateur de transaction est absent d'une demande doit utiliser l'identificateur de transaction zéro dans la réponse de transaction correspondante.

### **8.1.2 Identificateurs de contexte**

Les contextes sont identifiés par des identificateurs qui sont attribués par la passerelle MG et qui sont uniques dans le domaine de celle-ci. Le contrôleur MGC doit utiliser l'identificateur de contexte fourni par la passerelle MG dans toutes les transactions subséquentes qui se rapportent à ce contexte. Le protocole fait référence à une valeur distinctive que le contrôleur MGC peut utiliser pour se référer à une terminaison qui n'est pas actuellement associée à un contexte, c'est-à-dire l'identificateur de contexte *NULL*.

La structure générique CHOOSE sert à demander à la passerelle média de créer un nouveau contexte.

Le contrôleur MGC peut utiliser la structure générique ALL pour s'adresser à tous les contextes présents sur la passerelle MG. Le contexte NULL n'est pas inclus lorsque la structure générique ALL est utilisée.

Le contrôleur MGC n'emploiera pas les identificateurs de contexte partiellement spécifiés contenant les structures génériques CHOOSE ou ALL.

## 8.2 Interface de programmation d'application pour les transactions

L'interface de programmation d'application (API, *application programming interface*) ci-dessous décrit les transactions du protocole. Cette interface API vise à illustrer les transactions et leurs paramètres. Elle n'est pas destinée à spécifier une implémentation (par exemple au moyen d'appels de fonction de blocage). Elle décrira les paramètres d'entrée et les valeurs de retour censés devoir être utilisés par les diverses transactions du protocole à partir d'un niveau très élevé. La syntaxe et les codages des transactions seront spécifiés dans des paragraphes ultérieurs.

### 8.2.1 Demande de transaction (TransactionRequest)

La demande de transaction est invoquée par l'émetteur. Il y a une seule transaction par invocation de demande. Une demande contient une ou plusieurs actions, dont chacune spécifie son contexte cible.

```
TransactionRequest(TransactionID {  
    ContextID {Command ... Command},  
    ...  
    ContextID {Command ... Command}})
```

Le paramètre TransactionID doit toujours spécifier une valeur en vue d'une corrélation ultérieure avec la réponse de type transaction effectuée ou transaction en cours en provenance du récepteur.

Le paramètre ContextID doit toujours spécifier une valeur à rattacher à toutes les commandes qui font suite soit à la prochaine spécification d'un paramètre ContextID soit à la fin de la demande de transaction, selon le message qui arrive en premier.

Le paramètre Command désigne l'une des commandes mentionnées au § 7.2 ("Interface de programmation d'application pour les commandes").

### 8.2.2 Réponse de Transaction (TransactionReply)

La réponse de transaction (TransactionReply) effectuée est invoquée par le récepteur. Il y a une seule invocation de réponse par transaction. Une réponse contient une ou plusieurs actions, dont chacune doit toujours spécifier son contexte cible, ainsi qu'une ou plusieurs réponses par contexte. Une réponse peut être segmentée en plusieurs messages. La réponse de transaction est invoquée par le récepteur lorsqu'il a traité la demande de transaction.

Une demande de transaction est traitée lorsque:

- toutes les actions contenues dans la demande de transaction ont été exécutées;
- une erreur est produite au cours du traitement de la demande de transaction, ailleurs que dans une commande facultative.

Si un message contient plus de demandes de transaction TransactionRequests que le récepteur ne peut traiter, le récepteur renverra le descripteur "Error" 413 ("Dépassement du nombre maximal de transactions dans un message").

Une commande est traitée lorsque tous les descripteurs contenus dans cette commande ont été traités.

Un descripteur "Signals" est considéré comme ayant été traité lorsqu'il a été établi que sa syntaxe est valable, que les signaux demandés sont pris en charge et qu'ils ont été placés en file d'attente afin d'être appliqués.

Un descripteur "Events" ou "EventBuffer" est considéré comme ayant été traité lorsqu'il a été établi que sa syntaxe est valable, que les événements demandés peuvent être observés, que tout signal intégré peut être émis, que tout événement intégré peut être détecté et que la passerelle MG a été placée dans un état permettant la détection des événements.

```
TransactionReply(TransactionID {  
    ContextID {Response ... Response},  
    ...  
    ContextID {Response ... Response}})
```

Le paramètre TransactionID doit être le même que celui de la demande TransactionRequest correspondante.

Le paramètre ContextID doit toujours spécifier une valeur à rattacher à toutes les réponses pour l'action. Cet identificateur peut être spécifique, ALL ou NULL.

Chacun des paramètres de réponse représente une valeur de retour comme indiqué au § 7.2 ou un descripteur "Error" si l'exécution de la commande a rencontré une erreur. Les commandes postérieures au point de défaillance ne sont pas traitées et des réponses ne sont donc pas émises pour elles.

Une exception à cette règle existe si une commande a été marquée comme étant facultative dans la demande de transaction. Si la commande facultative produit une erreur, la transaction continue à s'exécuter de sorte que des réponses seront émises après une erreur.

Le paragraphe 7.1.20 ("Descripteur Error") décrit la production des descripteurs d'erreur. Le texte ci-après traite de plusieurs cas individuels.

Si le récepteur rencontre une erreur lors du traitement d'un identificateur de contexte mais qu'il puisse analyser syntaxiquement cet identificateur de contexte, la réponse d'action demandée sera composée de l'identificateur de contexte et d'un seul descripteur "Error", 422 ("Erreur de syntaxe dans la commande Action"). Si le récepteur ne peut pas analyser syntaxiquement l'identificateur de contexte, il renverra une réponse de transaction composée de l'identificateur de transaction et d'un seul descripteur "Error", 422 ("Erreur de syntaxe dans la commande Action").

Si le récepteur rencontre une erreur telle qu'il ne puisse pas déterminer une action autorisée, il renvoie une réponse de transaction composée de l'identificateur de transaction et d'un seul descripteur "Error", 422 ("Erreur de syntaxe dans la commande Action"). Si la fin d'une action ne peut pas être déterminée de manière fiable mais qu'il soit possible d'analyser syntaxiquement une ou plusieurs commandes, le récepteur les traite et envoie ensuite le code 422 ("Erreur de syntaxe dans la commande Action") en tant que dernière action pour la transaction. Si le récepteur rencontre une erreur telle qu'il ne puisse pas déterminer une transaction autorisée, il renvoie une réponse de transaction effectuée, composée d'un identificateur de transaction de valeur NULL et d'un seul descripteur "Error" 403 ("Erreur de syntaxe dans la demande de transaction").

Si la fin d'une transaction ne peut pas être déterminée de manière fiable mais qu'il soit possible d'analyser syntaxiquement une ou plusieurs actions, le récepteur les traite et envoie ensuite le code 403 ("Erreur de syntaxe dans la transaction") en tant que dernière réponse d'action pour la transaction. Si aucune action ne peut être analysée syntaxiquement, il renvoie le code 403 ("Erreur de syntaxe dans la transaction") en tant que seule réponse.

Si l'identificateur de terminaison (terminationID) ne peut pas être déterminé de manière fiable, il envoie le code 442 ("Erreur de syntaxe dans la commande") en tant que réponse d'action.

Si la fin d'une commande ne peut pas être déterminée de manière fiable, il renvoie le code 442 ("Erreur de syntaxe dans la commande") en tant que réponse à la dernière action qu'il peut analyser syntaxiquement.

### 8.2.3 Transaction en cours (TransactionPending)

C'est le récepteur qui invoque la réponse de transaction en cours. Celle-ci indique que la transaction est en cours de traitement actif mais que ce traitement n'est pas terminé. Cette réponse sert à éviter que l'émetteur ne tire la conclusion que la demande de transaction a été perdue lorsque la transaction demande un certain temps pour s'exécuter.

TransactionPending(TransactionID { } )

Le paramètre TransactionID doit être le même que celui de la demande TransactionRequest correspondante. Une propriété de terminaison racine (normalMGExecutionTime) est réglable par le contrôleur MGC afin d'indiquer l'intervalle pendant lequel la passerelle MG attend effectivement du contrôleur MGC une réponse à une transaction quelconque (à l'exclusion du retard dû au réseau). Une autre propriété de racine (normalMGCEExecutionTime) est réglable par le contrôleur MGC afin d'indiquer l'intervalle pendant lequel il convient que la passerelle MG attende du contrôleur MGC une réponse à une transaction quelconque (à l'exclusion du retard dû au réseau. La valeur MGProvisionalResponseTimerValue indique la durée pendant laquelle il convient que le contrôleur MGC attende une réponse en cours issue de la passerelle MG si une transaction ne peut pas être effectuée (cette valeur est initialement mise à la valeur normalMGExecutionTime plus le retard dû au réseau, mais elle peut être diminuée). La valeur MGProvisionalResponseTimerValue a la signification correspondant à celle de la passerelle MG. Les émetteurs peuvent recevoir plusieurs réponses de transaction en cours pour la même commande. Si une demande dupliquée est reçue pendant l'attente, le récepteur peut immédiatement envoyer une demande en attente ou continuer à attendre que son temporisateur déclenche une autre attente TransactionPending.

Une propriété de la terminaison racine (MGOriginatedPendingLimit) peut être fixée par le contrôleur MGC pour indiquer le nombre de transactions en cours qui peuvent être reçues en provenance de la passerelle MG. Lorsque la valeur indiquée par cette propriété est dépassée, la passerelle MG mettra fin au traitement de la transaction et renverra une réponse TransactionReply, afin que le contrôleur MGC ne pense pas que la transaction est fausse.

Une autre propriété de la terminaison racine (MGCOrganatedPendingLimit) peut être fixée par le contrôleur MGC pour indiquer le nombre de transactions en cours qui peuvent être reçues en provenance de lui. Lorsque la valeur indiquée par cette propriété est dépassée, le contrôleur MGC mettra fin au traitement de la transaction et renverra une réponse TransactionReply, afin que la passerelle MG ne suppose pas que la transaction est fausse.

Le nombre limite de transactions en cours émises xxxOriginatedPendingLimit (MGOriginatedPendingLimit ou MGCOrganatedPendingLimit) peut être dépassé en raison soit du traitement d'une longue commande, soit d'une erreur (par exemple, une commande provoquant une boucle). Dans les deux cas, le récepteur de la demande de transaction initiale émettra une réponse TransactionReply avec un descripteur "Error" comme paramètre de réponse, en rapport avec soit la longue commande incriminée, ou avec la commande ayant provoqué l'erreur. Les commandes suivantes de la transaction ne seront pas traitées. L'erreur 506 ("Nombre de transactions en cours dépassé") doit être utilisée.

NOTE – Afin d'éviter que le nombre limite de transactions en cours émises xxxOriginatedPendingLimit (MGOriginatedPendingLimit ou MGCOrganatedPendingLimit) soit dépassé en raison d'une erreur et que le récepteur de la demande de transaction initiale continue d'envoyer des transactions en cours, celui-ci devrait implémenter un mécanisme de protection de la gestion afin que soient déclenchées les mesures de récupération appropriées. L'émetteur de la demande de transaction initiale peut suivre l'évolution du nombre de transaction en cours reçues et commencer à prendre des mesures correctives.

## 8.3 Messages

Plusieurs transactions peuvent être concaténées pour introduction dans un message. Les messages ont un en-tête contenant l'identité de l'émetteur. L'identificateur de message (MID, *message identifier*) est réglé sur un nom fourni (par exemple adresse de domaine/nom de domaine/nom de

dispositif) de l'entité qui émet le message. Le nom de domaine est une valeur par défaut suggérée. Une entité H.248.1 (passerelle MG ou contrôleur MGC) doit employer de manière cohérente le même identificateur MID dans tous les messages qu'elle envoie pendant la durée de l'association en matière de commande avec son homologue (contrôleur MGC ou passerelle MG)

Chaque message contient un numéro de version désignant la version du protocole auquel le message est conforme. Les versions comportent un ou deux chiffres, en partant de la version 1. La version actuelle du protocole est la version 3.

Les transactions contenues dans un message sont traitées indépendamment. Aucun ordre n'est préétabli et il n'y a pas d'acquiescement de message par l'application ou le protocole. Un message est avant tout un mécanisme de transport. Par exemple, il peut être répondu au message X contenant les demandes de transaction A, B et C par le message Y contenant les réponses à A et C et par le message Z contenant la réponse à B. De même, il peut être répondu au message L contenant une demande D et au message M contenant une demande E par le message N contenant les réponses aux messages D et E.

## **9 Transport**

Le mécanisme de transport pour le protocole doit normalement permettre l'acheminement fiable de transactions entre un contrôleur MGC et une passerelle MG. Le transport doit rester indépendant des commandes particulières qui sont envoyées et doit être applicable à tous les états applicatifs. Plusieurs transports sont définis pour le protocole dans les annexes à la présente Recommandation et aux autres Recommandations de la sous-série H.248 (par exemple, H.248.4 et H.248.5). D'autres modes de transport pourront être définis dans des Recommandations supplémentaires de la sous-série H.248. Concernant le transport du protocole au-dessus du protocole IP, les contrôleurs MGC doivent implémenter aussi bien le protocole TCP que le protocole UDP/ALF et une passerelle MG doit implémenter le protocole TCP ou le protocole UDP/ALF ou les deux.

La passerelle MG est pourvue d'un nom ou d'une adresse (tel qu'un nom de service DNS ou une adresse IP) d'un contrôleur MGC primaire et de zéro, un ou plusieurs contrôleurs MGC secondaires (voir le § 7.2.8). Cette adresse est celle que la passerelle MG utilisera pour envoyer des messages au contrôleur MGC. Si le protocole TCP ou UDP est utilisé comme le transport de protocole et si le port auquel la demande initiale de changement de service doit être envoyée n'est pas autrement connu, il convient d'envoyer la demande au numéro de port par défaut du protocole. Ce numéro de port est 2944 pour une opération codée au format alphanumérique ou 2945 pour une opération codée au format binaire, tant pour le protocole UDP que pour le protocole TCP. Le contrôleur MGC reçoit de la passerelle MG le message contenant la demande de changement de service et peut en déterminer l'adresse de la passerelle MG. Comme il est décrit dans le § 7.2.8, la passerelle MG ou le contrôleur MGC peut fournir une adresse dans le paramètre ServiceChangeAddress à laquelle des demandes de transaction subséquentes doivent être adressées, mais les réponses (y compris la réponse à la demande initiale de changement de service) doivent toujours être renvoyées à l'adresse source de la demande correspondante. Par exemple, dans les réseaux IP, cela correspond à l'adresse de la source dans l'en-tête IP et du numéro de port de la source dans l'en-tête TCP/UDP/SCTP.

### **9.1 Ordonnancement des commandes**

La présente Recommandation ne prescrit pas que le protocole de transport sous-jacent garantisse la mise en séquence des transactions envoyées à une entité. Cette propriété tend à maximiser la ponctualité des actions mais elle comporte quelques inconvénients. Par exemple:

- les commandes Notify peuvent être retardées et arriver au contrôleur MGC après la transmission d'une nouvelle commande modifiant le descripteur "Events";
- si une nouvelle commande est transmise avant qu'une précédente soit acquittée, rien ne garantit que cette commande antérieure sera exécutée avant la nouvelle.

Les contrôleurs MGC qui souhaitent garantir un fonctionnement cohérent de la passerelle MG peuvent appliquer les règles ci-après. Ces règles se rapportent aux commandes qui se trouvent dans des transactions différentes. Les commandes qui se trouvent dans la même transaction sont exécutées dans l'ordre (voir le § 8).

- 1) Lorsqu'une passerelle MG gère plusieurs terminaisons, les commandes relatives à ces différentes terminaisons peuvent être envoyées en parallèle, par exemple selon un modèle où chaque terminaison (ou groupe de terminaisons) est commandée par son propre processus ou par son propre fil d'exécution individuelle.
- 2) Sur une terminaison, il convient normalement qu'il y ait au plus une seule commande en instance (Add, Modify ou Move), à moins que les commandes en instance ne se trouvent dans la même transaction. Une commande Subtract peut cependant être émise à tout moment. En conséquence, une passerelle MG peut parfois recevoir une commande Modify qui s'applique à une terminaison déjà soustraite. De telles commandes doivent être ignorées et un code d'erreur doit être renvoyé.
- 3) Pour les transports qui ne garantissent la livraison en séquence des messages (à savoir UDP), il convient normalement qu'il y ait au niveau d'une terminaison donnée au plus une seule commande Notify en instance à un moment donné.
- 4) Dans certains cas, une commande Subtract implicitement ou explicitement remplacée par une structure générique, qui s'applique à un groupe de terminaisons, peut se placer devant une commande Add en instance. Le contrôleur MGC devrait supprimer individuellement toutes les terminaisons pour lesquelles une commande Add était en instance au moment de la commande Subtract globale. Il convient également de ne pas envoyer de nouvelles commandes Add pour les terminaisons désignées par des structures génériques de remplacement (ou implicites dans un descripteur "Multiplex") jusqu'à ce que la commande Subtract remplacée par une structure générique soit acquittée.
- 5) Les commandes AuditValue et AuditCapability ne sont soumises à aucun séquençement.
- 6) La commande ServiceChange doit toujours être la première commande envoyée par une passerelle MG, comme défini par la procédure de redémarrage. Toute autre commande ou réponse doit être acheminée après cette commande ServiceChange.

Ces règles n'affectent pas le récepteur de la commande qui doit toujours répondre aux commandes.

## 9.2 Protection contre l'avalanche de redémarrage

Si un grand nombre de passerelles médias sont mises sous tension simultanément et si elles doivent toutes déclencher une transaction de changement de service, le contrôleur de passerelle média sera probablement submergé, ce qui provoquera des pertes de message et un encombrement du réseau pendant la période critique du rétablissement du service. Afin d'éviter de telles avalanches, le comportement suivant est suggéré:

- 1) lorsqu'une passerelle MG est mise sous tension, elle doit armer un temporisateur de redémarrage de valeur aléatoire uniformément répartie entre zéro et un temps d'attente maximal (MWD, *maximum waiting delay*). Il convient de veiller à éviter le synchronisme de production de nombres aléatoires entre plusieurs passerelles MG utilisant le même algorithme;
- 2) il convient que la passerelle MG attende soit l'expiration de cette temporisation soit la détection d'une activité d'utilisateur local, telle qu'une transition de décrochage sur une passerelle MG résidentielle;
- 3) lorsque la temporisation arrive à expiration ou lorsqu'une activité est détectée, il convient que la passerelle MG lance la procédure de redémarrage.

Celle-ci implique simplement que la passerelle MG garantisse que le premier message vu par le contrôleur MGC en provenance de cette passerelle sera un message de type ServiceChange informant le contrôleur MGC du redémarrage.

NOTE – La valeur du temps MWD est un paramètre de configuration qui dépend du type de passerelle MG. Le raisonnement suivant peut être utilisé afin de déterminer la valeur de ce temps dans les passerelles résidentielles.

Les contrôleurs MGC sont normalement dimensionnés de façon à absorber la charge de trafic à l'heure de pointe, pendant laquelle 10% des lignes en moyenne seront occupées à faire aboutir des communications dont la durée est normalement de trois minutes. Le traitement d'une communication implique normalement cinq transactions à six transactions de contrôleur MGC entre chaque passerelle MG et le contrôleur MGC. Ce simple calcul montre que le contrôleur MGC est censé traiter cinq transactions à six transactions pour chaque terminaison, toutes les trente minutes en moyenne. En d'autres termes, une transaction environ sera traitée toutes les cinq minutes à six minutes en moyenne dans chaque terminaison, ce qui permet d'estimer qu'une valeur raisonnable du temps MWD pour une passerelle résidentielle sera de dix minutes à douze minutes. En l'absence de configuration explicite, il y a lieu que les passerelles résidentielles adoptent une valeur de 600 s pour le temps MWD.

Le même raisonnement indique que la valeur du temps MWD devrait être beaucoup plus courte pour les passerelles de jonction ou pour les passerelles d'entreprise car elles gèrent un grand nombre de terminaisons dont le taux d'utilisation est aussi beaucoup plus élevé que 10% au cours de l'heure de pointe, une valeur typique étant 60%. Pendant l'heure de pointe, ces terminaisons sont censées contribuer à ce taux d'environ une transaction par minute à la charge du contrôleur MGC. Un algorithme rationnel consiste à rendre la valeur du temps MWD pour chaque terminaison "de jonction" six fois plus court que le temps MWD de chaque passerelle résidentielle et à rendre cette valeur inversement proportionnelle au nombre de terminaisons réinitialisées. Par exemple, le temps MWD devra être réglé à 2,5 s pour une passerelle qui gère une ligne au débit T1 ou à 60 ms pour une passerelle qui gère une ligne au débit T3.

### **9.3 Protection contre l'avalanche de notifications**

Si une accumulation de notifications se produit dans une passerelle MG, soit en raison de difficultés de transmission ou parce que la passerelle MG a reconnu un certain nombre d'événements dans un bref intervalle de temps, il convient que cette passerelle MG envoie les notifications de façon limitée jusqu'à ce que l'accumulation soit résorbée.

## **10 Considérations relatives à la sécurité**

Le présent paragraphe couvre la sécurité lors de l'utilisation du protocole dans un environnement IP.

### **10.1 Protection des connexions de protocole**

Il est évident qu'un mécanisme de sécurité est nécessaire pour empêcher des entités non autorisées d'utiliser le protocole défini dans la présente Recommandation afin d'établir des communications non autorisées ou de perturber des communications autorisées. Le mécanisme de sécurité pour le protocole transporté par des réseaux en protocole IP est IPSec (RFC 2401 à RFC 2411).

L'en-tête AH (RFC 2402) offre l'authentification de l'origine des données, l'intégrité du mode sans connexion et la protection facultative contre le redéfilement de messages transmis entre la passerelle MG et le contrôleur MGC. L'en-tête ESP (RFC 2406) assure au besoin la confidentialité des messages. Par exemple, le service de chiffrement de charge utile par encapsulation IP (ESP) doit être demandé si les descriptions de session sont utilisées pour acheminer des clés de session, comme défini dans le protocole SDP.

Les implémentations du protocole défini dans la présente Recommandation, utilisant l'en-tête du protocole ESP, doivent être conformes à la section 5 du Document RFC 2406 qui définit un ensemble minimal d'algorithmes pour le contrôle d'intégrité et le chiffrement. De même, les implémentations utilisant l'en-tête AH doivent être conformes à la section 5 du Document RFC 2402 qui définit un ensemble minimal d'algorithmes pour le contrôle d'intégrité au moyen de clés manuelles.

Les implémentations devraient utiliser le chiffrement IKE (RFC 2409) pour permettre des options de calcul de clé plus robustes. Les implémentations utilisant le chiffrement IKE devraient prendre en charge l'authentification par signatures à codage RSA et le chiffrement de clés publiques par codage RSA.

## 10.2 Système provisoire d'en-tête AH

L'implémentation du mécanisme IPSec nécessite que l'en-tête AH ou ESP soit inséré immédiatement après l'en-tête IP, ce qui ne peut pas être fait aisément au niveau de l'application. C'est donc un problème de déploiement pour le protocole défini dans la présente Recommandation lorsque l'implémentation du réseau sous-jacent ne prend pas en charge le mécanisme IPSec.

A titre de solution intérimaire, un en-tête AH facultatif est défini à l'intérieur de l'en-tête de protocole H.248.1. Les champs de cet en-tête sont exactement les mêmes que les champs SPI, SEQUENCE NUMBER et DATA qui sont définis dans le Document RFC 2402. La sémantique des champs d'en-tête est la même que celle du "mode de transport" du Document RFC 2402, sauf pour le calcul de la valeur de contrôle d'intégrité (ICV, *integrity check value*). Dans le mécanisme IPSec, la valeur ICV est calculée sur le paquet IP entier, y compris son en-tête IP. Cela empêche la simulation des adresses IP. Pour conserver la même capacité, le calcul de la valeur ICV doit normalement être effectué sur toutes les transactions (concaténées) dans le message préfixé d'un en-tête IP synthétisé à partir d'une adresse IP de source sur 32 bits, d'une adresse de destination sur 32 bits et d'un port de destination UDP de 16 bits codé sous forme de 20 chiffres hexadécimaux. Lorsque le mécanisme intérimaire d'en-tête AH est employé alors que le protocole TCP constitue la couche Transport, l'accès UDP ci-dessus devient l'accès TCP et toutes les autres opérations sont les mêmes.

Les implémentations du protocole H.248.1 doivent implémenter le mécanisme IPSec lorsque le système d'exploitation et le réseau de transport sous-jacents prennent en charge ce mécanisme. Les implémentations du protocole utilisant la version IPv4 doivent implémenter le schéma AH intérimaire. Ce système provisoire ne doit pas cependant être utilisé lorsque la couche Réseau sous-jacente prend en charge le mécanisme IPSec. Les implémentations en version IPv6 sont censées prendre en charge le mécanisme IPSec et ne doivent pas utiliser le système provisoire d'en-tête AH.

Toutes les implémentations du mécanisme intérimaire d'en-tête AH doivent être conformes à la section 5 du Document RFC 2402 qui définit un ensemble minimal d'algorithmes pour le contrôle de l'intégrité au moyen de clés manuelles.

Le système provisoire d'en-tête AH n'assure pas la protection contre les interceptions illicites, ce qui empêche des tierces parties de surveiller les connexions établies par une terminaison donnée. Il n'assure pas non plus de protection contre les attaques par redéfilement. Ces procédures ne protègent pas nécessairement contre les attaques par déni de service de passerelles MG ou de contrôleurs MGC à comportement erroné. Elles assureront cependant une identification de ces entités à comportement erroné, qui devront alors être privées de leur autorisation au moyen de procédures de maintenance.

### 10.3 Protection des connexions de médias

Le protocole permet au contrôleur MGC de fournir aux passerelles MG des "clés de session" pouvant servir à chiffrer les messages audio et à protéger contre les interceptions illicites.

Un problème propre aux réseaux en mode paquet est "l'intervention non contrôlée". Cette attaque peut être exécutée par un aiguillage des paquets médias vers l'adresse IP et l'accès UDP utilisés par une connexion. Si aucune protection n'est implémentée, les paquets doivent être décompressés et les signaux doivent être exécutés du côté "ligne".

Une protection de base contre cette attaque consiste à n'accepter que les paquets provenant de sources connues, en vérifiant par exemple que l'adresse source IP et l'accès source UDP correspondent aux valeurs annoncées dans le descripteur "Remote". Ce procédé présente deux inconvénients: il ralentit l'établissement de la connexion et il peut être contourné par une simulation de source:

- pour activer la protection fondée sur l'adresse, le contrôleur MGC doit toujours obtenir de la passerelle MG de sortie la description de la session distante puis la transmettre à la passerelle MG d'entrée. Cela nécessite au moins un aller et retour dans le réseau et pose un dilemme: soit laisser l'appel s'établir sans attendre la fin de l'aller et retour avec le risque, par exemple, d'une "mutilation" d'une annonce distante, soit attendre la fin de l'aller et retour et accepter des procédures d'établissement d'appel plus lentes;
- la simulation de source n'est effective que si l'attaquant peut obtenir des paires valides d'adresses et d'accès de source et de destination, par exemple par écoute d'une fraction du trafic. Pour combattre la simulation de source, l'on peut essayer de contrôler tous les points d'accès au réseau; mais cela est très difficile à réaliser en pratique.

Une autre solution pour vérifier l'adresse de source consiste à chiffrer et à authentifier les paquets au moyen d'une clé secrète qui est acheminée pendant la procédure d'établissement d'appel. Cela ne ralentira pas celui-ci et apportera une bonne protection contre la simulation de source.

## 11 Interface de commande MG-MGC

L'association de commande entre passerelle MG et contrôleur MGC est créée lors d'un démarrage à froid de la passerelle MG puis annoncée par un message de changement de service. Mais cette association peut être modifiée par des événements ultérieurs, comme des défaillances ou des événements de service manuel.

NOTE – Bien qu'il ne dispose pas d'un mécanisme explicite pour prendre en charge de multiples contrôleurs MGC commandant une passerelle MG physique, le protocole a été conçu de façon à prendre en charge les multiples passerelles MG logiques (contenues dans une même passerelle MG physique) qui peuvent être associées à différents contrôleurs MGC.

### 11.1 Multiples passerelles MG virtuelles

Une passerelle MG physique peut être partitionnée en une ou plusieurs passerelles MG virtuelles (VMG, *virtual MG*). Une passerelle MG virtuelle se compose d'un paquetage de terminaisons physiques statiquement partitionnées ou d'ensembles de terminaisons éphémères. Une terminaison physique est commandée par un seul contrôleur MGC. Le modèle n'exige pas que d'autres ressources -que les terminaisons- soient attribuées statiquement. Le mécanisme d'attribution des terminaisons aux passerelles MG virtuelles est une méthode de gestion hors du domaine d'application de la présente Recommandation. Chacune des passerelles MG virtuelles apparaît au contrôleur MGC comme un client de passerelle MG complet.

Une passerelle MG physique ne peut avoir qu'une seule interface avec le réseau, qui doit toujours être partagée entre passerelles MG virtuelles. Dans un tel cas, la terminaison du côté paquets/cellules est partagée. Il convient toutefois de noter qu'en exploitation, de telles interfaces

nécessitent une instance éphémère de la terminaison à créer dans chaque flux, ce qui simplifie le partage de la terminaison. Ce mécanisme implique toutefois une complication, à savoir que la passerelle MG doit toujours savoir lequel de ses contrôleurs MGC directeurs doit être averti de l'apparition d'un événement à l'interface.

En exploitation normale, la passerelle MG virtuelle sera appelée par le contrôleur MGC à créer des flux dans le réseau (si elle est du côté départ) ou à attendre des flux dans le réseau (si elle est du côté arrivée), de sorte qu'aucune confusion n'apparaîtra. Si cependant un événement imprévu se produit, la passerelle MG virtuelle doit savoir ce qu'il faut faire en ce qui concerne les ressources physiques qu'elle commande.

Si la reprise sur l'événement nécessite la manipulation d'un état de l'interface physique, il convient qu'un seul contrôleur MGC l'effectue. La solution de ces problèmes consiste à autoriser tout contrôleur MGC à créer des descripteurs d'événement qui recevront notification de ces événements. Mais un seul contrôleur MGC pourra avoir l'accès en lecture/écriture aux propriétés de l'interface physique, tous les autres contrôleurs MGC n'ayant qu'un accès en lecture. Le mécanisme de gestion est utilisé pour désigner le contrôleur MGC qui possède la capacité de lecture/écriture et qui est appelé contrôleur MGC maître.

Chaque passerelle MG virtuelle possède sa propre terminaison racine. Le plus souvent, les valeurs des propriétés de cette terminaison racine sont réglables indépendamment par chaque contrôleur MGC. Lorsqu'il ne peut y avoir qu'une seule valeur, ce paramètre est en lecture seulement pour tous les contrôleurs MGC sauf le contrôleur maître.

Une commande ServiceChange ne peut être appliquée qu'à une terminaison ou qu'à un ensemble de terminaisons partitionné(e) selon la passerelle MG virtuelle ou créé(e) (dans le cas de terminaisons éphémères) par cette passerelle MG virtuelle.

## **11.2 Démarrage à froid**

Une passerelle MG est préconfigurée par un mécanisme de gestion extérieur au domaine d'application de la présente Recommandation, avec un contrôleur MGC primaire et (facultativement) une liste ordonnée de contrôleurs MGC secondaires. Lors du démarrage à froid de la passerelle MG, celle-ci enverra à son contrôleur MGC primaire une commande ServiceChange avec méthode de type "redémarrage" (valeur "Restart") au sujet de la terminaison racine. Si le contrôleur MGC accepte la passerelle MG, il envoie une réponse de transaction ne contenant pas de paramètre ServiceChangeMgcID. S'il n'accepte pas l'enregistrement de la passerelle MG, il envoie une réponse de transaction fournissant l'adresse d'un autre contrôleur MGC à approcher, en incluant un paramètre ServiceChangeMgcID.

Si la passerelle MG reçoit une réponse de transaction qui comporte un paramètre ServiceChangeMgcID, elle envoie une commande ServiceChange au contrôleur MGC indiqué dans le paramètre ServiceChangeMgcID. Elle continue ce processus jusqu'à ce qu'elle obtienne un contrôleur MGC qui accepte son enregistrement; sinon, sa recherche de réponse, en provenance du contrôleur MGC primaire ou d'un successeur désigné, échoue. Dans ce dernier cas, la passerelle MG essaie, dans l'ordre, ses contrôleurs MGC secondaires préconfigurés. Si la passerelle MG ne peut pas établir une relation de commande avec un contrôleur MGC quelconque, elle doit attendre pendant un temps aléatoire comme décrit au § 9.2 et ensuite commencer à approcher de nouveau son contrôleur MGC primaire et, le cas échéant, ses contrôleurs MGC secondaires.

Il est possible qu'une réponse à un changement de service avec Restart soit perdu et que la passerelle MG reçoive une commande avant la réception de la réponse au changement de service. La passerelle MG doit émettre un code d'erreur 505 ("Demande de transaction reçue avant d'avoir reçu la réponse au changement de service").

### 11.3 Négociation de version de protocole

Une commande ServiceChange provenant d'une passerelle MG qui est enregistrée auprès du contrôleur MGC doit contenir le numéro de version du protocole pris en charge par la passerelle dans le paramètre ServiceChangeVersion. Quelle que soit la version indiquée dans le paramètre ServiceChangeVersion, le message contenant la commande sera codé comme un message de la version 1. A la réception d'un tel message, si le contrôleur MGC ne prend en charge qu'une version inférieure, le contrôleur MGC doit envoyer une réponse ServiceChangeReply avec la version inférieure et, par la suite, tous les messages entre la passerelle MG et le contrôleur MGC doivent être conformes à la version inférieure du protocole. Si la passerelle MG ne peut pas se conformer et si elle a établi une connexion de transport au contrôleur, il convient que la passerelle ferme cette connexion. Dans tous les cas, il convient qu'elle rejette toutes les demandes subséquentes provenant du contrôleur MGC avec le code d'erreur 406 ("Version non prise en charge").

Si le contrôleur MGC ne prend en charge qu'une ou plusieurs versions supérieures à celle de la passerelle MG, il doit rejeter l'association avec le code d'erreur 406 ("Version non prise en charge").

Si le contrôleur MGC prend en charge la version indiquée par la passerelle MG, il doit se conformer à cette version dans tous les messages subséquents. Dans ce cas, il est facultatif pour le contrôleur MGC de renvoyer une version dans la réponse de changement de service.

La négociation de la version de protocole peut également se produire aux changements de service "Handoff" (transfert) et "Failover" (reprise sur défaillance).

Lors de l'extension du protocole avec de nouvelles versions, il convient de suivre les règles ci-après:

- 1) il convient de ne pas modifier les éléments de protocole existants, à savoir procédures, paramètres, descripteur, propriété, valeur, à moins qu'il ne soit nécessaire de corriger une erreur de protocole ou de changer le fonctionnement du service qui est en cours de prise en charge par le protocole;
- 2) il convient de ne pas changer la sémantique d'une commande, d'un paramètre, d'un descripteur, d'une propriété ou d'une valeur;
- 3) il convient de ne pas modifier les règles établies de formatage et de codage des messages et des paramètres;
- 4) lorsque des éléments d'information sont obsolètes, on peut les marquer comme n'étant pas utilisés. Toutefois, l'identificateur pour cet élément d'information est marqué comme étant réservé. Ainsi, on ne peut pas l'utiliser dans de futures versions.

### 11.4 Défaillance d'une passerelle MG

Si une passerelle MG subit une défaillance mais reste capable d'envoyer un message au contrôleur MGC, elle envoie une commande ServiceChange avec une méthode appropriée (de valeur "Graceful" ou "Forced") puis elle spécifie l'identificateur de terminaison racine. Lorsqu'elle est remise en service, elle envoie une commande ServiceChange avec une méthode ServiceChangeMethod de type "Restart".

L'autorisation donnée au contrôleur MGC d'envoyer des messages en double à deux passerelles MG permet de gérer des paires de passerelles MG ayant la capacité de reprise redondante sur défaillance réciproque. Seule la passerelle MG en service doit accepter ou rejeter des transactions. En cas de reprise sur défaillance, la passerelle MG primaire envoie une commande ServiceChange avec une méthode de type "reprise sur défaillance" et une cause de type "défaillance imminente de passerelle MG". Le contrôleur MGC utilise alors la passerelle MG secondaire comme passerelle active. Lorsque l'état d'erreur est réparé, la passerelle MG en service peut envoyer une commande ServiceChange avec une méthode ServiceChangeMethod de type "Restart".

NOTE – Des passerelles à reprises sur défaillance redondantes nécessitent un transport fiable, parce que le protocole ne donne pas les moyens à une passerelle secondaire utilisant le verrouillage au niveau de la couche Application ALF d'accuser réception des messages envoyés par le contrôleur MGC.

### 11.5 Défaillance d'un contrôleur MGC

Si la passerelle MG détecte une défaillance de son contrôleur MGC, elle tente d'établir un contact avec le prochain contrôleur MGC figurant sur sa liste préconfigurée. Elle commence ses essais par le début de la liste (c'est-à-dire par le contrôleur MGC primaire, à moins que celui-ci ne soit en panne, auquel cas elle commence par son premier contrôleur MGC secondaire). Elle envoie un message ServiceChange avec une méthode de type "reprise sur défaillance" et une raison de type "défaillance imminente du contrôleur MGC". Si elle n'est pas en mesure d'établir une relation de commande avec un quelconque contrôleur MGC, elle attendra pendant un temps de durée aléatoire, tel que décrit au § 9.2 et reprendra ses contacts avec son contrôleur primaire et (si besoin est) ses contrôleurs secondaires. Lorsqu'elle approche les contrôleurs ayant effectué les contrôles précédents, elle envoie un message ServiceChange avec la valeur "Disconnected" (déconnexion) de la méthode ServiceChange.

En cas de défaillance partielle ou pour des raisons de maintenance manuelle, un contrôleur MGC peut souhaiter ordonner à ses passerelles MG contrôlées d'utiliser un autre contrôleur MGC. A cette fin, il envoie à la passerelle MG une commande ServiceChange avec une méthode de type "Handoff" (transfert) ainsi que, dans un paramètre ServiceChangeMgcID, la désignation du contrôleur qui le remplacera. Si une méthode de type "Handoff" est prise en charge, la passerelle enverra au contrôleur MGC ainsi désigné un message ServiceChange avec une méthode de type "Handoff" et une raison de type "changement ordonné par le contrôleur MGC". Si la passerelle ne parvient pas à obtenir une réponse du contrôleur MGC désigné, elle doit se comporter comme si son contrôleur MGC avait subi une défaillance et elle doit commencer à prendre contact avec ses contrôleurs MGC secondaires comme indiqué dans l'alinéa précédent. Si la passerelle ne peut pas établir de relation de commande avec un quelconque contrôleur MGC, elle doit attendre pendant un temps aléatoire comme il est décrit dans le § 9.2 et ensuite commencer à reprendre contact avec ses contrôleurs MGC primaires et, le cas échéant, secondaires.

Aucune recommandation n'est donnée sur la façon dont les contrôleurs MGC impliqués dans le transfert conservent les informations d'état; ce sujet est considéré comme ne s'inscrivant pas dans le domaine d'application de la présente Recommandation. Le contrôleur MGC et la passerelle MG peuvent suivre les étapes ci-après lorsque le transfert se produit. Lorsque le contrôleur MGC entreprend un transfert, celui-ci doit normalement être transparent aux opérations effectuées dans la passerelle MG. Les transactions peuvent être exécutées dans un ordre quelconque et peuvent être en cours lorsque la commande ServiceChange est exécutée. En conséquence, les commandes en cours se poursuivent et des réponses à toutes les commandes provenant du contrôleur MGC initial doivent être envoyées aux adresses de transport d'où celles-ci avaient été envoyées. Si la relation de service avec le contrôleur MGC expéditeur s'est achevée, les réponses devraient être ignorées. La passerelle MG peut recevoir des réponses concernant des transactions en cours envoyées par le nouveau contrôleur MGC. Aucun nouveau message ne doit être envoyé à partir du nouveau contrôleur MGC jusqu'à ce que l'association de commande soit établie. Les demandes de transaction répétées doivent être dirigées vers le nouveau contrôleur MGC. La passerelle doit conserver l'état de toutes les terminaisons et de tous les contextes.

Il est possible que le contrôleur MGC puisse être implémenté de telle façon qu'il soit, en cas de défaillance, remplacé par un nouveau contrôleur MGC en service dont l'identité est la même que celle du contrôleur défaillant. Dans un tel cas, le paramètre ServiceChangeMgcID est spécifié par la précédente valeur et la passerelle MG doit se comporter comme si la valeur avait été modifiée et doit envoyer un message de changement de service (ServiceChange), comme ci-dessus.

Des paires de contrôleurs MGC possédant la capacité de reprise redondante sur défaillance peuvent, au moyen du mécanisme ci-dessus, signaler les reprises sur défaillance aux passerelles MG contrôlées.

## **11.6 Surveillance de l'association de commande MGC-MG**

La surveillance de l'association de commande MGC-MG est essentielle pour les réseaux à disponibilité élevée. Elle peut être obtenue par un contrôle en continu de l'interconnexion (état de la liaison). De nombreux protocoles de transports offrent cette fonctionnalité, de sorte qu'un mécanisme protocolaire obligatoire n'est pas nécessaire.

Dans le cas des transports qui n'offrent pas de surveillance de l'état de la liaison, cette fonctionnalité peut être obtenue dans le protocole H.248 par l'utilisation de la messagerie existante. En l'absence de toute messagerie H.248, le contrôleur MGC peut utiliser une commande AuditValue sur la terminaison racine avec un descripteur "Audit" vide afin de détecter la perte de communications avec la passerelle MG. Celle-ci peut détecter elle-même la perte de communications au moyen du paquetage de temporisateur d'inactivité (Recommandation UIT-T H.248.14).

## **12 Définition des paquetages**

Le mécanisme primaire d'extension est celui des paquetages. Ceux-ci définissent des propriétés supplémentaires qui peuvent apparaître dans des terminaisons et dans des contextes, ainsi que des événements, des signaux et des statistiques qui peuvent apparaître dans des terminaisons.

Les paquetages définis par le groupe IETF apparaîtront dans des commentaires RFC distincts.

Les paquetages définis par l'UIT-T peuvent apparaître dans les Recommandations applicables (par exemple en tant que Recommandations de la sous-série H.248).

- 1) Il y a lieu de spécifier un document public ou un document de forum de normalisation pouvant être référencé comme étant le document décrivant chaque paquetage conforme aux directives ci-dessus.
- 2) Ce document doit spécifier la version du paquetage qu'il décrit.
- 3) Ce document devra être disponible sur un serveur Internet public et devra avoir une localisation URL stable. Ce site devra offrir la possibilité de formuler des observations et les réponses appropriées devront être renvoyées.

### **12.1 Directives pour la définition des paquetages**

Les paquetages définissent des propriétés, des événements, des signaux et des statistiques.

Les paquetages peuvent également définir des codes d'erreur conformément aux directives fournies dans le § 14.2. Il s'agit d'une question de commodité documentaire: la documentation des paquetages est soumise à l'autorité IANA en soutien à l'enregistrement des codes d'erreur. Si un paquetage est modifié, il n'est pas nécessaire de fournir à l'autorité IANA une nouvelle référence de document à l'appui des codes d'erreur, à moins que la description du code d'erreur ne soit elle-même modifiée.

Les noms de toutes les structures ainsi définies doivent se composer de l'identificateur de paquetage (qui désigne celui-ci de manière unique) et de l'identificateur de l'item (désignant de manière unique l'item contenu dans le paquetage). Dans un codage alphanumérique, ces deux identificateurs doivent être séparés par un caractère barre oblique ("/"). Par exemple, l'expression "tonegen/playtone" désigne le codage alphanumérique pour se référer au signal de tonalité d'exécution contenu dans le paquetage de production de tonalités.

Un paquetage doit contenir les sections et mots clés (indiqués en caractères gras) ci-après. Un gabarit définissant les paquetages est contenu dans l'Appendice II.

### 12.1.1 Paquetage

Description globale du paquetage, spécifiant:

**Nom du paquetage:** élément descriptif seulement

**Identificateur du paquetage:** identificateur

**Description:** description du paquetage

**Version:**

une nouvelle version d'un paquetage ne peut qu'ajouter des propriétés, événements, signaux et statistiques supplémentaires ainsi que d'éventuelles valeurs nouvelles pour un paramètre existant qui est décrit dans le paquetage initial. Aucune suppression ou modification n'est autorisée. Une version est un nombre entier dans la plage de 1 à 99.

**Conçu uniquement pour être étendu (facultatif):** oui

cela indique que le paquetage a été expressément conçu pour être étendu par d'autres paquetages, ne devant pas être référencés directement. Par exemple, le paquetage peut n'assurer aucune fonction lui-même ou être en soi dénué de sens. La passerelle MG ne doit pas publier cet identificateur de paquetage lorsqu'elle signale des paquetages.

**Extension (facultative):** identificateur et version du paquetage existant

un paquetage peut constituer une extension d'un paquetage existant. La version du paquetage initial doit être spécifiée. Lorsqu'un paquetage développe un autre paquetage, il ne peut qu'ajouter des propriétés, événements, signaux et statistiques supplémentaires ainsi que d'éventuelles valeurs nouvelles pour un paramètre existant qui est décrit dans le paquetage initial. Un paquetage étendu ne doit pas redéfinir ou recouvrir un identificateur défini dans le paquetage initial et dans les paquetages qu'il a pu développer (niveaux multiples d'extension). Si donc la version 1 du paquetage B développe la version 1 du paquetage A, la version 2 de B ne pourra pas développer la version 2 de A si celle-ci définit un nom qui existe déjà dans la version 1 de B. Si le paquetage ne développe aucun autre paquetage, il doit spécifier la valeur "aucune" (extension).

