Union internationale des télécommunications

UIT-T J.171.1

(11/2005)

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

SÉRIE J: RÉSEAUX CÂBLÉS ET TRANSMISSION DES SIGNAUX RADIOPHONIQUES, TÉLÉVISUELS ET AUTRES SIGNAUX MULTIMÉDIAS

IPCablecom

Protocole de commande de passerelle de jonction (TGCP) du système IPCablecom: profil 1

Recommandation UIT-T J.171.1

Recommandation UIT-T J.171.1

Protocole de commande de passerelle de jonction (TGCP) du système IPCablecom: profil 1

Résumé

La présente Recommandation décrit le profil IPCablecom d'une interface de programmation d'applications (API, *application programming interface*) appelée *interface de contrôle de passerelle média* (MGCI, *media gateway control interface*), ainsi que le protocole de contrôle de passerelle média (MGCP, *media gateway control protocol*) correspondant, afin de commander des passerelles de téléphonie utilisant le protocole Internet (VoIP, *voice-over-IP*) par réseau téléphonique commuté (RTC) à partir d'éléments extérieurs de commande d'appel. Il s'agit là de l'un des deux profils qui sont cités dans la Rec. UIT-T J.171.0. Le second profil est spécifié dans la Rec. UIT-T J.171.2.

Le protocole MGCP suppose une architecture de commande d'appel dans laquelle "l'intelligence" de la commande d'appel se trouve à l'extérieur des passerelles et est gérée par des éléments extérieurs de commande d'appel. Le profil IPCablecom qui est décrit dans la présente Recommandation est désigné par l'expression de *protocole de commande de passerelle de jonction* (TGCP, *trunking gateway control protocol*) IPCablecom.

Source

La Recommandation UIT-T J.171.1 a été approuvée le 29 novembre 2005 par la Commission d'études 9 (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 n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

TABLE DES MATIÈRES

Recommandation UIT-T J.171.1

Protocole de commande de passerelle de jonction (TGCP) du système IPCablecom: profil 1

1 Domaine d'application

La présente Recommandation décrit le profil IPCablecom d'une interface de programmation d'applications (API, *application programming interface*) appelée *interface de contrôle de passerelle média* (MGCI, *media gateway control interface*), ainsi que le protocole de contrôle de passerelle média (MGCP, *media gateway control protocol*) correspondant, afin de commander des passerelles de téléphonie utilisant le protocole Internet (VoIP, *voice-over-IP*) par réseau téléphonique commuté (RTC) à partir d'éléments extérieurs de commande d'appel. Il s'agit là de l'un des deux profils qui sont cités dans la Rec. UIT-T J.171.0. Le second profil est spécifié dans la Rec. UIT-T J.171.2.

Le protocole MGCP suppose une architecture de commande d'appel dans laquelle "l'intelligence" de la commande d'appel se trouve à l'extérieur des passerelles et est gérée par des éléments extérieurs de commande d'appel. Le profil IPCablecom qui est décrit dans la présente Recommandation est désigné par l'expression de *protocole de commande de passerelle de jonction* (TGCP, *trunking gateway control protocol*) IPCablecom.

La présente Recommandation est fondée sur la Rec. UIT-T J.162 concernant la signalisation d'appel fournie par le réseau IPCablecom ainsi que sur le document RFC 2705 du groupe IETF: *Media Gateway Control Protocol (MGCP)* (Protocole de contrôle de passerelle média). La présente Recommandation, qui définit le protocole TGCP IPCablecom, constitue une spécification qui est indépendante du protocole MGCP. Le profil TGCP du protocole MGCP est défini strictement et uniquement par le contenu de la présente Recommandation.

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 J.161 (Projet, version 2), *Caractéristiques des codecs audio pour la fourniture de services téléphoniques dans les réseaux de télévision par câble utilisant des câblo-modems*.

– Recommandation UIT-T J.162 (2005), *Protocole réseau de signalisation d'appel pour la fourniture de services à temps critique sur les réseaux de télévision par câble utilisant des câblo-modems*.

– Recommandation UIT-T J.170 (2005), *Spécification de la sécurité sur IPCablecom*.

– IETF RFC 2327 (1998), *SDP: Session Description Protocol* (Protocole de description de session).

2.2 Références informatives

– Recommandation UIT-T E.180/Q.35 (1998), *Caractéristiques techniques des tonalités du service téléphonique*.

– Recommandation UIT-T J.163 (2005), *Qualité de service dynamique pour la fourniture de services en temps réel sur les réseaux de télévision par câble utilisant des câblo-modems*.

– Recommandation UIT-T J.171.0 (2005), *Protocole de commande de passerelle de jonction (TGCP) du système IPCablecom – Aperçu général des profils.*

– IETF RFC 1889 (1996) *RTP: A Transport Protocol for Real-Time Applications*.

– IETF RFC 1890 (1996), *RTP Profile for Audio and Video Conferences with Minimal Control*.

– IETF RFC 2543 (1999), *SIP*: *Session Initiation Protocol*.

– IETF RFC 2705 (1999), *Media Gateway Control Protocol (MGCP) Version 1.0*.

– TCP/IP Illustrated, Volume 1 (2001), *The Protocols,* Addison-Wesley, 1994.

3 Définitions

La présente Recommandation définit les termes suivants:

3.1 modem-câble; cablo-modem: dispositif conforme aux Recommandations UIT-T J.83 et J.112 acheminant à grande vitesse des données d'accès à des sites de clients.

3.2 IPCablecom: projet de l'UIT-T comprenant une architecture et une série de Recommandations permettant de fournir des services en temps réel dans les réseaux de télévision par câble utilisant des câblo-modems.

4 Abréviations et acronymes

La présente Recommandation utilise les abréviations et acronymes suivants:

DNS système de dénomination de domaine (*domain name system*)

IP protocole Internet (*Internet protocol*)

IPSec sécurité IP (*Internet protocol security*)

ISUP sous-système utilisateur RNIS (*ISDN user part*)

MGC contrôleur de passerelle média (*media gateway controller*)

MGCP protocole de contrôle de passerelle média (*media gateway control protocol*)

MIB base d'informations de gestion (*management information base*)

MTA adaptateur de terminal média (*media terminal adapter*)

MWD temps d'attente maximal (*maximum waiting delay*)

NCS signalisation d'appel fondée sur le réseau (*network-based call signalling*)

NTP protocole relatif au temps dans le réseau (*network time protocol*)

QS qualité de service

RTCP protocole de commande en temps réel (*real-time control protocol*)

RTO temporisation de retransmission (*retransmission timeout*)

RTP protocole en temps réel (*real-time protocol*)

SDP protocole de description de session (*session description protocol*)

SG passerelle de signalisation (*signalling gateway*)

SPI indice de paramètre de sécurité (*security parameters index*)

5 Conventions

Si la présente Recommandation est implémentée, les mots clés "DOIT" (MUST ou SHALL, en anglais) et "REQUIS" doivent être interprétés comme indiquant un aspect obligatoire de la présente Recommandation.

Les mots clés indiquant un certain niveau d'importance de telle ou telle prescription utilisée dans la présente Recommandation sont résumés ci-dessous.

"DOIT"	Ce mot ainsi que l'adjectif "REQUIS" indiquent que l'article est une prescription absolue de la présente Recommandation.
"NE DOIT PAS"	Cette expression indique que l'article est une interdiction absolue de la présente Recommandation.
"DEVRAIT"	Cette forme verbale ainsi que l'adjectif "RECOMMANDÉ" indiquent qu'il peut, dans des circonstances particulières, exister des raisons valables pour ignorer cet article, mais qu'il convient, avant de faire ce choix, de prendre en considération la totalité des incidences et d'étudier soigneusement le cas.
"NE DEVRAIT PAS"	Cette expression indique qu'il peut, dans des circonstances particulières, exister des raisons valables pour que le comportement indiqué soit acceptable ou même utile, mais qu'il convient, avant de faire ce choix, de prendre en considération la totalité des incidences et d'étudier soigneusement le cas.
"PEUT"	Ce verbe ainsi que l'adjectif "FACULTATIF" indiquent que cet article est effectivement facultatif. Un fournisseur peut choisir d'inclure l'article p. ex. parce qu'il est requis sur un marché particulier ou parce qu'il améliore le produit, alors qu'un autre fournisseur peut choisir d'omettre ce même article.