### 12.1.2 Propriétés

Propriétés définies par le paquetage, spécifiant:

**Nom de la propriété:** élément descriptif seulement

**Identificateur de propriété:** identificateur

**Description:** description de la fonction de la propriété

**Type:** à choisir parmi les suivants:

booléen

chaîne: chaîne en format UTF-8

chaîne d'octets: nombre d'octets. Voir l'Annexe A et le § B.3 en ce qui concerne le codage

entier: entier signé de 4 octets

double: entier signé de 8 octets

caractère: codage de chaque lettre en Unicode UTF-8, éventuellement sur plusieurs octets

énumération: liste de valeurs uniques à choisir

sous-liste: liste de plusieurs valeurs tirées d'une liste. Ce type de sous-liste doit aussi être spécifié. Le type sera choisi parmi les types indiqués dans le présent paragraphe (à

l'exception du type sous-liste). Par exemple, la sous-liste peut être de type sous-liste d'énumération. Le codage des sous-listes est spécifié à l'Annexe A et au § B.2.

**Valeurs possibles:**

un paquetage doit toujours indiquer un ensemble particulier de valeurs ou donner une description de la manière de déterminer des valeurs. Il doit aussi spécifier une valeur par défaut ou le comportement par défaut lorsque la valeur n'est pas présente dans son descripteur. Par exemple, un paquetage peut indiquer que les procédures liées à la propriété sont suspendues lorsque la valeur est omise.

**Valeur par défaut:**

une valeur par défaut (mais non les procédures) peut être spécifiée comme étant susceptible d'être fournie.

**Définie dans:**

le descripteur H.248.1 dans lequel la propriété est définie. Le descripteur "LocalControl" définit les propriétés qui dépendent du flux. Le descripteur "TerminationState" définit les propriétés qui sont indépendantes du flux. Le descripteur "ContextAttribute" définit les propriétés qui affectent le contexte dans son ensemble, c'est-à-dire qui mélangent les propriétés. Ces descripteurs devraient couvrir les cas les plus courants mais il est possible que des propriétés soient définies dans d'autres descripteurs. Les propriétés de contexte doivent toujours être définies dans le descripteur "ContextAttribute".

**Caractéristiques:** en lecture seulement ou en lecture/écriture et (facultativement) globale:

indique si une propriété est en lecture seulement ou en lecture et écriture et si elle est globale. Si l'attribut "global" est omis, la propriété n'est pas globale. Si une propriété est déclarée globale, sa valeur est partagée par toutes les terminaisons réalisant le paquetage. Si une propriété de contexte est déclarée globale, cette propriété est partagée par tous les contextes réalisant le paquetage.

### 12.1.3 Événements

Événements définis par le paquetage, spécifiant:

**Nom de l'événement:** élément descriptif seulement

**Identificateur d'événement:** identificateur

**Description:** description de la fonction de l'événement

**Paramètres du descripteur "Events":**

paramètres utilisés par le contrôleur MGC pour configurer l'événement, contenus dans le descripteur "Events". Voir le § 12.2. S'il n'y a pas de paramètres pour le descripteur "Events", la valeur "Aucun" doit être spécifiée.

**Paramètres de descripteur "ObservedEvents":**

paramètres qui sont renvoyés au contrôleur MGC dans des demandes Notify et dans des réponses à des demandes de commande envoyées par les contrôleurs MGC contrôlant le descripteur "ObservedEvents", et qui se trouvent dans le descripteur "ObservedEvents". Voir le § 12.2. S'il n'y a pas de paramètres pour le descripteur "ObservedEvents", la valeur "Aucun" doit être spécifiée.

### 12.1.4 Signaux

Signaux définis par le paquetage, spécifiant:

**Nom de signal:** élément descriptif seulement

**Identificateur de signal:** identificateur utilisé dans un descripteur "Signals"

**Description:** description de la fonction du signal

**Type de signal:** à choisir parmi les suivants:

OO (commuté)

TO (temporisé)

BR (bref)

NOTE – Le type de signal peut être défini de façon qu'il dépende de la valeur d'un ou de plusieurs paramètres. Le paquetage doit toujours spécifier un type de signal par défaut. Si ce type par défaut est TO, le paquetage doit spécifier une durée par défaut qui peut être assurée. Une durée par défaut est dénuée de sens pour le type BR.

**Durée:** exprimée en centièmes de secondes

**Paramètres additionnels:** voir le § 12.2.

### 12.1.5 Statistiques

Statistiques définies par le paquetage, spécifiant:

**Nom de statistique:** élément descriptif seulement

**Identificateur de statistique:** identificateur utilisé dans un descripteur "Statistics"

**Description:** description de la statistique

**Type:** à choisir parmi les types suivants:

Booléen

Chaîne: chaîne UTF-8

Chaîne d'octets: un certain nombre d'octets. Voir l'Annexe A et le § B.3 pour le codage

Entier: entier signé de 4 octets

Double: entier signé de 8 octets

Caractère: codage Unicode UTF-8 d'une seule lettre, pouvant utiliser plus d'un octet

Énumération: valeur à choisir dans une liste de valeurs possibles

Sous-liste: liste de plusieurs valeurs tirées d'une liste. Ce type de sous-liste doit aussi être spécifié. Le type sera choisi parmi les types indiqués dans le présent paragraphe (à l'exception du type sous-liste). Par exemple, la sous-liste peut être de type sous-liste d'énumération. Le codage des sous-listes est spécifié à l'Annexe A et au § B.2.

**Valeurs possibles:**

un paquetage doit toujours indiquer l'unité de mesure, par exemple millisecondes, paquets, soit ici ou avec le type indiqué ci-dessus, ainsi que toute restriction apportée à l'étendue de valeurs.

**Niveau:** indique si la donnée statistique peut être conservée au niveau de la terminaison, au niveau de flux ou aux deux.

### 12.1.6 Codes d'erreur

Si le paquetage ne définit pas de codes d'erreur, ce paragraphe peut être omis. Sinon, il décrira les codes d'erreur définis par le paquetage, en spécifiant:

**Numéro du code d'erreur:** numéro du code d'erreur

**Nom:** nom de l'erreur

**Définition:** description du code d'erreur

**Texte d'erreur dans le descripteur "Error":**

description du texte à renvoyer dans le descripteur "Error"

**Commentaire:** tout autre commentaire sur l'utilisation du code d'erreur.

### 12.1.7 Procédures

Directives additionnelles concernant l'emploi du paquetage.

## 12.2 Directives de définition des paramètres relatifs aux événements et signaux

**Nom de paramètre:** élément descriptif seulement

**Identificateur de paramètre:** identificateur. L'identificateur alphanumérique des paramètres Events et Signals ne commencera pas par "EPA" et "SPA" respectivement. L'identificateur alphanumérique de paramètre ne sera pas non plus "ST", "Stream", "SY", "SignalType", "DR", "Duration", "NC", "NotifyCompletion", "KA", "KeepActive", "EB", "Embed", "DM", "DigitMap", "SPADI", "SPADirection", "SPARQ" ou "SPARquestID".

**Description:** description de la fonction du paramètre

**Type:** à choisir parmi les suivants:

booléen

chaîne: chaîne d'octets en codage UTF-8

chaîne d'octets: un certain nombre d'octets. Voir l'Annexe A et le § B.3 en ce qui concerne le codage

entier: entier signé de 4 octets

double: entier signé de 8 octets

caractère: codage de chaque lettre en Unicode UTF-8, éventuellement sur plusieurs octets.

énumération: liste de valeurs uniques à choisir.

sous-liste: liste de plusieurs valeurs tirées d'une liste (non prise en compte pour les statistiques). Ce type de sous-liste doit aussi être spécifié. Le type sera choisi parmi les types indiqués dans le présent paragraphe (à l'exception du type sous-liste). Par exemple, la sous-liste peut être de type sous-liste d'énumération. Le codage des sous-listes est spécifié à l'Annexe A et au § B.2.

**Facultatif:** oui/non

Indique si le paramètre peut être omis du signal ou de l'événement.

**Valeurs possibles:**

un paquetage doit toujours indiquer un ensemble particulier de valeurs ou donner une description de la manière de déterminer des valeurs. Il doit aussi spécifier une valeur par défaut ou le comportement par défaut lorsque la valeur n'est pas présente dans son descripteur. Par exemple, un paquetage peut indiquer que les procédures liées au paramètre sont suspendues lorsque la valeur est omise.

**Valeur par défaut:**

une valeur par défaut (mais non les procédures) peut être spécifiée comme étant susceptible d'être fournie.

### 12.3 Identificateurs

Les identificateurs codés en mode alphanumérique doivent être des chaînes de 64 caractères au plus, ne contenant pas d'espaces, commençant par un caractère alphabétique et contenant des caractères alphanumériques et/ou des chiffres avec possibilité d'insertion du caractère spécial de soulignement ("\_").

Les identificateurs codés en mode binaire ont une longueur de 2 octets.

Les valeurs alphanumériques et les valeurs binaires doivent toutes deux être spécifiées pour chaque identificateur, y compris les identificateurs utilisés comme valeurs dans les types énumérés.

## **12.4 Enregistrement de paquetage**

Un paquetage peut être enregistré auprès de l'autorité IANA pour des raisons d'interopérabilité. Voir le paragraphe 14 pour des considérations relatives à l'autorité IANA.

## **13 Définition des profils**

Des profils peuvent être spécifiés afin de mieux définir l'emploi du protocole H.248.1 et les fonctionnalités qui sont prises en charge par une passerelle MG. Un profil ne décrit que les capacités de l'interface H.248 entre MGC et MG. Le profil lui-même spécifie les options associées avec la présente Recommandation qui ont été employées. Par exemple: le transport et les paquetages utilisés pour une application.

Un profil est identifié par un nom (enregistré auprès de l'autorité IANA) et une version. Le nom consistera en une chaîne insensible à la hauteur de casse, d'au plus 64 caractères. La version peut varier de 1 à 99.

Le profil lui-même est un document qui indique les options pour une application particulière. Aucun format n'est fixé pour ce document. Seule est obligatoire la présence d'une section indiquant le nom et la version et d'un résumé du profil.

Les deux premiers éléments ci-dessous sont les seules sections obligatoires d'un profil:

- identification du profil: nom et version du profil envoyé dans la commande de modification du service;
- résumé: description du profil.

L'Appendice III contient un gabarit pour la définition des profils. Il convient de l'utiliser comme base d'une définition de profil.

## **14 Considérations relatives à l'autorité IANA**

### **14.1 Paquetages**

Les prescriptions suivantes doivent être observées pour enregistrer un paquetage auprès de l'autorité IANA:

- 1) pour chaque paquetage, une chaîne nominative unique, un numéro de série unique et un numéro de version doivent être enregistrés. La chaîne nominative est utilisée dans le cadre du codage en mode alphanumérique. Le numéro de série est utilisé dans le cadre du codage en mode binaire. Les numéros de série 0x8000 à 0xFFFF sont réservés à l'usage privé. Le numéro de série 0 est réservé;
- 2) un nom de contact, une adresse de courrier électronique et une adresse postale doivent être spécifiés pour ce contact. Ces informations de contact doivent être mises à jour, le cas échéant, par l'organisation chargée de la définition;
- 3) une référence à un document décrivant le paquetage, qui doit être public, doit être indiquée: Ce document doit spécifier la version du paquetage qu'il décrit.  
Si le document est public, il devra être disponible sur un serveur Internet public et avoir une localisation URL stable. Ce site devra offrir la possibilité de formuler des observations et les réponses appropriées devront être renvoyées;
- 4) les paquetages enregistrés par des organisations de normalisation autres que celles qui sont reconnues doivent avoir une longueur de chaîne nominative au moins égale à 8 caractères;

- 5) tous les autres noms de paquetage sont servis dans l'ordre d'arrivée si toutes les autres conditions sont satisfaites.

#### **14.2 Codes d'erreur**

Les prescriptions suivantes doivent être observées pour enregistrer un code d'erreur auprès de l'autorité IANA:

- 1) un numéro d'erreur et une seule ligne (chaîne de 80 caractères au plus) sont enregistrés pour chaque erreur;
- 2) une description complète des conditions dans lesquelles l'erreur est détectée doit être incluse dans un document disponible au niveau public. Cette description doit être suffisamment claire pour différencier l'erreur de tout autre code d'erreur existant;
- 3) ce document devra être disponible sur un serveur Internet public et devra avoir une localisation URL stable;
- 4) les numéros d'erreur enregistrés par des organismes de normalisation reconnus doivent se composer de 3 ou 4 caractères;
- 5) les numéros d'erreur enregistrés par tous les autres organismes ou individus doivent se composer de 4 caractères;
- 6) un numéro d'erreur ne doit pas être redéfini ni modifié sauf par l'organisation ou l'individu qui l'a défini initialement, ou par ses successeurs ou commissionnaires.

#### **14.3 Raisons de changement de service**

Les prescriptions suivantes doivent être observées pour enregistrer une raison de changement de service auprès de l'autorité IANA:

- 1) un code unique de raison, constitué d'une seule phrase de 80 caractères au plus, est enregistré pour chaque raison;
- 2) une description complète des conditions dans lesquelles la raison est utilisée doit être incluse dans un document disponible au niveau public. Cette description doit être suffisamment claire pour différencier la raison de tout autre code de raison existant;
- 3) ce document devra être disponible sur un serveur Internet public et devra avoir une localisation URL stable.

#### **14.4 Profils**

Les considérations suivantes doivent être satisfaites aux fins de l'enregistrement auprès de l'autorité IANA:

- 1) un seul nom de chaîne et un seul numéro de version (la version peut être omise lorsque le nom de profil contient une structure générique) sont enregistrés pour chaque profil;
- 2) le nom, l'adresse électronique et l'adresse postale d'une personne pouvant être approchée seront indiqués. Ces informations seront mises à jour, si besoin est, par l'organisme les ayant données;
- 3) les noms des profils enregistrés par des organismes autres que les organismes de normalisation agréés doivent comporter six caractères au moins;
- 4) les noms de profils se terminant par une structure générique "\*" doivent être acceptés si les six premiers caractères sont entièrement spécifiés. On suppose que l'organisme ayant reçu le nom de profil générera l'espace de ce nom, associé à la structure générique. L'autorité IANA ne délivrera pas d'autre nom de profil dans le domaine "nom\*".

Tous les autres noms de profil seront attribués selon l'ordre d'arrivée lorsque l'ensemble des autres conditions sont réunies.

## Annexe A

### Codage binaire du protocole

La présente annexe spécifie la syntaxe des messages au moyen de la notation définie dans la Rec. UIT-T X.680, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la numérotation de base*. Les messages doivent être codés pour la transmission en appliquant les règles de codage de base spécifiées dans la Rec. UIT-T X.690, *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives*.

#### A.1 Codage des structures génériques

L'utilisation des structures génériques ALL et CHOOSE est autorisée dans le protocole. Cela permet à un contrôleur MGC de spécifier partiellement des identificateurs de terminaison et à une passerelle MG d'effectuer un choix parmi les valeurs qui sont conformes à la spécification partielle. Les identificateurs de terminaison peuvent coder une hiérarchie de noms. Cette hiérarchie est configurée. Par exemple, un identificateur de terminaison peut se composer d'un faisceau de circuits, d'une jonction locale dans ce faisceau et d'un circuit dans cette jonction. Le remplacement par des structures génériques doit toujours être possible à tous les niveaux. Les alinéas suivants donnent les explications correspondantes.

La description en notation ASN.1 utilise des chaînes d'octets d'une longueur maximale de 8 octets pour les identificateurs de terminaison. C'est-à-dire qu'un identificateur de terminaison se compose d'au plus 64 bits. Un identificateur de terminaison complètement spécifié peut être précédé d'une séquence de champs de remplacement générique, d'une longueur égale à 1 octet. Le bit 7 (de plus fort poids) de cet octet spécifie le type de remplacement générique qui est invoqué: si la valeur du bit est 1, c'est la structure générique ALL qui est utilisée; si la valeur du bit est 0, c'est la structure générique CHOOSE qui est utilisée. Le bit 6 du champ de remplacement générique spécifie si la structure générique appartient à un seul niveau du système hiérarchique de nommage (valeur binaire 0) ou au niveau hiérarchique spécifié dans le champ de remplacement générique plus tous les niveaux inférieurs (valeur binaire 1). Les bits 0 à 5 du champ de remplacement générique spécifient la position binaire dans l'identificateur de terminaison à laquelle commence le remplacement générique.

Quelques exemples illustreront ce modèle. Dans ces exemples, le bit de plus fort poids dans une chaîne de bits apparaît à gauche.

Supposons que les identificateurs de terminaison aient une longueur de trois octets et que chaque octet représente un niveau dans un système hiérarchique de nommage. Un identificateur de terminaison valide sera:

00000001 00011110 01010101.

L'adressage de tous les noms avec le préfixe 00000001 00011110 sera effectué comme suit:

champ de remplacement générique: 10000111

identificateur de terminaison: 00000001 00011110 xxxxxxxx.

Les valeurs des bits représentés par la lettre "x" ne sont pas applicables et doivent être ignorées par le récepteur.

L'indication au récepteur qu'il doit choisir un nom dont le deuxième octet est 00011110 sera effectuée comme suit:

champs de remplacement générique: 00010111 suivi de 00000111

identificateur de terminaison: xxxxxxxx 00011110 xxxxxxxx.

Le premier champ de remplacement générique indique une structure CHOOSE pour le niveau hiérarchique de nommage commençant au bit 23, qui est le niveau le plus élevé dans notre système de nommage théorique. Le deuxième champ de remplacement générique indique une structure CHOOSE pour le niveau hiérarchique de nommage commençant au bit 7, qui est le niveau le moins élevé dans notre système de nommage théorique.

Finalement, un nom remplacé par la structure générique CHOOSE, de niveau hiérarchique de nommage le plus élevé égal à 000000001 est spécifié comme suit:

champ de remplacement générique: 01001111

identificateur de terminaison: 0000001 xxxxxxxx xxxxxxxx.

La valeur binaire 1 à la position binaire 6 du premier octet du champ de remplacement générique indique que le remplacement correspond au niveau spécifié dans la hiérarchie de nommage et à tous les niveaux inférieurs.

Les identificateurs de contexte peuvent également être remplacés par des structures génériques. Dans ce cas toutefois, la spécification de noms partiels n'est pas autorisée. L'identificateur ContextID 0x0 doit être utilisé pour indiquer le contexte NULL, l'identificateur ContextID 0xFFFFFFFF doit être utilisé pour indiquer une structure générique CHOOSE et l'identificateur ContextID 0xFFFFFFFF doit être utilisé pour indiquer une structure générique ALL.

L'identificateur de terminaison 0xFFFFFFFFFFFFFFFF doit être utilisé pour indiquer la terminaison racine.

## A.2 Spécification de syntaxe en notation ASN.1

La spécification en notation ASN.1 de la syntaxe du protocole H.248 est donnée dans le présent paragraphe.

NOTE 1 – Si l'on utilise un mécanisme de transport faisant appel au verrouillage des niveaux d'application, la définition ci-dessous de l'élément **Transaction** change. Se reporter à l'annexe ou à la Recommandation de la sous-série H.248.x définissant le mécanisme de transport pour la définition applicable à ce cas.

NOTE 2 – La présente spécification de syntaxe ne permet pas de faire respecter toutes les restrictions relatives aux inclusions et aux valeurs des éléments. Certaines restrictions supplémentaires sont mentionnées dans des commentaires, tandis que d'autres restrictions apparaissent dans le texte de la présente Recommandation. Ces restrictions supplémentaires font partie du protocole même si leur respect n'est pas assuré par la présente Recommandation.

NOTE 3 – Le module en notation ASN.1 de la présente annexe utilise des types de chaîne d'octets pour le codage des valeurs des paramètres de propriété, de signaux et d'événements et des statistiques. Les types effectifs de ces valeurs varient et sont spécifiés à l'Annexe C ou dans la définition de paquetage applicable.

Une valeur fait d'abord l'objet d'un codage BER en fonction de son type, au moyen du tableau ci-après. Le résultat de ce codage est ensuite codé sous la forme d'une chaîne d'octets en notation ASN.1, en effectuant un "double enveloppement" de la valeur. Le format spécifié à l'Annexe C ou dans le paquetage est lié au codage BER comme indiqué dans le tableau ci-après:

Type spécifié dans le paquetage	Type BER ASN.1
Chaîne	IA5String ou UTF8String (Note 4)
Entier (4 octets)	INTEGER
Double (entier signé à 8 octets)	INTEGER (Note 3)
Caractère (UTF-8) (Note 1)	IA5String
Énumération	ENUMERATED
Booléen	BOOLEAN
Entier non signé (Note 2)	INTEGER (Note 3)
Octet (chaîne)	OCTET STRING
<p>NOTE 1 – Peut avoir plus d'un octet.  NOTE 2 – L'entier non signé est référencé à l'Annexe C.  NOTE 3 – Le codage BER du type INTEGER n'implique pas l'emploi de 4 octets.  NOTE 4 – Le type "Chaîne" devrait être codé comme une chaîne IA5String lorsque la chaîne ne contient que des caractères ASCII, mais comme une chaîne UTF8String si elle contient un quelconque caractère non ASCII.</p>	

Voir le § 8.7/X.690 en ce qui concerne la définition du codage d'une valeur en chaîne d'octets.

```

MEDIA-GATEWAY-CONTROL {itu-t(0) recommendation(0) h(8) h248(248) modules(0)
media-gateway-control(0) version3(3)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
MegacoMessage      ::= SEQUENCE
{
    authHeader      AuthenticationHeader OPTIONAL,
    mess            Message
}

AuthenticationHeader ::= SEQUENCE
{
    secParmIndex    SecurityParmIndex,
    seqNum          SequenceNum,
    ad              AuthData
}

SecurityParmIndex  ::= OCTET STRING(SIZE(4))
SequenceNum        ::= OCTET STRING(SIZE(4))
AuthData           ::= OCTET STRING(SIZE(12..32))
Message            ::= SEQUENCE
{
    version          INTEGER(0..99),
    -- La version du protocole définie ici est égale à 3.
    mId             Mid, -- Nom/adresse de l'expéditeur du message
    messageBody     CHOICE
    {
        messageError ErrorDescriptor,
        transactions SEQUENCE OF Transaction
    },
    ...
}

Mid                ::= CHOICE
{
    ip4Address      IP4Address,
    ip6Address      IP6Address,
    domainName      DomainName,
    deviceName      PathName,

```

```

mtpAddress          OCTET STRING(SIZE(2..4)),
-- Structure d'adressage de mtpAddress:
--   25 - 15          0
--   | PC            | NI |
--   24 - 14 bits    2 bits
-- NOTE - 14 bits sont définis pour utilisation internationale.
-- Deux options nationales existent lorsque le code de point
-- est 16 ou 24 bits.
-- Afin d'aligner l'adresse mtpAddress en octets, les bits de
-- poids fort doivent être codés sous forme de zéros.
...
}

DomainName          ::= SEQUENCE
{
    name             IA5String,
-- Le nom commence par un caractère alphanumérique suivi par une
-- séquence de caractères alphanumériques, de tirets et de
-- points. Deux points ne doivent pas être consécutifs.
    portNumber       INTEGER(0..65535) OPTIONAL
}

IP4Address          ::= SEQUENCE
{
    address          OCTET STRING (SIZE(4)),
    portNumber       INTEGER(0..65535) OPTIONAL
}

IP6Address          ::= SEQUENCE
{
    address          OCTET STRING (SIZE(16)),
    portNumber       INTEGER(0..65535) OPTIONAL
}

PathName           ::= IA5String(SIZE (1..64))
-- Voir le § A.3

Transaction ::= CHOICE
{
    transactionRequest      TransactionRequest,
    transactionPending      TransactionPending,
    transactionReply        TransactionReply,
    transactionResponseAck  TransactionResponseAck,
-- l'utilisation des acquittements de réponse dépend du transport
-- sous-jacent
    ...,
    segmentReply            SegmentReply
}

TransactionId       ::= INTEGER(0..4294967295) -- entier non signé de 32 bits

TransactionRequest ::= SEQUENCE
{
    transactionId      TransactionId,
    actions            SEQUENCE OF ActionRequest,
    ...
}

TransactionPending ::= SEQUENCE
{
    transactionId      TransactionId,
    ...
}

```

```

TransactionReply      ::= SEQUENCE
{
    transactionId      TransactionId,
    immAckRequired     NULL OPTIONAL,
    transactionResult  CHOICE
    {
        transactionError      ErrorDescriptor,
        actionReplies         SEQUENCE OF ActionReply
    },
    ...,
    segmentNumber      SegmentNumber OPTIONAL,
    segmentationComplete NULL OPTIONAL
}

SegmentReply         ::= SEQUENCE
{
    transactionId      TransactionId,
    segmentNumber      SegmentNumber,
    segmentationComplete NULL OPTIONAL,
    ...
}

SegmentNumber        ::= INTEGER(0..65535)

TransactionResponseAck ::= SEQUENCE OF TransactionAck
TransactionAck       ::= SEQUENCE
{
    firstAck           TransactionId,
    lastAck            TransactionId OPTIONAL
}

ErrorDescriptor      ::= SEQUENCE
{
    errorCode          ErrorCode,
    errorText          ErrorText OPTIONAL
}

ErrorCode            ::= INTEGER(0..65535)
-- Voir au § 14 les considérations de l'autorité IANA concernant
-- les codes d'erreur

ErrorText            ::= IA5String

ContextID            ::= INTEGER(0..4294967295)
-- Valeur de contexte NULL: 0
-- Valeur de contexte CHOOSE: 4294967294 (0xFFFFFFFFE)
-- Valeur de contexte ALL: 4294967295 (0xFFFFFFFF)

ActionRequest        ::= SEQUENCE
{
    contextId          ContextID,
    contextRequest     ContextRequest OPTIONAL,
    contextAttrAuditReq ContextAttrAuditRequest OPTIONAL,
    commandRequests   SEQUENCE OF CommandRequest
}

ActionReply          ::= SEQUENCE
{
    contextId          ContextID,
    errorDescriptor    ErrorDescriptor OPTIONAL,
    contextReply       ContextRequest OPTIONAL,
    commandReply       SEQUENCE OF CommandReply
}

```

```

ContextRequest      ::= SEQUENCE
{
    priority          INTEGER(0..15) OPTIONAL,
    emergency         BOOLEAN OPTIONAL,
    topologyReq      SEQUENCE OF TopologyRequest OPTIONAL,
    ...,
    iepscallind      BOOLEAN OPTIONAL,
    contextProp      SEQUENCE OF PropertyParm OPTIONAL,
    contextList      SEQUENCE OF ContextID OPTIONAL
}
-- Lors du renvoi d'une liste de contexte contextList, l'identificateur
-- contextId contenu dans la production ActionReply renverra l'identificateur
-- de contexte issu de la requête d'action ActionRequest associée.

ContextAttrAuditRequest ::= SEQUENCE
{
    topology          NULL OPTIONAL,
    emergency         NULL OPTIONAL,
    priority          NULL OPTIONAL,
    ...,
    iepscallind      NULL OPTIONAL,
    contextPropAud   SEQUENCE OF IndAudPropertyParm OPTIONAL,
    selectpriority   INTEGER(0..15) OPTIONAL,
    -- afin de sélectionner la priorité indiquée
    selectemergency  BOOLEAN OPTIONAL,
    -- afin de sélectionner si l'urgence est activée/non activée (T/F)
    selectiepscalle  BOOLEAN OPTIONAL,
    -- afin de sélectionner si le plan IEPS est activé/non activé (T/F)
    selectLogic      SelectLogic OPTIONAL -- par défaut AND
}

SelectLogic        ::= CHOICE
{
    andAUDITSelect  NULL,      -- toutes les conditions de filtrage sont
    -- satisfaites
    orAUDITSelect   NULL,      -- au moins une condition de filtrage est
    -- satisfaite
    ...
}

CommandRequest     ::= SEQUENCE
{
    command          Command,
    optional         NULL OPTIONAL,
    wildcardReturn   NULL OPTIONAL,
    ...
}

Command           ::= CHOICE
{
    addReq          AmmRequest,
    moveReq         AmmRequest,
    modReq          AmmRequest,
    -- Les requêtes Add, Move, Modify ont les mêmes paramètres
    subtractReq     SubtractRequest,
    auditCapRequest AuditRequest,
    auditValueRequest AuditRequest,
    notifyReq       NotifyRequest,
    serviceChangeReq ServiceChangeRequest,
    ...
}

CommandReply       ::= CHOICE
{

```

```

    addReply             AmmsReply,
    moveReply           AmmsReply,
    modReply            AmmsReply,
    subtractReply       AmmsReply,
    -- Les réponses Add, Move, Modify, Subtract ont les mêmes paramètres
    auditCapReply       AuditReply,
    auditValueReply     AuditReply,
    notifyReply         NotifyReply,
    serviceChangeReply  ServiceChangeReply,
    ...
}
TopologyRequest ::= SEQUENCE
{
    terminationFrom      TerminationID,
    terminationTo        TerminationID,
    topologyDirection    ENUMERATED
    {
        bothway(0),
        isolate(1),
        oneway(2)
    },
    ...,
    streamID             StreamID OPTIONAL,
    topologyDirectionExtension  ENUMERATED
    {
        onewayexternal(0),
        onewayboth(1),
        ...
    }
}

AmmRequest ::= SEQUENCE
{
    terminationID        TerminationIDList,
    descriptors          SEQUENCE OF AmmDescriptor,
    -- Au plus un descripteur de chaque type (Voir AmmDescriptor)
    -- est permis dans la séquence.
    ...
}

AmmDescriptor ::= CHOICE
{
    mediaDescriptor      MediaDescriptor,
    modemDescriptor      ModemDescriptor,
    muxDescriptor        MuxDescriptor,
    eventsDescriptor     EventsDescriptor,
    eventBufferDescriptor EventBufferDescriptor,
    signalsDescriptor    SignalsDescriptor,
    digitMapDescriptor   DigitMapDescriptor,
    auditDescriptor      AuditDescriptor,
    ...,
    statisticsDescriptor StatisticsDescriptor
}

AmmsReply ::= SEQUENCE
{
    terminationID        TerminationIDList,
    terminationAudit     TerminationAudit OPTIONAL,
    ...
}

SubtractRequest ::= SEQUENCE
{
    terminationID        TerminationIDList,

```

```

    auditDescriptor      AuditDescriptor OPTIONAL,
    ...
}

AuditRequest           ::= SEQUENCE
{
    terminationID       TerminationID,
    auditDescriptor     AuditDescriptor,
    ...,
    terminationIDList   TerminationIDList OPTIONAL
}
-- l'identificateur terminationID doit contenir la première terminaison
-- de la liste lors de l'utilisation de la production terminationIDList
-- dans la commande AuditRequest

AuditReply             ::= CHOICE
{
    contextAuditResult  TerminationIDList,
    error               ErrorDescriptor,
    auditResult         AuditResult,
    ...,
    auditResultTermList TermListAuditResult
}
AuditResult           ::= SEQUENCE
{
    terminationID       TerminationID,
    terminationAuditResult TerminationAudit
}

TermListAuditResult   ::= SEQUENCE
{
    terminationIDList   TerminationIDList,
    terminationAuditResult TerminationAudit,
    ...
}

TerminationAudit      ::= SEQUENCE OF AuditReturnParameter

AuditReturnParameter  ::= CHOICE
{
    errorDescriptor     ErrorDescriptor,
    mediaDescriptor     MediaDescriptor,
    modemDescriptor     ModemDescriptor,
    muxDescriptor       MuxDescriptor,
    eventsDescriptor    EventsDescriptor,
    eventBufferDescriptor EventBufferDescriptor,
    signalsDescriptor   SignalsDescriptor,
    digitMapDescriptor  DigitMapDescriptor,
    observedEventsDescriptor ObservedEventsDescriptor,
    statisticsDescriptor StatisticsDescriptor,
    packagesDescriptor  PackagesDescriptor,
    emptyDescriptors    AuditDescriptor,
    ...
}

AuditDescriptor      ::= SEQUENCE
{
    auditToken          BIT STRING
    {
        muxToken(0),
        modemToken(1),
        mediaToken(2),
        eventsToken(3),
        signalsToken(4),

```

```

        digitMapToken(5),
        statsToken(6),
        observedEventsToken(7),
        packagesToken(8),
        eventBufferToken(9)
    } OPTIONAL,
    ...,
    auditPropertyToken      SEQUENCE OF IndAuditParameter OPTIONAL
}

IndAuditParameter ::= CHOICE
{
    indaudmediaDescriptor      IndAudMediaDescriptor,
    indaudeventsDescriptor     IndAudEventsDescriptor,
    indaudeventBufferDescriptor IndAudEventBufferDescriptor,
    indaudsignalsDescriptor    IndAudSignalsDescriptor,
    indauidigitMapDescriptor    IndAudDigitMapDescriptor,
    indaudstatisticsDescriptor  IndAudStatisticsDescriptor,
    indaudpackagesDescriptor    IndAudPackagesDescriptor,
    ...
}

IndAudMediaDescriptor ::= SEQUENCE
{
    termStateDescr            IndAudTerminationStateDescriptor OPTIONAL,
    streams                   CHOICE
    {
        oneStream              IndAudStreamParms,
        multiStream            SEQUENCE OF IndAudStreamDescriptor
    } OPTIONAL,
    ...
}

IndAudStreamDescriptor ::= SEQUENCE
{
    streamID                  StreamID,
    streamParms               IndAudStreamParms
}

IndAudStreamParms ::= SEQUENCE
{
    localControlDescriptor    IndAudLocalControlDescriptor OPTIONAL,
    localDescriptor           IndAudLocalRemoteDescriptor OPTIONAL,
    remoteDescriptor          IndAudLocalRemoteDescriptor OPTIONAL,
    ...,
    statisticsDescriptor      IndAudStatisticsDescriptor OPTIONAL
}

IndAudLocalControlDescriptor ::= SEQUENCE
{
    streamMode                NULL OPTIONAL,
    reserveValue              NULL OPTIONAL,
    reserveGroup              NULL OPTIONAL,
    propertyParms             SEQUENCE OF IndAudPropertyParm OPTIONAL,
    ...,
    streamModeSel             StreamMode OPTIONAL
}
-- les deux paramètres streamMode et streamModeSel ne doivent
-- pas être présents en même temps
-- s'ils sont présents tous les deux, seul streamModeSel
-- doit être pris en compte

IndAudPropertyParm ::= SEQUENCE
{

```

```

    name                PkgdName,
    ...,
    propertyParms       PropertyParm OPTIONAL
}

-- afin de sélectionner des valeurs fondées sur une propriété,
-- la logique de sélection ET/OU est spécifiée au niveau du contexte

IndAudLocalRemoteDescriptor ::= SEQUENCE
{
    propGroupID         INTEGER(0..65535) OPTIONAL,
    propGrps            IndAudPropertyGroup,
    ...
}

IndAudPropertyGroup     ::= SEQUENCE OF IndAudPropertyParm

IndAudTerminationStateDescriptor ::= SEQUENCE
{
    propertyParms       SEQUENCE OF IndAudPropertyParm,
    eventBufferControl  NULL OPTIONAL,
    serviceState        NULL OPTIONAL,
    ...,
    serviceStateSel     ServiceState OPTIONAL
}

-- les deux paramètres serviceState et serviceStateSel ne doivent pas être
-- présents en même temps. S'ils sont présents tous les deux, seul
-- serviceStateSel doit être pris en compte

IndAudEventsDescriptor  ::= SEQUENCE
{
    requestID           RequestID OPTIONAL,
    pkgdName            PkgdName,
    streamID            StreamID OPTIONAL,
    ...
}

IndAudEventBufferDescriptor ::= SEQUENCE
{
    eventName           PkgdName,
    streamID            StreamID OPTIONAL,
    ...
}

IndAudSignalsDescriptor ::= CHOICE
{
    signal              IndAudSignal,
    seqSigList          IndAudSeqSigList,
    ...
}

IndAudSeqSigList       ::= SEQUENCE
{
    id                  INTEGER(0..65535),
    signalList          IndAudSignal OPTIONAL
}

IndAudSignal           ::= SEQUENCE
{
    signalName          PkgdName,
    streamID            StreamID OPTIONAL,
    ...,
    signalRequestID     RequestID OPTIONAL
}

```

```

IndAudDigitMapDescriptor ::= SEQUENCE
{
    digitMapName          DigitMapName OPTIONAL
}

IndAudStatisticsDescriptor ::= SEQUENCE
{
    statName              PkgdName
}

IndAudPackagesDescriptor ::= SEQUENCE
{
    packageName           Name,
    packageVersion        INTEGER(0..99),
    ...
}

NotifyRequest           ::= SEQUENCE
{
    terminationID         TerminationIDList,
    observedEventsDescriptor ObservedEventsDescriptor,
    errorDescriptor       ErrorDescriptor OPTIONAL,
    ...
}

NotifyReply             ::= SEQUENCE
{
    terminationID         TerminationIDList,
    errorDescriptor       ErrorDescriptor OPTIONAL,
    ...
}

ObservedEventsDescriptor ::= SEQUENCE
{
    requestId             RequestID,
    observedEventList     SEQUENCE OF ObservedEvent
}

ObservedEvent          ::= SEQUENCE
{
    eventName             EventName,
    streamID              StreamID OPTIONAL,
    eventParList          SEQUENCE OF EventParameter,
    timeNotation          TimeNotation OPTIONAL,
    ...
}

EventName              ::= PkgdName

EventParameter         ::= SEQUENCE
{
    eventParameterName    Name,
    value                 Value,
    -- Concernant l'utilisation de extraInfo, voir le commentaire
    -- relatif à PropertyParm
    extraInfo             CHOICE
    {
        relation          Relation,
        range              BOOLEAN,
        sublist            BOOLEAN
    } OPTIONAL,
    ...
}

```

```

ServiceChangeRequest ::= SEQUENCE
{
    terminationID          TerminationIDList,
    serviceChangeParms    ServiceChangeParm,
    ...
}

ServiceChangeReply ::= SEQUENCE
{
    terminationID          TerminationIDList,
    serviceChangeResult    ServiceChangeResult,
    ...
}

-- Pour ServiceChangeResult, aucun paramètre n'est obligatoire.
-- D'où la distinction entre ServiceChangeParm et ServiceChangeResParm
ServiceChangeResult ::= CHOICE
{
    errorDescriptor        ErrorDescriptor,
    serviceChangeResParms  ServiceChangeResParm
}

WildcardField          ::= OCTET STRING(SIZE(1))

TerminationID          ::= SEQUENCE
{
    wildcard              SEQUENCE OF WildcardField,
    id                    OCTET STRING(SIZE(1..8)),
    ...
}

-- Voir le § A.1 pour l'explication du mécanisme de remplacement
-- par structure générique.
-- TerminationID 0xFFFFFFFFFFFFFFF indique la terminaison racine Root.

TerminationIDList      ::= SEQUENCE OF TerminationID

MediaDescriptor        ::= SEQUENCE
{
    termStateDescr        TerminationStateDescriptor OPTIONAL,
    streams                CHOICE
    {
        oneStream          StreamParms,
        multiStream        SEQUENCE OF StreamDescriptor
    } OPTIONAL,
    ...
}

StreamDescriptor        ::= SEQUENCE
{
    streamID              StreamID,
    streamParms           StreamParms
}

StreamParms            ::= SEQUENCE
{
    localControlDescriptor LocalControlDescriptor OPTIONAL,
    localDescriptor        LocalRemoteDescriptor OPTIONAL,
    remoteDescriptor        LocalRemoteDescriptor OPTIONAL,
    ...,
    statisticsDescriptor    StatisticsDescriptor OPTIONAL
}

LocalControlDescriptor ::= SEQUENCE

```

```

{
    streamMode          StreamMode OPTIONAL,
    reserveValue        BOOLEAN OPTIONAL,
    reserveGroup        BOOLEAN OPTIONAL,
    propertyParms       SEQUENCE OF PropertyParm,
    ...
}

```

```

StreamMode ::= ENUMERATED
{
    sendOnly(0),
    recvOnly(1),
    sendRecv(2),
    inactive(3),
    loopBack(4),
    ...
}

```

-- Dans PropertyParm, la valeur est une chaîne d'octets de type SEQUENCE OF.  
-- Lorsqu'elle est envoyée par un contrôleur MGC, l'interprétation est la  
-- suivante:  
-- une séquence vide désigne la structure CHOOSE;  
-- une séquence à 1 seul élément spécifie une valeur.  
-- Si le champ sublist n'est pas sélectionné, une séquence plus longue signifie  
-- "choisir une des valeurs" (c.-à-d. valeur1 OU valeur2 OU ...)  
-- Si le champ sublist est sélectionné,  
-- une séquence de plus d'un seul élément code la valeur  
-- d'une propriété à liste de valeurs (c.-à-d. valeur1 ET valeur2 ET ...).  
-- Le champ de relation ne peut être choisi que si la séquence de valeur a la  
-- longueur 1. Il indique que la passerelle média doit choisir une valeur pour  
-- la propriété. Par exemple, x > 3 indique que la passerelle média doit choisir  
-- n'importe quelle valeur supérieure à 3 pour la propriété x.  
-- Le champ d'étendue ne peut être choisi que si la séquence de valeur  
-- a la longueur 2. Il indique que la passerelle MG doit choisir une valeur  
-- dans l'étendue comprise entre le premier et le dernier octet de la  
-- séquence, y compris les valeurs des bornes.  
-- Lorsque la valeur est envoyée par la passerelle MG, seules les réponses à une  
-- demande AuditCapability peuvent contenir plusieurs valeurs, une plage ou  
-- un champ de relation.

```

PropertyParm ::= SEQUENCE
{
    name          PkgdName,
    value         SEQUENCE OF OCTET STRING,
    extraInfo     CHOICE
    {
        relation  Relation,
        range     BOOLEAN,
        sublist   BOOLEAN
    } OPTIONAL,
    ...
}

```

```

Name ::= OCTET STRING(SIZE(2))

```

```

PkgdName ::= OCTET STRING(SIZE(4))

```

-- représente un nom de paquetage (2 octets) plus une propriété,  
-- un événement, des noms de signal ou un identificateur de statistiques  
-- (2 octets).  
-- Pour remplacer génériquement un paquetage, utiliser 0xFFFF pour les  
-- 2 premiers octets.  
-- La structure CHOOSE n'est pas autorisée.  
-- Pour faire référence à l'étiquette de propriété initiale spécifiée dans  
-- l'Annexe C, utiliser 0x0000 dans les 2 premiers octets.