5.1 Travaux antérieurs

La présente Recommandation décrit le profil IPCablecom d'une interface de programmation d'applications appelée *interface de commande de passerelle média* (MGCI), ainsi que le protocole de commande de passerelle média (MGCP) correspondant, afin de commander des passerelles de voix sur IP (VoIP) par réseau téléphonique commuté (RTC) à partir d'éléments extérieurs de commande d'appel. Le protocole MGCP suppose une architecture de commande d'appel dans laquelle "l'intelligence" de la commande d'appel se trouve à l'extérieur des passerelles et est gérée par des éléments extérieurs de commande d'appel. Le profil IPCablecom qui est décrit dans la présente Recommandation sera désigné par l'expression de *protocole de commande de passerelle de jonction (TGCP) IPCablecom – Profil 1*.

La présente Recommandation est fondée sur la Rec. UIT-T J.162 concernant la signalisation d'appel fournie par le réseau IPCablecom (NCS, *network-based call signalling*), sur le document RFC 3435 du groupe IETF: *Media Gateway Control Protocol (MGCP)* (Protocole de commande de passerelle média, version 1.0, qui est le résultat de la fusion du projet IETF concernant le protocole simple de commande de passerelle avec le projet IETF concernant la famille des protocoles de commande de dispositif IP (IPDC, *IP device control*), et sur les contributions apportées par l'équipe thématique sur les passerelles RTC du système IPCablecom.

La présente Recommandation, qui définit le protocole TGCP IPCablecom, constitue une spécification indépendante du protocole MGCP afin d'offrir une base de référence stable tout en répondant aux délais actuellement imposés de mise sur le marché d'une telle référence. Cette Recommandation vise à s'aligner au mieux sur les protocoles NCS et MGCP 1.0 dans l'environnement IPCablecom de façon à éviter la mise au point de multiples protocoles pour résoudre le même problème. Cet objectif a été poursuivi et continuera à l'être grâce à la coopération avec les auteurs des Recommandations relatives aux protocoles NCS et MGCP. Le profil TGCP du protocole MGCP est toutefois défini strictement et uniquement par le contenu de la présente Recommandation.

Ce profil TGCP du protocole MGCP, qui sera désigné dans la présente Recommandation par les termes *TGCP 1.0* ou *Version 1.0 du protocole de signalisation d'appel par passerelle de jonction sur RTC* ou *profil TGCP* ou simplement *protocole TGCP*, a été modifié comme indiqué ci-après par rapport au document RFC 2435 du groupe IETF concernant le protocole MGCP 1.0:

- *le protocole TGCP ne vise qu'à prendre en charge les passerelles RTPC de voix sur IP IPCablecom.* Le protocole TGCP prend en charge les passerelles *RTPC* de voix sur IP comme défini par IPCablecom. La fonctionnalité présente dans le protocole MGCP 1.0, qui était superflue pour le protocole TGCP, a été supprimée;
- *le protocole TGCP contient des extensions et des modifications au protocole MGCP*. Les exigences propres à l'environnement IPCablecom sont prises en compte dans le protocole TGCP. L'architecture du protocole MGCP, ainsi que toutes les structures MGCP relatives aux passerelles RTC, ont cependant été conservées dans le protocole TGCP;
- *le protocole TGCP contient des simplifications mineures par rapport au protocole MGCP 1.0.* Lorsque plusieurs options sont offertes sans être nécessairement requises pour une passerelle *RTPC* dans l'environnement IPCablecom, quelques simplifications ont été faites pour les implémentations des passerelles de jonction.

Bien que le protocole MGCP ne soit pas le protocole TGCP et inversement, les dénominations *MGCP* et *TGCP* seront utilisées indistinctement dans la présente Recommandation car celle-ci est fondée sur le protocole MGCP. Sauf indication contraire dans le texte ou déduction contraire du contexte, le terme *protocole MGCP* sera considéré comme désignant ici le profil TGCP.

Le protocole TGCP est conçu de façon à répondre aux exigences protocolaires de l'interface entre contrôleur de passerelle média et passerelle média, qui est définie dans l'architecture IPCablecom.

6 Introduction

La présente Recommandation décrit le profil TGCP d'une interface de programmation d'application (API) appelée *commande de passerelle média* (MGCI) ainsi qu'un protocole correspondant (MGCP) pour la commande des passerelles de jonction à partir d'éléments extérieurs de commande d'appel. Une passerelle de jonction est un élément de réseau qui fournit un accès à un réseau de voix sur IP (VoIP) au moyen d'un circuit de jonction de signalisation par support analogique, pseudo-analogique, numérique ou voie par voie.

Les passerelles de jonction servent à assurer l'interface avec le RTPC et, à ce titre, sont censées être conformes aux normes électriques, opérationnelles et sémaphores applicables au type de jonction qu'elles mettent en œuvre. La Recommandation relative au protocole TGCP a été mise au point pour les opérateurs locaux alternatifs (CLEC, *competitive local exchange carriers*) et, dans sa première version, les types de jonction pris en charge étaient limités à ce qui suit:

- l'ISUP du système SS7;
  - les jonctions par voies supports normales;
- les jonctions multifréquences (MF) dans un commutateur local à accès en parallèle ou en série (EAEO/AT, *equal access end office/access tandem*);

- les jonctions de services d'opérateur[1];
- les accès d'abonné d'opérateur CLEC par l'intermédiaire d'un opérateur LEC;
- les accès d'opérateur à un commutateur EAEO/AT pour la vérification de ligne occupée et pour les interventions;
- les jonctions des services d'urgence pour l'accès à un tandem de services d'urgence.

Le protocole MGCP implique une architecture de commande d'appel dans laquelle "l'intelligence" de la commande d'appel se trouve à l'extérieur des passerelles et est gérée par des éléments extérieurs de commande d'appel, appelés *contrôleurs de passerelle média*. Le protocole MGCP part du principe que ces éléments de commande d'appel, ou contrôleurs de passerelle média (MGC), se synchroniseront les uns avec les autres afin d'envoyer des ordres cohérents aux passerelles qu'ils régissent. Le protocole MGCP défini dans la présente Recommandation ne définit pas de mécanisme pour synchroniser les contrôleurs de passerelle média bien que de futures Recommandations IPCablecom puissent spécifier de tels mécanismes.

Le protocole MGCP implique un modèle de connexion dans lequel les structures fondamentales sont des extrémités et des connexions. Une passerelle contient un ensemble d'extrémités qui sont des sources ou des puits de données et qui peuvent être physiques ou virtuelles.

Un exemple d'extrémité physique est un circuit interurbain aboutissant à une passerelle de jonction qui boucle une jonction de voies supports de l'ISUP allant vers un commutateur local. Un autre exemple est un client intégré ou une passerelle résidentielle qui boucle des lignes RTC résidentielles (vers des postes téléphoniques) bien que de tels dispositifs ne soient pas visés par la présente Recommandation.

Un exemple d'extrémité virtuelle est une source de signaux audio dans un serveur de fichiers audio. La création d'extrémités physiques requiert une installation matérielle tandis que la création d'extrémités virtuelles peut être réalisée par un logiciel. Le profil TGCP du protocole MGCP ne vise toutefois que les extrémités physiques.

Les connexions sont de type point à point. Une connexion point à point est une association entre deux extrémités en vue de transmettre des données de l'une à l'autre. Une fois cette association établie pour les deux extrémités, le transfert de données entre celles-ci peut avoir lieu. L'association est établie par création de deux moitiés de connexion: l'une à l'extrémité d'origine et l'autre à l'extrémité de destination.