-- Pour désigner génériquement un identificateur de propriété, d'événement,  
 -- de signal ou de statistique, utiliser 0xFFFF pour les deux derniers octets,  
 -- la structure CHOOSE n'est pas permise.  
 -- Le remplacement du nom d'un paquetage par une structure générique n'est  
 -- permis que si un identificateur de propriété, d'événement, de signal ou de  
 -- statistique est également remplacé par une structure générique.

```

Relation          ::= ENUMERATED
{
    greaterThan(0),
    smallerThan(1),
    unequalTo(2),
    ...}

LocalRemoteDescriptor ::= SEQUENCE
{
    propGrps          SEQUENCE OF PropertyGroup,
    ...
}

PropertyGroup      ::= SEQUENCE OF PropertyParm

TerminationStateDescriptor ::= SEQUENCE
{
    propertyParms      SEQUENCE OF PropertyParm,
    eventBufferControl EventBufferControl OPTIONAL,
    serviceState       ServiceState OPTIONAL,
    ...
}

EventBufferControl ::= ENUMERATED
{
    off(0),
    lockStep(1),
    ...
}

ServiceState       ::= ENUMERATED
{
    test(0),
    outOfSvc(1),
    inSvc(2),
    ...
}

MuxDescriptor      ::= SEQUENCE
{
    muxType           MuxType,
    termList          SEQUENCE OF TerminationID,
    nonStandardData   NonStandardData OPTIONAL,
    ...
}

MuxType            ::= ENUMERATED
{
    h221(0),
    h223(1),
    h226(2),
    v76(3),
    ...,
    nx64k(4)
}

StreamID           ::= INTEGER(0..65535) -- entier non signé de 16 bits
  
```

```

EventsDescriptor      ::= SEQUENCE
{
    requestID          RequestID OPTIONAL,
    -- RequestID doit être présent si eventList
    -- n'est pas vide
    eventList          SEQUENCE OF RequestedEvent,
    ...
}

RequestedEvent        ::= SEQUENCE
{
    pkgdName           PkgdName,
    streamID           StreamID OPTIONAL,
    eventAction        RequestedActions OPTIONAL,
    evParList          SEQUENCE OF EventParameter,
    ...
}

RegulatedEmbeddedDescriptor ::= SEQUENCE
{
    secondEvent        SecondEventsDescriptor OPTIONAL,
    signalsDescriptor  SignalsDescriptor OPTIONAL,
    ...
}

NotifyBehaviour      ::= CHOICE
{
    notifyImmediate    NULL,
    notifyRegulated    RegulatedEmbeddedDescriptor,
    neverNotify         NULL,
    ...
}

RequestedActions     ::= SEQUENCE
{
    keepActive         BOOLEAN OPTIONAL,
    eventDM            EventDM OPTIONAL,
    secondEvent        SecondEventsDescriptor OPTIONAL,
    signalsDescriptor  SignalsDescriptor OPTIONAL,
    ...,
    notifyBehaviour    NotifyBehaviour OPTIONAL,
    resetEventsDescriptor NULL OPTIONAL
}

EventDM              ::= CHOICE
{
    digitMapName       DigitMapName,
    digitMapValue      DigitMapValue
}

SecondEventsDescriptor ::= SEQUENCE
{
    requestID          RequestID OPTIONAL,
    eventList          SEQUENCE OF SecondRequestedEvent,
    ...
}

SecondRequestedEvent ::= SEQUENCE
{
    pkgdName           PkgdName,
    streamID           StreamID OPTIONAL,
    eventAction        SecondRequestedActions OPTIONAL,
    evParList          SEQUENCE OF EventParameter,
}

```

```

    ...
}
SecondRequestedActions ::= SEQUENCE
{
    keepActive          BOOLEAN OPTIONAL,
    eventDM             EventDM OPTIONAL,
    signalsDescriptor   SignalsDescriptor OPTIONAL,
    ...,
    notifyBehaviour     NotifyBehaviour OPTIONAL,
    resetEventsDescriptor NULL OPTIONAL
}

EventBufferDescriptor ::= SEQUENCE OF EventSpec

EventSpec ::= SEQUENCE
{
    eventName           EventName,
    streamID            StreamID OPTIONAL,
    eventParList        SEQUENCE OF EventParameter,
    ...
}

SignalsDescriptor ::= SEQUENCE OF SignalRequest

SignalRequest ::= CHOICE
{
    signal              Signal,
    seqSigList          SeqSigList,
    ...
}

SeqSigList ::= SEQUENCE
{
    id                  INTEGER(0..65535),
    signalList          SEQUENCE OF Signal
}

Signal ::= SEQUENCE
{
    signalName          SignalName,
    streamID            StreamID OPTIONAL,
    sigType             SignalType OPTIONAL,
    duration            INTEGER(0..65535) OPTIONAL,
    notifyCompletion    NotifyCompletion OPTIONAL,
    keepActive          BOOLEAN OPTIONAL,
    sigParList          SEQUENCE OF SigParameter,
    ...,
    direction           SignalDirection OPTIONAL,
    requestID           RequestID OPTIONAL,
    intersigDelay       INTEGER(0..65535) OPTIONAL
}

SignalType ::= ENUMERATED
{
    brief(0),
    onOff(1),
    timeOut(2),
    ...
}

SignalDirection ::= ENUMERATED
{
    internal(0),
    external(1),

```

```

    both(2),
    ...
}

SignalName ::= PkgdName

NotifyCompletion ::= BIT STRING
{
    onTimeOut(0), onInterruptByEvent(1),
    onInterruptByNewSignalDescr(2), otherReason(3), onIteration(4)
}

SigParameter ::= SEQUENCE
{
    sigParameterName Name,
    value Value,
    -- Pour l'utilisation de extraInfo, voir le commentaire
    -- relatif à PropertyParm
    extraInfo CHOICE
    {
        relation Relation,
        range BOOLEAN,
        sublist BOOLEAN
    } OPTIONAL,
    ...
}

-- Pour une réponse AuditCapReply avec tous les événements, l'identificateur
-- RequestID doit être ALL.
-- ALL est représenté par 0xffffffff.
RequestID ::= INTEGER(0..4294967295) -- entier non signé de 32 bits

ModemDescriptor ::= SEQUENCE
{
    mtl SEQUENCE OF ModemType,
    mpl SEQUENCE OF PropertyParm,
    nonStandardData NonStandardData OPTIONAL
}

ModemType ::= ENUMERATED
{
    v18(0),
    v22(1),
    v22bis(2),
    v32(3),
    v32bis(4),
    v34(5),
    v90(6),
    v91(7),
    synchISDN(8),
    ...
}

DigitMapDescriptor ::= SEQUENCE
{
    digitMapName DigitMapName OPTIONAL,
    digitMapValue DigitMapValue OPTIONAL
}

DigitMapName ::= Name

DigitMapValue ::= SEQUENCE
{
    startTimer INTEGER(0..99) OPTIONAL,

```

```

shortTimer          INTEGER(0..99) OPTIONAL,
longTimer           INTEGER(0..99) OPTIONAL,
digitMapBody       IA5String,
-- Les unités sont exprimées en secondes pour la temporisation de
-- début et les temporisations courtes/longues; et en centaines de
-- millisecondes pour la temporisation de durée. Les intervalles
-- courts/longs s'étendent donc de 1 à 99 s, tandis que ceux de
-- durée s'étendent de 100 ms à 9,9 s.
-- Voir le § A.3 pour l'explication de la syntaxe d'un script de
-- numérotation DigitMap
...,
durationTimer      INTEGER (0..99) OPTIONAL
}

ServiceChangeParm ::= SEQUENCE
{
    serviceChangeMethod      ServiceChangeMethod,
    serviceChangeAddress     ServiceChangeAddress OPTIONAL,
    serviceChangeVersion     INTEGER(0..99) OPTIONAL,
    serviceChangeProfile     ServiceChangeProfile OPTIONAL,
    serviceChangeReason      Value,
-- Un motif serviceChangeReason comporte un code numérique
-- indiquant la raison et une description alphanumérique
-- facultative. Le motif serviceChangeReason doit être une
-- chaîne comportant un code décimal indiquant la raison, suivi
-- facultativement d'un caractère d'espace et d'une
-- chaîne alphanumérique descriptive. Cette chaîne fait d'abord
-- l'objet d'un codage BER de forme IA5String. Le résultat de ce
-- codage BER est ensuite codé comme étant de type "ASN.1 OCTET
-- STRING", "à double enveloppement" de la valeur, comme
-- cela a été fait pour les éléments du paquetage
    serviceChangeDelay      INTEGER(0..4294967295) OPTIONAL,
-- entier non signé de 32 bits
    serviceChangeMgcId      Mid OPTIONAL,
    timeStamp               TimeNotation OPTIONAL,
    nonStandardData         NonStandardData OPTIONAL,
    ...,
    serviceChangeInfo       AuditDescriptor OPTIONAL,
    serviceChangeIncompleteFlag NULL OPTIONAL
}

ServiceChangeAddress ::= CHOICE
{
    portNumber              INTEGER(0..65535), -- numéro d'accès TCP/UDP
    ip4Address              IP4Address,
    ip6Address              IP6Address,
    domainName              DomainName,
    deviceName              PathName,
    mtpAddress              OCTET STRING(SIZE(2..4)),
    ...
}

ServiceChangeResParm ::= SEQUENCE
{
    serviceChangeMgcId      Mid OPTIONAL,
    serviceChangeAddress     ServiceChangeAddress OPTIONAL,
    serviceChangeVersion     INTEGER(0..99) OPTIONAL,
    serviceChangeProfile     ServiceChangeProfile OPTIONAL,
    timeStamp               TimeNotation OPTIONAL,
    ...
}

ServiceChangeMethod ::= ENUMERATED
{

```

```

    failover(0),
    forced(1),
    graceful(2),
    restart(3),
    disconnected(4),
    handOff(5),
    ...
}

ServiceChangeProfile ::= SEQUENCE
{
    profileName          IA5String(SIZE (1..67))
    -- 64 caractères pour le nom, 1 pour "/", 2 pour la version
    -- afin de correspondre au formalisme ABNF.
}

PackagesDescriptor ::= SEQUENCE OF PackagesItem

PackagesItem ::= SEQUENCE
{
    packageName          Name,
    packageVersion      INTEGER(0..99),
    ...
}

StatisticsDescriptor ::= SEQUENCE OF StatisticsParameter

StatisticsParameter ::= SEQUENCE
{
    statName             PkgdName,
    statValue            Value OPTIONAL
}
-- Si la statistique se compose d'une sous-liste, il y aura
-- plusieurs chaînes d'octets dans la valeur statValue.

NonStandardData ::= SEQUENCE
{
    nonStandardIdentifier NonStandardIdentifier,
    data                  OCTET STRING
}

NonStandardIdentifier ::= CHOICE
{
    object                OBJECT IDENTIFIER,
    h221NonStandard      H221NonStandard,
    experimental         IA5String(SIZE(8)),
    -- les deux premiers caractères devraient être "X-" ou "X+"
    ...
}

H221NonStandard ::= SEQUENCE
{
    t35CountryCode1      INTEGER(0..255),
    t35CountryCode2      INTEGER(0..255), -- pays, selon T.35
    t35Extension         INTEGER(0..255), -- attribué au niveau
    -- national
    manufacturerCode     INTEGER(0..65535), -- attribué au niveau
    -- national
    ...
}

TimeNotation ::= SEQUENCE
{
    date                 IA5String(SIZE(8)), -- format yyyyymmdd
    time                 IA5String(SIZE(8)) -- format hhmmssss
}

```

```

    -- selon l'ISO 8601:2004
}
Value ::= SEQUENCE OF OCTET STRING
END

```

### A.3 Noms des scripts de numérotation et des conduits

Du point de vue syntaxique, les scripts de numérotation "DigitMap" sont des chaînes soumises à des contraintes de syntaxe. La syntaxe des scripts de numérotation DigitMap valides est spécifiée en formalisme ABNF (RFC 2234). La syntaxe de script de numérotation DigitMap présentée dans le présent paragraphe n'a valeur que d'illustration. La définition de la structure digitMap dans l'Annexe B a priorité en cas de divergence entre les deux définitions.

```

digitMap          = (digitString / LWSP "(" LWSP digitStringList LWSP
                    ")" LWSP)
digitStringList   = digitString *(LWSP "|" LWSP digitString)
digitString       = 1*(digitStringElement)
digitStringElement = digitPosition [DOT]
digitPosition     = digitMapLetter / digitMapRange
digitMapRange     = ("x" / (LWSP "[" LWSP digitLetter LWSP "]" LWSP))
digitLetter       = *((DIGIT "-" DIGIT) / digitMapLetter)
digitMapLetter    = DIGIT                ; Symboles d'événement de base
                    / %x41-4B / %x61-6B ; a-k, A-K
                    / "L" / "S" / "T"   ; Temporisateurs entre événements;
                    ; (longs, courts, start)
                    / "Z"               ; modificateur; de longue durée
modifier
DOT               = %x2E                ; "." (point)
SP                = %x20                ; espace
HTAB              = %x09                ; tabulateur horizontal
CR                = %x0D                ; retour de chariot
LF                = %x0A                ; interligne
LWSP              = *(WSP / COMMENT / EOL)
EOL               = (CR [LF] / LF)
WSP               = SP / HTAB           ; espace blanc
SafeChar          = DIGIT / ALPHA / "+" / "-" / "&" /
                    "!" / " " / "/" / " " / "?" / "@" /
                    "^" / "`" / "~" / "*" / "$" / "\" /
                    "(" / ")" / "%" / "|" / "."
RestChar          = ";" / "[" / "]" / "{" / "}" / ":" / "," / "#" /
                    "<" / ">" / "="
DIGIT             = %x30-39             ; 0-9

```

ALPHA = %x41-5A / %x61-7A ; A-Z / a-z

Un nom de conduit est également une chaîne soumise à des contraintes syntaxiques. La production ABNF qui définit cette chaîne est reprise de l'Annexe B.

; La longueur totale de pathNAME ne doit jamais dépasser 64 caractères.

```
pathname = ["*"] NAME *("/" / "*" / ALPHA / DIGIT / "_" / "$" )
          ["@" pathDomainName ]
```

; Le formalisme ABNF permet deux ou plusieurs points "."  
; consécutifs, bien que cela n'ait pas de signification dans un nom  
; de domaine.

```
pathDomainName = (ALPHA / DIGIT / "*" ) *63(ALPHA / DIGIT / "-" / "*" / ".")
```

```
NAME = ALPHA *63(ALPHA / DIGIT / "_")
```

## Annexe B

### Codage alphanumérique du protocole

#### B.1 Codage des structures génériques

Lors du codage alphanumérique du protocole, les identificateurs de terminaison sont certes arbitraires mais un choix judicieux des noms permet de rendre plus utile le caractère générique "\*" (astérisque). Lorsque ce caractère générique est rencontré, il "correspond" à tous les identificateurs de terminaison dont les caractères précédents et suivants sont les mêmes (le cas échéant). Par exemple, en présence des identificateurs de terminaison R13/3/1, R13/3/2 et R13/3/3, l'identificateur de terminaison R13/3/\* leur correspondra tous. Dans certains cas, toutes les terminaisons doivent être désignées. L'identificateur de terminaison "\*" suffit et est désigné par la structure "ALL" (tous). Le caractère "\$" peut être utilisé comme structure "CHOOSE" (choix) dans un identificateur de terminaison afin de signaler à la passerelle MG qu'elle doit créer une terminaison éphémère ou choisir une terminaison physique au repos.

#### B.2 Spécification en formalisme ABNF

La syntaxe du protocole est présentée en formalisme ABNF conformément au Document RFC 2234.

NOTE 1 – La présente spécification de syntaxe ne permet pas de faire respecter toutes les restrictions relatives aux inclusions et aux valeurs des éléments. Certaines restrictions supplémentaires sont mentionnées dans des commentaires, tandis que d'autres restrictions apparaissent dans le texte de la présente Recommandation. Ces restrictions supplémentaires font partie du protocole même si leur respect n'est pas assuré par la présente Recommandation.

NOTE 2 – La syntaxe dépend du contexte. Par exemple, "Add" peut être le jeton AddToken ou un nom NAME, en fonction du contexte dans lequel il figure.

Le formalisme ABNF et le codage alphanumérique sont insensibles à la casse. Cela inclut les identificateurs de terminaison (TerminationID), les identificateurs de script de numérotation (DigitMapID), etc. Le protocole SDP est sensible à la casse en vertu du Document RFC 2327.

```
; NOTE - Le formalisme ABNF de la présente section emploie la structure VALUE
; (ou des listes de structures VALUE) pour le codage des diverses valeurs
; d'éléments de paquetage (propriétés, paramètres des signaux, etc.). Les types
; de ces valeurs varient et sont spécifiés dans la définition de paquetage
; applicable. Plusieurs de ces types sont décrits au § 12.2.
;
; La spécification du formalisme ABNF pour VALUE admet une forme quotedString
```

```

; (chaîne entre apostrophes doubles) ou un ensemble de caractères sûrs
; (SafeChar).Le codage des valeurs d'élément de paquetage en valeurs de type
; VALUES du formalisme ABNF est spécifié ci-après. Si le codage d'un type admet
; des caractères autres que de type "SafeChar", la forme quotedString doit
; être employée pour toutes les valeurs de ce type, même pour des valeurs
; spécifiques qui ne comportent que des caractères de type
; "SafeChar".
;
; Chaîne: une chaîne doit employer la forme quotedString de VALUE et peut
; contenir tout caractère admis dans cette forme quotedString.
;
; Entier, double et non signé: les valeurs décimales peuvent être codées au
; moyen des caractères 0 à 9. Les valeurs hexadécimales doivent être précédées
; de '0x' et peuvent utiliser les caractères 0 à 9, a à f et A à F. Le format
; octal n'est pas admis. Les entiers négatifs commençant pas le signe '-'
; doivent être décimaux. La forme SafeChar de VALUE doit être
; employée.
;
; Caractère: un codage UTF-8 d'une seule lettre précédée et suivie de
; guillemets.
;
; Énumération: une énumération doit employer la forme SafeChar de VALUE et
; peut contenir tout caractère admis dans cette forme SafeChar (sans
; échappement).
;
; Booléen: les valeurs booléennes sont codées comme "on" et "off" et ne sont pas
; sensibles à la casse. La forme SafeChar de VALUE doit être utilisée.
;
; Types ultérieurs: tout type défini doit satisfaire à la spécification ABNF
; de VALUE. En particulier, si le codage d'un type admet des
; caractères autres que SafeChars, la forme quotedString doit être employée
; pour toutes les valeurs de ce type, même pour des valeurs spécifiques qui ne
; comportent que des caractères de type SafeChar.
;
; Il convient de noter qu'il n'y a pas moyen d'employer
; des doubles apostrophes à l'intérieur d'une valeur.
;
; Il convient aussi de noter que le protocole SDP n'admet pas les espaces blancs
; en début de ligne, tandis que le formalisme ABNF du contrôleur MGC autorise
; les blancs avant que le protocole SDP ne débute au niveau du
; descripteur "Local" ou distant. Les analyseurs syntaxiques devraient
; accepter les espaces blancs entre l'accolade LBRKT suivant le jeton local
; ou distant et le début du protocole SDP.

megacoMessage          = LWSP [authenticationHeader SEP] message

authenticationHeader   = AuthToken EQUAL SecurityParmIndex COLON
                        SequenceNum COLON AuthData

SecurityParmIndex      = "0x" 8 (HEXDIG)
SequenceNum            = "0x" 8 (HEXDIG)
AuthData               = "0x" 24*64 (HEXDIG)

Message                = MegacopToken SLASH Version SEP mId SEP messageBody
; La version du protocole définie ici est égale à 3.

messageBody            = (errorDescriptor / transactionList)

transactionList        = 1*(transactionRequest / transactionReply /
                        transactionPending / transactionResponseAck /
                        segmentReply)
;L'utilisation d'acquittements de réponse dépend du transport sous-jacent.

transactionPending     = PendingToken EQUAL TransactionID LBRKT RBRKT

```

```

transactionResponseAck = ResponseAckToken LBRKT transactionAck
                        *(COMMA transactionAck) RBRKT
transactionAck         = TransactionID / (TransactionID "-" TransactionID)

transactionRequest     = TransToken EQUAL TransactionID LBRKT
                        actionRequest *(COMMA actionRequest) RBRKT

actionRequest          = CtxToken EQUAL ContextID LBRKT ((contextRequest
[COMMA commandRequestList]) /
commandRequestList) RBRKT

contextRequest         = ((contextProperties [COMMA contextAudit])
/ contextAudit)

contextProperties      = contextProperty *(COMMA contextProperty)

; une seule fois au plus
; EmergencyOff est à utiliser dans le sens MG-MGC seulement dans les versions 1
; et 2 de la Rec. UIT-T H.248.1
; EmergencyToken ou EmergencyOffToken, mais non les deux
contextProperty       = (topologyDescriptor / priority / EmergencyToken /
EmergencyOffToken / iepsValue /
contextAttrDescriptor)

contextAudit           = ContextAuditToken LBRKT (contextAuditProperties
*(COMMA contextAuditProperties)) /
indAudcontextAttrDescriptor RBRKT

; une seule fois au plus sauf contextAuditSelector
contextAuditProperties = (TopologyToken / EmergencyToken /
PriorityToken / IEPSToken/ pkgdName /
contextAuditSelector)

; une seule fois au plus
contextAuditSelector  = priority / emergencyValue / iepsValue /
contextAttrDescriptor / auditSelectLogic

auditSelectLogic      = [ AndAUDITselectToken / OrAUDITselectToken ]
; Si ce champ est vide, on part du principe que les conditions de sélection sont
; l'opérateur AND

indAudcontextAttrDescriptor
= ContextAttrToken LBRKT contextAuditProperties
*(COMMA contextAuditProperties) RBRKT

; "O-" indique une commande facultative
; "W-" indique une réponse générique à une commande
commandRequestList    = ["O-"] ["W-"] commandRequest *(COMMA ["O-"]
["W-"]commandRequest)

commandRequest         = (ammRequest / subtractRequest / auditRequest /
notifyRequest / serviceChangeRequest)

transactionReply       = ReplyToken EQUAL TransactionID [SLASH segmentNumber
[SLASH SegmentationCompleteToken]] LBRKT
[ImmAckRequiredToken COMMA]
(errorDescriptor / actionReplyList) RBRKT

segmentReply           = MessageSegmentToken EQUAL TransactionID SLASH
segmentNumber [SLASH SegmentationCompleteToken]

segmentNumber          = UINT16

```

```

actionReplyList      = actionReply *(COMMA actionReply)

actionReply          = CtxToken EQUAL ContextID [LBRKT (errorDescriptor /
commandReply /(commandReply COMMA
errorDescriptor)) RBRKT]

commandReply         = ((contextProperties [COMMA commandReplyList]) /
commandReplyList)

commandReplyList     = commandReplies *(COMMA commandReplies)

commandReplies       = (serviceChangeReply / auditReply / ammsReply /
notifyReply)

;Add, Move et Modify ont les mêmes paramètres de requête
ammRequest           = (AddToken / MoveToken / ModifyToken) EQUAL
termIDList [LBRKT ammParameter *(COMMA
ammParameter) RBRKT]

;une seule fois au plus
ammParameter         = (mediaDescriptor / modemDescriptor /muxDescriptor /
eventsDescriptor / signalsDescriptor /
digitMapDescriptor / eventBufferDescriptor /
auditDescriptor / statisticsDescriptor)

ammsReply            = (AddToken / MoveToken / ModifyToken / SubtractToken)
EQUAL termIDList [LBRKT terminationAudit RBRKT]

subtractRequest      = SubtractToken EQUAL termIDList [LBRKT
auditDescriptor RBRKT]

auditRequest         = (AuditValueToken / AuditCapToken) EQUAL
termIDList LBRKT auditDescriptor RBRKT

auditReply           = (AuditValueToken / AuditCapToken)
(contextTerminationAudit / auditOther)

auditOther           = EQUAL termIDList [LBRKT terminationAudit RBRKT]

terminationAudit     = auditReturnParameter *(COMMA auditReturnParameter)

contextTerminationAudit = EQUAL CtxToken (terminationIDList / LBRKT
errorDescriptor RBRKT)

auditReturnParameter = (mediaDescriptor / modemDescriptor / muxDescriptor /
eventsDescriptor / signalsDescriptor /
digitMapDescriptor / observedEventsDescriptor /
eventBufferDescriptor / statisticsDescriptor /
packagesDescriptor / errorDescriptor /
auditReturnItem)

auditReturnItem      = (MuxToken / ModemToken / MediaToken /
DigitMapToken / StatsToken / ObservedEventsToken /
PackagesToken)

auditDescriptor      = AuditToken LBRKT [auditItem *(COMMA auditItem)]
RBRKT

notifyRequest        = NotifyToken EQUAL termIDList LBRKT
(observedEventsDescriptor [COMMA errorDescriptor])
RBRKT

notifyReply          = NotifyToken EQUAL termIDList [LBRKT errorDescriptor
RBRKT]

```

```

serviceChangeRequest      = ServiceChangeToken EQUAL termIDList LBRKT
                           serviceChangeDescriptor RBRKT

serviceChangeReply        = ServiceChangeToken EQUAL termIDList [LBRKT
                           (errorDescriptor / serviceChangeReplyDescriptor)
                           RBRKT]

errorDescriptor           = ErrorToken EQUAL ErrorCode LBRKT [quotedString]
                           RBRKT
ErrorCode                 = 1*4(DIGIT) ; pourrait être étendu

TransactionID            = UINT32

mId                      = ((domainAddress / domainName) [":" portNumber]) /
                           mtpAddress / deviceName

; ABNF permet deux ou plus de 2 points "." consécutifs, bien que cela
; n'ait pas de signification dans un nom de domaine.
domainName                = "<" (ALPHA / DIGIT) *63(ALPHA / DIGIT / "-" / ".")
                           ">"

deviceName                = pathNAME

; Les valeurs 0x0, 0xFFFFFFFF et 0xFFFFFFFF sont réservées.
; '-' est utilisé pour le contexte NULL. '*' est ALL. '$' est CHOOSE.
ContextID                 = (UINT32 / "*" / "-" / "$")

domainAddress             = "[" (IPv4address / IPv6address) "]"

;RFC 2373 contient la définition de d'adresses IPv6.
IPv6address               = hexpart [":" IPv4address]

IPv4address               = V4hex DOT V4hex DOT V4hex DOT V4hex

V4hex                     = 1*3(DIGIT) ; "0".."255"

; cette production, bien qu'apparaissant dans le Document RFC 2373, n'est
; pas référencée
IPv6prefix                = hexpart SLASH 1*2DIGIT

hexpart                   = hexseq ":@" [hexseq] / ":@" [hexseq] / hexseq

hexseq                    = hex4 *(":" hex4)

hex4                      = 1*4HEXDIG

portNumber                 = UINT16

; Structure d'adressage de mtpAddress:
; 25 - 15                0
;   | PC                | NI |
; 24 - 14 bit           2 bit
; NOTE - 14 bits sont définis pour usage international.
; deux options nationales existent lorsque le code de point est 16 ou 24 bits.
; Afin d'aligner les octets de mtpAddress, les bits MSB doivent être codés par
; des zéros.
; Un octet doit être représenté par 2 caractères hexadécimaux.
mtpAddress                 = MPTToken LBRKT 4*8 (HEXDIG) RBRKT

termIDList                 = (TerminationID / LBRKT TerminationID 1*(COMMA
                           TerminationID) RBRKT)

terminationIDList         = LBRKT TerminationID *(COMMA TerminationID) RBRKT

```

```

; La longueur totale de pathNAME ne doit jamais dépasser 64 caractères.
pathNAME          = ["*"] NAME *("/" / "*" / ALPHA / DIGIT / "_" / "$")
                  ["@" pathDomainName]

; ABNF permet deux ou plus de 2 "." consécutifs, bien que cela n'ait
; pas de signification dans un nom de domaine de conduit.

pathDomainName    = (ALPHA / DIGIT / "*") *63(ALPHA / DIGIT / "-" /
                  "*" / ".")

; '*' est ALL. '$' est CHOOSE.
TerminationID    = "ROOT" / pathNAME / "$" / "*"

mediaDescriptor   = MediaToken LBRKT mediaParm *(COMMA mediaParm) RBRKT

; au plus un seul descripteur "terminationState"
; et soit streamParm(s) ou streamDescriptor(s) mais non les deux
mediaParm         = (streamParm / streamDescriptor /
                  terminationStateDescriptor)

; une seule fois au plus par item
streamParm        = (localDescriptor / remoteDescriptor /
                  localControlDescriptor / statisticsDescriptor)

streamDescriptor  = StreamToken EQUAL StreamID LBRKT streamParm *(COMMA
                  streamParm) RBRKT

localControlDescriptor = LocalControlToken LBRKT localParm *(COMMA localParm)
                  RBRKT

; une seule fois au plus par item sauf pour propertyParm
localParm         = (streamMode / propertyParm / reservedValueMode /
                  reservedGroupMode)

reservedValueMode = ReservedValueToken EQUAL ("ON" / "OFF")

reservedGroupMode = ReservedGroupToken EQUAL ("ON" / "OFF")

streamMode        = ModeToken EQUAL streamModes

streamModes       = (SendonlyToken / RecvonlyToken / SendrecvToken /
                  InactiveToken / LoopbackToken)

propertyParm      = pkgdName parmValue

; le caractère (sûr) '$' signifie CHOOSE (choix)
; le caractère (sûr) '*' signifie ALL (tous)
parmValue         = (EQUAL alternativeValue / INEQUAL VALUE)

alternativeValue  = (VALUE
                  / LSBRKT VALUE *(COMMA VALUE) RSBRKT
                  ; sous-liste (soit A ET B ET ...)
                  / LBRKT VALUE *(COMMA VALUE) RBRKT
                  ; options (soit A OU B OU ...)
                  / LSBRKT VALUE COLON VALUE RSBRKT)
                  ; range

INEQUAL          = LWSP ">" / "<" / "#") LWSP ; '#' signifie "not
                  ; equal"

LSBRKT           = LWSP "[" LWSP

RSBRKT           = LWSP "]" LWSP

```

```

; NOTE - L'octet zéro ne fait pas partie des caractères permis dans
; une chaîne d'octets. Comme la définition actuelle est
; limitée à SDP et qu'un octet zéro ne serait pas un caractère
; légal en SDP, cette restriction ne pose pas de problème.
localDescriptor          = LocalToken LBRKT octetString RBRKT

remoteDescriptor         = RemoteToken LBRKT octetString RBRKT

eventBufferDescriptor    = EventBufferToken [LBRKT eventSpec*(COMMA eventSpec)
RBRKT]

eventSpec                = pkgdName [LBRKT eventSpecParameter *(COMMA
eventSpecParameter) RBRKT]

eventSpecParameter       = (eventStream / eventOther)

eventBufferControl       = BufferToken EQUAL eventBufferControlValue
eventBufferControlValue = ("OFF" / LockStepToken)

terminationStateDescriptor
                        = TerminationStateToken LBRKT terminationStateParm
                          *(COMMA terminationStateParm) RBRKT

; une seule fois au plus par item sauf pour propertyParm
terminationStateParm    = (propertyParm / serviceStates / eventBufferControl)

terminationStateParm    = (propertyParm / serviceStates / eventBufferControl)

serviceStates           = ServiceStatesToken EQUAL serviceStatesValue
serviceStatesValue      = (TestToken / OutOfSvcToken / InSvcToken)

muxDescriptor           = MuxToken EQUAL MuxType terminationIDList

MuxType                 = (H221Token / H223Token / H226Token / V76Token /
extensionParameter / Nx64kToken)

StreamID                = UINT16

pkgdName                 = (PackageName SLASH ItemID);item spécifique
                          / (PackageName SLASH "*") ;tous
                          ; les items contenus dans le paquetage
                          / ("*" SLASH "*") ; tous les items
                          ; pris en charge par la passerelle MG

PackageName             = NAME

ItemID                  = NAME

eventsDescriptor        = EventsToken [EQUAL RequestID LBRKT requestedEvent
*(COMMA requestedEvent) RBRKT]

requestedEvent          = pkgdName [LBRKT eventParameter *(COMMA
eventParameter) RBRKT]

notifyRegulated         = NotifyRegulatedToken [LBRKT (embedWithSig/
embedNoSig) RBRKT]

notifyBehaviour         = NotifyImmediateToken / notifyRegulated /
NeverNotifyToken

; présence d'au plus 1 de chacune des valeurs KeepActiveToken,
; notifyBehaviour, eventDM, ResetEventsDescriptor et eventStream
; présence au plus de soit embedWithSig ou embedNoSig mais non des deux valeurs

```

```

; KeepActiveToken et embedWithSig ne doivent jamais être présentes toutes les
; deux eventParameter      = (embedWithSig / embedNoSig / KeepActiveToken /
                             eventDM / eventStream / eventOther /
                             notifyBehaviour / ResetEventsDescriptorToken)

embedWithSig                = EmbedToken LBRKT signalsDescriptor [COMMA
                             embedFirst] RBRKT

embedNoSig                  = EmbedToken LBRKT embedFirst RBRKT

; at-most-once of each
embedFirst                  = EventsToken [EQUAL RequestID LBRKT
                             secondRequestedEvent *(COMMA secondRequestedEvent)
                             RBRKT]

secondRequestedEvent        = pkgdName [LBRKT secondEventParameter *(COMMA
                             secondEventParameter) RBRKT]

; présence d'au plus 1 de chacune des valeurs embedSig, KeepActiveToken,
; notifyBehaviour, eventDM
; ResetEventsDescriptor ou eventStream
; KeepActiveToken et embedSig ne doivent jamais être présentes toutes les deux
secondEventParameter        = (embedSig / KeepActiveToken / eventDM /
                             eventStream / eventOther / notifyBehaviour /
                             ResetEventsDescriptorToken)

embedSig                    = EmbedToken LBRKT signalsDescriptor RBRKT

eventStream                  = StreamToken EQUAL StreamID

eventOther                   = eventParameterName parmValue

eventParameterName          = NAME

eventDM                      = DigitMapToken EQUAL((digitMapName) / (LBRKT
                             digitMapValue RBRKT))

signalsDescriptor            = SignalsToken [LBRKT signalParm *(COMMA signalParm)
                             RBRKT]

signalParm                   = signalList / signalRequest

signalRequest                = signalName [LBRKT sigParameter *(COMMA sigParameter)
                             RBRKT]

signalList                   = SignalListToken EQUAL signalListId LBRKT
                             signalListParm *(COMMA signalListParm) RBRKT

signalListId                 = UINT16

; exactement un seul signal signalType, durée une fois au plus et chaque signal
; parameter
signalListParm               = signalRequest

signalName                   = pkgdName

; une seule fois au plus sigStream, une seule fois au plus sigSignalType,
; une seule fois au plus sigDuration, une seule fois au plus sigDirection,
; une seule fois au plus sigRequestID, une seule fois au plus sigIntsigDelay
; chaque nom signalParameterName au plus une seule fois
sigParameter                 = sigStream / sigSignalType / sigDuration / sigOther /
                             notifyCompletion / KeepActiveToken /
                             sigDirection / sigRequestID / sigIntsigDelay

```

```

sigStream           = StreamToken EQUAL StreamID
sigOther            = sigParameterName parmValue
sigParameterName   = NAME
sigSignalType      = SignalTypeToken EQUAL signalType
signalType         = (OnOffToken / TimeOutToken / BriefToken)
sigDuration        = DurationToken EQUAL UINT16
sigDirection       = DirectionToken EQUAL direction
direction          = ExternalToken / InternalToken / BothToken
sigRequestID      = RequestIDToken EQUAL RequestID
sigIntsigDelay     = IntsigDelayToken EQUAL UINT16
notifyCompletion   = NotifyCompletionToken EQUAL (LBRKT
notificationReason *(COMMA notificationReason)
RBRKT)
notificationReason = TimeOutToken / InterruptByEventToken /
InterruptByNewSignalsDescrToken / OtherReasonToken /
IterationToken

observedEventsDescriptor
= ObservedEventsToken EQUAL RequestID LBRKT
observedEvent *(COMMA observedEvent) RBRKT

; durée par événement, car elle pourra être mise en mémoire tampon
observedEvent      = [TimeStamp LWSP COLON] LWSP pkgdName [LBRKT
observedEventParameter *(COMMA
observedEventParameter) RBRKT]

; eventStream une seule fois au plus, chaque nom eventParameterName une seule
; fois au plus
observedEventParameter = eventStream / eventOther

; pour une réponse AuditCapReply avec tous les événements, l'identificateur
; RequestID devrait être ALL.
RequestID          = UINT32 / "*"

modemDescriptor    = ModemToken ((EQUAL modemType) / (LSBRKT modemType
*(COMMA modemType) RSBKKT)) [LBRKT propertyParm
*(COMMA propertyParm) RBRKT]

; une seule fois au plus, sauf pour extensionParameter
modemType          = V32bisToken / V22bisToken / V18Token / V22Token /
V32Token / V34Token / V90Token / V91Token /
SynchISDNToken / extensionParameter

digitMapDescriptor = DigitMapToken EQUAL ((LBRKT digitMapValue RBRKT) /
(digitMapName [LBRKT digitMapValue RBRKT]))

digitMapName       = NAME

digitMapValue      = ["T" COLON Timer COMMA] ["S" COLON Timer COMMA] ["L"
COLON Timer COMMA] ["Z" COLON Timer COMMA] digitMap

```

```

Timer                = 1*2DIGIT
; Les unités sont les secondes pour les temporisateurs T, S, et L, et les
; centaines de millisecondes pour le temporisateur Z. T, S, et L vont donc de 1
; à 99 s et Z va donc de 100 ms à 9,9 s

digitMap              = (digitString / LWSP "(" LWSP digitStringList LWSP
                        ")" LWSP)

digitStringList       = digitString *(LWSP "|" LWSP digitString)

digitString           = 1*(digitStringElement)

digitStringElement    = digitPosition [DOT]

digitPosition         = digitMapLetter / digitMapRange

digitMapRange         = ("x" / (LWSP "[" LWSP digitLetter LWSP "]" LWSP))

digitLetter           = *((DIGIT "-" DIGIT) / digitMapLetter)

digitMapLetter        = DIGIT                ; Symboles d'événement de base
                        / %x41-4B / %x61-6B ; a-k, A-K
                        / "L" / "S" / "T"    ; Temporisateurs entre
                                                ; événements (longs,
                                                ; (long, courts, départ)
                        / "Z"                ; Modificateur de longue durée

; une seule fois au plus. Les valeurs DigitMapToken et PackagesToken ne sont pas
; autorisées dans la commande AuditCapabilities
auditItem             = auditReturnItem / SignalsToken / EventBufferToken /
                        EventsToken / indAudterminationAudit

indAudterminationAudit = indAudauditReturnParameter *(COMMA
                        indAudauditReturnParameter)

indAudauditReturnParameter
                        = indAudmediaDescriptor / indAudeventsDescriptor /
                        indAudsignalsDescriptor /
                        indAuddigitMapDescriptor /
                        indAudeventBufferDescriptor /
                        indAudstatisticsDescriptor /
                        indAudpackagesDescriptor

indAudmediaDescriptor = MediaToken LBRKT indAudmediaParm *(COMMA
                        indAudmediaParm) RBRKT

; soit streamParm ou streamDescriptor mais non les deux
indAudmediaParm       = indAudstreamParm / indAudstreamDescriptor /
                        indAudterminationStateDescriptor

; une seule fois au plus
indAudstreamParm      = (indAudlocalControlDescriptor /
                        indAudstatisticsDescriptor /
                        indAudremoteDescriptor / indAudlocalDescriptor )

indAudremoteDescriptor = RemoteToken LBRKT octetString RBRKT
indAudlocalDescriptor  = LocalToken LBRKT octetString RBRKT

indAudstreamDescriptor = StreamToken EQUAL StreamID LBRKT indAudstreamParm
                        RBRKT

```

```

indAudlocalControlDescriptor
    = LocalControlToken LBRKT indAudlocalParm
      *(COMMA indAudlocalParm) RBRKT

; une seule fois au plus par item
indAudlocalParm
    = ModeToken [(EQUAL/INEQUAL) streamModes]/ pkgdName /
      propertyParm / ReservedValueToken /
      ReservedGroupToken
; les valeurs propertyparm et streamModes ne sont utilisées que pour spécifier
; des critères de sélection d'analyse. La logique de sélection ET/OU est
; spécifiée au niveau du contexte

indAudterminationStateDescriptor
    = TerminationStateToken LBRKT
      indAudterminationStateParm RBRKT

; une seule fois au plus par item
indAudterminationStateParm
    = pkgdName / propertyParm / ServiceStatesToken

[(EQUAL/INEQUAL) serviceStatesValue ] / BufferToken
; Lorsque des valeurs sont incluses, une opération Select est impliquée.
; la logique ET/OU est spécifiée au niveau du contexte

indAudeventBufferDescriptor
    = EventBufferToken LBRKT indAudeventSpec RBRKT

indAudeventSpec
    = pkgdName [LBRKT indAudeventSpecParameter RBRKT]

indAudeventSpecParameter= eventStream / eventParameterName

indAudeventsDescriptor = EventsToken [EQUAL RequestID] LBRKT
    indAudrequestedEvent RBRKT

indAudrequestedEvent = pkgdName

indAudsignalsDescriptor = SignalsToken LBRKT [indAudsignalParm] RBRKT

indAudsignalParm
    = indAudsignalList / indAudsignalRequest

indAudsignalRequest
    = signalName [LBRKT indAudsignalRequestParm
      *(COMMA indAudsignalRequestParm) RBRKT]

indAudsignalRequestParm = sigStream / sigRequestID

indAudsignalList
    = SignalListToken EQUAL signalListId [LBRKT
      indAudsignalListParm RBRKT]

indAudsignalListParm = indAudsignalRequest

indAuddigitMapDescriptor= DigitMapToken EQUAL (digitMapName)

indAudstatisticsDescriptor
    = StatsToken LBRKT pkgdName RBRKT

indAudpackagesDescriptor= PackagesToken LBRKT packagesItem RBRKT

serviceChangeDescriptor = ServicesToken LBRKT serviceChangeParm *(COMMA
    serviceChangeParm) RBRKT

; présence de chaque paramètre une seule fois au plus, sauf auditItem
; au plus 1 valeur entre serviceChangeAddress ou serviceChangeMgcId mais
; non les deux
; les valeurs serviceChangeMethod et serviceChangeReason sont REQUISES.