Les contrôleurs de passerelle média donnent aux passerelles l'instruction de créer des connexions entre des extrémités et de détecter certains événements, comme des essais de continuité, et de produire certains signaux comme des retours d'appel. Il appartient strictement au contrôleur de passerelle média de spécifier comment et quand les connexions sont établies, entre quelles extrémités elles le sont ainsi que quels événements et signaux sont à détecter et à produire aux extrémités. La passerelle devient ainsi un simple dispositif sans aucun état d'appel, qui reçoit des instructions générales du contrôleur de passerelle média sans qu'il soit besoin de savoir ni même de connaître les concepts de communication, d'état d'appel, d'éléments de service ou d'interactions entre éléments de service. Lorsque de nouveaux services sont introduits, que des profils de client sont modifiés, etc., ces changements sont transparents pour la passerelle. Les contrôleurs de passerelle média implémentent ces changements et produisent le nouvel assortiment d'instructions appropriées à donner aux passerelles pour tenir compte des changements effectués. Chaque fois que la passerelle redémarre, elle se présente à l'état neuf et exécute simplement les instructions du contrôleur de passerelle média telles qu'elle les reçoit.

---

[1] Les services d'opérateur sont censés être fournis, directement ou indirectement, au moyen d'un opérateur de commutateur local (LEC, *local exchange carrier*).

6.1 Relation avec d'autres normes IPCablecom

Une passerelle RTPC conforme à l'environnement IPCablecom se compose des trois éléments fonctionnels suivants:

- le contrôleur de passerelle média (MGC), qui contient l'intelligence de la communication et auquel aboutit la signalisation d'appel. Ce composant est également désigné par le terme *agent d'appel*;
- la passerelle média (MG), à laquelle aboutissent les voies supports conformément aux instructions et aux commandes du contrôleur MGC. Cette fonction est également désignée par le terme *passerelle de jonction* (TGW);
- la passerelle de signalisation ou sémaphore (SG), qui connecte la signalisation d'appel au RTPC et qui offre une fonction de conversion des signaux de signalisation.

En plus des protocoles de passerelle RTPC, l'environnement IPCablecom comporte un protocole de signalisation d'appel fournie par le réseau (NCS), qui est un profil MGCP. Le protocole SDP a un rôle central à jouer dans cette architecture. Les deux protocoles NCS et TGCP font appel au protocole de description de session (SDP) afin de transporter les descriptions de session.

Les systèmes IPCablecom appliquant les protocoles NCS et TGCP conservent tous les états d'appel à l'intérieur des agents d'appel (MGC) et des serveurs CMS. La Figure 1 ci-dessous décrit les relations entre ces différents composants.

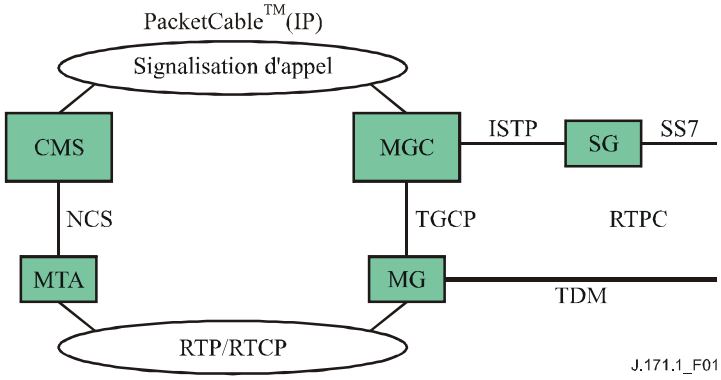


Figure 1/J.171.1 – Relation entre composants NCS et TGCP

6.2 Relation avec RFC 3435 et avec le formalisme ABNF

Le document RFC 3435 comporte une description formelle de la syntaxe du protocole MGCP conformément au "Formalisme BNF augmenté (ABNF) des spécifications de syntaxe". Cette description formelle est prise comme référence par les développeurs lors de la création de dispositifs interopérables. Une copie de la syntaxe du protocole MGCP, annotée et éditée afin d'indiquer son applicabilité aux Recommandations IPCablecom, est reproduite dans l'Appendice VI. L'application de ces directives peut améliorer l'interopérabilité en minimisant les défaillances dues à différentes interprétations syntaxiques et grammaticales.

7 Interface de commande de passerelle média (MGCI)

Les fonctions de l'interface MGCI assurent la commande de la connexion, la commande des extrémités, l'audit et la signalisation des descripteurs d'état. Elles emploient toutes le même modèle de système et les mêmes conventions de dénomination.

7.1 Modèle et conventions de dénomination

Le protocole MGCP prévoit un modèle de connexion dont les éléments fondamentaux sont les extrémités et les connexions. Les connexions sont regroupées en communications. Une même communication peut comporter une ou plusieurs connexions. Les connexions et les communications sont établies à l'initiative d'un ou de plusieurs contrôleurs de passerelle média (MGC). Il convient néanmoins de noter qu'en aucun de ces cas une "connexion" n'est établie dans un réseau IPCablecom au sens utilisé dans le RTPC (à commutation de circuits). Les termes "communication" (ou "appel") et "connexion" sont utilisés dans ce contexte (et dans l'ensemble de la présente Recommandation) afin de s'y référer facilement et non afin d'indiquer une quelconque similitude technique ou autre entre le réseau IPCablecom et le RTPC.

7.1.1 Noms des extrémités

Les noms des extrémités, alias *identificateurs d'extrémité*, sont composés de deux parties qui sont ici définies de manière à être insensibles à la hauteur de casse:

- le nom de domaine de la passerelle gérant l'extrémité;
- un nom d'extrémité local, situé à l'intérieur de cette passerelle.

Les noms des extrémités auront la forme suivante:

```
nom-d'extrémité-local@nom-de-domaine
```

où le nom-de-domaine est un nom absolu tel qu'il est défini dans le document IETF RFC 1034, qui comporte une partie relative au serveur local et dont un exemple pourrait être le suivant:

```
MyTrunkingGateway.cablelabs.com
```

La chaîne `domain-name` peut également être une adresse IPv4 en format décimal à séparation par points, représentée par une chaîne de texte entourée à gauche et à droite de crochets ("[" et "]") telle que "[128.96.41.1]" – Consulter le document IETF RFC 821 pour plus de détails. De façon générale, l'utilisation des adresses IP est toutefois déconseillée.

Les passerelles de jonction ont une ou plusieurs extrémités (p. ex. une extrémité par jonction) qui leur sont associées, et chacune de ces extrémités est identifiée par un nom local distinct d'extrémité. Comme dans le cas du nom de domaine, le nom d'extrémité local est insensible à la hauteur de casse. A ce nom d'extrémité local, on associe un type-d'extrémité qui définit le type de l'éxtrémité, p. ex. DS-0 ou ligne d'accès analogique. Le type peut être obtenu à partir du nom d'extrémité local, qui est un nom hiérarchique dont la composante la moins spécifique correspond au terme qui est situé à l'extrémité gauche, tandis que la composante la plus spécifique correspond au terme situé à l'extrémité droite. Plus formellement, le nom d'extrémité local doit respecter les règles suivantes en matière de dénomination:

- les différents termes du nom d'extrémité local doivent être séparés par une barre oblique unique ("/", ASCII 2F hex).
- les différents termes sont des chaînes de caractères ASCII composées de lettres, de chiffres et d'autres caractères imprimables, à l'exception des caractères employés pour délimiter les noms-d'extrémité ("/", "@"), des caractères génériques de remplacement ("*", "$") et des caractères d'espacement vide;
- les termes devant être remplacés dans le nom sont indiqués soit par un astérisque ("*") soit par un signe dollar ("$"). Donc, si l'ensemble du nom local d'extrémité a la forme suivante:

  ```
  term1/term2/term3
  ```

  et qu'un de ses termes soit remplacé, ce nom d'extrémité local aura la forme suivante:

`term1/term2/*`	lorsque `term3` est remplacé.
`term1/*/*`	lorsque `term2` et `term3` sont remplacés.

L'astérisque aurait pu être remplacé dans chacun des exemples par un signe dollar;

- le remplacement ne peut se faire qu'à partir de la droite. Donc, lorsqu'un terme est remplacé, tous ceux qui sont à sa droite doivent l'être également;
- lorsque le signe dollar est employé en même temps que l'astérisque, il doit se situer à sa droite. Donc, lorsqu'un terme est remplacé par un signe dollar, tous ceux qui sont à sa droite doivent l'être par lui également;
- on interprétera un terme représenté par un astérisque comme suit: "employer *toutes* les valeurs de ce terme qui s'appliquent à la passerelle de jonction concernée". Sauf spécification contraire, ce générique se rapporte à toutes les extrémités configurées pour le service, quel que soit leur état de service actuel, c'est-à-dire en service ou hors service;
- on interprétera un terme représenté par un signe dollar comme suit: "employer une valeur *quelconque* de ce terme qui s'applique à la passerelle de jonction concernée". Sauf spécification contraire, ce générique ne se rapporte qu'aux extrémités qui sont en service;
- chaque type-d'extrémité peut fournir des précisions supplémentaires en ce qui concerne les règles en matière de dénomination pour ce type d'extrémité, ces règles n'allant toutefois pas à l'encontre de celles qui sont mentionnées ci-dessus.

Il convient de noter que différents types-d'extrémité, ou même différents sous-termes comme des "lignes" dans le même type-d'extrémité, donneront deux noms d'extrémité différents. En conséquence, chaque "ligne" sera considérée comme une extrémité distincte. Etant donné que la portion nominative du domaine fait partie de l'identificateur d'extrémité, différentes formes ou différentes valeurs se rapportant à la même entité ne sont pas librement interchangeables. Après un redémarrage, la forme ou valeur fournie le plus récemment DOIT toujours être utilisée.

7.1.1.1 Noms des extrémités des passerelles de jonction

Les conventions supplémentaires en matière de dénomination qui sont spécifiées dans le présent paragraphe s'appliqueront aux extrémités dans les passerelles de jonction.

Les passerelles de jonction prendront en charge le type-d'extrémité de base suivant:

- `ds` circuit de jonction à débit DS-0.

  Il est prévu que le type d'extrémité de base soit configuré au moyen d'informations supplémentaires sur le type de signalisation pris en charge par le circuit de jonction et sur la fonction de système de commutation qu'il assure.

7.1.1.1.1 Extrémités des circuits de jonction

En plus des conventions de dénomination spécifiées ci-dessus, les noms locaux des extrémités dans les passerelles de jonction RTPC de type "`ds`" DOIVENT respecter les conventions suivantes:

- les noms locaux des extrémités comporteront une suite de termes séparés par une barre oblique ("/") qui décrivent la hiérarchie physique dans la passerelle:

  `ds/<unit-type1>-<unit #>/<unit-type2>-<unit #>/.../<channel #>`

- le premier terme (`ds`) identifie le système utilisé pour nommer les extrémités et le type d'extrémité de base;
- le dernier terme est un nombre décimal qui indique le numéro[2] de la *voie* au plus bas niveau hiérarchique;
- les termes intermédiaires entre le premier terme (`ds`) et le dernier terme (numéro de voie) indiquent des niveaux hiérarchiques intermédiaires et comportent le type d'unité <unit-type> et le numéro d'unité <unit #> séparés par un trait d'union ("-") où:

[2] Veuillez noter l'emploi du terme "voie" plutôt que le terme "intervalle de temps".

– le type d'unité <unit-type> identifie le niveau hiérarchique particulier. Les valeurs du type d'unité <unit-type> actuellement définies sont les suivantes: "s", "su", "oc3", "ds3", "e3", "ds2", "e2", "ds1" et "e1" où "s" est un numéro de créneau et où "su" est une sous-unité dans un créneau. D'autres valeurs indiquant des niveaux hiérarchiques physiques, dont il n'a pas été tenu compte dans cette liste mais auxquelles s'appliquent les mêmes règles fondamentales en matière de dénomination, seront également admises;

– le numéro d'unité <unit #> est un nombre décimal qui est utilisé pour renvoyer à une instance particulière d'un type d'unité <unit-type> à ce niveau hiérarchique.

• le nombre de niveaux et la dénomination de ces niveaux sont fondés sur la hiérarchie physique au sein de la passerelle média, comme illustré dans les exemples suivants:

– passerelle média ayant un certain nombre d'interfaces DS1:

```
ds/ds1-#/#
```

– passerelle média ayant un certain nombre d'interfaces OC3, qui contiennent des hiérarchies DS3 et DS1 dans les voies:

```
ds/oc3-#/ds3-#/ds1-#/#
```

– passerelle média contenant un certain nombre de créneaux, chacun d'eux ayant un certain nombre d'interfaces DS3:

```
ds/s-#/ds3-#/ds1-#/#
```

• certaines extrémités peuvent ne pas contenir tous les niveaux hiérarchiques possibles, mais tous les niveaux pris en charge par une extrémité donnée sont contenus dans la désignation de cette extrémité. P. ex. une interface DS3 sans verrouillage de trames DS1 pourrait être nommée comme suit:

```
ds/s-#/ds3-#/#
```

ce système de dénomination ne permet toutefois pas de représenter une interface DS3 *avec* verrouillage de trames DS1;

• la dénomination de remplacement est conforme aux conventions stipulées au § 7.1.1, le caractère astérisque ("*") se rapportant à "tous", tandis que le caractère dollar ("$") se rapporte à "un quelconque". Le remplacement d'une gamme "[N-M]" de voies allant de la voie N à la voie M, inclusivement, est également pris en charge:

– il convient de noter que le caractère de remplacement "tous" pour le premier terme (`ds`) se rapporte à tous les types d'extrémité dans la passerelle média, quels que soient ces types. Cette caractéristique est généralement destinée à être employée à des fins administratives, p. ex. l'audit ou le redémarrage;

– un nom d'extrémité local peut être sous-spécifié lorsque le nombre de termes fournis est inférieur au nombre normal à compter de la gauche du nom d'extrémité. Dans ce cas, les termes manquants à droite du dernier terme spécifié sont supposés être le caractère de remplacement "*", se rapportant à "tous", à moins que les termes spécifiés ne soient le caractère de remplacement "un quelconque", auquel cas les termes manquants à droite du dernier terme spécifié sont supposés être le caractère de remplacement "un quelconque";

– lorsque l'emploi du caractère de remplacement "tous" est admis, on peut utiliser au lieu de celui-ci le caractère de remplacement d'une gamme de voies "[N-M]" dans le dernier terme (c'est-à-dire le numéro de voie <channel-#>) du nom d'extrémité local. Ce caractère de remplacement "gamme" se rapportera alors à toutes les voies de N à M. Les règles et les restrictions qui s'appliquent à l'emploi du caractère de remplacement "tous" s'appliqueront également à l'emploi du caractère de remplacement "gamme".

Les exemples suivants illustrent l'emploi des caractères de remplacement:

`ds/ds1-3/*` toutes les voies de l'interface ds1 numéro 3 de la passerelle média concernée;

`ds/ds1-3/$` une quelconque voie de l'interface ds1 numéro 3 de la passerelle média concernée;

`ds/*` toutes les extrémités des circuits de jonction de la passerelle média concernée;

`*` toutes les extrémités (quel que soit le type d'extrémité) de la passerelle média concernée;

`ds/ds1-3/[1-24]` voies 1 à 24 de l'interface ds1 numéro 3 de la passerelle média concernée.