```

```

serviceChangeParm          = serviceChangeMethod / serviceChangeReason /
                             serviceChangeDelay / serviceChangeAddress /
                             serviceChangeProfile / extension / TimeStamp /
                             serviceChangeMgcId / serviceChangeVersion /
                             ServiceChangeIncompleteToken / auditItem)

serviceChangeReplyDescriptor
                             = ServicesToken LBRKT servChgReplyParm *(COMMA
                             servChgReplyParm) RBRKT

; une seule fois au plus. Version est REQUISE lors de la première réponse
ServiceChange
; au plus 1 valeur entre serviceChangeAddress ou serviceChangeMgcId mais
; non les deux.
servChgReplyParm          = serviceChangeAddress / serviceChangeMgcId /
                             serviceChangeProfile / serviceChangeVersion /
                             TimeStamp

serviceChangeMethod       = MethodToken EQUAL (FailoverToken / ForcedToken /
                             GracefulToken / RestartToken / DisconnectedToken /
                             HandOffToken / extensionParameter)

; Une valeur serviceChangeReason se compose d'un code numérique de
; raison et d'une description facultative en mode texte.
; Une valeur serviceChangeReason doit toujours être codée sous la forme
; d'une chaîne entre apostrophes doubles (quotedString) de l'élément VALUE.
; La chaîne entre apostrophes doubles doit contenir un code décimal de
; raison, facultativement suivi par un seul caractère d'espace et par une
; chaîne de description textuelle.
serviceChangeReason      = ReasonToken EQUAL VALUE

serviceChangeDelay       = DelayToken EQUAL UINT32

serviceChangeAddress     = ServiceChangeAddressToken EQUAL (mId / portNumber)

serviceChangeMgcId      = MgcIdToken EQUAL mId

serviceChangeProfile     = ProfileToken EQUAL NAME SLASH Version

serviceChangeVersion     = VersionToken EQUAL Version

extension                = extensionParameter parmValue

packagesDescriptor      = PackagesToken LBRKT packagesItem *(COMMA
                             packagesItem) RBRKT

Version                  = 1*2 (DIGIT)

packagesItem             = NAME "-" UINT16

TimeStamp                = Date "T" Time ; selon l'ISO 8601:2004

; Date = yyyyymmdd
Date                     = 8 (DIGIT)

; Heure = hhmmsssss
Time                     = 8 (DIGIT)

statisticsDescriptor     = StatsToken LBRKT statisticsParameter *(COMMA
                             statisticsParameter) RBRKT

;une seule fois au plus par item
statisticsParameter      = pkgdName [EQUAL VALUE /
                             (LSBRKT VALUE *(COMMA VALUE) RSBKRT)]

```

```

topologyDescriptor      = TopologyToken LBRKT topologyTriple *(COMMA
                        topologyTriple) RBRKT

topologyTriple          = terminationA COMMA terminationB COMMA
                        topologyDirection [COMMA eventStream]

terminationA            = TerminationID

terminationB            = TerminationID

topologyDirection      = BothwayToken / IsolateToken / OnewayToken /
                        OnewayExternalToken / OnewayBothToken

priority                = PriorityToken EQUAL UINT16

iepsValue               = IEPSToken EQUAL ("ON" / "OFF")

emergencyValue          = EmergencyValueToken EQUAL (EmergencyToken /
                        EmergencyOffToken)

contextAttrDescriptor  = ContextAttrToken LBRKT (contextIdList /
                        propertyParm *(COMMA propertyParm)) RBRKT

; Lors de l'utilisation de la production contextIdList, l'identificateur de
; contexte contenu dans la production actionReply doit être le même que
; l'identificateur de contexte contenu dans la requête actionRequest associée.
contextIdList           = ContextListToken EQUAL LBRKT ContextID
                        *(COMMA ContextID) RBRKT

extensionParameter     = "X" ("-" / "+") 1*6(ALPHA / DIGIT)

; Une chaîne d'octets sert à décrire le protocole
; SDP défini dans le Document RFC 2327. Des précautions sont à prendre en cas
; d'utilisation de la séquence CRLF (retour de chariot et interligne) mentionnée
; dans le Document RFC 2327. Par sécurité, utiliser la commande EOL (fin de
; ligne) dans ce formalisme ABNF.
; Chaque fois que le caractère "]" apparaît dans le protocole SDP, il est
; transformé par échappement en "\", par exemple "\]"
octetString             = *(nonEscapeChar)

nonEscapeChar           = ("\]" / %x01-7C / %x7E-FF)

; NOTE - Le caractère d'apostrophe ; double n'est pas autorisé à l'intérieur
; d'une chaîne entre apostrophes doubles ; (quotedString).
quotedString            = DQUOTE *(SafeChar / EOL / %x80-FF / RestChar / WSP)
                        DQUOTE

UINT16                  = 1*5(DIGIT) ; %x0-FFFF

UINT32                  = 1*10(DIGIT) ; %x0-FFFFFFFF

NAME                    = ALPHA *63(ALPHA / DIGIT / "_")

VALUE                   = quotedString / 1*(SafeChar / %x80-FF)

SafeChar                = DIGIT / ALPHA / "+" / "-" / "&" /
                        "!" / " " / "/" / "'" / "?" / "@" /
                        "^" / "`" / "~" / "*" / "$" / "\" /
                        "(" / ")" / "%" / "|" / "."

EQUAL                   = LWSP %x3D LWSP ; "="

COLON                   = %x3A ; ":"

```

```

LBRKT          = LWSP %x7B LWSP          ; "{"
RBRKT          = LWSP %x7D LWSP          ; "}"
COMMA          = LWSP %x2C LWSP          ; ","
DOT            = %x2E                    ; "."
SLASH          = %x2F                    ; "/"
ALPHA          = %x41-5A / %x61-7A      ; A-Z / a-z
DIGIT          = %x30-39                ; 0-9
DQUOTE         = %x22                    ; " (apostrophe double)
HEXDIG         = DIGIT / "A" / "B" / "C" / "D" / "E" / "F"
SP             = %x20                    ; espace
HTAB           = %x09                    ; tabulateur horizontal
CR             = %x0D                    ; retour de chariot
LF             = %x0A                    ; interligne
LWSP           = *(WSP / COMMENT / EOL)
EOL            = (CR [LF] / LF)
WSP            = SP / HTAB              ; espace blanc
SEP            = (WSP / EOL / COMMENT) LWSP
COMMENT        = ";" *(SafeChar/ RestChar / WSP / %x22) EOL
RestChar       = ";" / "[" / "]" / "{" / "}" / ":" / "," / "#" /
                "<" / ">" / "="

```

; Les nouveaux jetons ajoutés au paramètre sigParameter doivent prendre la  
; forme "SPA\*". L'astérisque peut être de forme quelconque, comme "SPAM".  
; Les nouveaux jetons ajoutés au paramètre eventParameter doivent prendre la  
; forme "EPA\*". L'astérisque peut être de forme quelconque, comme "EPAD".

```

AddToken        = ("Add"                  / "A")
AndAUDITSelectToken = ("ANDLgc")
AuditToken      = ("Audit"                / "AT")
AuditCapToken   = ("AuditCapability"      / "AC")
AuditValueToken = ("AuditValue"          / "AV")
AuthToken       = ("Authentication"      / "AU")
BothToken       = ("Both"                 / "B")
BothwayToken    = ("Bothway"              / "BW")
BriefToken      = ("Brief"                 / "BR")
BufferToken     = ("Buffer"                / "BF")
CtxToken        = ("Context"              / "C")
ContextAuditToken = ("ContextAudit"       / "CA")
ContextAttrToken = ("ContextAttr"         / "CT")
ContextListToken = ("ContextList"        / "CLT")
DigitMapToken   = ("DigitMap"             / "DM")
DirectionToken  = ("SPADirection"        / "SPADI")
DisconnectedToken = ("Disconnected"       / "DC")
DelayToken      = ("Delay"                 / "DL")
DurationToken   = ("Duration"             / "DR")

```

EmbedToken	= ("Embed"	/ "EM")
EmergencyToken	= ("Emergency"	/ "EG")
EmergencyOffToken	= ("EmergencyOff"	/ "EGO")
EmergencyValueToken	= ("EmergencyValue"	/ "EGV")
ErrorToken	= ("Error"	/ "ER")
EventBufferToken	= ("EventBuffer"	/ "EB")
EventsToken	= ("Events"	/ "E")
ExternalToken	= ("External"	/ "EX")
FailoverToken	= ("Failover"	/ "FL")
ForcedToken	= ("Forced"	/ "FO")
GracefulToken	= ("Graceful"	/ "GR")
H221Token	= ("H221")	
H223Token	= ("H223")	
H226Token	= ("H226")	
HandOffToken	= ("HandOff"	/ "HO")
IEPSToken	= ("IEPSCall"	/ "IEPS")
ImmAckRequiredToken	= ("ImmAckRequired"	/ "IA")
InactiveToken	= ("Inactive"	/ "IN")
InternalToken	= ("Internal"	/ "IT")
IntsigDelayToken	= ("Intersignal"	/ "SPAIS")
IsolateToken	= ("Isolate"	/ "IS")
InSvcToken	= ("InService"	/ "IV")
InterruptByEventToken	= ("IntByEvent"	/ "IBE")
InterruptByNewSignalsDescrToken	= ("IntBySigDescr"	/ "IBS")
IterationToken	= ("Iteration"	/ "IR")
KeepActiveToken	= ("KeepActive"	/ "KA")
LocalToken	= ("Local"	/ "L")
LocalControlToken	= ("LocalControl"	/ "O")
LockStepToken	= ("LockStep"	/ "SP")
LoopbackToken	= ("Loopback"	/ "LB")
MediaToken	= ("Media"	/ "M")
MegacopToken	= ("MEGACO"	/ "I")
MessageSegmentToken	= ("Segment"	/ "SM")
MethodToken	= ("Method"	/ "MT")
MgcIdToken	= ("MgcIdToTry"	/ "MG")
ModeToken	= ("Mode"	/ "MO")
ModifyToken	= ("Modify"	/ "MF")
ModemToken	= ("Modem"	/ "MD")
MoveToken	= ("Move"	/ "MV")
MTPToken	= ("MTP")	
MuxToken	= ("Mux"	/ "MX")
NeverNotifyToken	= ("NeverNotify"	/ "NBNN")
NotifyToken	= ("Notify"	/ "N")
NotifyCompletionToken	= ("NotifyCompletion"	/ "NC")
NotifyImmediateToken	= ("ImmediateNotify"	/ "NBIN")
NotifyRegulatedToken	= ("RegulatedNotify"	/ "NBRN")
Nx64kToken	= ("Nx64Kservice"	/ "N64")
ObservedEventsToken	= ("ObservedEvents"	/ "OE")
OnewayToken	= ("Oneway"	/ "OW")
OnewayBothToken	= ("OnewayBoth"	/ "OWB")
OnewayExternalToken	= ("OnewayExternal"	/ "OWE")
OnOffToken	= ("OnOff"	/ "OO")
OrAUDITselectToken	= ("ORLgc")	
OtherReasonToken	= ("OtherReason"	/ "OR")
OutOfSvcToken	= ("OutOfService"	/ "OS")
PackagesToken	= ("Packages"	/ "PG")
PendingToken	= ("Pending"	/ "PN")
PriorityToken	= ("Priority"	/ "PR")
ProfileToken	= ("Profile"	/ "PF")
ReasonToken	= ("Reason"	/ "RE")
RecvonlyToken	= ("ReceiveOnly"	/ "RC")
ReplyToken	= ("Reply"	/ "P")
ResetEventsDescriptorToken		

```

RestartToken          = ("ResetEventsDescriptor" / "RSE")
RestartToken          = ("Restart" / "RS")
RemoteToken           = ("Remote" / "R")
RequestIDToken        = ("SPARquestID" / "SPARQ")
ReservedGroupToken    = ("ReservedGroup" / "RG")
ReservedValueToken    = ("ReservedValue" / "RV")
SegmentationCompleteToken
SegmentationCompleteToken = ("END" / "&")
SendonlyToken         = ("SendOnly" / "SO")
SendrecvToken         = ("SendReceive" / "SR")
ServicesToken         = ("Services" / "SV")
ServiceStatesToken    = ("ServiceStates" / "SI")
ServiceChangeIncompleteToken
ServiceChangeIncompleteToken = ("ServiceChangeInc" / "SIC")
ServiceChangeToken    = ("ServiceChange" / "SC")
ServiceChangeAddressToken
ServiceChangeAddressToken = ("ServiceChangeAddress" / "AD")
SignalListToken       = ("SignalList" / "SL")
SignalsToken          = ("Signals" / "SG")
SignalTypeToken       = ("SignalType" / "SY")
StatsToken            = ("Statistics" / "SA")
StreamToken           = ("Stream" / "ST")
SubtractToken         = ("Subtract" / "S")
SynchISDNToken        = ("SynchISDN" / "SN")
TerminationStateToken = ("TerminationState" / "TS")
TestToken             = ("Test" / "TE")
TimeOutToken          = ("TimeOut" / "TO")
TopologyToken         = ("Topology" / "TP")
TransToken            = ("Transaction" / "T")
ResponseAckToken      = ("TransactionResponseAck" / "K")
V18Token              = ("V18")
V22Token              = ("V22")
V22bisToken           = ("V22b")
V32Token              = ("V32")
V32bisToken           = ("V32b")
V34Token              = ("V34")
V76Token              = ("V76")
V90Token              = ("V90")
V91Token              = ("V91")
VersionToken          = ("Version" / "V")

```

### B.3 Codage hexadécimal des octets

Le codage hexadécimal des octets est un moyen de représenter une chaîne d'octets comme une chaîne de chiffres hexadécimaux, chaque octet étant représenté par deux chiffres. Ce codage des octets devrait être employé lors du codage de chaînes d'octets dans la version alphanumérique du protocole.

Pour chaque octet, la séquence à 8 bits est codée au moyen de chiffres hexadécimaux. Le bit 0 est transmis en premier lieu, tandis que le bit 7 est transmis en dernier lieu.

Les bits 7 à 4 sont codés comme étant le premier chiffre hexadécimal, le bit 7 étant le bit de plus fort poids et le bit 4 étant celui de plus faible poids. Les bits 3 à 0 sont codés comme étant le second chiffre hexadécimal, le bit 3 étant le bit de plus fort poids et le bit 0 étant celui de plus faible poids.

Exemples:

Forme binaire des octets	Codage hexadécimal
00011011	D8
11100100	27
10000011 10100010 11001000 00001001	C1451390

## **B.4 Séquence hexadécimale pour les octets**

Une séquence hexadécimale pour les octets est un nombre pair de chiffres hexadécimaux, s'achevant par un caractère <CR>.

# **Annexe C**

## **Étiquettes des propriétés de flux média**

Les paramètres des descripteurs "Local", "Remote" et "LocalControl" sont spécifiés sous forme de paires de valeurs d'étiquette si le codage binaire est utilisé pour le protocole. La présente annexe contient les noms de propriétés (PropertyID), les étiquettes de propriétés (Property tag), le type de la propriété (Type) et les valeurs (Value). Les valeurs fournies dans le champ "Value" lorsque celui-ci contient des références doivent être considérées comme une "information". La référence contient les valeurs normatives. Si un champ de valeur ne contient pas de référence, les valeurs de ce champ doivent être considérées comme "normatives".

Les références aux propriétés décrites dans l'Annexe C suivent la structure d'identification PackageID/PropertyID. L'Annexe C n'est cependant pas, par elle-même, un paquetage. Elle est considérée comme disposant d'un identificateur PackageID 0x0000 pour le codage binaire et d'un identificateur "anxc" pour le codage textuel. Pour le codage textuel selon la Rec. UIT-T H.248.1, l'Annexe C ne doit être utilisée que si la propriété requise n'est pas définie par un paquetage ou n'est pas représentée par le protocole SDP. L'imbrication d'une propriété contenue dans l'Annexe C à l'intérieur d'une autre propriété est interdite.

Dans la présente annexe, les étiquettes sont exprimées en nombres hexadécimaux. Lors du réglage de la valeur d'une propriété, un contrôleur MGC peut sous-spécifier la valeur selon l'un des mécanismes spécifiés au § 7.1.1.

La prise en charge des propriétés de la présente annexe ou de ses paragraphes est facultative. Par exemple, seules trois propriétés du § C.3 et cinq propriétés du § C.8 pourraient être appliquées.

Pour le type "énumération", la valeur est représentée par la valeur entre parenthèses, par exemple Send(0), Receive(1). Les propriétés indiquées à l'Annexe C pour les types à "N bit" ou à "M octets" devraient être traitées comme des chaînes d'octets lors du codage du protocole, tandis que les propriétés à "Entier de N bit" seront traitées comme des entiers. Une valeur de type "String" sera traitée comme une chaîne IA5String lors du codage du protocole.

Lorsqu'un type présente une valeur inférieure à un octet, cette valeur doit être enregistrée dans les bits de plus faible poids d'une chaîne d'octet de longueur égale à un seul octet.

## C.1 Attributs généraux des médias

Identificateur de propriété	Etiquette de propriété	Type	Valeur
Média	1001	Enumération	Audio(0), Video(1), Data(2)
Mode de transmission	1002	Enumération	Send(0), Receive(1), Send&Receive(2)
Nombre de voies	1003	Entier non signé	0-255
Fréquence d'échantillonnage	1004	Entier non signé	0-2 <sup>32</sup>
Débit	1005	Entier	(0..4294967295) NOTE – Unités de 100 bit/s.
Codec A	1006	Chaîne d'octets	Audio Type de codec: Réf.: Rec. UIT-T Q.765.5 Les codecs non UIT-T sont définis par une organisation de normalisation appropriée sous un identificateur organisationnel défini.
Samplepp	1007	Entier non signé	Nombre maximal d'échantillons ou de trames pour chaque paquet: 0..65535
Silencesupp	1008	Booléen	Suppression de silence: Vrai/Faux
Encrypttype	1009	Chaîne d'octets	Réf.: Rec. UIT-T H.245
Encryptkey	100A	Longueur de la chaîne d'octets (0..65535)	Clé de chiffrement Réf.: Rec. UIT-T H.235.0
Echocanc	100B		Non utilisée. Voir le § E.13 pour un exemple de possibles propriétés de limitation d'écho.
Gain	100C		Valeur non utilisée. Voir au § E.13 une propriété de gain disponible.
Jitterbuff	100D	Entier non signé	Taille du tampon de gigue ms: 0..65535
PropDelay	100E	Entier non signé	Temps de propagation: 0..65535 Temps de propagation maximal en millisecondes pour la connexion support entre deux passerelles MG. Le temps maximal dépendra de la technique de support.
RTPpayload	100F	Entier	Type de charge utile dans le profil RTP pour conférences audio et vidéo avec commandes minimales Réf.: RFC 1890
Ptime	1010	Entier	Durée de paquetsation Cette propriété indique la durée en millisecondes représentée par le média dans un paquet. Réf.: IETF RFC 2327

## C.2 Propriétés relatives au multiplexage

Identificateur de propriété	Etiquette de propriété	Type	Valeur
H222	2001	Chaîne d'octets	H222LogicalChannelParameters Réf.: Rec. UIT-T H.245
H223	2002	Chaîne d'octets	H223LogicalChannelParameters Réf.: Rec. UIT-T H.245
V76	2003	Chaîne d'octets	V76LogicalChannelParameters Réf.: Rec. UIT-T H.245
H2250	2004	Chaîne d'octets	H2250LogicalChannelParameters Réf.: Rec. UIT-T H.245

## C.3 Propriétés générales de support

Identificateur de propriété	Etiquette de propriété	Type	Valeur
Mediatx	3001	Enumération	Type de transport de média Circuit(0) TDM, ATM(1), FR(2), Ipv4(3), Ipv6(4), ...
BIR	3002	4 octets	La valeur dépend de la technique de transport
NSAP	3003	1-20 octets	Voir NSAP. Réf.: Annexe A/X.213

## C.4 Propriétés générales du mode ATM

Identificateur de propriété	Etiquette de propriété	Type	Valeur
AESA	4001	20 octets	Adresse de système d'extrémité ATM
VPVC	4002	4 octets: VPCI dans les deux premiers octets de plus faible poids, VCI dans les deux octets suivants	VPCI/VCI Réf.: Rec. UIT-T Q.2931
SC	4003	Enumération	Catégorie de service: CBR(0), nrt-VBR1(1), nrt-VBR2(2), nrt-VBR3(3), rt-VBR1(4), rt-VBR2(5), rt-VBR3(6), UBR1(7), UBR2(8), ABR(9). Réf.: Forum ATM UNI 4.0
BCOB	4004	Entier de 5 bits	Classe de support large bande Réf.: Rec. UIT-T Q.2961.2
BBTC	4005	Entier de 7 bits	Capacité de transfert large bande Réf.: Rec. UIT-T Q.2961.1

Identificateur de propriété	Etiquette de propriété	Type	Valeur
ATC	4006	Enumération	Capacité de trafic ATM I.371 DBR(0), SBR1(1), SBR2(2), SBR3(3), ABT/IT(4), ABT/DT(5), ABR(6) Réf.: Rec. UIT-T I.371
STC	4007	2 bits	Sensibilité à la mutilation: Bits <u>2 1</u> 0 0 insensible à la mutilation 0 1 sensible à la mutilation Réf.: Rec. UIT-T Q.2931
UPCC	4008	2 bits	Configuration de la connexion de plan utilisateur: Bits <u>2 1</u> 0 0 point à point 0 1 point à multipoint Réf.: Rec. UIT-T Q.2931
PCR0	4009	Entier de 24 bits	Débit cellulaire crête (pour CLP = 0) Réf.: Rec. UIT-T Q.2931
SCR0	400A	Entier de 24 bits	Débit soutenable (pour CLP = 0) Réf.: Rec. UIT-T Q.2961.1
MBS0	400B	Entier de 24 bits	Taille maximale des rafales (pour CLP = 0) Réf.: Rec. UIT-T Q.2961.1
PCR1	400C	Entier de 24 bits	Débit cellulaire crête (pour CLP = 0 + 1) Réf.: Rec. UIT-T Q.2931
SCR1	400D	Entier de 24 bits	Débit soutenable (pour CLP = 0 + 1) Réf.: Rec. UIT-T Q.2961.1
MBS1	400E	Entier de 24 bits	Taille maximale des rafales (pour CLP = 0 + 1) Réf.: Rec. UIT-T Q.2961.1
BEI	400F	Booléen	Indicateur de connexion au mieux La valeur 1 indique que BEI doit être inclus dans la signalisation ATM; la valeur 0 indique que BEI ne doit pas être inclus dans la signalisation ATM. Réf.: Forum ATM UNI 4.0
TI	4010	Booléen	Indicateur d'étiquetage La valeur 0 indique que l'étiquetage n'est pas autorisé; la valeur 1 indique que l'étiquetage est demandé. Réf.: Rec. UIT-T Q.2961.1

Identificateur de propriété	Etiquette de propriété	Type	Valeur	
FD	4011	Booléen	Mise à l'écart de trame La valeur 0 indique qu'aucune mise à l'écart de trame n'est autorisée; la valeur 1 indique que la mise à l'écart de trame est autorisée. Réf.: Forum ATM UNI 4.0	
A2PCDV	4012	Entier de 24 bits	CDV entre 2 points acceptable Réf.: Rec. UIT-T Q.2965.2	
C2PCDV	4013	Entier de 24 bits	CDV entre 2 points cumulé Réf.: Rec. UIT-T Q.2965.2	
APPCDV	4014	Entier de 24 bits	CDV de P à P acceptable Réf.: Forum ATM UNI 4.0	
CPPCDV	4015	Entier de 24 bits	CDV de P à P cumulé Réf.: Forum ATM UNI 4.0	
ACLR	4016	Entier de 8 bits	Taux de perte de cellule acceptable Réf.: Rec. UIT-T Q.2965.2, Forum ATM UNI 4.0	
MEETD	4017	Entier de 16 bits	Temps de transit de bout en bout maximal Réf.: Rec. UIT-T Q.2965.2, Forum ATM UNI 4.0	
CEETD	4018	Entier de 16 bits	Temps de transit de bout en bout cumulé Réf.: Rec. UIT-T Q.2965.2, Forum ATM UNI 4.0	
QosClass	4019	Entier compris entre 0 et 5	Classe de qualité de service (QS)	
			Classe de QS	Signification
			0	QS par défaut associée à la capacité de transfert ATC telle que définie dans la Rec. UIT-T Q.2961.2
			1	Sévère
			2	Tolérante
			3	A deux niveaux
			4	Non bornée
			5	Sévère à deux niveaux
			Réf.: Rec. UIT-T Q.2965.1	
AALtype	401A	1 octet	Type de couche AAL Bits <u>8 7 6 5 4 3 2 1</u> 0 0 0 0 0 0 0 AAL pour voix 0 0 0 0 0 0 1 AAL type 1 0 0 0 0 0 1 0 AAL type 2 0 0 0 0 0 1 1 AAL type 3/4 0 0 0 0 1 0 1 AAL type 5 0 0 0 1 0 0 0 AAL définie par utilisateur Réf.: Rec. UIT-T Q.2931	

## C.5 Relais de trames

Identificateur de propriété	Etiquette de propriété	Type	Valeur
DLCI	5001	Entier non signé	Identificateur de connexion de liaison de données
CID	5002	Entier non signé	Identificateur de sous-voie
SID/Noiselevel	5003	Entier non signé	Descripteur d'insertion de silence
Primary Payload Type	5004	Entier non signé	Type de charge utile primaire couvre FAX et codecs

## C.6 IP

Identificateur de propriété	Etiquette de propriété	Type	Valeur
IPv4	6001	32 bits Ipv4Address	Adresse IPv4 Réf.: IETF RFC 791
IPv6	6002	128 bits	Adresse IPv6 Réf.: IETF RFC 2460
Port	6003	Entier non signé	0..65535
Porttype	6004	Enuméré	TCP(0), UDP(1), SCTP(2)
RtcpbwRS	6005	Entier	Le modificateur RS de largeur de bande RTCP indique la largeur de bande attribuée par le protocole RTCP à des expéditeurs de données actifs (tels que définis par la spécification du protocole RTP) Réf.: IETF RFC 3556
RtcpbwRR	6006	Entier	Le modificateur RR de largeur de bande RTCP indique la largeur de bande attribuée par le protocole RTCP à d'autres participants de la session RTP (c'est-à-dire, à des récepteurs) Réf.: IETF RFC 3556

## C.7 ATM AAL 2

Identificateur de propriété	Etiquette de propriété	Type	Valeur
AESA	7001	20 octets	Adresse d'extrémité de service AAL 2 comme défini dans la Recommandation référencée. ESEA NSEA Réf.: Rec. UIT-T Q.2630.1
BIR	Voir le § C.3	4 octets	Réf. produite par utilisateur du service comme défini dans la Recommandation référencée. SUGR Réf.: Rec. UIT-T Q.2630.1

Identificateur de propriété	Etiquette de propriété	Type	Valeur
ALC	7002	12 octets	Caractéristiques de liaison AAL 2 comme défini dans la Recommandation référencée. Débit CPS-SDU max./moyen; Longueur d'unité CPS-SDU max./moyenne Réf.: Rec. UIT-T Q.2630.1
SSCS	7003	I.366.2: audio (8 octets); multidébit (3 octets), ou I.366.1: SAR-assuré (14 octets); SAR non assuré (7 octets).	Informations de sous-couche de convergence propres au service comme défini dans: – Rec. UIT-T Q.2630.1, et utilisé dans: – Rec. UIT-T I.366.2: audio/multidébit; – Rec. UIT-T I.366.1: SAR-assuré/non assuré. Réf.: Recommandations UIT-T Q.2630.1, I.366.1 et I.366.2
SUT	7004	1..254 octets	Paramètre de transport d'utilisateur du service comme défini dans la Recommandation référencée. Réf.: Rec. UIT-T Q.2630.1
TCI	7005	Booléen	Indicateur de connexion d'essai comme défini dans la Recommandation référencée. Réf.: Rec. UIT-T Q.2630.1
Timer_CU	7006	Entier de 32 bits	Temporisateur-CU Millisecondes de maintien de cellule partiellement remplie avant émission.
MaxCPSSDU	7007	Entier de 8 bits	Unité de données de service CPS maximale Réf.: Rec. UIT-T Q.2630.1
CID	7008	8 bits	Identificateur de sous-voie: 0-255 Réf.: Rec. UIT-T I.363.2

## C.8 ATM AAL 1

Identificateur de propriété	Etiquette de propriété	Type	Valeur
BIR	Voir tableau au § C.3	4-29 octets	Transport d'identificateur générique (GIT, <i>generic identifier transport</i> ) Réf.: Rec. UIT-T Q.2941.1
AAL1ST	8001	1 octet	Sous-type AAL 1 Bits <u>8 7 6 5 4 3 2 1</u> 0 0 0 0 0 0 0 0 néant 0 0 0 0 0 0 0 1 transport de signaux en bande vocale à 64 kbit/s 0 0 0 0 0 0 1 0 transport en mode circuit 0 0 0 0 0 1 0 0 transport de signaux audio de haute qualité 0 0 0 0 0 1 0 1 transport de signaux vidéo Réf.: Rec. UIT-T Q.2931
CBRR	8002	1 octet	Débit CBR Bits <u>8 7 6 5 4 3 2 1</u> 0 0 0 0 0 0 0 1 64 kbit/s 0 0 0 0 0 1 0 0 1544 kbit/s 0 0 0 0 0 1 0 1 6312 kbit/s 0 0 0 0 0 1 1 0 32 064 kbit/s 0 0 0 0 0 1 1 1 44 736 kbit/s 0 0 0 0 1 0 0 0 97 728 kbit/s 0 0 0 1 0 0 0 0 2048 kbit/s 0 0 0 1 0 0 0 1 8448 kbit/s 0 0 0 1 0 0 1 0 34 368 kbit/s 0 0 0 1 0 0 1 1 139 264 kbit/s 0 1 0 0 0 0 0 0 n × 64 kbit/s 0 1 0 0 0 0 0 1 n × 8 kbit/s Réf.: Rec. UIT-T Q.2931
MULT	Voir tableau au § C.9		Multiplicateur de circuits, ou n × 64k/8k/300 Réf.: Rec. UIT-T Q.2931
SCRI	8003	1 octet	Méthode de recalage sur la fréquence d'horloge source Bits <u>8 7 6 5 4 3 2 1</u> 0 0 0 0 0 0 0 0 néant 0 0 0 0 0 0 0 1 SRTS 0 0 0 0 0 0 1 0 ACM Réf.: Rec. UIT-T Q.2931

Identificateur de propriété	Etiquette de propriété	Type	Valeur
ECM	8004	1 octet	Méthode de correction d'erreur Bits <u>8 7 6 5 4 3 2 1</u> 0 0 0 0 0 0 0 0 néant 0 0 0 0 0 0 0 1 FEC – Pertes 0 0 0 0 0 0 1 0 FEC – Temps Réf.: Rec. UIT-T Q.2931
SDTB	8005	Entier de 16 bits	Longueur de bloc de transfert de données structurées Longueur de bloc de service SDT CBR Réf.: Rec. UIT-T I.363.1
PFCI	8006	Entier de 8 bits	Identificateur de cellules partiellement remplies 1-47 Réf.: Rec. UIT-T I.363.1

### C.9 Capacités du circuit support

Les entrées du tableau qui font référence à la Rec. UIT-T Q.931 concernent le codage de l'élément d'information "Capacité support" de la Rec. UIT-T Q.931 et non pas l'élément d'information de couche inférieure.

Identificateur de propriété	Etiquette de propriété	Type	Valeur
TMR	9001	1 octet	Support de transmission requis (Rec. UIT-T Q.763) Bits <u>8 7 6 5 4 3 2 1</u> 0 0 0 0 0 0 0 0 parole 0 0 0 0 0 0 0 1 disponible 0 0 0 0 0 0 1 0 64 kbit/s sans restriction 0 0 0 0 0 0 1 1 3,1 kHz audio 0 0 0 0 0 1 0 0 réservé pour alternat parole (service 2)/données à 64 kbit/s sans restriction (service 1) 0 0 0 0 0 1 0 1 réservé pour alternat données à 64 kbit/s sans restriction (service 1)/parole (service 2) 0 0 0 0 0 1 1 0 64 kbit/s de préférence 0 0 0 0 0 1 1 1 2 × 64 kbit/s sans restriction 0 0 0 0 1 0 0 0 384 kbit/s sans restriction 0 0 0 0 1 0 0 1 1536 kbit/s sans restriction 0 0 0 0 1 0 1 0 1920 kbit/s sans restriction 0 0 0 0 1 0 1 1 à disponible 0 0 0 0 1 1 1 1 0 0 0 1 0 0 0 0 3 × 64 kbit/s sans restriction 0 0 0 1 0 0 0 1 4 × 64 kbit/s sans restriction 0 0 0 1 0 0 1 0 5 × 64 kbit/s sans restriction 0 0 0 1 0 0 1 1 disponible 0 0 0 1 0 1 0 0 7 × 64 kbit/s sans restriction

Identificateur de propriété	Etiquette de propriété	Type	Valeur
			0 0 0 1 0 1 0 1 8 × 64 kbit/s sans restriction 0 0 0 1 0 1 1 0 9 × 64 kbit/s sans restriction 0 0 0 1 0 1 1 1 10 × 64 kbit/s sans restriction 0 0 0 1 1 0 0 0 11 × 64 kbit/s sans restriction 0 0 0 1 1 0 0 1 12 × 64 kbit/s sans restriction 0 0 0 1 1 0 1 0 13 × 64 kbit/s sans restriction 0 0 0 1 1 0 1 1 14 × 64 kbit/s sans restriction 0 0 0 1 1 1 0 0 15 × 64 kbit/s sans restriction 0 0 0 1 1 1 0 1 16 × 64 kbit/s sans restriction 0 0 0 1 1 1 1 0 17 × 64 kbit/s sans restriction 0 0 0 1 1 1 1 1 18 × 64 kbit/s sans restriction 0 0 1 0 0 0 0 0 19 × 64 kbit/s sans restriction 0 0 1 0 0 0 0 1 20 × 64 kbit/s sans restriction 0 0 1 0 0 0 1 0 21 × 64 kbit/s sans restriction 0 0 1 0 0 0 1 1 22 × 64 kbit/s sans restriction 0 0 1 0 0 1 0 0 23 × 64 kbit/s sans restriction 0 0 1 0 0 1 0 1 disponible 0 0 1 0 0 1 1 0 25 × 64 kbit/s sans restriction 0 0 1 0 0 1 1 1 26 × 64 kbit/s sans restriction 0 0 1 0 1 0 0 0 27 × 64 kbit/s sans restriction 0 0 1 0 1 0 0 1 28 × 64 kbit/s sans restriction 0 0 1 0 1 0 1 0 29 × 64 kbit/s sans restriction 0 0 1 0 1 0 1 1 à          disponible 1 1 1 1 1 1 1 1 Réf.: Rec. UIT-T Q.763
TMRSR	9002	1 octet	Débit sous-multiple du support de transmission requis 0 non spécifié 1 8 kbit/s 2 16 kbit/s 3 32 kbit/s
Contcheck	9003	Booléen	Contrôle de continuité 0 non requis sur ce circuit 0 requis sur ce circuit Réf.: Rec. UIT-T Q.763
ITC	9004	5 bits	Capacité de transfert d'informations Bits <u>5 4 3 2 1</u> 0 0 0 0 parole 0 1 0 0 informations numériques sans restriction 0 1 0 1 informations numériques avec restriction 1 0 0 0 3,1 kHz audio 1 0 0 1 informations numériques sans restriction avec tonalités/annonces 1 1 0 0 vidéo Toutes les autres valeurs sont réservées. Réf.: Rec. UIT-T Q.763

Identificateur de propriété	Etiquette de propriété	Type	Valeur
TransMode	9005	2 bits	Mode de transfert Bits <u>2 1</u> 0 0 mode circuit 1 0 mode paquet Réf.: Rec. UIT-T Q.931
TransRate	9006	5 bits	Rapidité de transfert Bits <u>5 4 3 2 1</u> 0 0 0 0 ce code doit être utilisé pour les appels en mode paquet 1 0 0 0 64 kbit/s 1 0 0 1 2 × 64 kbit/s 1 0 0 1 384 kbit/s 1 0 1 0 1536 kbit/s 1 0 1 1 1920 kbit/s 1 1 0 0 multidébit (64 kbit/s au débit de base) Réf.: Rec. UIT-T Q.931
MULT	9007	7 bits	Multiplicateur de débit Toute valeur de 2 à n (nombre maximal de voies B) Réf.: Rec. UIT-T Q.931
layer1prot	9008	5 bits	Protocole de couche 1 des informations d'utilisateur Bits <u>5 4 3 2 1</u> 0 0 0 1 adaptation de débit normalisée UIT-T V.110 et X.30. 0 0 0 1 0 Rec. UIT-T G.711 – Loi $\mu$ 0 0 0 1 1 Rec. UIT-T G.711 – Loi A 0 0 1 0 0 Rec. UIT-T G.726 – MICDA à 32 kbit/s et Rec. UIT-T I.460 0 0 1 0 1 Rec. UIT-T H.221 et H.242 0 0 1 1 0 Rec. UIT-T H.223 et H.245 0 0 1 1 1 adaptation de débit normalisée non UIT-T. 0 1 0 0 0 adaptation de débit normalisée UIT-T V.120. 0 1 0 0 1 adaptation de débit normalisée UIT-T X.31 avec remplissage par fanions HDLC Toutes les autres valeurs sont réservées. Réf.: Rec. UIT-T Q.931

Identificateur de propriété	Etiquette de propriété	Type	Valeur
syncasync	9009	Booléen	Synchrone/Asynchrone 0 données synchrones 1 données asynchrones Réf.: Rec. UIT-T Q.931
negotiation	900A	Booléen	Négociation 0 négociation dans la bande possible 1 négociation dans la bande impossible Réf.: Rec. UIT-T Q.931
Userrate	900B	5 bits	Débit d'utilisateur Bits <u>5 4 3 2 1</u> 0 0 0 0 le débit est indiqué en bits E spécifiés dans la Rec. UIT-T I.460 ou peut être négocié dans la bande 0 0 0 1 0,6 kbit/s Recommandations UIT-T V.6 et X.1 0 0 0 1 0 1,2 kbit/s Rec. UIT-T V.6 0 0 0 1 1 2,4 kbit/s Recommandations UIT-T V.6 et X.1 0 0 1 0 0 3,6 kbit/s Rec. UIT-T V.6 0 0 1 0 1 4,8 kbit/s Recommandations UIT-T V.6 et X.1 0 0 1 1 0 7,2 kbit/s Rec. UIT-T V.6 0 0 1 1 1 8 kbit/s Rec. UIT-T I.460 0 1 0 0 0 9,6 kbit/s Recommandations UIT-T V.6 et X.1 0 1 0 0 1 14,4 kbit/s Rec. UIT-T V.6 0 1 0 1 0 16 kbit/s Rec. UIT-T I.460 0 1 0 1 1 19,2 kbit/s Rec. UIT-T V.6 0 1 1 0 0 32 kbit/s Rec. UIT-T I.460 0 1 1 0 1 38,4 kbit/s Rec. UIT-T V.110 0 1 1 1 0 48 kbit/s Recommandations UIT-T V.6 et X.1 0 1 1 1 1 56 kbit/s Rec. UIT-T V.6 1 0 0 1 0 57,6 kbit/s Rec. UIT-T V.14 étendue 1 0 0 1 1 28,8 kbit/s Rec. UIT-T V.110 1 0 1 0 0 24 kbit/s Rec. UIT-T V.110 1 0 1 0 1 0,1345 kbit/s Rec. UIT-T X.1 1 0 1 1 0 0,100 kbit/s Rec. UIT-T X.1 1 0 1 1 1 0,075/1,2 kbit/s Recommandations UIT-T V.6 et X.1 1 1 0 0 0 1,2/0,075 kbit/s Recommandations UIT-T V.6 et X.1 1 1 0 0 1 0,050 kbit/s Recommandations UIT-T V.6 et X.1 1 1 0 1 0 0,075 kbit/s Recommandations UIT-T V.6 et X.1 1 1 0 1 1 0,110 kbit/s Recommandations UIT-T V.6 et X.1 1 1 1 0 0 0,150 kbit/s Recommandations UIT-T V.6 et X.1 1 1 1 0 1 0,200 kbit/s Recommandations UIT-T V.6 et X.1 1 1 1 1 0 0,300 kbit/s Recommandations UIT-T V.6 et X.1 1 1 1 1 1 12 kbit/s Rec. UIT-T V.6 Toutes les autres valeurs sont réservées. Réf.: Rec. UIT-T Q.931

Identificateur de propriété	Etiquette de propriété	Type	Valeur
INTRATE	900C	2 bits	Débit intermédiaire Bits <u>2 1</u> 0 0 inutilisé 0 1 8 kbit/s 1 0 16 kbit/s 1 1 32 kbit/s Réf.: Rec. UIT-T Q.931
nictx	900D	Booléen	Horloge indépendante du réseau (NIC, <i>network independent clock</i> ) à l'émission 0 non requis pour envoyer des données avec horloge indépendante du réseau 1 requis pour envoyer des données avec horloge indépendante du réseau Réf.: Rec. UIT-T Q.931
nicrx	900E	Booléen	Horloge indépendante du réseau (NIC) à la réception 0 ne peut pas accepter de données avec horloge indépendante du réseau (c'est-à-dire que l'émetteur ne prend pas en charge cette procédure facultative) 1 peut accepter des données avec horloge indépendante du réseau (c'est-à-dire que l'émetteur prend bien en charge cette procédure facultative) Réf.: Rec. UIT-T Q.931
flowconttx	900F	Booléen	Commande de débit à l'émission (Tx, <i>transmission</i> ) 0 non requis pour envoyer des données avec mécanisme de commande de débit 1 requis pour envoyer des données avec mécanisme de commande de débit Réf.: Rec. UIT-T Q.931
flowcontrx	9010	Booléen	Commande de débit à la réception (Rx, <i>reception</i> ) 0 ne peut pas accepter de données avec mécanisme de commande de débit (c'est-à-dire que l'émetteur ne prend pas en charge cette procédure facultative) 1 peut accepter des données avec mécanisme de commande de débit (c'est-à-dire que l'émetteur prend bien en charge cette procédure facultative) Réf.: Rec. UIT-T Q.931
rateadapthdr	9011	Booléen	En-tête/pas d'en-tête d'adaptation de débit 0 en-tête d'adaptation de débit non inclus 1 en-tête d'adaptation de débit inclus Réf.: Rec. UIT-T Q.931

Identificateur de propriété	Etiquette de propriété	Type	Valeur
multiframe	9012	Booléen	Prise en charge de l'établissement de multitrames dans la liaison de données 0 établissement de multitrames non pris en charge. Seules les trames UI sont autorisées 1 prise en charge de l'établissement de multitrames Réf.: Rec. UIT-T Q.931
OPMODE	9013	Booléen	Mode de fonctionnement 0 mode de fonctionnement transparent aux bits 1 mode de fonctionnement sensible au protocole Réf.: Rec. UIT-T Q.931
llidnegot	9014	Booléen	Négociation d'identificateur de liaison logique 0 valeur par défaut, LLI = 256 seulement 1 négociation de protocole complète Réf.: Rec. UIT-T Q.931
assign	9015	Booléen	Commettant/Commissionnaire 0 l'expéditeur du message est "attribué par défaut" 1 l'expéditeur du message est "attributeur seulement" Réf.: Rec. UIT-T Q.931
inbandneg	9016	Booléen	Négociation dans la bande/hors bande 0 négociation effectuée avec les messages d'INFORMATION UTILISATEUR sur une connexion de signalisation temporaire 1 la négociation est effectuée dans la bande au moyen de la liaison logique zéro Réf.: Rec. UIT-T Q.931
stopbits	9017	2 bits	Nombre de bits d'arrêt Bits <u>2 1</u> 0 0 inutilisé 0 1 1 bit 1 0 1,5 bit 1 1 2 bits Réf.: Rec. UIT-T Q.931
databits	9018	2 bits	Nombre de bits de données, non compris le bit de parité si présent Bits <u>2 1</u> 0 0 inutilisé 0 1 5 bits 1 0 7 bits 1 1 8 bits Réf.: Rec. UIT-T Q.931

Identificateur de propriété	Etiquette de propriété	Type	Valeur
parity	9019	3 bits	Informations de parité Bits <u>3 2 1</u> 0 0 0 impaire 0 1 0 paire 0 1 1 aucune 1 0 0 forcée à 0 1 0 1 forcée à 1 Toutes les autres valeurs sont réservées. Réf.: Rec. UIT-T Q.931
duplexmode	901A	Booléen	Mode duplex 0 semi-duplex 1 duplex Réf.: Rec. UIT-T Q.931
modem	901B	6 bits	Type de modem Bits <u>6 5 4 3 2 1</u> 0 0 0 0 0 0 à usage national 0 0 0 1 0 1 0 1 0 0 0 1 Rec. UIT-T V.21 0 1 0 0 1 0 Rec. UIT-T V.22 0 1 0 0 1 1 Rec. UIT-T V.22 <i>bis</i> 0 1 0 1 0 0 Rec. UIT-T V.23 0 1 0 1 0 1 Rec. UIT-T V.26 0 1 1 0 0 1 Rec. UIT-T V.26 <i>bis</i> 0 1 0 1 1 1 Rec. UIT-T V.26 <i>ter</i> 0 1 1 0 0 0 Rec. UIT-T V.27 0 1 1 0 0 1 Rec. UIT-T V.27 <i>bis</i> 0 1 1 0 1 0 Rec. UIT-T V.27 <i>ter</i> 0 1 1 0 1 1 Rec. UIT-T V.29 0 1 1 1 0 1 Rec. UIT-T V.32 0 1 1 1 1 0 Rec. UIT-T V.34 1 0 0 0 0 0 à usage national 1 0 1 1 1 1 1 1 0 0 0 0 à spécifié par l'utilisateur 1 1 1 1 1 1 Réf.: Rec. UIT-T Q.931

Identificateur de propriété	Etiquette de propriété	Type	Valeur
layer2prot	901C	5 bits	Protocole de couche 2 des informations d'utilisateur Bits <u>5 4 3 2 1</u> 0 0 0 1 0 Rec. UIT-T Q.921/I.441 0 0 1 1 0 Rec. UIT-T X.25, couche Liaison 0 1 1 0 0 commande de liaison logique de réseau local (ISO/CEI 8802-2) Toutes les autres valeurs sont réservées. Réf.: Rec. UIT-T Q.931
layer3prot	901D	5 bits	Protocole de couche 3 des informations d'utilisateur Bits <u>5 4 3 2 1</u> 0 0 0 1 0 Rec. UIT-T Q.931 0 0 1 1 0 Rec. UIT-T X.25, couche paquet 0 1 0 1 1 ISO/CEI TR 9577 (identification de protocole dans la couche Réseau) Toutes les autres valeurs sont réservées. Réf.: Rec. UIT-T Q.931
addlayer3prot	901E	Octet	Protocole de couche 3 additionnel des informations d'utilisateur Bits <u>4 3 2 1</u> <u>4 3 2 1</u> 1 1 0 0 1 1 0 0 protocole Internet (RFC 791) (ISO/CEI TR 9577) 1 1 0 0 1 1 1 1 protocole point à point (RFC 1661) Réf.: Rec. UIT-T Q.931
DialledN	901F	30 octets	Numéro appelé
DiallingN	9020	30 octets	Numéro appelant
ECHOICI	9021		Non employé. Voir un exemple des propriétés possibles de limitation d'écho au § E.13
NCI	9022	1 octet	Nature des indicateurs de connexion Bits <u>2 1</u> <i>indicateur de satellite</i> 0 0 aucun circuit satellite dans la connexion 0 1 un seul circuit satellite dans la connexion 1 0 deux circuits satellites dans la connexion 1 1 disponible  Bits <u>4 3</u> <i>indicateur de contrôle de continuité</i> 0 0 contrôle de continuité non requis 0 1 contrôle de continuité requis sur ce circuit 1 0 contrôle de continuité effectué sur un circuit précédent 1 1 disponible

Identificateur de propriété	Etiquette de propriété	Type	Valeur
			Bits 5 <i>indicateur de limiteur d'écho</i> 0 <i>limiteur d'écho sortant non inclus</i> 1 <i>limiteur d'écho sortant inclus</i>  Bits 8 7 6 <i>disponible</i> Réf.: Rec. UIT-T Q.763
USI	9023	Chaîne d'octets	Informations relatives au service utilisateur Réf.: § 3.57/Q.763

### C.10 Propriétés de couche AAL 5

Identificateur de propriété	Etiquette de propriété	Type	Valeur
FMSDU	A001	Entier de 32 bits	Longueur maximale d'unité CPCS-SDU vers l'avant: Longueur maximale d'unité CPCS-SDU envoyée dans le sens de l'appelant vers l'appelé. Réf.: Rec. UIT-T Q.2931
BMSDU	A002	Entier de 32 bits	Longueur maximale d'unité CPCS-SDU vers l'arrière: Longueur maximale d'unité CPCS-SDU envoyée dans le sens de l'appelé vers l'appelant. Réf.: Rec. UIT-T Q.2931
SSCS	voir tableau au § C.7	voir tableau au § C.7	Voir le tableau du § C.7 Valeurs supplémentaires: VPI/VCI

### C.11 Equivalents SDP

Les équivalents du protocole SDP sont soumis aux exceptions SDP décrites dans le § 7.1.8 pour le codage textuel du protocole. Par exemple, la structure générique CHOOSE est autorisée à être utilisée dans le sens MGC vers MG quel que soit le codage (binaire ou textuel) du protocole.

Identificateur de propriété	Etiquette de propriété	Type	Valeur
SDP_V	B001	Chaîne	Version du protocole Réf.: RFC 2327
SDP_O	B002	Chaîne	Identificateur de propriétaire/créateur et identificateur session Réf.: RFC 2327
SDP_S	B003	Chaîne	Nom de session Réf.: RFC 2327
SDP_I	B004	Chaîne	Identificateur de session Réf.: RFC 2327

Identificateur de propriété	Etiquette de propriété	Type	Valeur
SDP_U	B005	Chaîne	URI de descripteur Réf.: RFC 2327
SDC_E	B006	Chaîne	Adresse électronique Réf.: RFC 2327
SDP_P	B007	Chaîne	Numéro de téléphone Réf.: RFC 2327
SDP_C	B008	Chaîne	Informations de connexion Réf.: RFC 2327
SDP_B	B009	Chaîne	Informations de largeur de bande Réf.: RFC 2327
SDP_Z	B00A	Chaîne	Réglage du fuseau horaire Réf.: RFC 2327
SDP_K	B00B	Chaîne	Clé de chiffrement Réf.: RFC 2327
SDP_A	B00C	Chaîne	Zéro, un ou plusieurs attributs de session Réf.: RFC 2327
SDP_T	B00D	Chaîne	Durée de session active Réf.: RFC 2327
SDP_R	B00E	Chaîne	Zéro, une ou plusieurs répétitions Réf.: RFC 2327
SDP_M	B00F	Chaîne	Type de média, port, transport et format Réf.: RFC 2327

## C.12 H.245

Identificateur de propriété	Etiquette de propriété	Type	Valeur
OLC	C001	Chaîne d'octets	Valeur de la structure H.245 OpenLogicalChannel. Réf.: Rec. UIT-T H.245
OLCack	C002	Chaîne d'octets	Valeur de la structure H.245 OpenLogicalChannelAck. Réf.: Rec. UIT-T H.245
OLCcnf	C003	Chaîne d'octets	Valeur de la structure H.245 OpenLogicalChannelConfirm. Réf.: Rec. UIT-T H.245
OLCrej	C004	Chaîne d'octets	Valeur de la structure H.245 OpenLogicalChannelReject. Réf.: Rec. UIT-T H.245
CLC	C005	Chaîne d'octets	Valeur de la structure H.245 CloseLogicalChannel. Réf.: Rec. UIT-T H.245

Identificateur de propriété	Etiquette de propriété	Type	Valeur
CLCack	C006	Chaîne d'octets	Valeur de la structure H.245 CloseLogicalChannelAck. Réf.: Rec. UIT-T H.245
LCN	C007	Entier	Valeur du numéro de voie locale H.245, de 0 à 65535. Réf.: Rec. UIT-T H.245

## Annexe D

### Transport sur protocole IP

#### D.1 Transport sur protocoles IP/UDP au moyen du verrouillage de niveau d'application

Les messages de protocole définis dans la présente Recommandation peuvent être transmis par protocole UDP. Si aucun port n'est fourni par l'homologue (voir le § 7.2.8), il convient d'envoyer les commandes au numéro de port par défaut: 2944 pour les opérations codées au format alphanumérique, ou 2945 pour celles qui sont codées au format binaire. Les réponses doivent être envoyées à l'adresse et au port d'origine des commandes correspondantes.