La forme canonique des noms est définie dans ce qui précède pour les extrémités d'une passerelle de jonction. Il est prévu que les alias pourront être pris en charge dans une version ultérieure de la présente Recommandation, p. ex. afin de prendre en charge la liaison entre des circuits DS-0 multiples pour les communications vidéo, p. ex. de la forme "`ds/ds1-1/H0-1`".

7.1.2 Noms des appels

Les appels (communications) sont identifiés par des identificateurs uniques et indépendants des plates-formes ou agents sous-jacents. Les identificateurs d'appel sont des chaînes hexadécimales qui sont établies par le contrôleur MGC. Des identificateurs d'appel d'une longueur maximale de 32 caractères DOIVENT être pris en charge.

Au minimum, les identificateurs d'appel DOIVENT être uniques dans l'ensemble des contrôleurs MGC qui commandent les mêmes passerelles. Toutefois, la coordination entre les contrôleurs MGC au sujet de ces identificateurs d'appel sort du cadre de la présente Recommandation. Lorsqu'un contrôleur MGC établit plusieurs connexions qui se rapportent au même appel, sur une même passerelle ou sur des passerelles différentes, ces connexions seront toutes liées au même appel au moyen de l'identificateur d'appel. Cet identificateur pourra ensuite être utilisé par des procédures de comptabilité ou celles de gestion, qui sortent du cadre du protocole MGCP.

7.1.3 Noms des connexions

Les identificateurs de connexion sont établis par la passerelle lorsqu'il est demandé à celle-ci d'établir une connexion. Ils identifient la connexion dans le cadre d'une extrémité. Les identificateurs de connexion sont traités dans le protocole MGCP comme des chaînes hexadécimales. La passerelle DOIT faire en sorte qu'une période d'attente correcte, d'au moins trois minutes, s'écoule entre la fin d'une connexion qui a employé cet identificateur et l'utilisation de celui-ci dans une nouvelle connexion pour la même extrémité. La longueur maximale du nom d'une connexion est de 32 caractères.

7.1.4 Noms des contrôleurs de passerelle média et d'autres entités

Le protocole de commande de passerelle média a été conçu afin d'améliorer la fiabilité du réseau afin de permettre l'implémentation de contrôleurs MGC redondants. Cela veut dire qu'il n'y a pas de liaison fixe entre les entités et les plates-formes matérielles ou les interfaces de réseau.

Les noms des contrôleurs MGC sont composés de deux parties, comme les noms des extrémités. La partie locale du nom ne révèle aucune structure interne. Un exemple de nom de contrôleur MGC est donné ci-après:

```
mgc1@mgc.whatever.net
```

Les précautions suivantes permettent d'assurer la fiabilité:

- des entités telles que les passerelles de jonction ou les contrôleurs MGC sont identifiées par leur nom de domaine, et non par leurs adresses dans le réseau. Plusieurs adresses peuvent

être associées à un nom de domaine. Si une commande ne peut être transmise à l'une des adresses dans le réseau, les implémentations DOIVENT réessayer la transmission au moyen d'une autre adresse;

- les entités peuvent passer à une autre plate-forme. L'association entre un nom logique (nom de domaine) et la plate-forme effective est conservée dans le service de dénomination de domaine (DNS, *domain name system*). Les contrôleurs MGC et les passerelles DOIVENT garder la trace de la lecture, faite dans le service DNS, de la durée de vie de l'enregistrement. Ils DOIVENT interroger le service DNS afin de renouveler les informations si la durée de vie a expiré.

Outre le traitement des données grâce à l'emploi des noms de domaine et du service DNS, la notion "d'entité notifiée" est essentielle pour la fiabilité et le basculement dans le protocole MGCP en cas de défaillance. "L'entité notifiée" pour une extrémité est le contrôleur MGC commandant effectivement cette extrémité. A tout moment, une extrémité dispose d'une et d'une seule "entité notifiée" qui lui est associée et, lorsqu'elle doit envoyer une commande au contrôleur MGC, elle DOIT l'envoyer à "l'entité notifiée" effective en indiquant la ou les extrémités auxquelles cette commande s'applique. Dès le démarrage, on DOIT attribuer une valeur fixée par la configuration à "l'entité notifiée". La plupart des commandes envoyées par le contrôleur MGC sont en mesure d'indiquer explicitement le nom de "l'entité notifiée" en utilisant un paramètre "NotifiedEntity". "L'entité notifiée" DOIT rester inchangée jusqu'à la réception d'un nouveau paramètre "NotifiedEntity" ou jusqu'à la réinitialisation de l'extrémité. S'il n'y a pas "d'entité notifiée" pour une extrémité ou qu'elle n'ait pas été explicitement[3] fixée, l'adresse de "l'entité notifiée" passera alors par défaut à celle de l'expéditeur de la dernière commande de traitement de la connexion ou de la dernière demande de notification reçue pour l'extrémité. L'audit ne modifiera donc pas "l'entité notifiée".

Le paragraphe 7.4 contient une description plus détaillée de la fiabilité et du basculement en cas de défaillance.

7.1.5 Scripts de numérotation

Dans le protocole MGCP, le contrôleur MGC peut demander à la passerelle de recueillir les chiffres qui ont été composés par un utilisateur. Cette fonctionnalité est généralement utilisée par des lignes d'accès analogique avec des passerelles résidentielles afin de recueillir les numéros que compose un utilisateur. Elle peut également être utilisée pour les interfaces de commutateur privé de signalisation voie par voie. Plutôt que d'envoyer chaque chiffre au contrôleur MGC dès sa détection, celui-ci peut fournir une grammaire décrivant combien de chiffres devraient être accumulés avant qu'il en soit notifié. Cette grammaire est désignée comme étant un *script de numérotation*.

Aucun type de circuit pris en charge par la version actuelle de la Recommandation relative au protocole TGCP ne nécessite de script de numérotation, et celui-ci ne fait donc pas partie de la présente Recommandation.

7.1.6 Evénements et signaux

La notion d'événements et de signaux est essentielle pour le protocole MGCP. Un contrôleur MGC peut demander à être notifié lorsque certains événements se produisent à une extrémité, p. ex. des décrochages de poste téléphonique. Il peut également demander que certains signaux soient appliqués à une extrémité, comme le retour d'appel sonore.

Les événements et les signaux sont regroupés dans des paquetages à l'intérieur desquels ils se partagent le même espace nominatif, auquel nous nous référerons ci-après par le terme de *noms*

---

[3] Ceci pourrait être dû au fait qu'aucune valeur n'a été attribuée au paramètre NotifiedEntity.

d'événement. Un paquetage est un ensemble d'événements et de signaux qui sont pris en charge par un type d'extrémité endpoint-type particulier. P. ex. un paquetage peut prendre en charge un certain groupe d'événements et de signaux pour les circuits ISUP, tandis qu'un autre paquetage peut prendre en charge un autre groupe d'événements et de signaux pour les circuits multifréquences. Il peut exister un ou plusieurs paquetages pour un type d'extrémité endpoint-type donné, et chaque type d'extrémité endpoint-type possède un paquetage par défaut auquel il est associé.

Les noms d'événement sont composés d'un nom de paquetage et d'un code d'événement. Etant donné que chaque paquetage définit un espace nominatif distinct, les mêmes codes d'événement peuvent être utilisés dans différents paquetages. Les noms de paquetage et les codes d'événement sont des chaînes de lettres, de chiffres et de trait d'unions insensibles à la hauteur de casse, et soumis à la restriction que le trait d'union NE DOIT PAS être le premier ou le dernier caractère d'un nom. Certains codes d'événement doivent parfois être paramétrés au moyen de données supplémentaires, ce qui peut être fait par l'adjonction de paramètres entre parenthèses. Le nom du paquetage est séparé du code d'événement par une barre oblique ("/"). Il peut ne pas être contenu dans le nom d'événement, auquel cas le nom du paquetage par défaut est attribué au type d'extrémité concerné. P. ex. pour un circuit de jonction ISUP, le paquetage ISUP (nom du paquetage: "IT") étant le paquetage par défaut, les deux noms d'événement suivants sont considérés comme équivalents:

`IT/oc` opération achevée dans le paquetage ISUP pour un circuit de jonction ISUP.