Le verrouillage ALF consiste en un ensemble de techniques qui permettent à une application, contrairement à un empilement, de décider de la manière dont seront envoyés les messages vers l'autre côté. Une technique de verrouillage ALF caractéristique est de permettre à une application de modifier l'ordre d'envoi des messages lorsque, après la mise en file d'attente de ceux-ci, cette file s'allonge encore. Il n'existe aucune spécification formelle concernant le verrouillage ALF. Les procédures indiquées à l'Annexe D.1 contiennent un ensemble minimal de propositions de comportements de verrouillage ALF.

Il convient que les réalisateurs utilisant IP/UDP avec ALF connaissent les restrictions de MTU relatives à la taille maximale des messages.

##### D.1.1 Fourniture de la capacité fonctionnelle d'une fois au plus

Les messages transportés sur le protocole UDP peuvent subir des pertes. En l'absence de réponse opportune, les commandes sont répétées. La plupart des commandes ne sont pas idempotentes. L'état de la passerelle MG deviendra imprévisible si, par exemple, des commandes de type Add sont exécutées plusieurs fois. Les procédures de transmission doivent donc offrir une capacité d'une fois au plus.

Les entités de protocole homologues sont censées mémoriser une liste des réponses qu'elles ont envoyées à de récentes transactions ainsi qu'une liste des transactions qui restent en cours d'exécution. L'identificateur de transaction de chaque message entrant est comparé aux identificateurs de transaction des réponses récentes envoyées avec le même paramètre MID. S'il y a une correspondance, l'entité n'exécute pas la transaction mais répète simplement la réponse. S'il n'y a aucune correspondance, le message est comparé à la liste des transactions couramment en attente. S'il y a une correspondance dans cette liste, indiquant une transaction dupliquée, l'entité n'exécute pas la transaction (voir le § D.1.4 pour les procédures concernant l'envoi de TransactionPending (Transaction en attente ou en cours)).

La procédure utilise une valeur de temporisation plus longue, appelée ci-après "LONG-TIMER". La temporisation doit être réglée de façon à être supérieure à la durée maximale d'une transaction afin de tenir compte du nombre maximal de répétitions, de la valeur maximale du compteur de

répétitions et du temps de propagation maximal d'un paquet dans le réseau. Une valeur de 30 s est suggérée.

La copie des réponses peut être détruite soit LONG-TIMER secondes après l'envoi de la réponse ou lorsque la passerelle de l'entité reçoit – au moyen du "paramètre d'acquiescement de réponse" – confirmation du fait que la réponse a été reçue. Pour les transactions qui sont acquiescées au moyen de ce paramètre, l'entité doit conserver une copie de l'identificateur de transaction (TransactionID) pendant LONG-TIMER secondes après l'envoi de la réponse, afin de détecter et d'ignorer les copies miroirs de la demande de transaction qui pourraient être produites par le réseau.

## **D.1.2 Identificateurs de transaction et dialogue à trois**

### **D.1.2.1 Identificateurs de transaction**

Les identificateurs de transaction sont des nombres entiers de 32 bits. Un contrôleur MGC peut décider d'utiliser un espace numérique spécifique pour chacune des passerelles MG qu'il gère. Il peut également utiliser le même espace numérique pour toutes les passerelles MG qui appartiennent à un même groupe arbitraire. Les contrôleurs MGC peuvent décider de partager entre plusieurs processus indépendants la charge de gérer une grande passerelle MG. Ces processus partageront le même espace numérique de transaction. Il existe de multiples implémentations possibles de cette mise en commun, par exemple une attribution centralisée des identificateurs de transaction ou une préaffectation à différents processus d'étendues d'identificateurs ne se chevauchant pas. Les implémentations doivent garantir que des identificateurs de transaction uniques sont attribués à toutes les transactions issues d'un contrôleur MGC logique (identificateurs MID identiques). Les passerelles MG peuvent simplement détecter les transactions en double en verrouillant au niveau de l'identificateur de transaction et de l'identificateur MID seulement.

### **D.1.2.2 Dialogue à trois**

Le paramètre d'acquiescement de réponse de transaction peut se trouver dans un message quelconque. Ce paramètre contient le paquetage des "étendues d'identificateurs de transaction confirmés". Les entités peuvent choisir de supprimer les copies des réponses aux transactions dont l'identificateur est inclus dans les "étendues d'identificateurs de transaction confirmés" qui ont été reçues dans les messages de réponse concernant les transactions. Ces entités doivent ignorer sans notification les commandes ultérieures si l'identificateur de transaction ne se trouve pas dans ces étendues.

Les valeurs des "étendues d'identificateurs de transaction confirmés" ne doivent pas être utilisées si plus de LONG-TIMER secondes se sont écoulées depuis que la passerelle MG a envoyé sa dernière réponse au contrôleur MGC ou si une passerelle MG reprend son fonctionnement. Dans cette situation, il y a lieu d'accepter et de traiter les transactions sans effectuer de test sur l'identificateur de transaction.

Les messages qui acheminent le paramètre "Acquiescement de réponse de transaction (TransactionResponseAcknowledgement)" peuvent être transmis dans un ordre quelconque. L'entité doit conserver "les étendues d'identificateurs de transaction confirmés" reçues, pendant LONG-TIMER secondes.

Dans le codage binaire, si seul l'élément `firstAck` est présent dans l'acquiescement d'une réponse (voir le § A.2), une seule transaction est acquiescée. Si les deux éléments `firstAck` et `lastAck` sont présents, c'est l'étendue des transactions comprises entre le premier et le dernier acquiescement qui est acquiescée. Dans le codage alphanumérique, un trait horizontal est utilisé pour indiquer une plage de transactions en cours d'acquiescement (voir le § B.2).

### **D.1.3 Calcul des temporisations de retransmission**

Il appartient à l'entité émettrice de la demande de fournir des temporisations appropriées pour toutes les transactions en cours et de réessayer les transactions dont la temporisation a été dépassée. Par ailleurs, lorsque des transactions répétées n'obtiennent pas d'acquiescement, il appartient à l'entité

émettrice de la demande de rechercher des services redondants et de libérer des connexions actives ou en instance.

La spécification évite à dessein de prescrire de quelconques valeurs pour les temporisateurs de retransmission. Ces valeurs dépendent normalement du réseau. Il y a lieu que les temporisateurs de retransmission estiment normalement leur valeur en mesurant la durée écoulée entre l'envoi d'une commande et le retour d'une réponse. Les implémentations doivent garantir que l'algorithme utilisé pour calculer la temporisation de retransmission augmente exponentiellement la temporisation de retransmission pour chaque retransmission ou pour chaque répétition après la première retransmission.

NOTE – Une possibilité consiste à utiliser l'algorithme implémenté dans le protocole TCP-IP, qui fait appel à deux variables comme suit:

- le temps d'acquiescement moyen (AAD, *average acknowledgement delay*), qui est estimé par un calcul de moyenne lissée exponentiellement des temps observés;
- l'écart type d'Allan (ADEV, *Allan deviation*) qui est estimé par un calcul de moyenne lissée exponentiellement de la valeur absolue de la différence entre le temps observé et la moyenne actuelle. Dans le protocole TCP, le temporisateur de retransmission est réglé à la somme du temps moyen plus N fois l'écart type d'Allan. Il convient de limiter la valeur maximale du temporisateur pour le protocole défini dans la présente Recommandation afin de garantir qu'aucun paquet répété ne sera reçu par les passerelles après LONG-TIMER secondes. Une valeur maximale de 4 s est suggérée.

Après toute retransmission, il convient que l'entité effectue les opérations suivantes:

- doubler la valeur estimée du temps d'acquiescement moyen (AAD);
- calculer une valeur aléatoire, répartie uniformément entre 0,5 AAD et AAD;
- régler le temporisateur de retransmission sur la somme de cette valeur aléatoire et N fois l'écart type d'Allan.

Cette procédure a deux effets. Comme elle comporte un composant qui augmente exponentiellement, elle ralentit automatiquement le flux de messages en cas d'encombrement. Comme elle comporte également un composant aléatoire, elle rompt l'éventuelle synchronisation entre les notifications déclenchées par le même événement externe.

#### **D.1.4 Réponses provisoires**

L'exécution de certaines transactions peut demander beaucoup de temps. De longs délais d'exécution peuvent entrer en interaction avec la procédure de retransmission temporisée et donner lieu soit à un nombre anormal de retransmissions ou à des valeurs de temporisation devenant trop longues pour être efficaces. Les entités qui peuvent prédire qu'une transaction nécessitera un long temps d'exécution peuvent envoyer une réponse provisoire, de type "transaction en cours". Il convient qu'elles envoient cette réponse si elles reçoivent une répétition d'une transaction encore en cours d'exécution.

Les entités qui reçoivent un message de "transaction en cours" doivent commuter sur un temporisateur de répétition différent pour les transactions répétitives. A la réception d'une réponse finale suivant la réception de réponses provisoires, une confirmation immédiate doit être envoyée et les temporisateurs de répétition normaux envoyés par la suite. Une entité qui envoie une réponse provisoire doit inclure le champ `immAckRequired` dans la réponse finale qui suit en indiquant qu'une confirmation immédiate est attendue. La réception d'une transaction en attente après une réponse doit être ignorée.

#### **D.1.5 Répétitions des demandes, réponses et acquiescements**

Le protocole est organisé sous la forme d'un paquetage de transactions dont chacune se compose d'une demande et d'une réponse couramment appelée *acquiescement*. Les messages du protocole, acheminés par protocole UDP, peuvent subir des pertes. En l'absence de réponse opportune, les

transactions sont répétées. Les entités sont censées mémoriser une liste des réponses qu'elles ont envoyées à des transactions récentes, c'est-à-dire une liste de toutes les réponses qu'elles ont envoyées au cours des dernières LONG-TIMER secondes, ainsi qu'une liste des transactions qui sont en cours d'exécution.

Le mécanisme de répétition est utilisé pour éviter trois types d'erreur possibles:

- les erreurs de transmission, lorsque par exemple un paquet est perdu à cause du bruit en ligne ou d'un encombrement dans une file d'attente;
- les défaillances de composants, lorsque par exemple une interface avec une entité devient indisponible;
- les défaillance d'entité, lorsque par exemple une entité entière devient indisponible.

Les entités doivent être en mesure de déduire, de l'historique enregistré, une estimation du taux de pertes de paquets en raison d'erreurs de transmission. Dans un système correctement configuré, ce taux de pertes doit être maintenu à une valeur très basse, normalement inférieure à 1%. Si un contrôleur MGC ou une passerelle MG doit répéter un message plus d'un certain nombre de fois, il est tout à fait normal de partir du principe qu'il se produit autre chose qu'une erreur de transmission. Par exemple, pour un taux de pertes de 1%, la probabilité que cinq tentatives de transmission consécutives échouent est de  $10^{-11}$ , soit un événement qui devrait se produire moins souvent que tous les 10 jours pour une passerelle MG traitant 1000 transactions par seconde. (En fait, le nombre de répétitions qui est considéré comme excessif devrait être fonction du taux de pertes de paquets observé.) Il convient de remarquer que le "seuil de suspicion", que l'on appellera "Max1" est normalement inférieur au "seuil de déconnexion", qui devrait être fixé à une valeur beaucoup plus grande.

Un algorithme classique de retransmission comptera simplement le nombre de répétitions successives et conclura que l'association est rompue après la réémission du même paquet un nombre de fois excessif (normalement entre 7 et 11 fois). Pour tenir compte de la possibilité d'une reprise sur défaillance non détectée ou en cours, l'on modifie l'algorithme classique de façon que, si la passerelle MG reçoit un message ServiceChange valide, annonçant une reprise sur défaillance, cette passerelle commence à envoyer les commandes en instance à ce nouveau contrôleur MGC. Les réponses aux commandes continueront d'être envoyées à l'adresse source de chaque commande.

Afin d'assurer une adaptation automatique à la charge du réseau, la présente Recommandation spécifie des temporisateurs à croissance exponentielle. Si la temporisation initiale est réglée à 200 ms, la perte d'une cinquième retransmission sera détectée au bout d'environ 6 s. C'est sans doute un temps d'attente acceptable pour détecter une reprise sur défaillance. Les répétitions devront continuer après ce délai, non seulement pour résoudre un éventuel problème de connectivité transitoire mais une attente totale de 30 s est sans doute acceptable aussi pour donner un peu plus de temps pour l'exécution d'une reprise sur défaillance.

Il importe cependant que le temps maximal alloué aux retransmissions soit limité. Avant toute retransmission, l'on vérifie que le temps écoulé depuis l'envoi du datagramme initial n'est pas plus grand que T-MAX. Si une durée supérieure à T-MAX s'est écoulée, la passerelle MG en conclut que le contrôleur MGC a subi une défaillance et elle commence son processus de reprise comme décrit au § 11.5. La passerelle MG doit utiliser une commande ServiceChange avec le paramètre ServiceChangeMethod réglé sur Disconnected de façon que le nouveau contrôleur MGC sache que la passerelle MG a perdu une ou plusieurs transactions. La valeur T-MAX est associée à la valeur LONG-TIMER, qui est obtenue par addition à T-MAX du temps maximal de propagation dans le réseau.

## **D.2 Utilisation du protocole TCP**

Les messages de protocoles définis dans la présente Recommandation peuvent être transmis sur le protocole TCP. Si aucun accès n'est spécifié par l'autre côté (voir le § 7.2.8), les commandes doivent être envoyées à l'accès par défaut. Le protocole a des messages pour unité de transfert alors que le protocole TCP est un protocole orienté vers le flux. TPKT, conformément à RFC 1006, doit être utilisé pour détailler les messages à l'intérieur du flux TCP.

Dans un protocole orienté vers les transactions, il reste possible que des demandes ou des réponses transactionnelles soient perdues. A ce propos, il est recommandé que les entités utilisant le transport TCP implémentent des temporisateurs de niveau d'application pour chaque demande et chaque réponse, comme ceux qui sont spécifiés pour le verrouillage des niveaux d'application en protocole UDP.

### **D.2.1 Fourniture de la capacité fonctionnelle d'une fois au plus**

Les messages acheminés par le protocole TCP ne sont pas exposés aux pertes de transport mais, dans les implémentations concrètes, on peut noter des pertes de demande ou de réponse transactionnelle. En l'absence de réponse opportune, les commandes sont répétées. La plupart des commandes ne sont pas idempotentes. L'état de la passerelle MG deviendra imprévisible si, par exemple, des commandes de type Add sont exécutées plusieurs fois.

Afin d'éviter de telles pertes, il est recommandé que les entités suivent les procédures du § D.1.1.

### **D.2.2 Identificateurs de transaction et dialogue à trois**

Pour les mêmes raisons, il est possible que des réponses de transaction effectuée soient perdues malgré un protocole d'acheminement fiable comme le TCP. Il est recommandé que les entités suivent les procédures du § D.1.2.2.

### **D.2.3 Calcul des temporisations de retransmission**

Avec un acheminement fiable, l'incidence des pertes de demandes/réponses de transaction est censée être très faible. De simples mécanismes de temporisation seront donc suffisants. Des algorithmes de ralentissement exponentiel ne devraient pas être nécessaires bien qu'ils puissent être employés lorsque, dans un contrôleur MGC par exemple, le code correspondant est déjà requis car les contrôleurs MGC doivent implémenter le protocole ALF/UDP en plus du protocole TCP.

### **D.2.4 Réponses provisoires**

Comme avec le protocole UDP, l'exécution de certaines transactions peut demander beaucoup de temps. Les entités qui peuvent prédire qu'une transaction nécessitera un long temps d'exécution peuvent envoyer une réponse provisoire, de type "transaction en cours". Elles doivent envoyer cette réponse si elles reçoivent une répétition d'une transaction qui est encore en cours d'exécution.

Les entités qui reçoivent un message de "transaction en cours" doivent commuter sur un temporisateur de répétition plus long pour cette transaction.

Les entités doivent conserver les transactions et les réponses jusqu'à ce qu'elles aient été confirmées. La procédure de base du § D.1.4 devrait être suivie mais de simples valeurs de temporisation devraient suffire. Il n'est pas nécessaire d'envoyer une confirmation immédiate à la réception d'une réponse finale.

### **D.2.5 Ordonnancement des commandes**

Le protocole TCP assure un acheminement ordonné des transactions. Aucune procédure spéciale n'est requise. Il convient de noter que le protocole ALF/UDP permet à une entité expéditrice de modifier son comportement en cas d'encombrement. Il permet en particulier à cette entité de réordonner les transactions lorsqu'un encombrement est détecté. Le protocole TCP ne peut pas donner les mêmes résultats.

## Annexe E

### Paquetages de base

La présente annexe contient les définitions de quelques paquetages à utiliser dans le cadre de la présente Recommandation.

#### E.1 Paquetage générique

**Nom du paquetage:** paquetage générique

**Identificateur de paquetage:** g (0x0001)

**Description:** paquetage générique pour les items communément rencontrés

**Version:** 2

**Extension:** aucune

##### E.1.1 Propriétés

Aucune.

##### E.1.2 Événements

###### E.1.2.1 Cause

**Nom de l'événement:** cause

**Identificateur d'événement:** cause (0x0001)

**Description:** événement d'erreur générique

**Paramètres de descripteur "Events" ():** aucun

**Paramètres de descripteur "ObservedEvents" (événements observés):**

###### Cause d'ordre général

**Nom du paramètre:** cause générale

**Identificateur de paramètre:** Generalcause (0x0001)

**Description:** ce paramètre répartit les défaillances en six groupes sur lesquels le contrôleur MGC peut agir.

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:**

"NR" (0x0001)	libération normale	(normal release)
"UR" (0x0002)	ressources non disponibles	(unavailable resources)
"FT" (0x0003)	défaillance, temporaire	(failure, temporary)
"FP" (0x0004)	défaillance, permanente	(failure, permanent)
"IW" (0x0005)	erreur d'interfonctionnement	(interworking error)
"UN" (0x0006)	événement non pris en charge	(unsupported)

**Valeur par défaut:** aucune

### **Cause de la défaillance**

**Nom du paramètre:** cause de défaillance

**Identificateur de paramètre:** Failurecause (0x0002)

**Description:** la cause de la défaillance est la valeur produite par l'équipement libéré, c'est-à-dire une connexion réseau libérée. La valeur concernée est définie dans le protocole approprié de commande de support.

**Type:** OCTET STRING

**Facultatif:** oui

**Valeurs possibles:** OCTET STRING

**Valeur par défaut:** aucune

### **E.1.2.2 Achèvement de signal**

**Nom de l'événement:** achèvement de signal

**Identificateur d'événement:** sc (0x0002)

**Description:** indique la fin d'un signal pour lequel le paramètre NotifyCompletion était réglé de manière à permettre le rapport d'un événement d'achèvement. Pour une description procédurale plus poussée, voir les § 7.1.1, 7.1.11 et 7.2.7.

**Paramètres descripteur "Events":** aucun

**Paramètres de descripteur ObservedEvents (événements observés):**

#### **Identité du signal**

**Nom du paramètre:** identité du signal

**Identificateur de paramètre:** SigID (0x0001)

**Description:** ce paramètre identifie le signal qui s'est achevé. Lorsqu'un signal est contenu dans une liste de signaux (SignalList), le paramètre d'identification de cette liste (SignalListID), indiquant la liste appropriée, devrait aussi être renvoyé.

**Type:** binaire: octet (chaîne); texte: chaîne

**Facultatif:** non

**Valeurs possibles:** un signal qui s'est achevé. Un signal doit être identifié à l'aide de la syntaxe de pkgdName sans utiliser de remplacement par des structures génériques.

**Valeur par défaut:** aucune

#### **Méthode de terminaison**

**Nom du paramètre:** méthode de terminaison

**Identificateur de paramètre:** Meth (0x0002)

**Description:** indique comment le signal s'est achevé.

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:**

"TO" (0x0001) signal ayant expiré ou s'étant achevé autrement de lui-même

"EV" (0x0002) interrompu par un événement  
"SD" (0x0003) arrêté par un nouveau descripteur "Signals"  
"NC" (0x0004) inachevé pour une autre raison  
"PI" (0x0005) de la première à l'avant-dernière itération. Pour la dernière itération, utiliser TO.

**Valeur par défaut:** aucune

#### **Identificateur de la liste de signaux**

**Nom du paramètre:** identificateur de la liste de signaux

**Identificateur de paramètre:** SLID (0x0003)

**Description:** indique la liste de signaux SignalList à laquelle un signal appartient. L'identificateur de la liste de signaux n'est renvoyé que si le signal figure sur une liste de signaux SignalList.

**Type:** entier

**Facultatif:** oui (car il n'est pris en charge que si des listes de signaux sont utilisées)

**Valeurs possibles:** de 1 à 65535

**Valeur par défaut:** aucune

#### **Identificateur de requête**

**Nom du paramètre:** identificateur requête

**Identificateur de paramètre:** RID (0x0004)

**Description:** indique la requête NotifyCompletion à laquelle l'identificateur de signal est associé

**Type:** entier

**Facultatif:** oui (car il n'est pris en charge que si des listes de signaux sont utilisées)

**Valeurs possibles:** de 1 à 4294967295

**Valeur par défaut:** aucune

### **E.1.3 Signaux**

Aucun.

### **E.1.4 Statistiques**

Aucune.

## **E.2 Paquetage racine de base**

**Nom du paquetage:** paquetage racine de base

**Identificateur de paquetage:** root (0x0002)

**Description:** ce paquetage définit les propriétés à l'échelle de la passerelle.

**Version:** 2

**Extension:** aucune

## **E.2.1 Propriétés**

### **E.2.1.1 Nombre maximal de contextes**

**Nom de la propriété:** MaxNrOfContexts

**Identificateur de propriété:** maxNumberOfContexts (0x0001)

**Description:** la valeur de cette propriété donne le nombre maximal de contextes qui peuvent exister à n'importe quel moment. Le contexte NULL (néant) n'est pas compris dans ce nombre.

**Type:** double

**Valeurs possibles:** 1 et plus

**Valeur par défaut:** aucune

**Défini dans:** TerminationState

**Caractéristiques:** lecture seulement (ReadOnly)

### **E.2.1.2 Nombre maximal de terminaisons par contexte**

**Nom de la propriété:** MaxTerminationsPerContext

**Identificateur de propriété:** maxTerminationsPerContext (0x0002)

**Description:** indique le nombre maximal de terminaisons autorisées dans un contexte (voir le § 6.1)

**Type:** entier

**Valeurs possibles:** n'importe quel entier positif

**Valeur par défaut:** fournie

**Défini dans:** TerminationState

**Caractéristiques:** lecture/seule (ReadOnly)

### **E.2.1.3 Durée normale d'exécution par la passerelle MG**

**Nom de la propriété:** normalMGExecutionTime

**Identificateur de la propriété:** normalMGExecutionTime (0x0003)

**Description:** réglable par le contrôleur MGC afin d'indiquer l'intervalle pendant lequel le contrôleur MGC s'attend à une réponse à toute transaction en provenance de la passerelle MG (à l'exclusion du retard dû au réseau)

**Type:** entier

**Valeurs possibles:** tout entier positif; représente des millisecondes

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

### **E.2.1.4 Durée normale d'exécution par le contrôleur MGC**

**Nom de la propriété:** normalMGCExecutionTime

**Identificateur de la propriété:** normalMGCExecutionTime (0x0004)

**Description:** réglable par le contrôleur MGC afin d'indiquer l'intervalle pendant lequel la passerelle MG devrait s'attendre à une réponse à toute transaction issue du contrôleur MGC (à l'exclusion du retard dû au réseau).

**Type:** entier

**Valeurs possibles:** tout entier positif; représente des millisecondes

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

#### **E.2.1.5 Valeur provisoire du temporisateur de réponse issue de la passerelle MG**

**Nom de la propriété:** MGProvisionalResponseTimerValue

**Identificateur de la propriété:** MGProvisionalResponseTimerValue (0x0005)

**Description:** indique la durée pendant laquelle le contrôleur MGC devrait s'attendre à une réponse en cours (de type Pending) en provenance de la passerelle MG si une transaction ne peut pas être effectuée. Valeur initialement mise à normalMGExecutionTime plus le retard dû au réseau, mais cette durée peut être diminuée.

**Type:** entier

**Valeurs possibles:** tout entier positif; représente des millisecondes

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

#### **E.2.1.6 Valeur provisoire du temporisateur de réponse issue du contrôleur MGC**

**Nom de la propriété:** MGCProvisionalResponseTimerValue

**Identificateur de la propriété:** MGCProvisionalResponseTimerValue (0x0006)

**Description:** indique la durée pendant laquelle la passerelle MG devrait s'attendre à une réponse en cours (de type Pending) en provenance du contrôleur MGC si une transaction ne peut pas être effectuée. Valeur initialement mise à normalMGCExecutionTime plus le retard dû au réseau, mais cette durée peut être diminuée.

**Type:** entier

**Valeurs possibles:** tout entier positif; représente des millisecondes

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

#### **E.2.1.7 Limite de réponses en cours issues du contrôleur MGC**

**Nom de la propriété:** MGCOriinatedPendingLimit

**Identificateur de la propriété:** MGCOriinatedPendingLimit (0x0007)

**Description:** indique le nombre de réponses en cours (de type TransactionPending) qui peuvent être reçues en provenance du contrôleur MGC. Une fois cette limite dépassée, le contrôleur MGC devrait émettre une réponse TransactionReply avec le code d'erreur 506 ("Nombre de transactions en cours dépassé"), sinon la passerelle MG peut considérer que la transaction est erronée.

**Type:** entier

**Valeurs possibles:** tout entier positif

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

### **E.2.1.8 Limite du nombre de réponses en cours issues de la passerelle MG**

**Nom de la propriété:** MGOriatedPendingLimit

**Identificateur de la propriété:** MGOriatedPendingLimit (0x0008)

**Description:** cette propriété indique le nombre de transactions en cours qui peuvent être reçues en provenance de la passerelle MG. Lorsque cette limite est dépassée, la passerelle MG devrait émettre une réponse TransactionReply avec l'erreur 506 ("Nombre de transactions en cours dépassé"), sinon le contrôleur MGC peut considérer que la transaction est erronée.

**Type:** entier

**Valeurs possibles:** tout entier positif

**Valeur par défaut:** fournie

**Défini dans:** TerminationState

**Caractéristiques:** lecture/écriture

### **E.2.2 Evénements**

Aucun.

### **E.2.3 Signaux**

Aucun.

### **E.2.4 Statistiques**

Aucune.

### **E.2.5 Procédures**

Aucune.

## **E.3 Paquetage de générateur de tonalités**

**Nom du paquetage:** paquetage de générateur de tonalités

**Identificateur de paquetage:** tonegen (0x0003)

**Description:** ce paquetage définit des signaux afin de produire des tonalités audio. Il ne spécifie pas de valeurs paramétriques. Il est destiné à être extensible. Généralement, les tonalités sont définies comme un signal individuel assorti d'un paramètre, "ind", représentant le délai entre chiffres et un identificateur de tonalité à utiliser avec les tonalités restituables. La compatibilité de chaque identificateur de tonalité devrait être conservée avec toute production de la même tonalité. Les passerelles MG sont censées être configurées avec les caractéristiques des tonalités appropriées au pays dans lequel ces passerelles sont situées.

**Version:** 2

**Conçu pour être étendu seulement:** oui

**Extension:** aucune

### **E.3.1 Propriétés**

Aucune.

### **E.3.2 Evénements**

Aucun.

### **E.3.3 Signaux**

#### **E.3.3.1 Tonalité restituable**

**Nom du signal:** tonalité restituable

**Identificateur de signal:** pt (0x0001)

**Description:** restitution d'une tonalité audio dans une voie audio

**Type de signal:** bref

**Durée:** fournie

**Paramètres additionnels:**

##### **Liste d'identificateurs de tonalité**

**Nom du paramètre:** liste d'identificateurs de tonalité

**Identificateur de paramètre:** tl (0x0001)

**Description:** liste de tonalités à restituer en séquence.

**Type:** sous-liste d'énumération

**Facultatif:** non

**Valeurs possibles:** cette sous-liste doit contenir un ou plusieurs identificateurs de tonalité.

**Valeur par défaut:** aucune

##### **Durée entre signaux**

**Nom du paramètre:** durée entre signaux

**Identificateur de paramètre:** ind (0x0002)

**Description:** temporisation entre deux tonalités consécutives, en millisecondes

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** tout entier supérieur ou égal à 0

**Valeur par défaut:** fournie

##### **Sens de la tonalité**

**Nom du paramètre:** sens de la tonalité

**Identificateur de paramètre:** btd (0x0003)

**Description:** sens dans lequel il convient que la tonalité progresse à partir de la terminaison.

**Type:** énumération

**Facultatif:** oui

**Valeurs possibles:** "EXT" (0x0001) (vers l'extérieur)  
"INT" (0x0002) (vers l'intérieur)  
"BOTH" (0x0003) (dans les deux sens)

**Valeur par défaut:** vers l'extérieur

#### E.3.4 Statistiques

Aucune.

#### E.3.5 Procédures

Aucun identificateur de tonalité n'est spécifié dans ce paquetage. Les paquetages qui étendent ce paquetage peuvent ajouter des valeurs possibles pour l'identificateur de tonalité; ils peuvent également ajouter des signaux de tonalité individuels.

### E.4 Paquetage de détection de tonalité

**Nom du paquetage:** paquetage de détection de tonalité

**Identificateur de paquetage:** tonedet (0x0004)

**Description:** ce paquetage définit les événements relatifs à la détection de tonalité audio. Les tonalités sont sélectionnées par leur nom (identificateur de tonalité). Les passerelles MG sont censées être profilées avec les caractéristiques des tonalités appropriées au pays dans lequel la passerelle MG est située.

Ce paquetage ne spécifie pas de valeurs paramétriques. Il est destiné à être extensible.

**Version:** 1

**Conçu pour être étendu seulement:** oui

**Extension:** aucune

#### E.4.1 Propriétés

Aucune.

#### E.4.2 Événements

##### E.4.2.1 Détection de début de tonalité

**Nom de l'événement:** détection de début de tonalité

**Identificateur d'événement:** std (0x0001)

**Description:** détecte le début d'une tonalité. Les caractéristiques d'une détection positive de tonalité dépendent de l'implémentation.

**Paramètres de descripteur "Events":**

##### Liste d'identificateurs de tonalité

**Identificateur de paramètre:** tl (0x0001)

**Type:** sous-liste d'énumération

**Facultatif:** non

**Valeurs possibles:** le seul identificateur de tonalité défini dans ce paquetage est "structure générique", qui est l'astérisque "\*" en codage alphanumérique et qui est "0x0000" en codage binaire. Les extensions de ce paquetage ajouteront des valeurs possibles d'identificateur de tonalité. Si le paramètre tl a la valeur "structure générique", tout identificateur de tonalité sera détecté.

**Valeur par défaut:** aucune

## Paramètres de descripteur "ObservedEvents":

### Identificateur de tonalité

**Nom du paramètre:** identificateur de tonalité

**Identificateur de paramètre:** tid (0x0003)

**Description:** identificateur de la tonalité détectée.

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:** la valeur "structure générique" définie ci-dessus est la seule valeur définie dans ce paquetage. Les extensions de ce paquetage ajouteront d'autres valeurs possibles d'identificateur de tonalité.

**Valeur par défaut:** aucune

## E.4.2.2 Détection de fin de tonalité

**Nom de l'événement:** détection de fin de tonalité

**Identificateur d'événement:** etd (0x0002)

**Description:** détecte la fin d'une tonalité.

**Paramètres de descripteur "Events":**

### Liste d'identificateurs de tonalité

**Nom du paramètre:** liste d'identificateurs de tonalité

**Identificateur de paramètre:** tl (0x0001)

**Description:** liste d'identificateurs de tonalité à détecter.

**Type:** sous-liste de types énumérés

**Facultatif:** non

**Valeurs possibles:** aucune valeur possible n'est spécifiée dans ce paquetage. Des extensions de ce paquetage ajouteront des valeurs possibles d'identificateur de tonalité.

**Valeur par défaut:** aucune

**Paramètres du descripteur "ObservedEvents":**

### Identificateur de tonalité

**Nom du paramètre:** identificateur de tonalité

**Identificateur de paramètre:** tid (0x0003)

**Description:** identificateur de la tonalité détectée.

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:** la valeur "structure générique" définie ci-dessus est la seule valeur définie dans ce paquetage. Des extensions de ce paquetage ajouteront des valeurs possibles d'identificateur de tonalité.

**Valeur par défaut:** aucune

### Durée

**Nom du paramètre:** durée

**Identificateur de paramètre:** dur (0x0002)

**Description:** ce paramètre contient la durée de la tonalité, de sa première détection à son arrêt.

**Type:** entier  
**Facultatif:** oui  
**Valeurs possibles:** tout entier positif en millisecondes  
**Valeur par défaut:** aucune

#### **E.4.2.3 Détection de tonalité longue**

**Nom de l'événement:** détection de tonalité longue

**Identificateur d'événement:** ltd (0x0003)

**Description:** détecte qu'une tonalité est restituée pendant au moins une certaine durée

**Paramètres de descripteur "Events":**

##### **Liste d'identificateurs de tonalité**

**Nom du paramètre:** liste d'identificateurs de tonalité

**Identificateur de paramètre:** tl (0x0001)

**Description:** liste des identificateurs de tonalité à détecter.

**Type:** sous-liste d'énumération

**Facultatif:** non

**Valeurs possibles:** la valeur "structure générique" définie ci-dessus est la seule valeur définie dans ce paquetage. Des extensions de ce paquetage ajouteront des valeurs possibles d'identificateur de tonalité.

**Valeur par défaut:** aucune

##### **Durée:**

**Nom du paramètre:** durée

**Identificateur de paramètre:** dur (0x0002)

**Description:** durée de référence pour l'essai

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** tout entier positif, exprimé en millisecondes

**Valeur par défaut:** fournie

**Paramètres du descripteur "ObservedEvents":**

##### **Identificateur de tonalité:**

**Nom du paramètre:** identificateur de tonalité

**Identificateur de paramètre:** tid (0x0003)

**Description:** identificateur de la tonalité détectée

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:** aucune valeur possible n'est spécifiée dans ce paquetage. Des extensions de ce paquetage ajouteront des valeurs possibles d'identificateur de tonalité.

**Valeur par défaut:** aucune

#### **E.4.3 Signaux**

Aucun.

#### E.4.4 Statistiques

Aucune.

#### E.4.5 Procédures

Aucune.

### E.5 Paquetage de générateur de tonalités DTMF de base

**Nom du paquetage:** paquetage de générateur de tonalités DTMF de base

**Identificateur de paquetage:** dg (0x0005)

**Description:** ce paquetage définit les tonalités DTMF de base sous la forme de signaux et constitue une extension des valeurs autorisées du paramètre tl de tonalités restituables dans le paquetage tonegen.

**Version:** 2

**Extension:** tonegen version 2

#### E.5.1 Propriétés

Aucune.

#### E.5.2 Evénements

Aucun.

#### E.5.3 Signaux

##### E.5.3.1 Caractère DTMF 0

**Nom du signal:** caractère DTMF 0

**Identificateur de signal:** d0 (0x0010)

**Description:** produit une tonalité DTMF 0 dont les caractéristiques sont définies dans la passerelle.

**Type de signal:** bref

**Durée:** fournie

**Paramètres additionnels:**

##### Sens de la tonalité

**Nom du paramètre:** sens de la tonalité

**Identificateur du paramètre:** btd (0x0001)

**Description:** la tonalité devrait progresser vers l'extérieur de la terminaison

**Type:** énumération

**Facultatif:** oui

**Valeurs possibles:** "EXT" (x0001) vers l'extérieur  
"INT" (x0002) vers l'intérieur  
"BOTH" (0x0003) dans les deux sens

**Valeur par défaut:** vers l'extérieur

**Valeurs additionnelles:** d0 (0x0010) est défini comme identificateur de tonalité restituable

Les autres caractères DTMF sont spécifiés exactement de la même façon. Un tableau avec tous les noms de signaux et tous les identificateurs de signaux est inclus. Il convient de noter que chaque caractère DTMF est défini à la fois comme signal et comme identificateur de tonalité (toneID), ce

qui étend le paquetage de production de tonalités de base. Il convient également de noter que les identificateurs de signaux DTMF sont différents des noms utilisés dans un script de numérotation (DigitMap).

Nom de signal	Ident. de signal/ident. de tonalité
Caractère DTMF 0	d0 (0x0010)
Caractère DTMF 1	d1 (0x0011)
Caractère DTMF 2	d2 (0x0012)
Caractère DTMF 3	d3 (0x0013)
Caractère DTMF 4	d4 (0x0014)
Caractère DTMF 5	d5 (0x0015)
Caractère DTMF 6	d6 (0x0016)
Caractère DTMF 7	d7 (0x0017)
Caractère DTMF 8	d8 (0x0018)
Caractère DTMF 9	d9 (0x0019)
Caractère DTMF *	ds (0x0020)
Caractère DTMF #	do (0x0021)
Caractère DTMF A	da (0x001a)
Caractère DTMF B	db (0x001b)
Caractère DTMF C	dc (0x001c)
Caractère DTMF D	dd (0x001d)

#### E.5.4 Statistiques

Aucune.

#### E.5.5 Procédures

Aucune.

### E.6 Paquetage de détection de tonalités DTMF

**Nom du paquetage:** paquetage de détection de tonalités DTMF

**Identificateur de paquetage:** dd (0x0006)

**Description:** ce paquetage définit la détection des tonalités DTMF de base. Il constitue une extension des valeurs possibles d'identificateur de tonalité contenues dans les événements "détection de début de tonalité", "détection de fin de tonalité" et "détection de tonalité longue".

Les valeurs additionnelles d'identificateur de tonalité sont tous les identificateurs de tonalité décrits dans le paquetage dg (générateur de tonalités DTMF de base).

Le tableau ci-après mappe les événements DTMF avec les symboles du script de numérotation comme il est décrit au § 7.1.14.

DTMF	Symbole d'événement
d0	"0"
d1	"1"
d2	"2"
d3	"3"
d4	"4"
d5	"5"
d6	"6"
d7	"7"
d8	"8"
d9	"9"
da	"A" ou "a"
db	"B" ou "b"
dc	"C" ou "c"
dd	"D" ou "d"
ds	"E" ou "e"
do	"F" ou "f"

**Version:** 1

**Extension:** tonedet version 1

### E.6.1 Propriétés

Aucune.

### E.6.2 Événements

#### E.6.2.1 Chiffres DTMF

**Nom de l'événement:** chiffres DTMF

**Identificateur d'événement:** les identificateurs d'événements sont définis avec les mêmes noms que les identificateurs de signaux définis dans le tableau du § E.5.3.

**Description:** cet événement est produit lorsque la passerelle MG détecte un chiffre

**Paramètres de descripteur "Events":** aucun

**Paramètres de descripteur "ObservedEvents":** aucun

#### E.6.2.2 Événement d'achèvement de script de numérotation

**Nom de l'événement:** événement d'achèvement de script de numérotation

**Identificateur d'événement:** ce (0x0004)

**Description:** événement produit lorsqu'un script de numérotation s'achève comme décrit dans le § 7.1.14.

**Paramètres de descripteur "Events":** aucun

## Paramètres de descripteur "ObservedEvents":

### Chaîne de chiffres

**Nom du paramètre:** DigitString

**Identificateur de paramètre:** ds (0x0001)

**Description:** partie de la chaîne de numérotation actuelle comme décrit au § 7.1.14 qui correspond à tout ou partie de la séquence d'événement en variante spécifiée dans le script de numérotation.

**Type:** chaîne de symboles de script de numérotation (éventuellement vide) retournée sous forme de chaîne entre doubles apostrophes.

**Facultatif:** non

**Valeurs possibles:** séquence des caractères "0" à "9", "A" à "F", et le modificateur de longue durée "Z".

**Valeur par défaut:** aucune

### Méthode de terminaison

**Nom du paramètre:** méthode de terminaison

**Identificateur de paramètre:** Meth (0x0003)

**Description:** indique la raison de la création de l'événement. Voir les procédures dans le § 7.1.14.

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:**

"UM" (0x0001)	correspondance non ambiguë
"PM" (0x0002)	correspondance partielle, achèvement par l'expiration de la temporisation ou par un événement sans correspondance "FM"
"FM" (0x0003)	correspondance totale, achèvement par l'expiration de la temporisation ou par un événement sans correspondance

**Valeur par défaut:** aucune

### E.6.3 Signaux

Aucun.

### E.6.4 Statistiques

Aucune.

### E.6.5 Procédures

Le script de numérotation est activé uniquement si un descripteur "Events" est activé, qui contient un événement de fin de script de numérotation, comme défini au § E.6.2, et si cet événement de fin de script de numérotation contient un champ "EventDM" dans les actions requises, tel que défini au § 7.1.9. D'autres paramètres tels que KeepActive ou des descripteurs "Signals" ou "Events" imbriqués peuvent aussi être présents dans le descripteur "Events", sans affecter le traitement du script de numérotation.

## E.7 Paquetage de générateur de tonalités de progression d'appel

**Nom du paquetage:** paquetage de générateur de tonalités de progression d'appel

**Identificateur de paquetage:** cg (0x0007)

**Description:** ce paquetage définit les tonalités de progression d'appel de base en tant que signaux. Il constitue une extension des valeurs autorisées du paramètre tl de tonalités restituables dans le paquetage tonegen.

**Version:** 2

**Extension:** tonegen version 2

### E.7.1 Propriétés

Aucune.

### E.7.2 Evénements

Aucun.

### E.7.3 Signaux

#### E.7.3.1 Tonalité d'invitation à numéroté

**Nom du signal:** tonalité d'invitation à numéroté

**Identificateur de signal:** dt (0x0030)

**Description:** produit une tonalité d'invitation à numéroté dont les caractéristiques sont disponibles dans la passerelle.

**Type de signal:** temporisé

**Durée:** fournie

**Paramètres additionnels:** aucun

**Valeurs additionnelles:** dt (0x0030) est défini comme identificateur des tonalités restituables

Les autres tonalités de ce paquetage sont définies exactement de la même façon. Un tableau avec tous les noms de signaux et tous les identificateurs de signaux est inclus. Il convient de noter que chaque tonalité est définie à la fois comme signal et comme identificateur de tonalité, ce qui étend le paquetage de production de tonalités de base.

Nom de signal	Ident. de signal/ident. de tonalité
Tonalité d'invitation à numéroté	dt (0x0030)
Tonalité de retour d'appel	rt (0x0031)
Tonalité d'occupation	bt (0x0032)
Tonalité d'encombrement	ct (0x0033)
Tonalité spéciale d'information	sit (0x0034)
Tonalité d'avertissement (d'enregistrement)	wt (0x0035)
Tonalité d'identification de publiphone	prt (0x0036)
Tonalité d'appel en attente	cw (0x0037)
Tonalité d'attente de l'appelant	cr (0x0038)

#### **E.7.4 Statistiques**

Aucune.

#### **E.7.5 Procédures**

NOTE – L'ensemble requis d'identificateurs de tonalité correspond à celui qui est défini dans la Rec. UIT-T E.180/Q.35. L'explication de ces tonalités est donnée dans la Rec. UIT-T E.180/Q.35.

#### **E.8 Paquetage de détection de tonalités de progression d'appel**

**Nom du paquetage:** paquetage de détection de tonalités de progression d'appel

**Identificateur de paquetage:** cd (0x0008)

**Description:** ce paquetage définit les tonalités de détection de progression d'appel de base. Il constitue une extension des valeurs possibles d'identificateur de tonalité contenues dans les événements "détection de début de tonalité", "détection de fin de tonalité" et "détection de tonalité longue".

##### **Valeurs additionnelles**

Les valeurs d'identificateur de tonalités sont définies pour la détection de début de tonalité, la détection de fin de tonalité et la détection de tonalité longue avec les mêmes valeurs que celles du paquetage cg (production de tonalités de progression d'appel).

L'ensemble requis d'identificateurs de tonalité correspond à celui qui est défini dans la Rec. UIT-T E.180/Q.35. L'explication de ces tonalités est donnée dans la Rec. UIT-T E.180/Q.35.

**Version:** 1

**Extension:** tonedet version 1

#### **E.8.1 Propriétés**

Aucune.

#### **E.8.2 Événements**

Les événements sont définis comme dans le paquetage "Générateur de tonalités de progression d'appel" (cg) pour les tonalités qui sont énumérées dans le tableau du § E.7.3.

#### **E.8.3 Signaux**

Aucun.

#### **E.8.4 Statistiques**

Aucune.

#### **E.8.5 Procédures**

Aucune.

#### **E.9 Paquetage de supervision de ligne analogique**

**Nom du paquetage:** paquetage de supervision de ligne analogique

**Identificateur de paquetage:** al, (0x0009)

**Description:** ce paquetage définit les événements et les signaux pour une ligne analogique.

**Version:** 1

**Extension:** aucune

## E.9.1 Propriétés

Aucune.

## E.9.2 Evénements

### E.9.2.1 Raccrochage

**Nom de l'événement:** raccrochage

**Identificateur d'événement:** on (0x0004)

**Description:** détecte le raccrochage du combiné. Toutes les fois qu'un descripteur "Events" est activé pour demander une surveillance afin de détecter un événement de raccrochage et que la ligne est déjà raccrochée, la passerelle MG doit se comporter conformément au réglage du paramètre "strict".

**Paramètres de descripteur "Events":**

#### Transition stricte

**Nom du paramètre:** transition stricte

**Identificateur de paramètre:** strict (0x0001)

**Description:** indique la façon dont l'événement de raccrochage est détecté.

**Type:** énumération

**Facultatif:** oui

**Valeurs possibles:** "exact" (0x0000) signifie que seule une transition effective à l'état du crochet commutateur "raccroché" doit être reconnue;

"state" (0x0001) signifie que l'événement doit être reconnu si la transition à l'état du crochet commutateur est détectée ou si l'état du crochet commutateur est déjà "raccroché";

"failWrong" (0x0002) signifie quesi l'état du crochet commutateur est déjà "raccroché", la commande échoue et une erreur est renvoyée.

**Valeur par défaut:** "exact"

**Paramètres de descripteur "ObservedEvents":**

#### Etat initial

**Nom du paramètre:** état initial

**Identificateur de paramètre:** init (0x0002)

**Description:** raison pour laquelle la transition à l'état raccroché a été signalée. Cette valeur n'est renvoyée que si le paramètre de transition stricte est mis à la valeur "state".

**Type:** booléen

**Valeurs possibles:** "True" (Vrai) signifie que l'événement a été signalé parce que la ligne était déjà raccrochée lorsque le descripteur "Events" contenant cet événement a été activé;

"False" (Faux) signifie qu'une transition effective à l'état "raccroché" a été détectée.

**Valeur par défaut:** aucune

### E.9.2.2 Décrochage

**Nom de l'événement:** décrochage

**Identificateur d'événement:** of (0x0005)

**Description:** détecte le décrochage du combiné. Toutes les fois qu'un descripteur "Events" est activé pour demander une surveillance afin de détecter un événement de décrochage et que la ligne est déjà décrochée, la passerelle MG doit se comporter conformément au réglage du paramètre "strict".

**Paramètres de descripteur "Events":**

**Transition stricte**

**Nom du paramètre:** transition stricte

**Identificateur de paramètre:** strict (0x0001)

**Description:** indique la façon dont l'événement de décrochage est détecté.

**Type:** énumération

**Valeurs possibles:** "exact" (0x0000) signifie que seule une transition effective de l'état de raccrochage à l'état "décroché" doit être reconnue;

"state" (0x0001) signifie que l'événement doit être reconnu si la transition à l'état de raccrochage est détectée ou si l'état du crochet commutateur est déjà "décroché";

"failWrong" (0x0002) signifie que si l'état du crochet commutateur est déjà "décroché", la commande échoue et une erreur est renvoyée.

**Valeur par défaut:** exact

**Paramètres de descripteur "ObservedEvents":**

**Etat initial**

**Nom du paramètre:** état initial

**Identificateur de paramètre:** init (0x0002)

**Description:** raison pour laquelle la transition à l'état décroché a été signalée. Cette valeur n'est renvoyée que si le paramètre de transition stricte est mis à la valeur "state".

**Type:** booléen

**Facultatif:** oui

**Valeurs possibles:** "True" (Vrai) signifie que la ligne était déjà décrochée lorsque le descripteur "Events" contenant cet événement a été activé;

"False" (Faux) signifie qu'une transition effective à l'état "décroché" a été détectée.

**Valeur par défaut:** aucune

**E.9.2.3 Manipulation de crochet**

**Nom de l'événement:** manipulation de crochet

**Identificateur d'événement:** fl (0x0006)

**Description:** détecte la manipulation de crochet. Celle-ci correspond à un raccrochage suivi d'un décrochage dans un délai compris entre une durée minimale et une durée maximale.

**Paramètres de descripteur "Events"**

**Durée minimale**

**Nom du paramètre:** durée minimale

**Identificateur de paramètre:** mindur (0x0004)

**Description;** temps minimal, entre un raccrochage et un décrochage, qui peut être détecté comme une manipulation de crochet commutateur.

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** tout entier positif, en millisecondes

**Valeur par défaut:** fournie.

#### **Durée maximale**

**Nom du paramètre:** durée maximale

**Identificateur de paramètre:** maxdur (0x0005)

**Description :** temps maximal, entre un raccrochage et un décrochage, qui peut être détecté comme une manipulation de crochet commutateur.

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** tout entier positif, en millisecondes

**Valeur par défaut:** fournie.

**Paramètres de descripteur "ObservedEvents":** aucun

### **E.9.3 Signaux**

#### **E.9.3.1 Sonnerie**

**Nom du signal:** sonnerie

**Identificateur de signal:** ri (0x0002)

**Description:** s'applique au retour d'appel sur la ligne

**Type du signal:** temporisé

**Durée:** fournie

**Paramètres additionnels:**

##### **Cadence**

**Nom du paramètre:** cadence

**Identificateur de paramètre:** cad (0x0006)

**Description:** représente des durées de segments commutés alternativement, qui constituent un cycle de sonnerie complet à partir d'une commutation active. Les passerelles MG à fonctions restreintes peuvent ne pas tenir compte des valeurs de cadence qu'elles ne sont pas en mesure de produire.

**Type:** sous-liste d'entiers

**Facultatif:** oui

**Valeurs possibles:** tout entier positif, en millisecondes

**Valeur par défaut:** fournie

##### **Fréquence**

**Nom du paramètre:** fréquence

**Identificateur de paramètre:** freq (0x0007)

**Description:** les passerelles MG à fonctions restreintes peuvent ne pas tenir compte des valeurs de cadence qu'elles ne sont pas en mesure de produire.

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** tout entier positif, en hertz

**Valeur par défaut:** fournie

#### E.9.4 Statistiques

Aucune.

#### E.9.5 Codes d'erreur

**Code d'erreur n°:** 540

**Nom:** état initial du crochet commutateur inattendu

**Définition:** cette erreur est produite lorsque le contrôleur MGC a essayé de demander un événement de transition d'état du crochet commutateur avec le paramètre "strict" mis à la valeur "failWrong" et lorsque l'état du crochet est déjà celui que la transition implique.

**Texte d'erreur dans le descripteur "Error":** –

**Commentaire:** –

#### E.9.6 Procédures

Si le contrôleur MGC définit un descripteur "Events" contenant un événement de transition d'état du crochet commutateur (raccroché ou décroché) avec le paramètre "strict" (0x0001) réglé sur "failWrong", et que l'état du crochet commutateur corresponde déjà à ce que la transition implique, l'exécution de la commande contenant le descripteur "Events" échoue. La passerelle MG doit inclure dans sa réponse le code d'erreur 540 ("Etat initial du crochet commutateur inattendu").

#### E.10 Paquetage de continuité de base

**Nom du paquetage:** paquetage de continuité de base

**Identificateur de paquetage:** ct (0x000a)

**Description:** ce paquetage définit des événements et des signaux pour l'essai de continuité, qui comprend la fourniture d'un bouclage ou d'une fonctionnalité d'émetteur-récepteur.

**Version:** 1

**Extension:** aucune

##### E.10.1 Propriétés

Aucune.

##### E.10.2 Événements

###### E.10.2.1 Exécution de l'essai de continuité

**Nom du paramètre:** exécution

**Identificateur d'événement:** cmp (0x0005)

**Description:** cet événement détecte la fin d'un essai de continuité.

**Paramètres de descripteur "Events":** aucun

**Paramètres de descripteur "ObservedEvents":**

###### Résultat de l'essai de continuité

**Nom du paramètre:** résultat

**Identificateur de paramètre:** res (0x0008)

**Description:** indique le résultat de l'essai de continuité.

**Type:** énumération

**Facultatif:** non

**Valeurs possibles:** "SUCCESS" (0x0001), succès  
"FAILURE" (0x0000), échec

**Valeur par défaut:** aucune

### E.10.3 Signaux

#### E.10.3.1 Essai de continuité

**Nom du signal:** essai de continuité

**Identificateur de signal:** ct (0x0003)

**Description:** déclenche l'envoi d'une tonalité d'essai de continuité vers la terminaison à laquelle il est appliqué.

**Type du signal:** temporisé

**Valeur par défaut:** fournie

**Paramètres additionnels:** aucun

#### E.10.3.2 Réponse à l'essai de continuité

**Nom du signal:** réponse

**Identificateur de signal:** rsp (0x0004)

**Description:** ce signal est utilisé pour répondre à un essai de continuité. Voir le § E.10.5 pour une explication plus poussée.

**Type du signal:** commuté

**Valeur par défaut:** fournie

**Paramètres additionnels:** aucun

### E.10.4 Statistiques

Aucune.

### E.10.5 Procédures

Lorsqu'il veut déclencher un essai de continuité, un contrôleur MGC envoie à la passerelle MG une commande qui contient:

- un descripteur "Signals" avec le signal ct;
- un descripteur "Events" contenant l'événement cmp.

A la réception d'une commande contenant le signal ct et l'événement cmp, la passerelle MG déclenche la tonalité d'essai de continuité pour la terminaison spécifiée. Si la tonalité de retour est détectée et que toutes les autres conditions requises sont remplies avant l'expiration de la temporisation du signal, l'événement cmp doit être créé avec la valeur du paramètre de résultat égale à "success" (succès). Dans tous les autres cas, l'événement cmp doit être créé avec la valeur du paramètre de résultat égale à "failure" (échec).

Lorsqu'un contrôleur MGC veut que la passerelle MG réponde à un essai de continuité, il envoie à cette passerelle une commande qui contient un descripteur "Signals" avec le signal rsp. A la réception d'une commande avec le signal rsp, la passerelle MG applique soit un bouclage ou (pour les circuits à 2 fils) attend la réception d'une tonalité d'essai de continuité. Dans le cas du bouclage, toute information entrante sera renvoyée comme information sortante. Dans le cas du circuit à 2 fils, chaque fois que la tonalité d'essai appropriée est reçue, la tonalité de réponse appropriée doit être envoyée. Le contrôleur MGC détermine quand supprimer le signal rsp.

Lorsqu'un essai de continuité est exécuté sur une terminaison, aucun dispositif d'écho ou codec ne doit être actif sur cette terminaison.

L'exécution d'une assurance de conduit vocal en tant que partie d'un essai de continuité est configurée par accord bilatéral entre les opérateurs de réseau.

NOTE – Des exemples de tonalités et les détails des procédures d'essai sont donnés dans les § 7 et 8/Q.724, dans le § 2.1.8/Q.764 et dans la Rec. UIT-T Q.1902.4.

## **E.11 Paquetage de réseau**

**Nom du paquetage:** paquetage de réseau

**Identificateur de paquetage:** nt (0x000b)

**Description:** ce paquetage définit les propriétés de terminaisons de réseau indépendantes du type de réseau. Ceci couvre entre autres les réseaux TDM, IP et ATM.

**Version:** 1

**Extension:** aucune

### **E.11.1 Propriétés**

#### **E.11.1.1 Tampon maximal de gigue**

**Nom de la propriété:** tampon maximal de gigue

**Identificateur de propriété:** jit (0x0007)

**Description:** cette propriété fixe une capacité maximale au tampon de gigue.

**Type:** entier

**Valeurs possibles:** tout entier positif, en millisecondes.

**Valeur par défaut:** fournie

**Défini dans:** descripteur "LocalControl"

**Caractéristiques:** lecture/écriture

### **E.11.2 Evénements**

#### **E.11.2.1 Défaillance du réseau**

**Nom de l'événement:** défaillance du réseau

**Identificateur d'événement:** netfail (0x0005)

**Description:** la terminaison produit cet événement lorsqu'elle détecte une défaillance due à des raisons externes ou internes du réseau.

**Paramètres de descripteur "Events":** aucun

**Paramètres de descripteur "ObservedEvents":**

##### **Cause**

**Nom du paramètre:** cause

**Identificateur de paramètre:** cs (0x0001)

**Description:** ce paramètre peut être inclus avec l'événement de défaillance afin de fournir des informations de diagnostic sur la raison de la défaillance.

**Type:** chaîne

**Facultatif:** oui

**Valeurs possibles:** n'importe quelle chaîne alphanumérique

**Valeur par défaut:** aucune

### E.11.2.2 Alerte de qualité

**Nom de l'événement:** alerte de qualité

**Identificateur d'événement:** qualert (0x0006)

**Description:** cette propriété permet à la passerelle MG d'indiquer une perte de qualité de la connexion réseau. A cette fin, la passerelle peut mesurer la perte de paquet, la gigue entre arrivées et le temps de propagation, puis indiquer ces valeurs sous la forme d'un pourcentage de perte de qualité.

**Paramètres de descripteur "Events":**

#### Seuil

**Nom du paramètre:** seuil

**Identificateur de paramètre:** th (0x0001)

**Description:** seuil relatif de la perte de qualité mesurée, calculé sur la base d'une méthode configurée qui peut prendre en compte, par exemple, la perte de paquets, la gigue et le temps de propagation. Cet événement est déclenché lorsque la valeur calculée dépasse le seuil.

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** 0 à 99

**Valeur par défaut:** fournie

**Paramètres de descripteur "ObservedEvents":**

#### Seuil

**Nom du paramètre:** seuil

**Identificateur de paramètre:** th (0x0001)

**Description:** seuil relatif de la perte de qualité mesurée, calculé sur la base d'une méthode configurée qui peut prendre en compte, par exemple, la perte de paquets, la gigue et le temps de propagation.

**Type:** entier

**Facultatif:** oui

**Valeurs possibles:** 0 à 99

**Valeur par défaut:** aucune

### E.11.3 Signaux

Aucun.

### E.11.4 Statistiques

#### E.11.4.1 Durée

**Nom de la statistique:** durée

**Identificateur de statistique:** dur (0x0001)

**Description:** indique la durée depuis laquelle la terminaison existe ou est hors du contexte NULL.

**Type:** nombre à double précision

**Valeurs possibles:** tout entier positif, en millisecondes

**Niveau:** l'un ou l'autre

#### E.11.4.2 Octets envoyés

**Nom de la statistique:** octets envoyés

**Identificateur de statistique:** os (0x0002)

**Description:** fournit le nombre d'octets envoyés à partir de la terminaison ou du flux depuis que la terminaison existe ou est hors du contexte NULL. Les octets représentent le flux de média en sortie à l'exclusion de tout surdébit de transport. Au niveau de la terminaison, il est égal à la somme des flux en sortie dans tous les flux binaires.

Pour les médias à transmission analogique, le décompte d'octets doit être choisi de valeur égale à zéro.

**Type:** nombre à double précision

**Valeurs possibles:** tout entier de 64 bits supérieur à 0

**Niveau:** l'un ou l'autre

#### E.11.4.3 Octets reçus

**Nom de la statistique:** octets reçus

**Identificateur de statistique:** or (0x0003)

**Type:** nombre à double précision

**Description:** fournit le nombre d'octets envoyés à partir de la terminaison ou du flux depuis que la terminaison existe ou est hors du contexte NULL. Les octets représentent le flux de média en entrée à l'exclusion de tout surdébit de transport. Au niveau de la terminaison, il est égal à la somme des flux en entrée dans tous les flux binaires.

Pour les médias à transmission analogique, le décompte d'octets doit être choisi de valeur égale à zéro.

**Valeurs possibles:** tout entier de 64 bits supérieur à 0

**Niveau:** l'un ou l'autre

### E.11.5 Procédures

#### E.11.5.1 Procédures pour application des événements de défaillance de réseau et d'alerte de qualité

L'application des deux événements définis par le paquetage dépend de ce qui suit:

- le type de support spécifique (par exemple TDM, ligne analogique, RTP, IP, AAL 2, AAL 1, etc.);
- la technique de couche de transmission sous-jacente (par exemple analogique, PDH, SDH, SONET, xDSL, 802.3, 802.11, etc.);
- la question de savoir si l'événement est associé aux ressources internes ou externes de la passerelle MG.

##### E.11.5.1.1 Procédure commune aux deux événements

L'application d'un événement ou de l'autre requiert une compréhension mutuelle entre le contrôleur MGC et la passerelle MG. Si la passerelle MG et le contrôleur MGC n'ont pas une vue cohérente de la signification des deux événements, le contrôleur MGC ne sera pas en mesure d'agir sur eux efficacement.

Des recommandations explicites sont hors du domaine d'application du protocole central de la présente Recommandation. L'usage spécifique de ces événements devrait être plutôt précisé dans les spécifications de profil H.248.

### E.11.5.1.2 Procédure spécifique à l'événement "Défaillance de réseau"

Le paramètre "Cause" peut servir à distinguer entre différents modes de défaillance. Le type de terminaison contribue également à l'interprétation de l'événement.

### E.11.5.1.3 Procédure spécifique à l'événement "Alerte de qualité"

La logique de détection devrait être fondée sur une ou plusieurs mesures de performance relatives à la qualité de service, à partir de laquelle ou desquelles une valeur comparable au paramètre de seuil peut être calculée.

Les détails du processus sous-jacent de mesurage et d'estimation (les variables d'observation relatives à la qualité de service, les mécanismes de filtrage, la longueur des périodes de mise à jour, etc.) peuvent être spécifiés dans un profil H.248.

### E.11.5.1.4 Relation avec les procédures relatives au changement de service

Un événement de défaillance de réseau ou d'alerte de qualité peut déclencher une procédure de changement de service à la ou aux terminaisons affectées. La transition à l'état hors service (*OutOfService*) peut être lancée par la passerelle MG ou par le contrôleur MGC, selon les circonstances.

## E.12 RTP Package

**Package Name:** RTP Package

## E.12 Paquetage du protocole de transport en temps réel (RTP)

**Nom du paquetage:** paquetage de protocole RTP

**Identificateur de paquetage:** rtp (0x000c)

**Description:** ce paquetage est utilisé pour assurer le transfert en mode paquet de données multimédias au moyen du protocole de transport en temps réel (RTP, *real-time transport protocol*) (RFC 3550).

**Version:** 1

**Extension:** version 1 nt

### E.12.1 Propriétés

Aucune.

### E.12.2 Evénements

#### E.12.2.1 Transition de charge utile

**Nom de l'événement:** transition de charge utile

**Identificateur d'événement:** pltrans (0x0001)

**Description:** cet événement détecte et signale la transition du format de charge utile RTP à un autre format.

**Paramètres de descripteur "Events":** aucun

**Paramètres de descripteur "ObservedEvents":**

#### Type de charge utile RTP

**Nom du paramètre:** rtppayload

**Identificateur de paramètre:** rtppltype (0x01)

**Description:** format de charge utile dans lequel la transition a été effectuée.

**Type:** sous-liste d'énumération

**Facultatif:** non

**Valeurs possibles:** la méthode de codage doit être spécifiée au moyen d'un ou de plusieurs noms de codage valides, comme définis dans le profil AV du protocole RTP ou comme enregistrés auprès de l'autorité IANA.

**Valeur par défaut:** aucune

### E.12.3 Signaux

Aucun.

### E.12.4 Statistiques

#### E.12.4.1 Paquets envoyés

**Nom de la statistique:** paquets envoyés

**Identificateur de statistique:** ps (0x0004)

**Description:** fournit le nombre de paquets envoyés à partir de la terminaison ou du flux depuis que la terminaison existe ou est hors du contexte NULL.

**Type:** nombre à double précision

**Valeurs possibles:** tout entier de 64 bits supérieur à 0

**Niveau:** l'un ou l'autre

#### E.12.4.2 Paquets reçus

**Nom de la statistique:** paquets reçus

**Identificateur de statistique:** pr (0x0005)

**Description:** fournit le nombre de paquets reçus par la terminaison ou par le flux depuis que la terminaison existe ou est hors du contexte NULL.

**Type:** nombre à double précision

**Valeurs possibles:** tout entier de 64 bits supérieur à 0

**Niveau:** l'un ou l'autre

#### E.12.4.3 Perte de paquets

**Nom de la statistique:** perte de paquets

**Identificateur de statistique:** pl (0x0006)

**Description:** décrit le taux de perte de paquets actuel dans un flux RTP, comme défini dans le document RFC 3550. La perte de paquets est exprimée en pourcentage calculé comme suit: nombre de paquets perdus dans l'intervalle entre deux rapports de réception, divisé par le nombre de paquets attendus au cours de cet intervalle.

**Type:** nombre à double précision

**Valeurs possibles:** nombre entier de 32 bits et fraction de 32 bits.

**Niveau:** l'un ou l'autre

#### E.12.4.4 Gigue

**Nom de la statistique:** gigue

**Identificateur de statistique:** jit (0x0007)

**Description:** demande la valeur actuelle de la gigue entre arrivées dans un flux RTP tel que défini par le document RFC 3550. La gigue mesure la variation du temps entre arrivées de paquets de données RTP.

**Type:** nombre à double précision

**Valeurs possibles:** tout entier de 64 bits supérieur à 0

**Niveau:** l'un ou l'autre

#### E.12.4.5 Temps de propagation

**Nom de la statistique:** temps de propagation

**Identificateur de statistique:** delay (0x0008)

**Description:** demande la valeur actuelle du temps de propagation des paquets, exprimée en unités de marqueurs temporels. Il s'agit de la même grandeur que la latence moyenne.

**Type:** nombre à double précision

**Valeurs possibles:** tout entier de 64 bits supérieur à 0

**Niveau:** l'un ou l'autre

#### E.12.5 Procédures

Lorsque le protocole RTCP est associé à un flux RTP, le protocole RTCP doit rester insensible à la propriété de mode H.248.1 contenue dans le descripteur "LocalControl".

Lorsque le protocole RTCP est associé à un flux RTP et que la passerelle MG reçoit un descripteur "EmptyRemote" pour ce flux, la passerelle MG doit interrompre le flux RTCP de même que le flux RTP correspondant.

#### E.13 Paquetage de circuit TDM

**Nom du paquetage:** paquetage de circuit TDM

**Identificateur de paquetage:** tdmc (0x000d)

**Description:** ce paquetage peut être utilisé par toute terminaison qui prend en charge le gain et la limitation de l'écho. Il était initialement destiné à être utilisé sur les circuits TDM mais son utilisation peut être étendue plus largement

Les nouvelles versions ou extensions de ce paquetage devraient pouvoir tenir compte des circuits non TDM.

**Version:** 1

**Extension:** version 1 du paquetage de réseau

#### E.13.1 Propriétés

##### E.13.1.1 Annulation d'écho

**Nom de la propriété:** annulation d'écho

**Identificateur de propriété:** ec (0x0008)

**Type:** opérateur booléen

**Valeurs possibles:** "True" (lorsque l'annulation d'écho est requise)  
"False" (lorsque l'annulation d'écho n'est pas requise.)

**Valeur par défaut:** fournie

**Défini dans:** descripteur "LocalControl"

**Caractéristiques:** lecture/écriture

### E.13.1.2 Commande de gain

**Nom de la propriété:** commande de gain

**Identificateur de propriété:** gain (0x000a)

**Description:** la commande de gain, ou l'utilisation de l'adaptation du niveau des signaux et de la réduction du niveau de bruit, est utilisée pour adapter le niveau du signal en accès sortant. Il est cependant nécessaire, par exemple pour les communications par modem, de désactiver cette fonction. Lorsque la valeur est réglée à "automatic", la terminaison fonctionne comme une commande automatique de niveau (ALC, *automatic level control*) avec une cible de niveau fournie dans la passerelle MG, le sens étant: vers l'extérieur.

**Type:** entier

**Valeurs possibles:** la commande de gain spécifie le gain en décibels (valeur positive ou négative), l'entier positif maximal 214748647 (0x7fffffff) étant réservé à la représentation de la valeur "automatic".

**Valeur par défaut:** fournie

**Défini dans:** descripteur "LocalControl"

**Caractéristiques:** lecture/écriture

### E.13.2 Evénements

Aucun.

### E.13.3 Signaux

Aucun.

### E.13.4 Statistiques

Aucune.

### E.13.5 Procédures

Aucune.

## E.14 Paquetage de segmentation

**Nom du paquetage:** paquetage de segmentation

**Identificateur de paquetage:** seg (0x00A3)

**Description:** ce paquetage définit des propriétés à utiliser lors de l'exécution d'une segmentation de type H.248 dans des protocoles de transport non segmenteurs.

**Version:** 1

**Extension:** version 2 radicale

### E.14.1 Propriétés

#### E.14.1.1 Valeur du temporisateur de segmentation par la passerelle MG

**Nom de la propriété:** MGSegmentationTimerValue

**Identificateur de la propriété:** MGSegmentationTimerValue (0x0009)

**Description:** indique la durée pendant laquelle le contrôleur MGC devrait s'attendre à la réception de tous segments de message à venir de la passerelle MG une fois le jeton SegmentationCompleteToken reçu. Valeur initialement mise à MGProvisionalResponseTimerValue, mais cette valeur peut être modifiée.

**Type:** entier

**Valeurs possibles:** tout entier positif en millisecondes

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

#### **E.14.1.2 Valeur du temporisateur de segmentation par le contrôleur MGC**

**Nom de la propriété:** MGCSegmentationTimerValue

**Identificateur de la propriété:** MGCSegmentationTimerValue (0x000A)

**Description:** indique la durée pendant laquelle la passerelle MG devrait s'attendre à la réception de tous segments de message à venir du contrôleur MGC une fois le jeton SegmentationCompleteToken reçu. Valeur initialement mise à MGCProvisionalResponseTimerValue, mais cette valeur peut être modifiée.

**Type:** entier

**Valeurs possibles:** tout entier positif en millisecondes

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

#### **E.14.1.3 Longueur maximale d'unité PDU de passerelle MG**

**Nom de la propriété:** MGMaxPDUSize

**Identificateur de la propriété:** MGMaxPDUSize (0x000B)

**Description:** indique la longueur maximale de l'unité de données de protocole entrant dans la passerelle MG pour le protocole de transport de l'association de commande. Le contrôleur MGC devrait éviter de construire des messages qui dépassent cette longueur.

**Type:** entier

**Valeurs possibles:** tout entier positif en octets

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture seule (ReadOnly)

#### **E.14.1.4 Longueur maximale d'unité PDU de contrôleur MGC**

**Nom de la propriété:** MGCMMaxPDUSize

**Identificateur de la propriété:** MGCMMaxPDUSize (0x000C)

**Description:** indique la longueur maximale de l'unité de données protocolaire entrant dans le contrôleur MGC pour le protocole de transport de l'association de commande. La passerelle MG devrait éviter construire des messages qui dépassent cette longueur.

**Type:** entier

**Valeurs possibles:** tout entier positif en octets

**Valeur par défaut:** fournie

**Propriété définie dans l'état:** TerminationState

**Caractéristiques:** lecture/écriture

#### **E.14.2 Evénements**

Aucun.

### E.14.3 Signaux

Aucun.

### E.14.4 Statistiques

Aucune.

### E.14.5 Codes d'erreur

**Code d'erreur n°:** 459

**Nom:** segments non reçus

**Définition:** ce code d'erreur indique que le destinataire d'une réponse TransactionReply est arrivé à l'expiration de son délai d'attente pour tous les segments à remettre.

**Texte de l'erreur dans le descripteur "Error":** les numéros de segment manquants sont inclus.

**Commentaire:** –

### E.14.6 Procédures

Etant donné que certains transports ne segmentent pas automatiquement les messages qui dépassent la longueur de l'unité de transmission maximale (MTU, *maximal transmission unit*) du protocole de transport, des messages peuvent finir par être tronqués. Si le message en question contient de multiples réponses de transaction, l'expéditeur devrait envoyer chaque réponse de transaction dans un message distinct. S'il reste des messages qui dépassent la longueur de l'unité MTU, l'expéditeur peut implémenter les procédures de segmentation suivantes pour les réponses de transaction. Les requêtes de transaction (TransactionRequest) ne doivent pas être segmentées.

Lors de la segmentation d'une réponse de transaction, l'expéditeur doit vérifier que le message contient des commandes et/ou actions complètes. Chaque segment est indiqué par des informations de segment additionnelles, au niveau transactionnel. Chaque segment doit utiliser le même identificateur de transaction, et ne doit pas répéter, dans des segments subséquents, des commandes ou actions déjà effectuées. Chaque segment doit être numéroté consécutivement à partir de 1 et ainsi de suite jusqu'au dernier segment, lequel est repéré par inclusion du jeton SegmentationCompleteToken. Dans tous les cas, les messages segmentés doivent être des productions valides sur le plan de la syntaxe. Le récepteur répond à chaque segment à son tour, au moyen de la réponse de segmentation. Etant donné que les segments sont des messages complets par eux-mêmes, l'entité réceptrice n'a pas besoin d'attendre de nouveaux segments avant de traiter un segment donné.

Dès réception du segment final, le récepteur doit déterminer s'il a reçu tous les segments de message. Si c'est le cas, le récepteur doit ensuite répondre par un acquittement TransactionResponseAcknowledgement.

Si le récepteur n'a pas reçu tous les segments, il doit armer un temporisateur, et attendre les segments manquants. La durée du temporisateur doit pouvoir être fournie au récepteur, mais devrait être égale à la valeur ProvisionalResponseTimerValue (voir le § E.2.1) des autres entités. Si les segments manquants n'arrivent pas avant que le temporisateur expire, le récepteur doit répondre par le code d'erreur 459 ("Segments non reçus").

Exemple 1:

```
Expéditeur:    !/3 [12.34.56.78]:2944 P=1/1{C=1{AV=term1{...}, AV=term2{...}}}  
Récepteur:    !/3 [12.34.56.79]:2944 SM=1/1  
Expéditeur:    !/3 [12.34.56.78]:2944 P=1/2{C=1{AV=term3{...}}, C=2{AV=term4{...}}}  
Récepteur:    !/3 [12.34.56.79]:2944 SM=1/2  
Expéditeur:    !/3 [12.34.56.78]:2944 P=1/3/#{C=3{AV=term5{...}}}  
Récepteur:    !/3 [12.34.56.79]:2944 SM=1/3/#  
Récepteur:    !/3 [12.34.56.79]:2944 K=1
```

## Exemple 2:

```
Expéditeur:    !/3 [12.34.56.78]:2944 P=1/1{C=1{AV=term1{...}, AV=term2{...}}}
Récepteur:    !/3 [12.34.56.79]:2944 SM=1/1
Expéditeur:    !/3 [12.34.56.78]:2944 P=1/4/{C=3{AV=term5{...}}}
Récepteur:    !/3 [12.34.56.79]:2944 SM=1/4/#
/* La temporisation de segmentation expire */
Récepteur:    !/3 [12.34.56.79]:2944 ER=459{"2,3"}
```

## E.15 Comportement de notification

**Nom du paquetage:** paquetage de comportement de notification

**Identificateur de paquetage:** nb (0x009a)

**Description:** le paquetage possède une capacité qui permet à la passerelle MG, sur demande du contrôleur MGC, de régler l'envoi de commandes de notification. Ce paquetage entre en interaction avec le fanion NotifyBehaviour décrit dans le § 7.1.9. Cette version du paquetage décrit un comportement de régulation fondé sur un pourcentage. D'autres types de comportement de régulation feront l'objet d'un complément d'étude.

**Version:** 1

**Extension:** aucune

### E.15.1 Propriétés

#### E.15.1.1 Régulation des notifications

**Nom de la propriété:** régulation des notifications

**Identificateur de la propriété:** notreg (0x0001)

**Description:** indique le pourcentage de notifications de la passerelle MG qui devraient être supprimées. Une valeur de 0% indique: aucune suppression. Une valeur de 100% indique que toutes les notifications devraient être supprimées. Cette propriété ne doit être définie qu'à la terminaison racine. Lorsque la commande "Regulated Notify" est envoyée au sujet d'un événement particulier, la passerelle MG utilise la propriété de régulation des notifications afin de déterminer le pourcentage correct de notifications à régler (c'est-à-dire à supprimer).

**Type:** entier

**Valeurs possibles:** 0-100

**Valeur par défaut:** aucune

**Propriété définie dans l'état:** terminationState

**Caractéristiques:** lecture/écriture

#### E.15.2 Événements

Aucun.

#### E.15.3 Signaux

Aucun.

#### E.15.4 Statistiques

Aucune.

#### E.15.5 Procédures

##### E.15.5.1 NotifyBehaviour

Le mécanisme NotifyBehaviour autorise le contrôleur MGC à gérer la façon dont il devrait recevoir des commandes Notify.request en provenance de la passerelle MG.

### **E.15.5.1.1 NeverNotify**

Il peut y avoir des cas où le contrôleur MGC n'est pas intéressé à recevoir une commande Notify.request. Par exemple lorsque la détection de l'occurrence de cet événement est essentielle afin de déclencher un signal ou événement imbriqué; mais le contrôleur MGC n'a pas besoin d'en être informé. Le réglage du paramètre NotifyBehaviour à la valeur "NeverNotify" au sujet d'un événement garantit que la détection de l'événement ne sera pas notifiée au contrôleur MGC; cependant, tous les signaux ou événements imbriqués seront exécutés. La commande NeverNotify n'a aucun rapport avec la propriété de régulation des notifications (nb/notreg) et les événements ainsi marqués ne sont pas inclus dans les calculs de régulation.

### **E.15.5.1.2 NotifyImmediate**

Le contrôleur MGC peut souhaiter être averti d'événements importants tels qu'un événement d'appel d'urgence', bien que certains autres événements soient régulés. Cela est réalisé par réglage du paramètre NotifyBehaviour à la valeur "NotifyImmediate" pour cet événement particulier. C'est le comportement par défaut actuel lorsque les événements sont détectés. La commande NotifyImmediate n'a aucun rapport avec la propriété de régulation des notifications (nb/notreg) et les événements ainsi marqués ne sont pas inclus dans les calculs de régulation.

### **E.15.5.1.3 RegulatedNotify**

#### **E.15.5.1.3.1 Notifications régulées sur la base du pourcentage**

Lorsque le paramètre NotifyBehaviour est mis à la valeur "NotifyRegulated" au sujet d'un événement particulier, la décision d'envoyer au contrôleur MGC une notification pour toute occurrence d'un événement particulier est régie par la propriété de régulation des notifications (nb/notreg) à la terminaison racine. La valeur de cette propriété est déterminée et réglée par le contrôleur MGC. Par exemple, le taux de régulation pourrait correspondre au niveau d'encombrement du contrôleur MGC. La passerelle MG sera ensuite chargée de déterminer le pourcentage correct de commandes de notification à supprimer parmi les événements marqués par le paramètre "NotifyRegulated" dans l'ensemble de la passerelle MG.

La passerelle MG peut implémenter tout algorithme qui réalise le pourcentage fixé de régulation des notifications. L'algorithme devrait être choisi de façon que le risque de synchronisation entre différentes passerelles MG soit minimisé. Un algorithme est suggéré ci-dessous:

Pour un pourcentage de régulation donné (par exemple 10%) la passerelle MG fera ce qui suit pour les événements qui possédaient le fanion RegulatedNotify:

- 1) supprimer un certain nombre de notifications (par exemple la notification 1);
- 2) accepter un certain nombre de notifications subséquentes jusqu'à ce que le pourcentage de régulation des notifications soit atteint (par exemple accepter 9 notifications);
- 3) répéter les étapes 1) et 2) jusqu'à ce que le pourcentage de régulation soit modifié;
- 4) afin d'éviter des problèmes de synchronisation, la passerelle MG devrait commencer par une position aléatoire dans la séquence suggérée ci-dessus.

Si un événement a le paramètre NotifyBehaviour mis à la valeur "RegulatedNotify" et que la propriété de régulation des notifications soit mise à zéro, cela aura le même effet que si l'événement avait été mis à la valeur "ImmediateNotify", ce qui est le comportement par défaut pour la notification des événements.

Le mécanisme NotifyBehaviour devrait être utilisé avec précaution afin d'éviter des états incorrects. Par exemple, si la notification d'un événement de décrochage est supprimée par la passerelle MG, alors le contrôleur MGC ne sera pas en mesure de fournir la réponse correcte (par exemple tonalité d'invitation à numéroté, tonalité d'encombrement). Ce comportement incorrect est évité par l'utilisation du mécanisme de signal imbriqué ou d'événements imbriqués afin de renvoyer une

réponse appropriée sans que le contrôleur MGC soit impliqué à ce moment. Le contrôleur MGC peut régler, d'une part, un descripteur "Signals" ou "Events" imbriqué, qui est à déclencher lorsqu'un événement marqué "NotifyRegulated" est détecté et accepté (c'est-à-dire lorsque la notification n'est pas supprimée) et, d'autre part, un descripteur "Signals" ou "Events" imbriqué et réglé séparément, qui est à déclencher lorsque l'événement est détecté et régulé (c'est-à-dire lorsque la notification est supprimée).

NOTE – Même dans le cas ci-dessus, la passerelle MG reste non informée de ce qui constitue un appel. Elle est simplement en train d'exécuter une tâche programmée plutôt qu'une capacité de commande d'appel.

### E.15.5.2 Exemple de scénario

Soit le scénario où un contrôleur MGC et une passerelle MG sont l'aboutissement de lignes analogiques: l'unité MGC/MG peut manifester différents comportements selon que le contrôleur MGC est ou non en état de surcharge ou et selon que la ligne analogique (ALN) est ou non à l'origine d'un appel d'urgence. Les différents comportements sont représentés dans le Tableau E.1 ci-dessous:

**Tableau E.1/H.248.1 – Eventuel comportement en surcharge de MGC/MG**

Comportement de passerelle MG Tâche: "Traitement de signalisation d'appel sur ligne analogique (ALS)" = trafic de signalisation de commande d'appel pour terminaux RTC (ALN; interfaces avec ligne analogique)		1) Type d'appel/de contexte	
		Appel d'urgence	Appel autre que d'urgence
2) Etat de charge de MGC NOTE – Modèle supposé à deux états	Etat de non-surcharge (par exemple charge faible, moyenne, élevée)	I) MG: Traitement normal de signalisation ALS avec priorité <i>la plus élevée</i>	II) MG: Traitement normal de signalisation ALS avec priorité <i>inférieure</i>
	Etat de surcharge	III) MG: Traitement normal de signalisation ALS ==> signalisation de commande d'appel réexpédiée vers le contrôleur MGC dans le "mode H.248 habituel"	IV) MG: Traitement programmé de signalisation ALS locale ==> MGC <i>pas</i> impliqué ==> protocole d'appel conforme rejeté via la "tonalité d'encombrement"

#### E.15.5.2.1 Comportement dans les quadrants I et II

Aucun comportement spécial de notification n'est requis sur la passerelle MG car le contrôleur MGC n'est pas en surcharge et peut recevoir tous les messages (notifications) en provenance de la passerelle MG. Cependant, une fois que le contrôleur MGC a déterminé le type d'appel, il peut régler une priorité dans la passerelle MG au moyen d'un attribut de contexte approprié.

#### E.15.5.2.2 Comportement dans les quadrants III et IV

Lorsque le contrôleur MGC a déterminé qu'il est dans un état surchargé, il peut envoyer la commande suivante à toutes les terminaisons ou à un certain groupe de terminaisons choisies dans le contexte NULL.

Exemple de message:

```
Transaction=1234{
  Context = - {
    Modify = aln/* {          ; choisir les terminaisons comme souhaité
      Events = 1234 {
        al/of {
          RegulatedNotify {
            Embed {
              Signals {cg/dt} ,
; Si le décrochage est régulé, appliquer la tonalité d'invitation à numéroté.
              Events = 1235 {
                xdd/xce {
                  DigitMap = PriorityDialPlan1,
                  bc = 20,
                  mp = enhanced,
                  ImmediateNotify
                },
; Si l'urgence produit une notification vers le contrôleur MGC
                xdd/xce {
                  DigitMap = NonPriorityDialPlan1,
                  bc = 20,
                  mp = enhanced,
                  NeverNotify,
                  Embed{
                    Signals {cg/ct}
                  }
                },
; Si la non-priorité envoie la tonalité d'encombrement et jamais de notification
                al/on {
                  NeverNotify,
                  ResetEventsDescriptor
                }
; Si un raccrochage est détecté, alors réinitialiser le descripteur "Events"
; et ne jamais envoyer de notification
            }
          }
        }
      }
    }
  }
}
```

NOTE – L'identité du contexte peut être spécifique ou de valeur NULL et l'identificateur de terminaison peut être remplacé par une structure générique ou être spécifique selon le mode et le moment du réglage de ce comportement dans la passerelle MG. L'emploi d'une structure générique en réponse limitera le nombre de réponses si le comportement en surcharge est chargé lorsque le contrôleur MGC entre en surcharge.

#### **E.15.5.2.3 Cas où la propriété de régulation des notifications n'est pas mise à zéro (état surchargé)**

Le comportement de la passerelle MG dans le cas où un décrochage est détecté et régulé consistera à:

- supprimer la notification de décrochage adressée au contrôleur MGC;
- appliquer la tonalité d'invitation à numéroté;
- recevoir les chiffres composés par l'abonné;
- s'il a été déterminé que l'utilisateur a composé un numéro d'urgence, toujours produire la notification de la concordance avec le "plan de numérotation en cas d'urgence";

- s'il a été déterminé que l'utilisateur a composé un numéro non prioritaire, toujours supprimer la notification de la concordance avec le "plan de numérotation hors cas d'urgence" et appliquer la tonalité d'encombrement;
- activer la détection de raccrochage (avec le fanion NeverNotify) ce qui, dès réception d'un raccrochage, fera que le descripteur "Events" situé à la terminaison (ou le descripteur "Events" fourni par la passerelle MG) sera réinitialisé à l'état qu'il avait avant que l'imbrication ait été activée.

#### **E.15.5.2.4 Cas où la propriété de régulation des notifications est mise à zéro (état non surchargé)**

Le comportement de la passerelle MG dans le cas ci-dessus consistera à:

- envoyer une notification de décrochage au contrôleur MGC.

Si le descripteur "Events" du comportement en surcharge est réglé au moyen de commandes Modify explicites à partir du contrôleur MGC (par opposition à l'utilisation de la configuration de la passerelle MG) alors, bien que le contrôleur MGC reste en condition de surcharge, le descripteur "Events" du comportement en surcharge sera rechargé pour chaque terminaison dans le cadre de la séquence de relève de tout appel ou de toute tentative d'appel notifié(e) au contrôleur MGC.

Lorsque la condition de surcharge du contrôleur MGC a cessé, celui-ci peut réinitialiser les terminaisons se trouvant dans le contexte NULL, afin de les remettre au comportement de non-surcharge par défaut. En variante, le contrôleur MGC devrait régler la propriété de régulation des notifications à une valeur égale à zéro.

## **Annexe F**

### **Procédures relatives au changement de service**

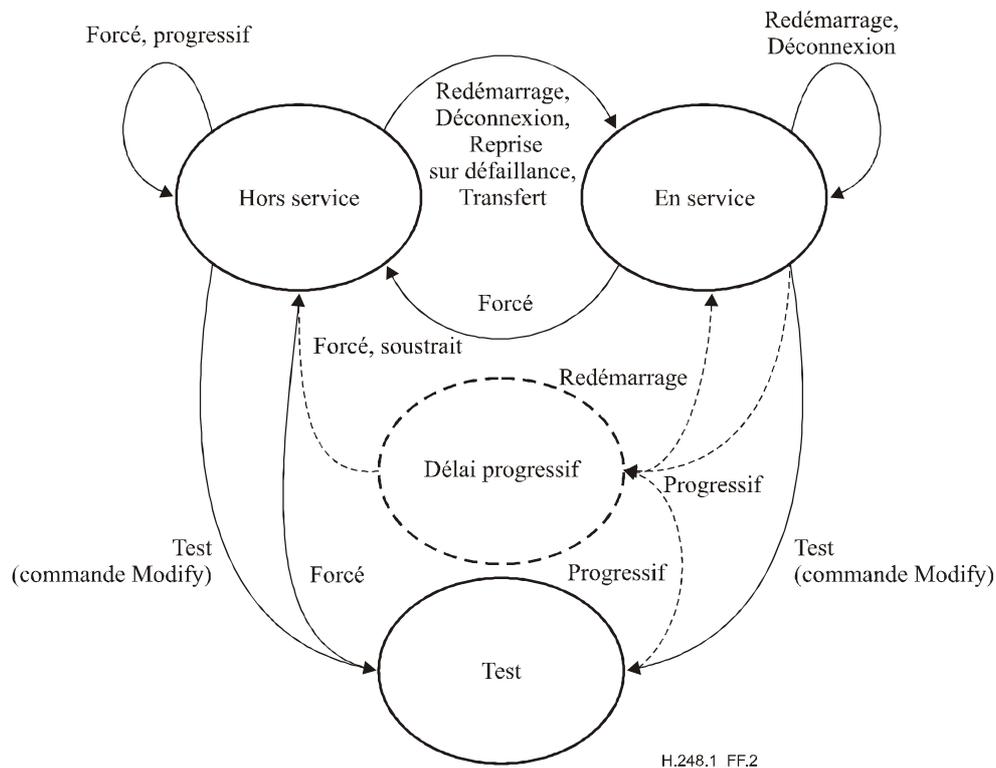
#### **F.1 Introduction**

La présente annexe décrit l'utilisation de procédures de changement de service lorsque certains événements se produisent dans une passerelle MG ou dans un contrôleur MGC. Elle vise à clarifier les procédures relatives au changement de service qui sont décrites dans les § 7.2.8 et 11 ci-dessus. Lorsqu'il y a des divergences entre la présente annexe et la présente Recommandation, les procédures de celle-ci ont priorité sur celles qui sont décrites dans la présente annexe.