`oc` opération achevée dans le paquetage ISUP (par défaut) pour un circuit de jonction ISUP.

Le Tableau 1 ci-dessous énumère des types d'extrémité de passerelle de jonction ainsi que les paquetages définis pour ces types dans la présente Recommandation. Le contrôleur MGC DOIT prendre en charge tous les paquetages énumérés dans le Tableau 1. La passerelle MG DOIT prendre en charge les paquetages "IT", "FXR" et "XRM". La passerelle MG DEVRAIT prendre en charge les paquetages "MO" et "MT".

Tableau 1/J.171.1 – Paquetages associés aux types d'extrémité

Type d'extrémité	Paquetage	Nom du paquetage	Paquetage par défaut?	Première version qui a introduit l'exigence de prise en charge du paquetage
DS-0	Jonction ISUP	IT	Oui	1.0
DS-0	OSS à MF	MO	Non	1.5
DS-0	MF à l'arrivée	MT	Non	1.5
DS-0	FAX	FXR	Non	1.5
DS-0	Mesures VoIP	XRM	Non	1.5

Des noms de paquetage et des codes d'événement supplémentaires peuvent être définis par l'architecture IPCablecom ou y être enregistrés. Toute modification des paquetages définis dans la présente Recommandation DOIT entraîner un changement de nom du paquetage, ou un changement de numéro de la version du profil TGCP, ou éventuellement les deux.

Chaque paquetage DOIT être défini, c'est-à-dire que son nom DOIT être défini ainsi que chaque événement qui en fait partie. La définition des événements DOIT inclure le nom précis de l'événement, c'est-à-dire son code d'événement, une définition claire de l'événement et, selon le cas, la définition précise des signaux correspondants, p. ex. les fréquences exactes des signaux audio tels que les tonalités de retour d'appel sonore ou de télécopie. Les événements doivent en outre spécifier s'ils sont durables (voir § 7.3.1) et s'ils contiennent des états d'événement contrôlables (voir § 7.3.8.1). Les signaux DOIVENT également être de type défini (activé/désactivé, temporisé,

ou bref) et les signaux de temporisation DOIVENT avoir une valeur de temporisation par défaut – voir § 7.3.1.

Outre les paquetages IPCablecom, les responsables de l'implémentation PEUVENT trouver profitable de définir des paquetages expérimentaux. Le nom de ces paquetages DOIT commencer par les deux caractères "x-" ou "X-". Dans l'architecture IPCablecom, les noms de paquetage qui commencent par ces deux caractères ne DOIVENT PAS être enregistrés. Une passerelle qui reçoit une commande se rapportant à un paquetage non pris en charge DOIT renvoyer une erreur (code d'erreur 518 – paquetage non pris en charge).

Les noms de paquetage et les codes d'événement prennent chacun en charge une seule notation de remplacement. Le caractère de remplacement "*" (astérisque) peut être utilisé pour renvoyer à tous les paquetages pris en charge par l'extrémité concernée, tandis que le code d'événement "tous" peut l'être pour renvoyer à tous les événements contenus dans le paquetage concerné. P. ex.:

`IT/all` renvoie à tous les événements contenus dans le paquetage de jonction ISUP pour un circuit de jonction ISUP.

`*/all` renvoie à tous les paquetages et à tous les événements contenus dans ces paquetages qui sont pris en charge par l'extrémité concernée pour un circuit de jonction ISUP.

En conséquence, le nom de paquetage "*" NE DOIT PAS être attribué à un paquetage, et le code d'événement "tous" NE DOIT PAS être utilisé dans un quelconque paquetage.

Des événements et des signaux sont détectés et produits par défaut aux extrémités: toutefois, certains événements et signaux peuvent être détectés et produits dans les connexions en plus ou au lieu de ceux qui sont détectés ou produits à un extrémité. P. ex. il peut être demandé que les extrémités fournissent une tonalité de retour d'appel sonore lors d'une connexion. Afin qu'un événement ou un signal puisse être détecté ou produit dans une connexion, la définition de l'événement ou du signal DOIT explicitement établir que l'événement ou le signal peut être détecté ou produit dans une connexion.

Lorsqu'un signal est appliqué à une connexion, le nom de celle-ci est ajouté au nom de l'événement, au moyen du signe "arobase" (@) en tant que signe de délimitation, comme dans l'expression suivante:

```
IT/rt@0A3F58
```

Si la connexion doit être supprimée pendant qu'un événement ou signal y est en cours de détection ou lui est appliqué, cette détection d'événement ou cette production de signal particulière DOIT s'arrêter. Selon le type de signal, l'extrémité DEVRAIT produire une défaillance, c'est-à-dire que si le type de signal est TO, l'événement "échec de l'opération" sera produit car la connexion associée au signal a été supprimée avant l'expiration de la temporisation du signal. L'action de notification associée à la signalisation de la défaillance doit être conforme aux opérations de notification comme défini pour le traitement des demandes de notification (§ 7.3.1).

Le caractère de remplacement "*" (astérisque) peut servir à indiquer "toutes connexions" à l' ou aux extrémités affectées. Quand cette convention est utilisée, la passerelle DOIT produire ou détecter l'événement sur toutes les connexions qui sont connectées à l' ou aux extrémités. Un exemple de cette convention est le suivant:

```
IT/ma@*
```

Cependant, quand l'événement est effectivement observé, la passerelle DOIT comprendre le nom de la connexion spécifique sur laquelle l'événement s'est produit. Le caractère de remplacement "$" (signe dollar) peut être utilisé pour indiquer la "connexion actuelle". Cette convention NE DOIT PAS être utilisée, à moins que la demande de notification d'événement ne soit "intégrée"dans une commande CreateConnection ou ModifyConnection. Lorsque la convention est employée, la

passerelle DOIT produire ou détecter l'événement dans la connexion qui est effectivement en cours d'établissement ou de modification. Un exemple de cette convention est donné ci-après:

```
IT/rt@$
```

Lors du traitement d'une commande au moyen du générique "connexion actuelle", le caractère générique "$" DOIT être étendu par la passerelle à la valeur de la connexion actuelle. Si une commande subséquente se rapporte explicitement (p. ex. par audit de vérification) ou implicitement (p. ex. par persistence) à un tel événement, la valeur étendue DOIT être utilisée par la passerelle. En d'autres termes, le générique "connexion actuelle" est étendu une seule fois, c'est-à-dire au traitement initial de la commande dans laquelle il a été explicitement inclus.

L'identificateur de connexion ou un caractère de remplacement peut être utilisé conjointement avec les conventions "tous les paquetages" et "tous les événements". P. ex. la notation suivante:

```
*/all@*
```

peut être utilisée pour désigner tous les événements dans toutes les connexions à l'extrémité ou aux extrémités concernées. Cependant, l'utilisation des caractères génériques "tous les paquetages" et "tous les événements" est fortement déconseillée. Les agents d'appel doivent être en mesure de fonctionner dans un environnement où certaines extrémités ne prennent pas en charge tous les paquetages. Une extrémité qui reçoit une commande faisant référence à un paquetage qu'elle ne prend pas en charge va répondre par un code d'erreur 518 (Paquetage non pris en charge ou inconnu). Dès réception de cette réponse d'erreur, l'agent d'appel pourrait essayer la commande de nouveau sans le paramètre de paquetage bien que, si la commande initiale contenait des paramètres pour des paquetages multiples, l'agent d'appel n'eût aucun moyen de savoir quel ou quels paquetages spécifiques seraient à exclure. L'agent d'appel peut également utiliser la commande AuditEndpoint afin de déterminer l'ensemble des paquetages pris en charge par une extrémité.

7.2 Utilisation du protocole de description de session (SDP)

Le contrôleur MGC emploie le protocole MGCP pour fournir aux passerelles la description de paramètres de connexion tels que les adresses IP, les ports UDP et les profils de protocole en temps réel (RTP, *real-time protocol*). Sauf mention ou implication contraire dans la présente Recommandation, les descriptions du protocole SDP DOIVENT se faire suivant les conventions établies dans le protocole de description de session (SDP) qui est maintenant une norme proposée par l'IETF et décrite dans le document IETF RFC 2327. Par ailleurs, tous les contrôleurs MGC et toutes les passerelles médias DOIVENT ignorer d'éventuels paramètres, attributs ou champs SDP qui ne seraient pas compris par l'agent d'appel ou par la passerelle.

Le protocole SDP permet de décrire les conférences multimédias. Le profil TGCP ne prendra en charge que l'établissement de connexions audio au moyen du type de média "audio".

Le protocole SDP permet la description de la télécopie en temps réel au moyen du type de média "image". Le profil TGCP prendra en charge l'établissement des connexions de télécopie utilisant le type de média "image".

7.3 Fonctions de commande de passerelle

Le présent paragraphe décrit les commandes du protocole MGCP sous la forme d'un appel de procédure à distance (RPC, *remote procedure call*) telle que l'interface API, que nous désignerons comme étant l'interface de commande de passerelle média (MGCI). Une fonction de l'interface MGCI est définie pour chaque commande du protocole MGCP, cette fonction prenant et renvoyant les mêmes paramètres que la commande correspondante du protocole MGCP. Les fonctions indiquées dans le présent paragraphe fournissent une description de haut niveau de l'exploitation du protocole MGCP et donnent un exemple d'une interface API de type appel RPC qui PEUT être utilisée pour l'implémentation du protocole MGCP. Bien que l'interface API de type MGCI soit

simplement un exemple d'interface API, le comportement sémantique défini par l'interface MGCI fait partie intégrante de la Recommandation, et toutes les réalisations DOIVENT être conformes à la sémantique spécifiée pour l'interface MGCI. Les messages du protocole MGCP effectivement échangés, y compris les formats et les codages des messages utilisés, sont définis au § 8. Les passerelles de jonction DOIVENT permettre de les implémenter exactement comme spécifié.

Le service de l'interface MGCI se compose des commandes de prise en charge des connexions et des extrémités. Un aperçu des commandes est donné ci-après:

- le contrôleur MGC peut lancer une commande NotificationRequest à une passerelle, ordonnant à celle-ci de surveiller des événements particuliers tels qu'une tonalité de prise de ligne ou de télécopie se produisant à une extrémité spécifiée;
- la passerelle utilisera ensuite la commande Notify pour informer le contrôleur MGC lorsque les événements demandés se produisent à l'extrémité spécifiée;
- le contrôleur MGC peut employer la commande CreateConnection pour établir une connexion qui aboutit à une extrémité à l'intérieur de la passerelle;
- le contrôleur MGC peut utiliser la commande ModifyConnection pour modifier les paramètres associés à une connexion établie précédemment;
- le contrôleur MGC peut employer la commande DeleteConnection pour supprimer une connexion existante. Dans certaines circonstances, la commande DeleteConnection peut aussi être utilisée par une passerelle pour indiquer qu'une connexion ne peut pas être maintenue plus longtemps;
- le contrôleur MGC peut employer les commandes AuditEndpoint et AuditConnection pour vérifier l'état d'une "extrémité" et toutes les connexions qui lui sont associées. Une gestion du réseau plus poussée que celle qui est obtenue au moyen de ces commandes est généralement souhaitable, p. ex. en ce qui concerne des informations sur l'état de la passerelle de jonction et de chacun des circuits de jonction. On prévoit que ces capacités seront prises en charge lors de l'utilisation du protocole simple de gestion de réseau (SNMP, *simple network management protocol*) et de la définition d'une base d'informations de gestion (MIB, *management information base*), sujets qui sortent du cadre de la présente Recommandation;
- la passerelle peut employer la commande RestartInProgress pour notifier au contrôleur MGC que l'extrémité (ou un groupe d'extrémités gérés par elle) est mise hors service ou a été remise en service.

Ces services permettent au contrôleur (normalement le contrôleur MGC) de communiquer à une passerelle l'établissement des connexions qui aboutissent à une extrémité reliée à la passerelle, et d'être informé des événements qui se produisent à cette extrémité. Actuellement, une extrémité de passerelle de jonction est limitée à un circuit de jonction particulier dans une passerelle de jonction.

Les connexions sont regroupées en "appels" (communications). Plusieurs connexions, qui appartiennent éventuellement au même appel, peuvent aboutir à la même extrémité. Chaque connexion est spécifiée par un paramètre "mode", dont la valeur peut être fixée à "envoi seulement" (sendonly), "réception seulement" (recvonly), "envoi/réception" (sendrecv), "inactif" (inactive), "bouclage" (loopback), "essai de continuité" (conttest), "bouclage en réseau" (netwloop) ou "essai de continuité en réseau" (netwtest). Le paramètre "mode" détermine si les paquets médias peuvent être envoyés ou reçus par l'intermédiaire de la connexion; le RTPCP n'est toutefois pas concerné.

Les signaux audio reçus en provenance d'une extrémité DOIVENT être envoyés par toute connexion à cette extrémité dont le mode est soit "envoi seulement" soit "envoi/réception", à moins que l'extrémité ait une connexion en mode "bouclage" ou "essai de continuité". Les données audio produites par l'application d'un signal à une connexion DOIVENT cependant être envoyées par la connexion dans tous les modes, sauf "bouclage en réseau".

Le traitement des signaux audio qui sont reçus par ces connexions est également déterminé par les paramètres de mode comme suit:

- les signaux audio reçus en paquets de données par l'intermédiaire de connexions en mode "inactif", "bouclage" ou "essai de continuité" DOIVENT être ignorés;
- les signaux audio reçus en paquets de données par l'intermédiaire de connexions en mode "réception seulement" ou "envoi/réception" DOIVENT être combinés puis envoyés à l'extrémité[4], à moins que l'extrémité ait une autre connexion en mode "bouclage" ou "essai de continuité";
- les signaux audio en provenance d'une extrémité DOIVENT être transmis par l'intermédiaire de toutes les connexions dont le mode est "envoi seulement" ou "envoi/réception", à moins que cette extrémité ait une autre connexion en mode "bouclage" ou "essai de continuité".

Noter qu'afin de détecter des événements sur une connexion, la connexion DOIT par défaut être dans un des modes "réception seulement", "envoi/réception", "bouclage en réseau" ou "essai de continuité en réseau". La détection d'événement ne s'applique qu'à l'audio entrant. Les connexions en mode "envoi seulement", "inactif", "bouclage", ou "essai de continuité" ne vont donc normalement pas détecter d'événements quelconques, bien qu'une demande en ce sens ne soit pas considérée comme une erreur.

Les modes "bouclage" et "essai de continuité" sont utilisés au cours d'opérations de maintenance et d'essai de continuité. Une extrémité peut avoir plus d'une seule connexion en mode "bouclage" ou "essai de continuité". Tant qu'il y a au moins une connexion dans ce mode particulier et qu'aucune autre connexion à cette extrémité n'est placée dans un mode différent de maintenance ou d'essai, l'opération de maintenance ou d'essai DOIT continuer sans perturbation. Il existe deux variantes d'essai de continuité (COT, *continuity test*), l'une destinée à un usage général et l'autre employée dans le cas de plusieurs réseaux nationaux. Dans le premier cas, l'essai est un essai de bouclage. Le commutateur de départ enverra une tonalité (la tonalité d'aller) sur le circuit support et attendra que le commutateur d'arrivée reboucle le circuit. Si le commutateur de départ observe que la même tonalité lui est renvoyée (la tonalité de retour), l'essai COT a réussi. Dans le cas contraire, il a échoué. Dans le second cas, les tonalités d'aller et de retour sont différentes. Le commutateur de départ envoie une certaine tonalité d'aller. Le commutateur d'arrivée détecte la tonalité d'aller, mais impose une tonalité de retour différente dans le sens arrière. Lorsque le commutateur de départ détecte la tonalité de retour, l'essai COT a réussi. S'il ne la détecte pas dans un certain délai, l'essai COT a échoué.