Les événements aboutissant à l'envoi d'une commande de changement de service telle que décrite dans la présente annexe sont les suivants:

- 1) enregistrement de la passerelle MG – la passerelle MG s'enregistre auprès d'un contrôleur MGC et une association de commande est établie;
- 2) réenregistrement de la passerelle MG – la passerelle MG s'enregistre auprès d'un contrôleur MGC après y avoir été invitée par un contrôleur MGC;
- 3) annulation du service de passerelle MG – la passerelle MG informe le contrôleur MGC du fait que les terminaisons ou la passerelle MG dans son ensemble doivent être mises hors service;
- 4) rétablissement du service de passerelle MG – la passerelle MG a effectué une reprise sur défaillance et en informe le contrôleur MGC;
- 5) reprise redondante de service de passerelle MG – une passerelle MG secondaire prend le relais en cas de défaillance ou de maintenance d'une passerelle MG primaire;





**Figure F.2/H.248.1 – Modèle d'états d'un changement de service simple**

## F.2 Définition de l'association de commande

Une association de commande est une relation de communication par laquelle un contrôleur MGC va commander une passerelle MG. Une association de commande est instanciée au moyen d'un enregistrement. L'association de commande est terminée dès que la passerelle MG passe à l'état OutOfService, qu'elle est correctement transférée à un autre contrôleur MGC, ou qu'elle se reprend correctement sur défaillance vers un autre contrôleur MGC. Une passerelle MG doit avoir au plus une seule association de commande à tout moment sauf lorsqu'une passerelle MG physique est subdivisée en une ou plusieurs passerelles MG virtuelles. Dans le cas d'une passerelle MG virtuelle, chaque passerelle MG virtuelle doit avoir au plus une seule association de commande. Seules les passerelles MG peuvent instancier des associations de commande. Celles qui sont définies ici s'appliquent seulement à l'intérieur du contexte de signalisation de commande. Cette définition ne devrait pas être confondue avec celle des associations de commande se trouvant dans le contexte de transport (TCP, UDP, etc.).

L'utilisation de commandes de changement de service à la terminaison racine affecte seulement la passerelle MG virtuelle à laquelle la terminaison racine appartient. Les autres passerelles MG virtuelles de la même passerelle MG physique ne sont pas affectées.

## F.3 Événements conduisant à des procédures de changement de service

### F.3.1 Enregistrement de passerelle MG

La passerelle MG peut s'enregistrer dans une des trois façons suivantes:

- annoncer sa présence initiale, ou son "enregistrement" au moyen de la valeur "Restart" de la méthode de changement de service, avec le code 901 ("Démarrage à froid") ou 902 ("Démarrage à chaud") de raison du changement de service;
- indiquer qu'elle est en train de changer son association au moyen de la valeur "Failover" (reprise sur défaillance) de la méthode de changement de service, avec le code 909

("Défaillance imminente du MGC") ou 908 ("Défaillance imminente de la MG") de raison du changement de service;

- indiquer qu'elle avait reçu l'ordre de changer son association au moyen d'une valeur "HandOff" (transfert) de la méthode de changement de service, avec le code 903 ("Changement ordonné par le MGC") (§ F.3.11) de raison du changement de service.

L'enregistrement commence lorsque la commande initiale de changement de service est envoyée de la passerelle MG au contrôleur MGC et il se termine lorsque l'ordre a fait l'objet d'une réponse par le contrôleur MGC sans adresse de remplacement ou sans erreur et lorsque le profil de changement de service est agréé entre contrôleur MGC et passerelle MG. L'association de commande est établie dès l'achèvement de l'enregistrement. Celui-ci apparaît toujours à la terminaison racine. Toute association de commande existant entre le contrôleur MGC et la passerelle MG est terminée dès le lancement d'un nouvel enregistrement. Toutes les commandes issues de la précédente association de commande doivent être ignorées.

Le profil de changement de service est utilisé par l'enregistrement. Voir le § F.5.5 pour de plus amples détails.

La version du changement de service est utilisée par l'enregistrement. Voir le § F.5.6 pour de plus amples détails.

### **F.3.2 Réenregistrement de passerelle MG**

Le réenregistrement de la passerelle MG apparaît dans deux cas:

- 1) lorsque le contrôleur MGC demande à la passerelle MG de se réenregistrer (§ F.3.8). La passerelle MG doit ensuite envoyer une commande de changement de service avec la valeur "Handoff" (transfert) de la méthode de changement de service, Le code de raison du changement de service 903 ("Changement ordonné par le MGC") et l'identificateur désigné "ServiceChangeMgcID";
- 2) lorsque le contrôleur MGC lance le rétablissement du service (§ F.3.9). La passerelle MG doit se réenregistrer en envoyant une commande de changement de service avec la valeur "Restart" (redémarrage) de la méthode de changement de service.

Le réenregistrement est achevé lorsque l'ordre a fait l'objet d'une réponse par un contrôleur MGC sans adresse de remplacement ni erreur et que le profil de changement de service est agréé entre le contrôleur MGC et la passerelle MG. Si aucune réponse n'est reçue du contrôleur MGC primaire, alors la passerelle MG doit suivre les procédures du § F.3.1. Les codes de raison du changement de service 903 ("Changement ordonné par le MGC") ou 909 ("Défaillance imminente du MGC") sont recommandés.

L'association de commande est établie dès l'achèvement de l'enregistrement. Celui-ci apparaît toujours à la terminaison racine. Toute association de commande existant entre le contrôleur MGC et la passerelle MG est terminée dès le lancement du réenregistrement. Toutes les commandes issues de la précédente association de commande doivent être ignorées.

Le profil de changement de service est utilisé par le réenregistrement. Voir le § F.5.5 pour de plus amples détails.

La version du changement de service est utilisée par le réenregistrement. Voir le § F.5.6 pour de plus amples détails.

### **F.3.3 Annulation de service de passerelle MG**

Afin de se mettre hors service, la passerelle MG envoie une commande de changement de service avec une valeur "Forced" (changement forcé) ou "Graceful" (changement progressif) dans la méthode de changement de service à la terminaison racine. Voir le § F.4.1.1 pour de plus amples détails.

Afin de mettre une terminaison ou un groupe de terminaisons hors service, la passerelle MG envoie une commande de changement de service avec une valeur "Forced" (changement forcé) ou "Graceful" (changement progressif) dans la méthode de changement de service à la ou aux terminaisons en question. Voir les § F.4.1.2 et F.4.1.3 pour de plus amples détails.

Le paramètre "ServiceChangeDelay" (délai de changement de service) indique la durée qui s'écoule avant que l'annulation du service se produise. Voir le § F.5.3 pour de plus amples détails.

Afin d'annuler une valeur précédemment émise (et acquittée) "Graceful" (changement progressif) pour l'ensemble de la passerelle MG, celle-ci envoie une commande de changement de service à la terminaison racine avec une valeur "Restart" et le code de raison du changement de service 918, "Annulation de la progressivité". La passerelle MG doit rester en service et toutes les terminaisons déjà réglées sur la mise hors service sont remises en service sauf signalisation contraire par la passerelle MG. Si une annulation de la progressivité est reçue après l'expiration du temporisateur de délai, la passerelle MG doit être considérée comme s'étant réenregistrée, exactement comme si elle avait envoyé un démarrage à froid.

Afin d'annuler une valeur précédemment émise (et acquittée) "Graceful" (changement progressif) à une terminaison ou à un groupe de terminaisons, la passerelle MG envoie une commande de changement de service avec une valeur "Restart" et une raison de changement de service "Annulation de la progressivité" à la ou aux terminaisons en question. La terminaison doit rester en service. Si la terminaison a déjà passé à l'état hors service, elle doit être remise en service exactement comme elle le serait avec une valeur de redémarrage (Restart) du changement de service.

#### **F.3.4 Rétablissement du service de passerelle MG**

L'événement de rétablissement du service de passerelle MG apparaît lorsqu'une passerelle MG reprend son service après une défaillance ou une action de maintenance.

Lorsqu'elle doit le faire, la passerelle MG envoie une commande de changement de service au contrôleur MGC afin d'annoncer qu'elle a redémarré ou souhaite renégocier la version du protocole ou le profil H.248. La valeur "Restart" (redémarrage) de la méthode de changement de service et le code de raison du changement de service 900 ("Service rétabli") sont utilisés dans ce cas. Le code de raison du changement de service indiquera quelles actions peuvent devoir être effectuées par le contrôleur MGC.

#### **F.3.5 Reprise redondante de passerelle MG**

Il y a deux cas de reprise redondante du service par une passerelle MG:

- 1) reprise du service lancée par une passerelle MG primaire:  
lorsqu'une passerelle MG passe à l'état OutOfService et souhaite remettre le traitement à une certaine passerelle MG secondaire, elle envoie une commande de changement de service au contrôleur MGC avec la valeur de méthode de changement de service "Failover" (reprise sur défaillance) et une raison de changement de service 908 ("Défaillance imminente de la passerelle MG"). Le contrôleur MGC cessera la messagerie vers la passerelle MG primaire et mettra fin à l'association de commande. La passerelle MG secondaire doit ensuite envoyer une commande de changement de service avec la valeur "Restart" (redémarrage) de la méthode de changement de service et une raison de changement de service 900 ("Service rétabli"). Le contrôleur MGC doit ensuite établir une nouvelle association de commande avec la passerelle MG secondaire;
- 2) reprise du service lancée par une passerelle MG secondaire:  
lorsqu'une unité secondaire facultative d'une passerelle MG détecte un délestage dû à une action de maintenance ou à une défaillance de l'unité primaire de la passerelle MG et lorsque cette unité primaire est incapable de signaler la défaillance au contrôleur MGC, la

passerelle MG secondaire doit envoyer une commande de changement de service au contrôleur MGC avec la valeur de méthode de changement de service "Failover" et une raison de changement de service 919 ("Reprise à chaud sur défaillance") ou 920 ("Reprise à froid sur défaillance"), selon le cas. Les procédures du § F.3.4 sont applicables.

Le contrôleur MGC doit ignorer toutes les tentatives de reprise redondante du service par des passerelles MG inconnues. Le contrôleur MGC doit toujours avoir connaissance, par fourniture ou autrement, du fait que la passerelle MG secondaire annonçant la reprise du service est associée à la passerelle MG primaire qui a eu une défaillance et a donc l'autorisation de reprendre le service de la passerelle MG primaire.

Si la passerelle MG secondaire est connue et a l'autorisation de reprendre le service de la passerelle MG primaire, le contrôleur MGC doit essayer de communiquer avec la passerelle MG primaire afin de déterminer si elle fonctionne encore. Cela peut être effectué au moyen d'une commande AuditValue vide ou par une autre méthode appropriée (voir le § 11.5). Si la passerelle MG primaire répond, le contrôleur MGC doit rejeter la tentative de reprise sur défaillance. Si la passerelle MG primaire ne réussit pas à répondre, le contrôleur MGC doit supposer qu'elle a subi une défaillance et doit accepter la reprise à chaud ou à froid sur défaillance. Lorsque le contrôleur MGC accepte la reprise à chaud ou à froid sur défaillance, l'association de commande avec la passerelle MG primaire est supprimée et une nouvelle association de commande est établie avec la passerelle MG secondaire.

### **F.3.6 Perte de communication de passerelle MG**

Lorsque la passerelle MG a détecté une perte et un rétablissement subséquent de communication avec le contrôleur MGC, la passerelle MG envoie une commande de changement de service avec une valeur "Disconnected" (déconnexion) au contrôleur MGC se trouvant dans l'association de commande actuelle. Si ce contrôleur MGC ne réussit pas à répondre, la passerelle MG envoie alors une commande de changement de service avec une valeur "Failover" (reprise sur défaillance) et le code de raison du changement de service 909 ("Défaillance imminente du MGC") à chaque contrôleur MGC figurant dans sa liste tour à tour, jusqu'à ce qu'elle ait correctement établi une nouvelle association de commande ou qu'elle ait épuisé sa liste de contrôleurs MGC.

Si le contrôleur MGC qui se trouvait dans l'association de commande initiale répond à la commande de changement de service, l'association de commande continue sans interruption, et toutes les commandes sont traitées comme s'il n'y avait eu aucune perte de communication. Sinon, l'association de commande initiale est terminée lorsque la passerelle MG envoie une commande de changement de service à un nouveau contrôleur MGC, et toutes les commandes issues de la précédente association de commande sont ignorées. Une nouvelle association de commande est établie dès que le nouveau contrôleur MGC répond à la commande de changement de service, ce qui complète l'enregistrement.

Si la passerelle MG épuise sa liste de contrôleurs MGC sans établir correctement une association de commande, la passerelle MG attend pendant une durée aléatoire puis essaie encore de s'enregistrer auprès des contrôleurs MGC figurant dans sa liste, en commençant par le contrôleur MGC issu de l'association initiale de commande. La passerelle MG enverra une commande de changement de service avec une valeur "Disconnected" au contrôleur MGC issu de l'association initiale de commande chaque fois que cette passerelle MG essaiera de prendre contact avec lui. La passerelle MG envoie une commande de changement de service avec une valeur "Failover" à tous les autres contrôleurs MGC.

Les contrôleurs MGC qui reçoivent des commandes de changement de service avec une valeur "Disconnected" devraient examiner la passerelle MG afin de déterminer si une discordance d'états s'est produite à cause d'une perte de messages.

Dans ces scénarios, le code de raison du changement de service 900 ("Service rétabli") est recommandé, bien que des scénarios particuliers puissent nécessiter différents codes de raison du changement de service.

### **F.3.7 Changement de capacité de passerelle MG**

Afin d'annoncer un changement de capacité de la passerelle MG, celle-ci envoie une commande de changement de service avec une valeur appropriée de la méthode de changement de service et une raison de changement de service 916 ("Changement de paquetage") ou 917 ("Changement de capacité"). Le contrôleur MGC devrait examiner la passerelle MG afin de déterminer les nouvelles capacités de la passerelle MG. Pour une passerelle MG en service qui envoie une commande de changement de service avec la valeur "Restart" (redémarrage) de la méthode de changement de service, l'association de commande continue sans interruption après un changement de capacité, et la passerelle MG n'est pas considérée comme ayant subi un redémarrage du système.

### **F.3.8 Réenregistrement de passerelle MG lancé par le contrôleur MGC**

Le contrôleur MGC peut demander à la passerelle MG de se réenregistrer en émettant une commande de changement de service à la terminaison racine avec la valeur "Handoff" (transfert) de la méthode de changement de service, le code de raison du changement de service 903 ("Changement ordonné par le MGC") et son propre identificateur ServiceChangeMgcID (c'est-à-dire celui du contrôleur MGC actuel).

Voir au § F.3.2 l'action effectuée par la passerelle MG.

### **F.3.9 Rétablissement de service lancé par le contrôleur MGC**

Afin de demander que la passerelle MG procède à son propre redémarrage, le contrôleur MGC envoie une commande de changement de service à cette passerelle MG avec la valeur "Restart" (redémarrage) de la méthode de changement de service et un code approprié de raison du changement de service tel que 900 ("Service rétabli") ou 901 ("Démarrage à froid"). La passerelle MG doit toujours établir une nouvelle association de commande conformément au § F.3.2 en utilisant la raison indiquée du changement de service.

Afin de rétablir le service à une terminaison ou dans un groupe de terminaisons, le contrôleur MGC envoie une commande de changement de service avec la valeur "Restart" (redémarrage) de la méthode de changement de service à la ou aux terminaisons en question. Voir aux § F.4.1.2 et F.4.1.3 les actions effectuées par le contrôleur MGC.

Le paramètre "ServiceChangeDelay" (délai de changement de service) indique la durée avant que le rétablissement du service se produise. Voir au § F.5.3 de plus amples détails.

### **F.3.10 Annulation de service lancée par le contrôleur MGC**

#### **F.3.10.1 Terminaison racine**

Afin de mettre la passerelle MG hors service, le contrôleur MGC envoie une commande de changement de service avec une valeur "Forced" (changement forcé) ou "Graceful" (changement progressif) dans la méthode de changement de service à la terminaison racine. Les codes appropriés de raison du changement de service peuvent inclure 905 ("Terminaison mise hors service"), entre autres codes. Voir au § F.4.1.1 les actions effectuées par le contrôleur MGC.

Le paramètre "ServiceChangeDelay" (délai de changement de service) indique la durée à la fin de laquelle l'annulation du service se produit. Voir au § F.5.3 de plus amples détails.

Afin d'annuler une valeur précédemment émise (et acquittée) "Graceful" (changement progressif) pour l'ensemble de la passerelle MG, le contrôleur MGC envoie une commande de changement de service à la terminaison racine avec une valeur "Restart" et le code de raison du changement de service 918 ("Annulation de la progressivité"). La passerelle MG doit rester en service, et toutes les

terminaisons déjà réglées sur la mise hors service sont remises en service sauf signalisation contraire par la passerelle MG. Si une annulation de la progressivité est reçue après l'expiration du temporisateur de délai, la passerelle MG doit signaler un code d'erreur 502 ("Pas encore prête") au contrôleur MGC. La passerelle MG sera tenue de se réenregistrer afin de reprendre son service.

### **F.3.10.2 Terminaisons physiques**

Afin de mettre une terminaison ou un groupe de terminaisons hors service, le contrôleur MGC envoie à la ou aux terminaisons en question une commande de changement de service avec une valeur "Forced" (changement forcé) ou "Graceful" (changement progressif) dans la méthode de changement de service. Les codes appropriés de raison du changement de service peuvent inclure 904 ("Terminaison défectueuse"), 905 ("Terminaison mise hors service"), 906 ("Perte de connectivité de couche inférieure"), ou 907 ("Panne de transmission"), entre autres codes. Voir aux § F.4.2 et F.4.3 les actions effectuées par le contrôleur MGC.

Le paramètre "ServiceChangeDelay" (délai de changement de service) indique la durée à la fin de laquelle l'annulation du service se produit. Voir au § F.5.3 de plus amples détails.

Afin d'annuler une valeur précédemment émise (et acquittée) "Graceful" (changement progressif) à une terminaison ou dans un groupe de terminaisons, la passerelle MG envoie une commande de changement de service avec une valeur "Restart" et le code de raison du changement de service 918 ("Annulation de la progressivité") à la ou aux terminaisons en question. La terminaison doit rester en service. Si la terminaison a déjà passé à l'état hors service, elle doit être remise en service exactement comme elle le serait avec une valeur de redémarrage (Restart) du changement de service.

### **F.3.10.3 Terminaisons éphémères**

Le contrôleur MGC ne doit pas utiliser la procédure de changement de service afin d'annuler le service sur des terminaisons éphémères. Le fait de soustraire la terminaison du contexte suffit à éliminer cette terminaison.

## **F.3.11 Reprise redondante sur défaillance du contrôleur MGC**

Lorsque le contrôleur MGC contenu dans une association de commande rencontre une condition de maintenance ou de défaillance telle qu'il doit absolument passer à l'état hors service, ce contrôleur peut diriger la passerelle MG vers un contrôleur MGC spécifique en envoyant une commande de changement de service avec une valeur "Handoff", une raison de changement de service 903 ("Changement ordonné par le MGC") et l'adresse du nouveau contrôleur MGC dans le paramètre "ServiceChangeMgcID". L'association de commande est terminée dès réception de la commande "Reply" en provenance de la passerelle MG. Celle-ci envoie ensuite une commande de changement de service au contrôleur MGC spécifié avec une valeur "Handoff" et une raison de changement de service 903 ("Changement ordonné par le MGC"). Une nouvelle association de commande est établie dès réception de la commande "Reply" issue du contrôleur MGC.

Si le contrôleur MGC spécifié rejette la tentative de transfert ou s'il est incapable d'envoyer une commande de changement de service avec la valeur de méthode de changement de service "Handoff", la passerelle MG doit revenir aux procédures relatives à la perte de communication qui sont décrites dans le § F.3.6, en commençant par son unité primaire configurée.

## **F.4 Description de l'élément ServiceChange**

### **F.4.1 Méthode ServiceChangeMethod**

Le présent paragraphe décrit le comportement de la méthode ServiceChangeMethod sur les différents types de terminaison. Le présent paragraphe est subdivisé comme suit:

- la terminaison racine;

- les terminaisons physiques;
- les terminaisons éphémères.

#### **F.4.1.1 Comportement de la méthode de changement de service à la terminaison racine**

La commande de changement de service à la terminaison racine a différents effets selon la méthode de changement de service. Les résultats de chaque méthode de changement de service sont décrits ci-dessous:

- 1) Restart (redémarrage) – valeur envoyée par la passerelle MG afin d'annoncer qu'elle a redémarré, qu'elle souhaite renégocier la version du protocole, le profil, ou qu'un changement de capacité va avoir lieu. Le code de raison du changement de service indique quelles actions peuvent devoir être effectuées par le contrôleur MGC. Lorsque cette valeur est envoyée par le contrôleur MGC, la passerelle MG doit procéder à son propre redémarrage en utilisant la raison indiquée du changement de service.
- 2) Forced (changement forcé) – lorsque cette valeur est envoyée par la passerelle MG, celle-ci indique qu'elle passe immédiatement à l'état OutOfService (hors service). L'association de commande est terminée par la passerelle MG dès réception de la commande "Reply" issue du contrôleur MGC. Lorsque cette valeur est envoyée par le contrôleur MGC, la passerelle MG doit passer elle-même à l'état OutOfService et mettre fin à l'association de commande après avoir envoyé la commande "Reply". Noter qu'un contrôleur MGC ne peut pas remettre une passerelle MG en service, bien qu'il puisse lui demander de passer à l'état hors service. La passerelle MG lance son enregistrement auprès du contrôleur MGC afin d'établir une nouvelle association de commande et d'indiquer le rétablissement du service, exactement comme elle le ferait pour tout autre enregistrement. Le paramètre "ServiceChangeDelay" (délai de changement de service) n'a aucun effet sur la valeur "Forced" de la méthode de changement de service.
- 3) Graceful (changement progressif) – lorsque cette valeur est envoyée par la passerelle MG, elle indique que celle-ci passe à l'état OutOfService à la fin de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). L'association de commande est terminée à la fin de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). Lorsque cette valeur est envoyée par le contrôleur MGC, la passerelle MG doit passer elle-même à l'état OutOfService et mettra fin à l'association de commande à la fin de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). L'utilisation du paramètre "ServiceChangeDelay" (délai de changement de service) de valeur égale à zéro ou l'absence du paramètre "ServiceChangeDelay" (délai de changement de service) indique que la passerelle MG doit passer à l'état hors service et mettre fin à l'association de commande lorsque le dernier contexte est retiré par soustraction de ses terminaisons, et que le contrôleur MGC ne doit pas ajouter de nouvelles connexions. La passerelle MG devrait régler à la valeur "OutOfService" la propriété "ServiceStates" de la terminaison racine lors de l'expiration du paramètre "ServiceChangeDelay" (délai de changement de service) ou lors du retrait de toutes les terminaisons de leur contexte actif (selon ce qui intervient en premier). Afin d'annuler une commande de changement de service déjà envoyée (et acquittée) contenant la valeur de méthode de changement de service "Graceful", l'entité lançant la méthode de changement de service progressive envoie une commande de changement de service avec la valeur de méthode de changement de service "Restart" (redémarrage) et le code de raison du changement de service 918 ("Annulation de la progressivité").
- 4) Failover (reprise sur défaillance) – lorsque cette valeur est envoyée par la passerelle MG à un contrôleur MGC qui n'est plus dans l'association de commande actuelle, la passerelle MG indique que son contrôleur MGC actuel a subi une défaillance, et elle va essayer de s'enregistrer auprès du destinataire de la commande. Lorsque cette valeur est envoyée par la

passerelle MG au contrôleur MGC dans l'association de commande actuelle, la passerelle MG primaire indique qu'une passerelle MG secondaire a repris le service pour l'unité primaire défaillante. Dans un cas comme dans l'autre, l'association précédente de commande est terminée par l'envoi de la commande de changement de service, et la nouvelle association de commande est établie entre la passerelle MG et le nouveau contrôleur MGC, ou entre celui-ci et la nouvelle passerelle MG dès réception de la commande "Reply". Les contrôleurs MGC ne doivent pas envoyer de commandes de changement de service avec la méthode de changement de service mise à la valeur "Failover".

- 5) Handoff (transfert) – lorsque cette valeur est envoyée par le contrôleur MGC, celui-ci indique que la passerelle MG va être transférée à un nouveau contrôleur MGC. Cette opération met fin à l'association de commande actuelle, dès réception de la commande "Reply". Lorsque cette valeur est envoyée par la passerelle MG au contrôleur MGC, cela indique que la passerelle MG va essayer d'établir une nouvelle association de commande conformément à une commande de transfert reçue du contrôleur MGC dans sa précédente association de commande. Lorsque cette valeur est envoyée par la passerelle MG au contrôleur MGC, un transfert est un enregistrement, et une nouvelle association de commande est établie dès réception de la commande "Reply". La passerelle MG ne doit pas utiliser la valeur "Handoff" sans y avoir été invitée par un contrôleur MGC.
- 6) Disconnected (déconnexion) – lorsque cette valeur est envoyée par la passerelle MG, elle indique que la communication dans l'association de commande actuelle avait été perdue, mais qu'elle est maintenant rétablie. L'association de commande actuelle est renouvelée. Les contrôleurs MGC ne doivent pas envoyer de commandes de changement de service avec la méthode de changement de service mise à la valeur "Disconnected".

#### **F.4.1.2 Comportement de la méthode de changement de service aux terminaisons physiques**

Alors que la passerelle MG est dans une association active de commande, l'émission de la commande de changement de service à une terminaison physique a différents effets selon la méthode de changement de service. Les résultats de chaque méthode sont décrits ci-dessous:

- 1) Restart (redémarrage) – lorsque cette valeur est envoyée par la passerelle MG, celle-ci annonce que la ou les terminaisons ont redémarré, ou qu'un changement de capacité va avoir lieu. Le code de raison du changement de service indique quelles actions peuvent devoir être effectuées par le contrôleur MGC. Lorsque cette valeur est envoyée par le contrôleur MGC, la passerelle MG doit procéder au redémarrage de la ou des terminaisons en utilisant la raison indiquée du changement de service.
- 2) Forced (changement forcé) – lorsque cette valeur est envoyée par la passerelle MG, elle indique que la terminaison passe immédiatement à l'état OutOfService. Lorsque cette valeur est envoyée par le contrôleur MGC, la passerelle MG doit immédiatement mettre hors service la terminaison. Dans les deux cas, le paramètre "ServiceStates" doit être mis à la valeur "OutOfService" et le contrôleur MGC est chargé de nettoyer tous contextes ou ressources associés à la terminaison. Le paramètre "ServiceChangeDelay" (délai de changement de service) n'a aucun effet sur la méthode de changement de service forcé.
- 3) Graceful (changement progressif) – lorsque cette valeur est envoyée par la passerelle MG, elle indique que la ou les terminaisons passent à l'état OutOfService après la durée indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). Lorsque cette valeur est envoyée par le contrôleur MGC, la passerelle MG doit mettre la ou les terminaisons hors service à la fin de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). La propriété "ServiceStates" doit être mise à la valeur "OutOfService" à l'expiration de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service) ou lorsque la ou les terminaisons sont retirées d'un contexte actif (selon ce qui intervient en premier) et le contrôleur MGC

est chargé de nettoyer tous contextes ou ressources associés à la ou aux terminaisons. L'utilisation du paramètre "ServiceChangeDelay" (délai de changement de service) de valeur égale à zéro ou l'absence du paramètre "ServiceChangeDelay" (délai de changement de service) indique que la terminaison doit passer à l'état hors service lorsqu'elle est retirée du contexte par soustraction. Le contrôleur MGC ne doit pas utiliser la ou les terminaisons indiquées pour une connexion tant que la valeur "Graceful" n'est pas annulée ou tant que la terminaison n'a pas été remise en service par une commande subséquente de changement de service. Afin d'annuler une commande de changement de service déjà envoyée (et acquittée) avec la valeur de méthode de changement de service égale à "Graceful", l'entité lançant le changement progressif envoie une commande de changement de service avec la valeur "Restart" (redémarrage) pour la méthode de changement de service et le code de raison du changement de service 918 ("Annulation de la progressivité").

- 4) Failover (reprise sur défaillance) – la méthode de changement de service "Failover" ne doit pas être utilisée avec des terminaisons autres que la terminaison racine.
- 5) Handoff (transfert) – la méthode de changement de service "Handoff" ne doit pas être utilisée avec des terminaisons autres que la terminaison racine.
- 6) Disconnected (déconnexion) – la méthode de changement de service "Disconnected" ne doit pas être utilisée avec des terminaisons autres que la terminaison racine.

#### **F.4.1.3 Comportement de la méthode de changement de service aux terminaisons éphémères**

Lorsque la passerelle MG est dans une association active de commande, l'émission de la commande de changement de service à une terminaison éphémère a différents effets selon la méthode de changement de service. Les résultats de chaque méthode sont décrits ci-dessous:

- 1) Restart (redémarrage) – lorsque cette valeur est envoyée par la passerelle MG, celle-ci annonce que la ou les terminaisons ont redémarré, ou vont annoncer un changement de capacité. Le code de raison du changement de service indique quelles actions peuvent devoir être effectuées par le contrôleur MGC. Celui-ci ne doit pas envoyer la valeur "Restart" (redémarrage) de la méthode de changement de service pour des terminaisons éphémères.
- 2) Forced (changement forcé) – lorsque cette valeur est envoyée par la passerelle MG, elle indique que la ou les terminaisons passent immédiatement à l'état OutOfService. Le contrôleur MGC est chargé de soustraire la terminaison. Le contrôleur MGC ne doit pas envoyer la valeur "Forced" de la méthode de changement de service pour des terminaisons éphémères. Le paramètre "ServiceChangeDelay" (délai de changement de service) n'a aucun effet sur la valeur "Forced" de la méthode de changement de service.
- 3) Graceful (changement progressif) – lorsque cette valeur est envoyée par la passerelle MG, elle indique que la ou les terminaisons passent à l'état OutOfService à la fin de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). Le contrôleur MGC est chargé de soustraire la ou les terminaisons à l'expiration de la période indiquée par le paramètre "ServiceChangeDelay" (délai de changement de service). Le contrôleur MGC ne doit pas envoyer la valeur "Graceful" de la méthode de changement de service pour des terminaisons éphémères. L'utilisation du paramètre "ServiceChangeDelay" (délai de changement de service) de valeur égale à zéro indique que la terminaison doit être détruite lorsqu'elle est retirée du contexte par soustraction. La passerelle MG devrait régler à la valeur "OutOfService" la propriété "ServiceStates" de la terminaison dès l'expiration du paramètre "ServiceChangeDelay" (délai de changement de service) ou dès la suppression de la terminaison d'un contexte actif (selon ce qui intervient en premier). Afin d'annuler une commande de changement de service déjà envoyée (et acquittée) avec la valeur "Graceful", l'entité lançant le changement progressif envoie une commande de changement de service avec la valeur de méthode de changement de service "Restart" et avec le code de raison du changement de service 918: "Annulation de la progressivité".

- 4) Failover (reprise sur défaillance) – la méthode de changement de service "Failover" ne doit pas être utilisée avec des terminaisons autres que la terminaison racine.
- 5) Handoff (transfert) – la méthode de changement de service "Handoff" ne doit pas être utilisée avec des terminaisons autres que la terminaison racine.
- 6) Disconnected (déconnexion) – la méthode de changement de service "Disconnected" ne doit pas être utilisée avec des terminaisons autres que la terminaison racine.

## F.5 Utilisation des paramètres de l'élément ServiceChange

### F.5.1 ServiceChangeMethod

Voir l'usage décrit dans le § F.4.

### F.5.2 ServiceChangeMethod

Les codes de raison du changement de service permettent au destinataire de modifier ses comportements afin de s'adapter à une situation particulière. Par exemple, si une passerelle MG a envoyé une raison de changement de service 901 ("Démarrage à froid") au contrôleur MGC lorsqu'elle a redémarré, le contrôleur MGC pourrait supposer que cette passerelle MG a perdu la totalité de son état: dans ces conditions, il ne va pas effectuer de maintenance mais va examiner les tâches afin d'évaluer et de nettoyer la passerelle MG en vue de la remettre dans un état utilisable.

Le Tableau F.1 montre avec quelles méthodes de changement de service chaque code de raison de changement de service peut être envoyé.

**Tableau F.1/H.248.1 – Mappage de la méthode de changement de service avec le code de raison du changement de service**

Raison du changement	Méthode de changement de service						Description
	Redémarrage	Forcé	Progressif	Déconnexion	Reprise sur déf.	Transfert	
900	X			Racine seule MG seule			Service rétabli
901	Racine seule						Démarrage à froid
902	Racine seule						Démarrage à chaud
903						Racine seule	Changement ordonné par le MGC
904		X	X				Terminaison défectueuse
905		X	X				Terminaison mise hors service
906		X	X				Perte de connexité de couche inférieure
907		X	X				Panne de transmission
908		Racine seule MG seule	Racine seule MG seule		Racine seule MG seule		Défaillance imminente de la passerelle MG

**Tableau F.1/H.248.1 – Mappage de la méthode de changement de service avec le code de raison du changement de service**

Raison du changement	Méthode de changement de service						Description
	Redémarrage	Forcé	Progressif	Déconnexion	Reprise sur déf.	Transfert	
909					Racine seule MG seule		Défaillance imminente du contrôleur MGC
910	MG seule	X	X				Défaillance de capacité de média
911	MG seule	X	X				Défaillance de capacité de modem
912	MG seule	X	X				Défaillance de capacité de multiplexage
913	MG seule	X	X				Défaillance de capacité de signal
914	MG seule	X	X				Défaillance de capacité d'événement
915		X	X				Perte d'état
916	X			Racine seule MG seule	Racine seule MG seule		Changement de paquetage
917	X			Racine seule MG seule	Racine seule MG seule		Changement de capacité
918	X						Annulation de la progressivité
919					Racine seule MG seule		Reprise à chaud sur défaillance
920					Racine seule MG seule		Reprise à froid sur défaillance

### F.5.3 ServiceChangeDelay

Le paramètre "ServiceChangeDelay" (délai de changement de service) sert à fournir un délai avant que la commande de changement de service prenne effet et que la propriété "ServiceStates" de la passerelle MG ou de la ou des terminaisons soit modifiée. La commande de changement de service n'a fait l'objet d'une réponse que si l'exécution a eu lieu dès réception du message; mais en fait, la passerelle MG ou la ou les terminaisons ne modifient pas l'état tant que le paramètre "ServiceChangeDelay" (délai de changement de service) n'a pas expiré. Un paramètre "ServiceChangeDelay" (délai de changement de service) de valeur égale à zéro est équivalent à une absence de délai de changement de service, à l'exception de son utilisation combinée avec la méthode de changement de service "Graceful". Voir au § F.4 les effets du paramètre "ServiceChangeDelay" (délai de changement de service) sur les diverses méthodes de changement de service.

#### **F.5.4 ServiceChangeAddress**

L'utilisation du paramètre "ServiceChangeAddress" est décrite dans le § 7.2.8 ci-dessus. L'utilisation de ce paramètre est déconseillée. S'il est présent, il ne peut être utilisé qu'avec des commandes de changement de service à la terminaison racine, et toutes nouvelles transactions doivent être envoyées à la nouvelle adresse et/ou au numéro d'accès spécifié. Les réponses doivent être renvoyées à l'adresse à partir de laquelle la requête correspondante est arrivée.

#### **F.5.5 ServiceChangeProfile**

Le paramètre de profil de changement de service autorise le contrôleur MGC et la passerelle MG à négocier le profil H.248 à utiliser dans l'association de commande. Le contrôleur MGC peut toujours examiner la passerelle MG afin de déterminer si elle prend en charge d'autres capacités. Le profil de changement de service ne doit être envoyé que par des commandes d'enregistrement ou de réenregistrement.

#### **F.5.6 ServiceChangeVersion**

La version du changement de service sert à négocier la version du protocole H.248 à utiliser entre contrôleur MGC et passerelle MG. La version du changement de service est obligatoire lors de l'enregistrement initial, et devrait être envoyée par toutes les autres commandes d'enregistrement lorsqu'une négociation de la version de protocole doit avoir lieu. La version du changement de service ne doit pas être envoyée dans des commandes autres que d'enregistrement. Voir au § F.3.1 les procédures d'enregistrement.

#### **F.5.7 ServiceChangeMgcID**

Le contrôleur MGC peut envoyer ce paramètre dans une commande de changement de service dirigée vers la terminaison racine. Dès réception pendant une tentative d'enregistrement, la passerelle MG doit tenter de s'enregistrer auprès du contrôleur MGC à l'adresse spécifiée. Lorsque ce paramètre est reçu dans une commande de transfert en provenance du contrôleur MGC primaire de la passerelle MG, celle-ci doit utiliser les procédures décrites dans le § 11.5.

#### **F.5.8 TimeStamp**

L'utilisation du paramètre facultatif TimeStamp (marqueur temporel) est décrite dans le § 7.2.8 ci-dessus. Le paramètre TimeStamp n'a aucun effet sur l'exécution de la commande de changement de service, mais peut servir au récepteur de la commande de changement de service à d'autres fins, comme la facturation ou la coordination du rythme.

### **F.6 Changement de service par rapport à état de terminaison**

La propriété "ServiceStates" contenue dans le descripteur "TerminationState" conserve la valeur actuelle de l'état de la terminaison. La propriété ServiceStates peut avoir trois valeurs: "InService", "OutOfService" et "Test".

Le contrôleur MGC ne doit utiliser que des commandes Modify afin de régler la propriété "ServiceStates" à la valeur "Test" ou à partir de celle-ci. Toutes les autres modifications de la propriété "ServiceStates" sont une violation du protocole et doivent recevoir le code d'erreur 401 ("Erreur de protocole"). Afin de faire passer l'état "InService" à la valeur "OutOfService" ou vice versa, la commande de changement de service doit toujours être utilisée.

Pour les terminaisons dans l'état "Test", la passerelle MG ne doit envoyer au contrôleur MGC que des commandes de changement de service avec une valeur "Forced" (changement forcé) ou "Graceful" (changement progressif) dans la méthode de changement de service. Des commandes de changement de service ultérieures pourront ensuite régler la propriété "ServiceStates" de la terminaison à la valeur "InService".

Les récepteurs devraient faire tout leur possible pour accepter les commandes de changement de service, mais dans certains cas un rejet de la commande est garanti. Par exemple, si une passerelle MG a envoyé une commande de changement de service afin de remettre une terminaison en service, mais que le contrôleur MGC ne se soit pas fait attribuer les ressources nécessaires afin de desservir cette terminaison en raison d'un manque de fourniture ou pour une autre raison, alors ce contrôleur MGC peut rejeter cette commande. Si l'entité réceptrice rejette la commande de changement de service, l'état de la terminaison reste sans changement. L'expéditeur peut décider d'attendre un certain temps puis de réessayer, ou d'attendre que le récepteur tente de modifier l'état de la terminaison. Les demandes visant à mettre une terminaison hors service devraient être suivies d'effet.

Le Tableau F.2 montre les commandes qui peuvent servir à effectuer un changement d'état à une terminaison.

**Tableau F.2/H.248 – Commandes de transition d'état et leurs effets**

Etat actuel	Nouvel état	Commande	MGC autorisé?	MG autorisée?
InService	Test	ServiceChange	Impossible	Impossible
		Modify	Oui	Impossible
InService	OutOfService	ServiceChange	Oui, terminaisons physiques et terminaison racine seulement	Oui
		Modify	No	Impossible
OutOfService	InService	ServiceChange	Oui, mais pas à la terminaison racine	Oui
		Modify	Non	Impossible
OutOfService	Test	ServiceChange	Impossible	Impossible
		Modify	Oui	Impossible
Test	InService	ServiceChange	Non	Non
		Modify	Oui	Impossible
Test	OutOfService	ServiceChange	Non	Oui
		Modify	Oui	Impossible

## Appendice I

### Exemples de flux de communication

Tous les réalisateurs du protocole de la présente Recommandation doivent en lire attentivement la partie normative avant de l'appliquer à des implémentations. Aucun d'eux ne devrait utiliser les exemples donnés dans le présent appendice en tant qu'explications autonomes sur la façon de créer des messages de protocole.

Les exemples donnés dans le présent appendice font appel au protocole SDP pour le codage des descripteurs de flux "Local" et "Remote". Le protocole SDP est défini dans le document RFC 2327. En cas de divergence dans les exemples entre le protocole SDP et le document RFC 2327, c'est ce dernier qui doit être consulté comme référence. Les profils audio utilisés sont définis dans le document RFC 1890 ou enregistrés auprès de l'autorité IANA. Par exemple, la loi A de la Rec. UIT-T G.711 est désignée par "MIC-A" dans le protocole SDP et est attribuée au profil 0. Le codage G.723.1 est nommé "G723" et correspond au profil 4; le codage H.263 est nommé "H263" et correspond au profil 34. Voir également <http://www.iana.org/assignments/rtp-parameters>.

## I.1 Communication de passerelle résidentielle à passerelle résidentielle

Cet exemple de scénario décrit l'emploi des éléments du protocole afin d'établir une communication de passerelle résidentielle à passerelle résidentielle sur un réseau en protocole IP. Par concision, cet exemple part du principe que les deux passerelles résidentielles impliquées dans la communication sont régies par le même contrôleur MGC.

### I.1.1 Programmation du comportement de repos de terminaisons de ligne analogique passant par une passerelle résidentielle

On trouvera ci-dessous la description des invocations d'interface API issues du contrôleur MGC et des passerelles MG pour faire en sorte que les terminaisons de ce scénario soient programmées au comportement de repos. La passerelle MG d'origine comme son homologue de destination a des terminaisons de ligne analogique au repos qui sont programmées pour rechercher les événements de lancement d'appel (c'est-à-dire de décrochage) au moyen de la commande Modify contenant les paramètres appropriés. Le contexte NULL est utilisé pour indiquer que les terminaisons ne sont pas encore associées à un contexte. La terminaison racine est utilisée pour renvoyer à la passerelle MG entière et non pas à une terminaison contenue dans cette passerelle.

Dans cet exemple, la passerelle MG1 possède l'adresse IP 124.124.124.222. La passerelle MG2 possède l'adresse IP 125.125.125.111 et le contrôleur MGC 123.123.123.4. L'accès Megaco par défaut est 55555 pour les trois entités.

- 1) Une passerelle MG s'enregistre avec un contrôleur MGC au moyen de la commande ServiceChange:

MG1 à MGC:

```
MEGACO/1 [124.124.124.222]
Transaction = 9998 {
  Context = - {
    ServiceChange = ROOT {Services {
      Method=Restart, Version=3,
      ServiceChangeAddress=55555, Profile=ResGW/1}
    }
  }
}
```

- 2) Le contrôleur MGC envoie la réponse suivante:

MGC à MG1:

```
MEGACO/1 [123.123.123.4]:55555
Reply = 9998 {
  Context = - {ServiceChange = ROOT {
    Services {ServiceChangeAddress=55555, Profile=ResGW/1} } }
}
```

- 3) Le contrôleur MGC programme une terminaison dans le contexte NULL. L'identificateur de cette terminaison est A4444, l'identificateur de flux est 1, l'identificateur de demande contenu dans le descripteur "Events" est 2222. L'identificateur MID est celui de l'expéditeur de ce message; dans ce cas, il s'agit de l'adresse IP et de l'accès [123.123.123.4]:55555. La propriété "Mode" pour ce flux est réglée à "SendRecv" (envoyer et recevoir). Le paramètre "al" est le paquetage de supervision de la ligne analogique. On suppose que les descripteurs "Local" et "Remote" sont fournis.

MGC à MG1:

```
MEGACO/3 [123.123.123.4]:55555
Transaction = 9999 {
  Context = - {
    Modify = A4444 {
```

```

Media { Stream = 1 {
    LocalControl {
        Mode = SendRecv,
        tdmc/gain=2, ; in dB,
        tdmc/ec=on
    },
},
Events = 2222 {al/of {strict=state}}
}
}

```

Le script de numérotation peut avoir été chargé préalablement dans la passerelle MG. Sa fonction sera d'attendre le décrochage, d'activer la tonalité d'invitation à numérotter et de commencer à acquérir des chiffres par tonalités DTMF. Dans cet exemple cependant, l'on utilise le script de numérotation, qui est mis en place après la détection du décrochage (étape 5) ci-dessous).

Noter que le descripteur "Events" imbriqué peut avoir été utilisé pour combiner les étapes 3) et 4) avec les étapes 8) et 9), ce qui élimine les étapes 6) et 7).

4) La passerelle MG1 accepte la commande Modify avec cette réponse:

```

MG1 à MGC:
MEGACO/3 [124.124.124.222]:55555
Reply = 9999 {
    Context = - {Modify = A4444}
}

```

5) Un échange similaire se déroule entre la passerelle MG2 et le contrôleur MGC, ce qui produit une terminaison au repos appelée A5555.

### I.1.2 Acquisition des chiffres de l'expéditeur et de la terminaison d'origine

La procédure suivante est fondée sur les conditions indiquées ci-dessus. Elle décrit les transactions issues du contrôleur MGC et de la passerelle média d'origine (MG1, *originating media gateway*) afin d'acquérir la terminaison d'origine (A4444) en passant par les étapes d'acquisition de chiffres nécessaires pour ouvrir une connexion vers la passerelle média de destination (MG2, *terminating media gateway*).

6) La passerelle MG1 détecte un événement de décrochage en provenance de l'utilisateur 1 et le signale au contrôleur MGC au moyen de la commande Notify.

```

MG1 à MGC:
MEGACO/3 [124.124.124.222]:55555
Transaction = 10000 {
    Context = - {
        Notify = A4444 {ObservedEvents =2222 {
            19990729T22000000:al/of(init=OFF)}}
    }
}

```

7) Et la commande Notify est acquittée.

```

MGC à MG1:
MEGACO/3 [123.123.123.4]:55555
Reply = 10000 {
    Context = - {Notify = A4444}
}

```

- 8) Le contrôleur MGC modifie la terminaison afin de jouer la tonalité de numérotation, de rechercher des chiffres conformément à Dialplan0 et de chercher maintenant un événement de raccrochage.

```
MGC à MG1:
MEGACO/3 [123.123.123.4]:55555
Transaction = 10001 {
  Context = - {
    Modify = A4444 {
      Events = 2223 {
        al/on(strict=state), dd/ce {DigitMap=Dialplan0}
      },
      Signals {cg/dt},
      DigitMap= Dialplan0{
(0| 00| [1-7]xxx|8xxxxxxxx|Fxxxxxxxx|Exx|91xxxxxxxxxxxx|9011x.)}
      }
    }
  }
}
```

- 9) Et la commande Modify est acquittée.

```
MG1 à MGC:
MEGACO/3 [124.124.124.222]:55555
Reply = 10001 {
  Context = - {Modify = A4444}
}
```

- 10) Ensuite, les chiffres sont accumulés par la passerelle MG1 au fur et à mesure qu'ils sont composés par l'utilisateur 1. La tonalité d'invitation à numéroter est arrêtée à la détection du premier chiffre. Lorsqu'une correspondance appropriée est réalisée pour les chiffres recueillis avec le plan de numérotation couramment programmé pour A4444, une autre commande Notify est envoyée au contrôleur de passerelle média.

```
MG1 à MGC:
MEGACO/3 [124.124.124.222]:55555
Transaction = 10002 {
  Context = - {
    Notify = A4444 {ObservedEvents =2223 {
      19990729T22010001:dd/ce{ds="916135551212",Meth=UM}}}
  }
}
```

- 11) Et la commande Notify est acquittée.

```
MGC à MG1:
MEGACO/3 [123.123.123.4]:55555
Reply = 10002 {
  Context = - {Notify = A4444}
}
```

- 12) Le contrôleur analyse ensuite les chiffres et détermine qu'une connexion doit être établie de la passerelle MG1 à la passerelle MG2. Aussi bien la terminaison TDM A4444 qu'une terminaison RTP sont ajoutées à un nouveau contexte dans la passerelle MG1. Le mode est réception seulement (RecvOnly) car les valeurs du descripteur "Remote" ne sont pas encore spécifiées. Les codecs préférés sont énumérés dans l'ordre de préférence du contrôleur MGC.

```
MGC à MG1:
MEGACO/3 [123.123.123.4]:55555
Transaction = 10003 {
  Context = $ {
    Add = A4444,
    Add = $ {
```



```

        Events=1234{al/of(strict=state)},
        Signals {al/ri}
    },
    Add = $ {Media {
        Stream = 1 {
            LocalControl {
                Mode = SendRecv,
                nt/jit=40 ; in ms
            },
            Local {
v=0
c=IN IP4 $
m=audio $ RTP/AVP 4
a=ptime:30
                },
                Remote {
v=0
c=IN IP4 124.124.124.222
m=audio 2222 RTP/AVP 4
a=ptime:30
                } ; le profil RTP pour G.723.1 est 4
            }
        }
    }
}

```

- 15) Cette connexion est acquittée. Le numéro d'accès au flux est différent du numéro d'accès de commande. Dans ce cas, ce numéro est 1111 (dans le protocole SDP).

```

MG2 à MGC:
MEGACO/3 [125.125.125.111]:55555
Reply = 50003 {
    Context = 5000 {
        Add = A5555,
        Add = A5556{
            Media {
                Stream = 1 {
                    Local {
v=0

o=- 7736844526 7736842807 IN IP4 125.125.125.111
s=-
t= 0 0
c=IN IP4 125.125.125.111
m=audio 1111 RTP/AVP 4
                }
            } ; le profil RTP pour G.723.1 est 4
        }
    }
}

```

- 16) L'adresse IP et l'accès UDP ci-dessus définis doivent maintenant être indiqués à la passerelle MG1.

```

MGC à MG1:
MEGACO/3 [123.123.123.4]:55555
Transaction = 10005 {
    Context = 2000 {
        Modify = A4444 {
            Signals {cg/rt}
        },
        Modify = A4445 {

```

```

Media {
  Stream = 1 {
    Remote {
v=0

o=- 7736844526 7736842807 IN IP4 125.125.125.111
s=-
t= 0 0
c=IN IP4 125.125.125.111
m=audio 1111 RTP/AVP 4
    }
  } ; le profil RTP pour G.723.1 est 4
}
}
}
}
}

```

MG1 à MGC:

MEGACO/3 [124.124.124.222]:55555

Reply = 10005 {

Context = 2000 {Modify = A4444, Modify = A4445}

}

- 17) Les deux passerelles sont maintenant connectées et l'utilisateur 1 entend le retour d'appel. La passerelle MG2 attend maintenant que l'utilisateur 2 soulève le combiné pour que la communication soit établie dans les deux sens.

De MG2 à MGC:

MEGACO/3 [125.125.125.111]:55555

Transaction = 50005 {

Context = 5000 {

Notify = A5555 {ObservedEvents =1234 {  
19990729T22020002:al/of(init=off)}}

}

}

De MGC à MG2:

MEGACO/3 [123.123.123.4]:55555

Reply = 50005 {

Context = - {Notify = A5555}

}

De MGC à MG2:

MEGACO/3 [123.123.123.4]:55555

Transaction = 50006 {

Context = 5000 {

Modify = A5555 {

Events = 1235 {al/on(strict=state)},  
Signals ; pour couper la sonnerie

}

}

}

De MG2 à MGC:

MEGACO/3 [125.125.125.111]:55555

Reply = 50006 {

Context = 5000 {Modify = A4445}

}

- 18) Dans la passerelle MG1, le mode est commuté sur émission-réception et le retour d'appel est arrêté.

```
MGC à MG1:
MEGACO/3 [123.123.123.4]:55555
Transaction = 10006 {
  Context = 2000 {
    Modify = A4445 {
      Media {
        Stream = 1 {
          LocalControl {
            Mode=SendRecv
          }
        }
      }
    },
    Modify = A4444 {
      Signals
    }
  }
}
```

```
De MG1 à MGC:
MEGACO/3 [124.124.124.222]:55555
Reply = 10006 {
  Context = 2000 {Modify = A4445, Modify = A4444}}
```

- 19) Le contrôleur MGC décide d'envoyer la commande Audit à la terminaison RTP de la passerelle MG2.

```
MEGACO/3 [123.123.123.4]:55555
Transaction = 50007 {
  Context = 5000 {AuditValue = A5556{
    Audit{Media, DigitMap, Events, Signals, Packages, Statistics }}
  }
}
```

- 20) La passerelle MG2 répond.

```
MEGACO/3 [125.125.125.111]:55555
Reply = 50007 {
  Context = 5000 {
    AuditValue = A5556 {
      Media {
        TerminationState { ServiceStates = InService,
          Buffer = OFF },
        Stream = 1 {
          LocalControl { Mode = SendRecv,
            nt/jit=40 },
          Local {
```

v=0

```
o=- 7736844526 7736842807 IN IP4 125.125.125.111
s=-
t= 0 0
c=IN IP4 125.125.125.111
m=audio 1111 RTP/AVP 4
a=ptime:30
```

```
},
Remote {
```

v=0

```
o=- 2890844526 2890842807 IN IP4 124.124.124.222
s=-
```

```

t= 0 0
c=IN IP4 124.124.124.222
m=audio 2222 RTP/AVP 4
a=ptime:30
    } } },
    Events,
    Signals,
    DigitMap,
    Packages {nt-1, rtp-1},
    Statistics { rtp/ps=1200,    ; paquets envoyés
                 nt/os=62300,   ; octets envoyés
                 rtp/pr=700,    ; paquets reçus
                 nt/or=45100,   ; octets reçus
                 rtp/pl=0.2,    ; % perte de paquets
                 rtp/jit=20,
                 rtp/delay=40 } ; latence moyenne
    }
}

```

- 21) Lorsque le contrôleur MGC reçoit un signal de raccrochage en provenance d'une des passerelles MG, il libère la communication. Dans cet exemple, l'utilisateur situé du côté de la passerelle MG2 raccroche le premier.

De MG2 à MGC:

```

MEGACO/3 [125.125.125.111]:55555
Transaction = 50008 {
  Context = 5000 {
    Notify = A5555 {ObservedEvents =1235 {
      19990729T24020002:al/on(init=off)}
    }
  }
}

```

De MGC à MG2:

```

MEGACO/3 [123.123.123.4]:55555
Reply = 50008 {
  Context = - {Notify = A5555}
}

```

- 22) Le contrôleur MGC envoie ensuite aux deux passerelles MG une commande Subtract afin de libérer la communication. Seules les commandes de soustraction envoyées à la passerelle MG2 sont représentées ici. Chaque terminaison possède son propre paquetage de statistiques recueillies. Un contrôleur MGC peut ne pas avoir besoin de demander que les deux commandes lui soient renvoyées. La terminaison A5555 est physique, tandis que la terminaison A5556 est en protocole RTP.

De MGC à MG2:

```

MEGACO/3 [123.123.123.4]:55555
Transaction = 50009 {
  Context = 5000 {
    Subtract = A5555 {Audit{Statistics}},
    Subtract = A5556 {Audit{Statistics}}
  }
}

```

De MG2 à MGC:

MEGACO/3 [125.125.125.111]:55555

```
Reply = 50009 {
  Context = 5000 {
    Subtract = A5555 {
      Statistics {
        nt/os=45123, ; octets envoyés
        nt/or=45123, ; octets reçus
        nt/dur=40000 ; en millisecondes
      }
    },
    Subtract = A5556 {
      Statistics {
        rtp/ps=1245, ; paquets envoyés
        nt/os=62345, ; octets envoyés
        rtp/pr=780, ; paquets reçus
        nt/or=45123, ; octets reçus
        rtp/pl=10, ; % de perte de paquets
        rtp/jit=27,
        rtp/delay=48, ; latence moyenne
        nt/dur=38000 ; en millisecondes
      }
    }
  }
}
```

- 23) Le contrôleur MGC configure ensuite les deux passerelles MG1 et MG2 de façon qu'elles soient prêtes à détecter le prochain événement de décrochage. (Voir l'étape 1)). Noter que cet état pourrait être défini comme étant la valeur par défaut d'une terminaison dans le contexte NULL. Dans ce cas, aucun message n'a besoin d'être envoyé du contrôleur MGC à la passerelle MG. Une fois qu'une terminaison revient au contexte NULL, elle reprend ses valeurs par défaut assignées.

## Appendice II

### Gabarit de paquetage H.248

Des paquetages H.248 nouveaux et mis à jour devraient être définis au moyen du gabarit suivant, qui est dans le format d'une Recommandation de l'UIT. Les éditeurs issus d'organisations autres que l'UIT devraient, au minimum, utiliser les structures décrites dans le § 6. Les mots clés des paquetages H.248 sont indiqués en caractères gras. Les sections à remplir sont indiquées par des chevrons "< >". On trouvera au § 12 ci-dessus des informations détaillées sur la façon de créer un nouveau paquetage.

NOTE – Les codes d'erreur spécifiques des paquetages nécessitent un enregistrement auprès de l'autorité IANA. L'attribution d'un code d'erreur doit suivre les règles décrites dans le § 4.1/H.248.8, "*Attribution des codes d'erreur*".

#### Recommandation UIT-T H.248.<xxx>

##### Protocole de commande de passerelle: paquetage <xxx>

#### 1 Domaine d'application

<Le domaine d'application du paquetage>

#### 2 Références normatives

La présente Recommandation renvoie à certaines dispositions des Recommandations UIT-T ou à d'autres textes de référence qui en font ainsi partie intégrante. Les versions indiquées étaient en vigueur au moment de la publication de la présente Recommandation. Les Recommandations ou textes de référence énumérés ci-après sont tous sujets à révision; les utilisateurs de la présente Recommandation sont invités à se reporter, si possible, à leurs versions les plus récentes. La liste des Recommandations UIT-T en vigueur est publiée régulièrement. La référence à un document figurant dans cette Recommandation ne donne pas à ce document, en tant que tel, le statut d'une Recommandation.

- Recommandation UIT-T H.248.1 (<xx/xxxx>), *Protocole de commande de passerelle: Version< x>*
- <Autres références>

#### 3 Termes et définitions

La présente Recommandation utilise les termes et définitions ci-après:

<termes et définitions>

#### 4 Abréviations

La présente Recommandation utilise les abréviations suivantes:

<abréviations>

#### 5 Conventions

<conventions>

<b>6</b>	<b>&lt;Titre du paquetage&gt;</b>	
	<b>Nom du paquetage:</b>	<nom>
	<b>Identificateur de paquetage:</b>	<identificateur du paquetage: "texte", binaire <0x????> >
	<b>Description:</b>	<description>
	<b>Version:</b>	<version>
	<b>Extension:</b>	<extension d'identification du paquetage et version>
<b>6.1</b>	<b>Propriétés</b>	
<b>6.1.1</b>	<b>&lt;Titre de la propriété&gt;</b>	
	<b>Nom de la propriété:</b>	<nom>
	<b>Identificateur de la propriété:</b>	<identificateur textuel, identificateur binaire (0x????)>
	<b>Description:</b>	<description>
	<b>Type:</b>	<type>
	<b>Valeurs possibles:</b>	<valeurs>
	<b>Valeur par défaut:</b>	<valeur>
	<b>Propriété définie dans l'état:</b>	<Local, Remote, LocalControl, TerminationState, ContextAttribut>
	<b>Caractéristiques:</b>	<Lecture seule, lecture/écriture>
<b>6.2</b>	<b>Evénements</b>	
<b>6.2.1</b>	<b>&lt;Titre de l'événement&gt;</b>	
	<b>Nom de l'événement:</b>	<nom d'événement>
	<b>Identificateur d'événement:</b>	<(identificateur textuel), (identificateur binaire (0x????)>
	<b>Description:</b>	<description>
<b>6.2.1.1</b>	<b>Paramètres du descripteur "Events":</b>	
<b>6.2.1.1.1</b>	<b>&lt;Titre du paramètre&gt;</b>	
	<b>Nom du paramètre:</b>	<nom>
	<b>Identificateur du paramètre:</b>	<(identificateur textuel), (identificateur binaire (0x????)>
	<b>Description:</b>	<description>
	<b>Type:</b>	<types>
	<b>Facultatif:</b>	<oui/non>
	<b>Valeurs possibles:</b>	<valeurs>
	<b>Valeur par défaut:</b>	<valeur>
<b>6.2.1.2</b>	<b>Paramètres du descripteur "ObservedEvents":</b>	
<b>6.2.1.2.1</b>	<b>&lt;Titre du paramètre&gt;</b>	
	<b>Nom du paramètre:</b>	<nom>
	<b>Identificateur du paramètre:</b>	<(identificateur textuel), (identificateur binaire (0x????)>
	<b>Description:</b>	<description>

<b>Type:</b>	<types>
<b>Facultatif:</b>	<oui/non>
<b>Valeurs possibles:</b>	<valeurs>
<b>Valeur par défaut:</b>	<valeur>

## 6.3 Signaux

### 6.3.1 <Titre du signal>

<b>Nom du signal:</b>	<nom>
<b>Identificateur du signal:</b>	<(identificateur textuel), (identificateur binaire (0x????))>
<b>Description:</b>	<description>
<b>Type de signal:</b>	<type>
<b>Durée:</b>	<durée>

#### 6.3.1.1 Paramètres additionnels

##### 6.3.1.1.1 <Titre du paramètre>

<b>Nom du paramètre:</b>	<nom>
<b>Identificateur du paramètre:</b>	<(identificateur textuel), (identificateur binaire (0x????))>
<b>Description:</b>	<description>
<b>Type:</b>	<types>
<b>Facultatif:</b>	<oui/non>
<b>Valeurs possibles:</b>	<valeurs>
<b>Valeur par défaut:</b>	<valeur>

## 6.4 Statistiques

### 6.4.1 <Titre de la statistique>

<b>Nom de la statistique:</b>	<nom>
<b>Statistic ID:</b>	<(identificateur textuel), (identificateur binaire (0x????))>
<b>Description:</b>	<description>
<b>Type:</b>	<type>
<b>Valeurs possibles:</b>	<valeurs>
<b>Niveau:</b>	<terminaison, flux, l'un ou l'autre>

## 6.5 Codes d'erreur

### 6.5.1 <Titre du code d'erreur>

<b>Numéro du code d'erreur:</b>	<nombre>
<b>Nom:</b>	<nom>
<b>Définition:</b>	<définition>
<b>Texte de l'erreur dans le descripteur "Error":</b>	<texte de l'erreur à renvoyer>
<b>Commentaire:</b>	<commentaire>

## 6.6 Procédures

<Les procédures associées au paquetage>

### Appendice III

#### Gabarit de définition de profil H.248

De nouveaux profils H.248 devraient être définis au moyen du gabarit de profil ci-dessous, qui est dans le format d'une Recommandation de l'UIT. Les éditeurs issus d'organisations autres que l'UIT devraient, au minimum, utiliser les structures décrites dans le § 5 ci-dessous. Si ce format de gabarit n'est pas utilisé, les éditeurs de profils devraient alors vérifier que les titres et rubriques décrits dans le présent appendice sont repris par leur profil.

Les titres énumérés dans la structure ci-dessous représentent des éléments qui peuvent être considérés comme facultatifs. Dans les définitions de profil, un certain élément H.248.1 peut être inutilisé bien qu'il soit obligatoire dans la présente Recommandation. Ces éléments sont également inclus dans la structure ci-dessous. Les éléments non énumérés doivent être considérés comme obligatoires par le protocole de la présente Recommandation. Dans le gabarit de profil ci-dessous, des éléments peuvent être définis comme étant "facultatifs" ou "obligatoires". Le terme "facultatif" signifie que l'expéditeur ou le récepteur a le choix d'inclure l'élément dans un message. Si l'entité réceptrice reçoit un élément facultatif qu'elle n'a pas implémenté conformément au § 6.2.3 ci-dessus, cette entité devrait envoyer le code d'erreur 501 ("Non implémenté").

L'éditeur devrait fournir, dans chacun des paragraphes ci-dessous, une description écrite indiquant s'il développe plus précisément le comportement H.248.1. Par exemple, si la commande "Move" (§ 6.8.4) est limitée à certains types de terminaison, cela devrait être indiqué.

Le texte en caractères italiques est à supprimer.

Le <texte> entre chevrons doit être complété.

#### Recommandation UIT-T <xxx>

#### <Titre du profil>

##### 1 **Domaine d'application**

<Le domaine d'application du profil>

##### 2 **Références**

<Références>

##### 3 **Termes et définitions**

La présente Recommandation utilise les termes et définitions ci-après:

<termes et définitions>

## 4 Abréviations

La présente Recommandation utilise les abréviations suivantes:

<abréviations>

## 5 Conventions

<conventions>

## 6 Description du profil

### 6.1 Identification du profil

Nom du profil:	<nom de 1 à 64 caractères>
Version:	<version 1 à 99>

*Le nom et la version du profil qui sont envoyés dans la commande de changement de service.*

### 6.2 Résumé

<Description>

*Description de la nature du profil.*

### 6.3 Version du protocole de commande de passerelle

<Numéro de version>

*Numéro minimal de la version H.248 requise afin de prendre en charge le profil. Ce numéro devrait être fondé sur la prise en charge de la syntaxe de base et non pas sur une assignation arbitraire de version. Ce point est en rapport avec le paramètre "ServiceChangeVersion" décrit dans le § 6.8.8.*

### 6.4 Modèle de connexion

<à décrire en mots et diagrammes>

*Description des configurations de terminaison autorisées dans un contexte.*

<b>Nombre maximal de contextes:</b>	<Entier>
<b>Nombre maximal de terminaisons pour chaque contexte:</b>	<Entier>
<b>Combinaisons autorisées de types de terminaison dans un contexte:</b>	<par exemple Contexte[a](IP,TDM), Contexte[b](TDM T,AAL 2), etc.>

### 6.5 Attributs de contexte

Attribut de contexte	Pris en charge	Valeurs prises en charge
<b>Topologie</b>	<Oui/non>	Voir au § 6.7.8
<b>Indicateur de priorité</b>	<Oui/non>	<1 à 15>
<b>Indicateur d'urgence</b>	<Oui/non>	NA
<b>Indicateur de plan IEPS</b>	<Oui/non>	NA
<b>Descripteur "ContextAttribute"</b>	<Oui/non>	Si "Oui", voir au § 6.8.9 de plus amples détails sur les attributs pris en charge.
<b>Paramètre "ContextIDList"</b>	<Oui/non>	NA

*Est-ce que l'attribut de contexte d'opérateur de choix ET/OU est pris en charge?*

<b>Attribut de contexte ET/OU</b>	<Oui/non>	<ET/OUX/OU>
-----------------------------------	-----------	-------------

## 6.6 Terminaisons

### 6.6.1 Noms de terminaison

<La structure de l'identificateur de terminaison>

*Identifier les identités de terminaison associées aux terminaisons physiques, éphémères et virtuelles.*

### 6.6.2 Terminaisons multiplexées

<b>Terminaisons multiplexées prises en charge?</b>	<Oui/non>
--	-----------

*Si oui:*

<b>Types de multiplexage pris en charge</b>	< H.221, H.223, H.226, V.76, N × 64K>
<b>Nombre maximal de terminaisons connectées à un multiplex</b>	<Entier>

*Des terminaisons multiplexées sont-elles utilisées? Si oui, décrire.*

## 6.7 Descripteurs

### 6.7.1 Descripteur "Stream"

<b>Nombre maximal de flux par type de terminaison</b>	<TerminationType>	<Entier>
---	-------------------	----------

*Si plus de 1:*

<b>Configuration du flux:</b>	<Décrire les configurations autorisées. Est-ce que plusieurs flux audio sont autorisés? Etc.>
-------------------------------	---

#### 6.7.1.1 Descripteur "LocalControl"

*Est-ce que les propriétés ReserveGroup et ReserveValue sont utilisées?*

<i>En l'absence de structure générique, énumérer les types appropriés de terminaison et de flux</i>		<b>Type de terminaison</b>	<b>Type de flux</b>
<b>Groupe de réserve utilisé:</b>	<Oui/non>	<Type>	<Type>
<b>Valeur de réserve utilisée:</b>	<Oui/non>	<Type>	<Type>

*Quelles valeurs de mode de flux (StreamMode) sont utilisées?*

<b>Type de terminaison</b>	<b>Type de flux</b>	<b>Valeurs autorisées de mode de flux</b>
<Type>	<Type>	<SendOnly, RecvOnly, SendRecv, Loopback>

## 6.7.2 Descripteur "Events"

Tous les événements contenus dans ce profil peuvent être choisis dans toute terminaison ou dans tout flux [avec les exceptions suivantes].

NOTE – Le texte entre crochets [] est facultatif et seulement inclus s'il y a de telles exceptions.

<b>Événements réglables selon les types de terminaison et de flux:</b>	<Oui/non>		
<i>Si oui</i>	<b>Identificateur d'événement</b>	<b>Type de terminaison</b>	<b>Type de flux</b>
	<Nom et identité d'événement, par exemple événement d'erreur générique (g/cause, 0x0001/0x0001)>	<Type>	<Type de flux, par exemple audio/vidéo/données ou identificateur de flux>

*Un tampon d'événements est-il utilisé?*

<b>Commande "EventBuffer" utilisée:</b>	<Oui/non>
---	-----------

*La commande "KeepActive" est-elle utilisée pour des événements?*

<b>Commande KeepActive utilisée pour des événements:</b>	<Oui/non>
--	-----------

*L'imbrication dans les événements est-elle utilisée?*

<b>Événements imbriqués dans un descripteur "Events":</b>	<Oui/non>
<b>Signaux imbriqués dans un descripteur "Events":</b>	<Oui/non>

*La régulation des événements imbriqués est-elle prise en charge?*

<b>La régulation des événements imbriqués est déclenchée lors:</b>	<Aucune régulation/spécifier un événement particulier>
--	--

*Le fanion "ResetEventsDescriptor" est-il utilisé?*

<b>Fanion "ResetEventsDescriptor" utilisé avec événements:</b>	<Tous les événements/aucun/spécifier un événement particulier>
--	--

*Quel comportement de notification est pris en charge?*

<b>NotifyImmediate:</b>	<Tous les événements (ALL)/aucun/spécifier un événement particulier>
<b>NotifyRegulated:</b>	<Tous les événements (ALL)/aucun/spécifier un événement particulier>
<b>NeverNotify:</b>	<Tous les événements (ALL)/aucun/spécifier un événement particulier>

### 6.7.3 Descripteur "EventBuffer"

*Ce descripteur est-il pris en charge?*

<b>Descripteur "EventBuffer" utilisé:</b>	<Oui/non>	
<i>Si oui</i>	<b>Identificateur d'événements</b>	<Nom et identité d'événement, par exemple événement d'erreur générique (g/cause, 0x0001/0x0001) ou ALL (tous les événements)>

### 6.7.4 Descripteur "Signals"

*Tous les signaux contenus dans ce profil peuvent être choisis dans toute terminaison/tout flux [avec les exceptions suivantes].*

*NOTE – Le texte entre crochets [] est facultatif et seulement inclus s'il y a de telles exceptions.*

<b>Le réglage de signaux dépend des types de terminaison ou de flux:</b>	<Oui/non> NOTE – "Non" signifie que tous les signaux peuvent être restitués dans une terminaison ou dans un flux quelconque. Si "Oui", tout signal non énuméré ci-dessous peut être restitué dans une terminaison ou dans un flux quelconque.		
<i>Si oui</i>	<b>Identificateur du signal</b>	<b>Type de terminaison</b>	<b>Type de flux / ID</b>
	<Nom et identité du signal, par exemple tonalité restituable (tonegen/pt, 0x0003/0x0001)>	<Type>	<Type de flux, par exemple audio/vidéo/données ou identificateur de flux>

*Les listes de signaux sont-elles prises en charge? Si oui, quel est le nombre maximal de signaux par liste et par type de terminaison/flux prenant en charge des listes?*

<b>Listes de signaux prises en charge:</b>	<Oui/non>	
<i>Si oui</i>	<b>Type de terminaison prenant en charge des listes:</b>	<Type/ALL>
	<b>Type de flux prenant en charge des listes:</b>	<Type/ALL>
	<b>Nombre maximal de signaux dans une liste de signaux:</b>	<Entier>
	<b>Paramètre de retard entre signaux pris en charge:</b>	<Oui/non>

*Est-ce que la priorité sur le type et la durée d'un signal est prise en charge?*

<b>Type de signal et durée pris en charge:</b>	<Oui/non>	
<i>Si oui</i>	<b>Identificateur du signal</b>	<b>Priorité sur type ou durée</b>
	<Nom et identité du signal, par exemple tonalité restituable (tonegen/pt, 0x0003/0x0001) ou "ALL">	<Type, durée, les deux>

*Le sens du signal est-il pris en charge?*

<b>Sens du signal pris en charge:</b>	<Oui/non>
---------------------------------------	-----------

*Est-ce que l'achèvement de notification "notifyCompletion" est pris en charge? Quels types sont pris en charge? Est-ce que l'identificateur "RequestID" est utilisé avec "NotifyCompletion"?*

<b>Achèvement de notification pris en charge:</b>	<Oui/non>	
<i>Si oui</i>	<b>Identificateur du signal</b>	<b>Type d'achèvement pris en charge</b>
	<Nom et identité du signal, par exemple tonalité restituable (tonegen/pt, 0x0003/0x0001) ou ALL>	<ALL, TO, EV, ED, NC, PI>
<b>Paramètre "RequestID" pris en charge:</b>	<Oui/non>	

*Est-ce que des signaux multiples peuvent être restitués simultanément?*

<b>Signaux restitués simultanément:</b>	<Oui/non>	
<i>Si oui</i>	<b>Identificateurs de signaux qui peuvent être restitués simultanément:</b>	<Nom et identité du signal, par exemple tonalité restituable (tonegen/pt, 0x0003/0x0001) ou ALL>

*La commande "KeepActive" est-elle prise en charge pour les signaux?*

<b>Commande KeepActive utilisée pour les signaux:</b>	<Oui/non>
---	-----------

### 6.7.5 Descriptor "DigitMap"

*Les scripts de numérotation sont-ils pris en charge? Si oui, décrire les noms, structures et temporisateurs.*

<b>Scripts de numérotation pris en charge:</b>	<Oui/non>		
<i>Si oui</i>	<b>Nom du script</b>	<b>Structure</b>	<b>Temporisateurs</b>
	<nom>	<Décrire>	<temporisateurs>

### 6.7.6 Descripteur "Statistics"

*Est-ce que les statistiques sont prises en charge dans les terminaisons, les flux ou les deux?*

<b>Statistiques prises en charge dans:</b>	<Terminaison/flux/les deux>
--	-----------------------------

*Les statistiques sont-elles signalées?*

<b>Statistiques signalées par commande Subtract:</b>	<Oui/non>	
<i>Si oui</i>	Identificateurs de statistique signalés:	<Nom et identité de la statistique, par exemple paquets envoyés (rtp/ps, 0x00c/0x0004) ou ALL>

### 6.7.7 Descripteur "ObservedEvents"

*L'instant de détection est-il pris en charge?*

<b>Instant de détection d'événement pris en charge:</b>	<Oui/non>
---	-----------

### 6.7.8 Descripteur "Topology"

*Si ce descripteur est utilisé, quels sont les réglages autorisés?*

<b>Triplets autorisés:</b>	<(T1, T2, oneway) etc.>
----------------------------	-------------------------

### 6.7.9 Descripteur "Error"

*Quels codes d'erreur définis dans la Rec. UIT-T H.248.8 et dans des paquetages sont pris en charge?*

**Codes d'erreur envoyés par le contrôleur MGC:**

<b>Codes d'erreur H.248.8 pris en charge:</b>	<ALL (tous ceux de la Rec.) H.248.8, liste de nombres individuels>
<b>Codes d'erreur définis dans des paquetages pris en charge:</b>	Pour une liste des codes d'erreur, voir le § 6.14.x <Référence au paragraphe approprié dans le § 6.14 ci-dessous>

**Codes d'erreur envoyés par la passerelle MG:**

<b>Codes d'erreur H.248.8 pris en charge:</b>	<ALL (tous ceux de la Rec. ) H.248.8, liste de nombres individuels>
<b>Codes d'erreur définis dans des paquetages pris en charge:</b>	Pour une liste des codes d'erreur, voir le § 6.14.x <Référence au paragraphe approprié dans le § 6.14 ci-dessous>

## 6.8 Interface API de commande

*NOTE – Il est supposé qu'un descripteur "Error" peut être renvoyé dans toute commande "Reply".*

### 6.8.1 Add

*Quels descripteurs peuvent être utilisés dans une requête Add?*

<b>Descripteurs utilisés par une requête Add:</b>	<Media, Mux, Events, EventBuffer, Signals, DigitMap, Audit>
---	---

*Quels descripteurs peuvent être utilisés dans une réponse Add?*

<b>Descripteurs utilisés par une réponse Add:</b>	<Media, Mux, Events, EventBuffer, Signals, DigitMap, Audit, Statistics>
---	---

### 6.8.2 Modify

*Quels descripteurs peuvent être utilisés dans une requête Modify?*

<b>Descripteurs utilisés par une requête Modify:</b>	<Media, Mux, Events, EventBuffer, Signals, DigitMap, Audit>
--	---

*Quels descripteurs peuvent être utilisés dans une réponse Modify?*

<b>Descripteurs utilisés par une réponse Modify:</b>	<Media, Mux, Events, EventBuffer, Signals, DigitMap, Audit>
--	---

### 6.8.3 Subtract

*Un descripteur "Audit" peut-il être utilisé dans une requête Subtract?*

<b>Descripteurs utilisés par une requête Subtract:</b>	<Audit>
--	---------

Un descripteur "Statistics" peut-il être utilisé dans une réponse Subtract?

<b>Descripteurs utilisés par une réponse Subtract:</b>	<Statistics>
--	--------------

#### 6.8.4 Move

La commande Move est-elle utilisée? Certaines configurations de contexte ne peuvent pas l'utiliser.

<b>Commande Move utilisée:</b>	<Oui/non>
--------------------------------	-----------

Si utilisée:

<b>Descripteurs utilisés par une requête Move:</b>	<Media, Mux, Events, EventBuffer, Signals, DigitMap, Audit, Statistics>
<b>Descripteurs utilisés par une réponse Move:</b>	<Media, Mux, Events, EventBuffer, Signals, DigitMap, Audit, Statistics>

#### 6.8.5 AuditValue

Quels descripteurs et/ou propriétés, signaux, événements ou statistiques peuvent être examinés individuellement?

<b>Propriétés examinées:</b>	<Nom et identité de la propriété, par exemple nombre maximal de contextes (Root/maxNumberOfContexts, 0x0002/0x0001), ALL ou aucun contexte>	<Descripteur: Local, Remote, LocalControl, TerminationState>
<b>Statistiques examinées:</b>	<Nom et identité de la statistique, par exemple paquets envoyés (rtp/ps, 0x00c/0x0004), ALL ou aucune>	
<b>Signaux examinés:</b>	<Nom et identité du signal, par exemple tonalité restituable (tonegen/pt, 0x0003/0x0001), ALL ou aucune>	
<b>Événements examinés:</b>	<Nom et identité d'événement, par exemple événement d'erreur générique (g/cause, 0x0001/0x0001), ALL ou aucun événement>	
<b>Examen de paquetage possible:</b>	Le descripteur de paquetage peut-il être examiné? <Oui/non>	

#### 6.8.6 AuditCapabilities

Quels descripteurs et/ou propriétés, signaux, événements ou statistiques peuvent être examinés individuellement?

<b>Propriétés examinées:</b>	<Nom et identité de la propriété, par exemple nombre maximal de contextes (Root/maxNumberOfContexts, 0x0002/0x0001), ALL ou aucun contexte>	<Descripteur: Local, Remote, LocalControl, TerminationState>
<b>Statistiques examinées:</b>	<Nom et identité de la statistique, par exemple paquets envoyés (rtp/ps, 0x00c/0x0004), ALL ou aucun paquet>	
<b>Signaux examinés:</b>	<Nom et identité du signal, par exemple tonalité restituable (tonegen/pt, 0x0003/0x0001), ALL ou aucune tonalité>	
<b>Événements examinés:</b>	<Nom et identité d'événement, par exemple événement d'erreur générique (g/cause, 0x0001/0x0001), ALL ou aucun événement>	

L'examen par détection est-il possible?

<b>Propriétés examinées/Attributs de contexte utilisés pour un examen par détection:</b>	<Aucune/ALL/à spécifier individuellement>
--	---

## 6.8.7 Notify

Quels descripteurs peuvent être utilisés dans une commande Notify?

<b>Descripteurs utilisés par une requête/réponse Notify:</b>	<ObservedEvents, Error>
--	-------------------------

## 6.8.8 ServiceChange

Quelles méthodes et raisons de changement de service sont prises en charge?

**Méthodes de changement de service et codes de raison du changement de service envoyés par le contrôleur MGC:**

Méthodes de changement de service prises en charge	Raisons de changement de service prises en charge
< Graceful, Forced, Restart, Handoff, Failover, ALL, Autre?>	< 900 à 920 >

**Méthodes de changement de service et codes de raison du changement de service envoyés par la passerelle MG:**

Méthodes de changement de service pris en charge	Raisons de changement de service prises en charge
< Graceful, Forced, Restart, Disconnected, Handoff, Failover, ALL, autre?>	< 900 à 920 >

Le paramètre "ServiceChangeAddress" est-il utilisé?

<b>Paramètre ServiceChangeAddress utilisé:</b>	<Oui/non>
--	-----------

Le paramètre "ServiceChangeDelay" (délai de changement de service) est-il utilisé?

<b>Paramètre ServiceChangeDelay utilisé:</b>	<Oui/non>
Si oui	<b>Période valide:</b> <0 à x> ms

Le fanion "ServiceChangeIncomplete" est-il utilisé?

<b>Fanion ServiceChangeIncomplete utilisé:</b>	<Oui/non>
--	-----------

Quelle version de la présente Rec. UIT-T H.248.1 est utilisée par la commande ServiceChangeVersion? La plus faible valeur indiquée ici devrait être la version minimale qui est définie dans le § 6.3.

<b>Version utilisée dans ServiceChangeVersion:</b>	<1, 2, 3>
--	-----------

Des profils multiples peuvent-ils être pris en charge conformément à la Rec. UIT-T H.248.18?

<b>Négociation de profil conformément à la Rec. UIT-T H.248.18:</b>	<Oui/non>
---	-----------

## 6.8.9 Manipulation et examen des attributs de contexte

Quels attributs de contexte peuvent être manipulés et/ou examinés?

<b>Attributs de contexte manipulés:</b>	<topologie, urgence, priorité, indicateur de plan IEPS, descripteur "ContextAttribute" (énumération des noms d'attribut), ALL>
<b>Attributs de contexte examinés:</b>	<topologie, urgence, priorité, indicateur de plan IEPS, descripteur "ContextAttribute" (énumération des noms d'attribut), ALL>

## 6.9 Syntaxe et codage de commande générique

Spécifier les codages qui sont pris en charge par le profil.

<b>Codages pris en charge:</b>	<Texte et binaire, binaire, texte>
--------------------------------	------------------------------------

## 6.10 Transactions

Quel est le nombre maximal de requêtes/réponses de transaction par message?

<b>Nombre maximal de requêtes/réponses de transaction/Acquittements de réponse de transaction/Réponses de segmentation par message:</b>	<Entier>
---	----------

Quel est le nombre maximal de commandes par requête de transaction?

<b>Nombre maximal de commandes par requête de transaction:</b>	<Entier>
--	----------

Quel est le nombre maximal de commandes par réponse de transaction?

<b>Nombre maximal de commandes par réponse TransactionReply:</b>	<Entier>
--	----------

Certaines commandes peuvent-elles être marquées "Facultatives"? Décrire.

<b>Commandes pouvant être marquées "Facultatives":</b>	<Add, Modify, Move, Subtract, AuditValue, AuditCapability, ServiceChange, toutes les commandes, aucune>
--	---

Spécifier les valeurs des temporisateurs de transaction

<b>Temporisateur de transaction:</b>	<b>Valeur</b>
<b>normalMGExecutionTime</b>	<Entier ou "Fournie">
<b>normalMGCExecutionTime</b>	<Entier ou "Fournie">
<b>MGOriinatedPendingLimit</b>	<Entier ou "Fournie">
<b>MGCOriinatedPendingLimit</b>	<Entier ou "Fournie">
<b>MGProvisionalResponseTimerValue</b>	<Entier ou "Fournie">
<b>MGCProvisionalResponseTimerValue</b>	<Entier ou "Fournie">

## 6.11 Messages

Conventions de nommage MGC/MG: identificateur MID associé aux noms de MGC/MG.

<Décrire>

Indiquer le nombre maximal de transactions par message (ce tableau peut être omis s'il n'est pas significatif)

<b>Nombre maximal de transactions par message:</b>	<Entier >
--	-----------

## 6.12 Transport

Spécifie quels transports de sous-série H.248 sont pris en charge par le profil.

<b>Transports pris en charge:</b>	<UDP, TCP, SCTP, MTP3B, SSCOP/AAL 5, ALF/AAL 5>
-----------------------------------	---

Est-ce que la segmentation est prise en charge et si oui, par quelle méthode?

<b>Segmentation prise en charge:</b>	<Non, propre au transport, segmentation H.248>
--------------------------------------	--

Est-ce que la surveillance d'association de commande (§ 11.6 ci-dessus) est utilisée et si oui, par quelle méthode?

<b>Surveillance d'association de commande prise en charge:</b>	<Non, propre au transport, valeur AuditValue vide dans la terminaison racine, H.248.14>
--	---

### 6.13 Sécurité

Spécifie les mécanismes de sécurité utilisés.

<b>Sécurité prise en charge:</b>	<Aucun mécanisme, système AH intérimaire, IPSec>
----------------------------------	--

### 6.14 Paquetages

Spécifie les paquetages qui sont pris en charge dans ce profil.

Obligatoire: spécifie les paquetages qui doivent être pris en charge dans ce profil.

Paquetages obligatoires:		
Nom du paquetage	Identificateur du paquetage	Version
<nom>	<xxxx, (0x00xx)>	<1, 2, 3, ...>

Facultatif: spécifie les paquetages qui peuvent être pris en charge dans le profil.

Paquetages facultatifs:			
Nom du paquetage	Identificateur du paquetage	Version	Prise en charge dépendant de:
<nom>	<xxx, 0x00??>	<1, 2, 3, ...>	<Décrire>

#### Informations sur l'usage du paquetage

Ce tableau spécifie comment les paquetages ci-dessus seront utilisés. Par exemple:

- il énumère les propriétés/signaux/événements/statistiques selon qu'ils sont facultatifs ou obligatoires;
- il indique si la valeur de la propriété/du signal/de l'événement qui est fournie devrait être spécifiée (par exemple noms et nombre de cycles pour une annonce H.248.7).

Spécifie les valeurs de propriété qui sont spécifiées comme étant fournies.

#### Informations sur l'usage du paquetage

##### 6.14.x <Nom du paquetage>

Propriétés	Obligatoire/ Facultatif	Utilisé dans la commande:	Valeurs prises en charge:	Valeur fournie:
<nom et identité, par exemple paquets envoyés (rtp/ps, 0x000c/0x0004), ALL ou aucune>	<M/O>	<ADD, MOD, MOVE, AUDITVALUE, AUDITCAP>	<Valeurs/ALL>	<Valeur/non applicable>

Signaux	Obligatoire/ Facultatif	Paquetage utilisé dans la commande:		Valeur de durée fournie:
<nom et identité >	<M/O>	<ADD, MOD, MOVE, AUDITVALUE, AUDITCAP>		<Valeur/non applicable>
	Paramètres de signal	Obligatoire/ Facultatif	Valeurs prises en charge:	Valeur de durée fournie:
	<nom et identité>	<M/O>	<Valeurs/ALL>	<Valeur/non applicable>
Evénements	Obligatoire/ Facultatif	Paquetage utilisé dans la commande		
<nom et identité >	<M/O>	<ADD, MOD, MOVE, NOTIFY, AUDITVALUE, AUDITCAP>		
	Paramètres "Events"	Obligatoire/ Facultatif	Valeurs prises en charge:	Valeur fournie:
	<nom et identité>	<M/O>	<Valeurs/ALL>	<Valeur/non applicable>
	Paramètres "ObservedEvent"	Obligatoire/ Facultatif	Valeurs prises en charge:	Valeur fournie:
	<nom et identité>	<M/O>	<Valeurs/ALL>	<Valeur/non applicable>
Statistiques	Obligatoire/ Facultatif	Paquetage utilisé dans la commande:	Valeurs prises en charge:	
<nom et identité >	<M/O>	<ADD, MOD, MOVE, SUBTRACT, AUDITVALUE, AUDITCAP>	<Valeurs/ALL>	
Codes d'erreur	Obligatoire/Facultatif			
<nombre>	<M/O>			

*Des restrictions supplémentaires peuvent être inscrites dans ce tableau selon les souhaits de l'utilisateur.*

### 6.15 Prise en charge obligatoire du protocole SDP et des éléments d'information de l'Annexe C

*Spécifie quels attributs SDP et éléments d'information de l'Annexe C sont à prendre en charge.*

Eléments d'information de l'Annexe C et attributs SDP pris en charge:		
Elément d'information	Prise en charge de l'Annexe C	Prise en charge du protocole SDP
<nom>	<Propriété selon l'Annexe C>	<Décrire>

### 6.16 Prise en charge facultative des éléments d'information SDP et de l'Annexe C

*Spécifie quels attributs SDP et éléments d'information de l'Annexe C peuvent être pris en charge.*

Eléments d'information facultatifs de l'Annexe C et du protocole SDP:			
Elément d'information	Prise en charge de l'Annexe C	Prise en charge du protocole SDP	Prise en charge dépendant de:
<nom>	<Propriété selon l'Annexe C>	<Décrire>	<Décrire>

## 6.17 Procédures

*Spécifie les procédures qui sont associées au profil.*

*Il est recommandé que les procédures adoptent le format suivant:*

### 6.17.1 <Nom de la procédure>

*Lorsque le <nom> de la procédure est requis, ce qui suit est lancé.*

Une commande <ADD.req, MOD.req, MOV.req, SUB.req, AuditValue.req, AuditCapability.req, ServiceChange.req, Notify.req> est envoyée par <MGC/MG> vers <MG/MGC> avec les informations suivantes.

<Insérer les informations dans un format approprié. Par exemple descripteurs, propriété, signal, événement, nom de la statistique ou une abstraction des paramètres réels en termes d'éléments d'information.
---

Dès réception de la commande, le <MG/MGC> doit:

- <Indiquer les actions>

Dès l'achèvement de la commande de traitement (1) une commande <ADD.reply, MOD.reply, MOV.reply, SUB.reply, AuditValue.reply, AuditCapability.reply, ServiceChange.reply, Notify.reply> est envoyée de <MG/MGC> à <MGC/MG> avec les informations suivantes.

< Insérer les informations dans un format approprié. Par exemple descripteurs, propriété, signal, événement, nom de la statistique ou une abstraction des paramètres réels en termes d'éléments d'information.
--



## SÉRIES DES RECOMMANDATIONS UIT-T

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