Si le mode est mis à "bouclage", la passerelle DOIT renvoyer le signal entrant, issu d'une extrémité, vers cette même extrémité. Cette procédure sera normalement utilisée pour vérifier la continuité de circuits de jonction conformément aux spécifications de l'UIT. Si le mode est mis à "essai de continuité", la passerelle est informée que l'autre bout du circuit a entamé une procédure d'essai de continuité conformément aux procédures spécifiées dans le cas de plusieurs réseaux nationaux. La passerelle placera le circuit dans le mode transpondeur nécessaire aux essais de continuité à deux tonalités.

En outre, lorsqu'une connexion d'extrémité est en mode "bouclage" ou "essai de continuité":

- les signaux audio reçus dans toute connexion d'extrémité NE DOIVENT *pas* être envoyés à cette extrémité;
- les signaux audio reçus à l'extrémité NE DOIVENT *pas* être envoyés vers une quelconque connexion de cette extrémité.

---

[4] Les extrémités conformes au protocole TGCP ne sont actuellement pas exigées pour la prise en charge de la combinaison.

Si le mode est "bouclage en réseau", les signaux audio reçus par l'intermédiaire de la connexion DOIVENT être renvoyés en écho dans cette même connexion. Le mode "bouclage en réseau" DEVRAIT simplement fonctionner comme un réflecteur de paquets conformes au protocole RTP. Le média NE DOIT PAS être réexpédié vers l'extrémité.

Le mode "essai de continuité en réseau" est employé pour vérifier la continuité à travers le réseau IP. Un signal propre au type d'extrémité est envoyé vers les extrémités par l'intermédiaire du réseau IP, et les extrémités sont ensuite censées renvoyer en écho ce signal dans le réseau IP après l'avoir fait passer par l'équipement interne de la passerelle pour en vérifier le fonctionnement correct. Le signal DOIT passer par un décodage et un recodage internes avant d'être renvoyé. Pour les extrémités en hiérarchie DS-0, le signal est de type audio et NE DOIT PAS être envoyé dans un circuit connecté à l'extrémité, quel que soit l'état effectif de prise de ce circuit.

Des connexions existantes et nouvelles à l'extrémité NE DOIVENT PAS être affectées par des connexions placées en mode "bouclage en réseau" ou "essai de continuité en réseau". Toutefois, des contraintes locales relatives aux ressources peuvent limiter le nombre de nouvelles connexions qui peuvent être établies.

Voir l'Appendice I en ce qui concerne les illustrations d'interactions modales.

7.3.1 Commande NotificationRequest

La commande NotificationRequest sert à demander que la passerelle envoie une notification lorsque des événements spécifiés se produisent à une extrémité. P. ex. une notification peut être demandée lorsque des tonalités associées à une communication destinée à la télécopie sont détectées à l'extrémité. L'entité recevant cette notification, habituellement le contrôleur MGC, peut alors décider qu'un type de codage différent devrait être utilisé dans les connexions aboutissant à cette extrémité et en informer la passerelle en conséquence[5].

```
ReturnCode
     ← NotificationRequest(EndpointId
                         [, NotifiedEntity]
                         [, RequestedEvents]
                         , RequestIdentifier
                         [, SignalRequests]
                         [, QuarantineHandling]
                         [, DetectEvents])
```

Le paramètre **EndpointId** est l'identificateur de l'extrémité ou des extrémités de la passerelle où la commande NotificationRequest est exécutée. Il DOIT suivre les règles applicables aux noms d'extrémité spécifiées au § 7.1.1. Le caractère de remplacement "un quelconque" NE DOIT PAS être employé. Une passerelle qui reçoit une demande de notification avec la convention générique "un quelconque" DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse. La convention générique "tous" DOIT être prise en charge pour les demandes de notification où chacun des paramètres RequestedEvents, SignalsRequest et DetectEvents est soit vide ou omis. Par concision, certaines passerelles peut décider de ne pas prendre en charge le générique "tous" pour les demandes de notification où un ou plusieurs de ces paramètres n'est ni vide ni omis. De telles passerelles doivent répondre par le code d'erreur 503 s'ils reçoivent une demande de notification remplacée par la convention générique "tous" qu'ils sont dans l'incapacité de traiter pour ce motif.

Le paramètre **NotifiedEntity** est un paramètre facultatif qui spécifie une nouvelle "entité notifiée" pour l'extrémité. Quand il est utilisé, le nom entier du contrôleur de passerelle média DOIT être spécifié. Il doit comprendre à la fois le nom local et nom de domaine – même si une adresse IP entre parenthèses est utilisée pour le nom de domaine. Voir les § 7.1.1 et 7.1.4 pour de plus amples

[5] La nouvelle instruction consisterait en une commande ModifyConnection.

informations. Si cependant, seul le nom de domaine est fourni, la passerelle MG DEVRAIT utiliser le nom de domaine en tant qu'identificateur d'agent d'appel.

Le paramètre **RequestIdentifier** est employé pour associer cette demande à la notification qu'elle peut déclencher. Il sera répété dans la commande Notify correspondante.

Le paramètre **SignalRequests** est un paramètre qui contient un ensemble de signaux que la passerelle est invitée d'appliquer. Sauf spécification contraire, les signaux sont appliqués à l'extrémité, mais certains signaux peuvent être appliqués à une connexion. Des exemples de signaux[6] sont donnés ci-après:

- essai de continuité;
- appel multifréquence de démarrage au système d'aide à l'exploitation.

Les signaux sont divisés en différents types en fonction de leur comportement:

- `On/off (OO)` (à désactivation) – Lorsqu'ils sont appliqués, ces signaux fonctionnent jusqu'à ce qu'ils soient arrêtés. Cela ne peut se produire que lorsqu'un nouveau paramètre SignalRequests est défini, indiquant que le signal est désactivé (voir ci-dessous). Des signaux du type OO sont définis comme étant idempotents, donc plusieurs demandes d'activation (ou de désactivation) d'un signal OO donné sont parfaitement valables et NE DOIVENT PAS entraîner d'erreur. Lorsqu'ils sont activés, ces signaux NE DOIVENT PAS être désactivés avant que le contrôleur MGC l'ait explicitement ordonné, ou que l'extrémité ait été redémarré.
- `Time-out (TO)` (à temporisation) – Lorsqu'ils sont appliqués, ces signaux durent jusqu'à ce qu'ils soient arrêtés (en raison de la présence d'un événement ou du fait qu'ils n'ont pas été inclus dans une liste suivante [éventuellement vide] de signaux), ou qu'un intervalle de temps propre au signal s'est écoulé. Un signal qui est interrompu produira un événement "opération achevée" (voir le § A.1 pour une définition plus approfondie de cet événement). Un signal TO pourrait consister en un " placement d'appel multifréquence" qui est interrompu après 16 s. Si un événement se produit avant la fin des 16 s, le signal sera arrêté[7] par défaut. Si le signal n'est pas arrêté, il arrivera à expiration, s'arrêtera et produira un événement "opération achevée" dont le contrôleur MGC peut éventuellement avoir demandé à être notifié. S'il a demandé à en être notifié, l'événement "opération achevée" qui lui est envoyé comprendra le ou les noms du ou des signaux qui sont arrivés à expiration[8]. Le ou les signaux produits dans une connexion comporteront le nom de celle-ci. Une temporisation par défaut, qui peut être modifiée par le processus de préconfiguration, est définie pour les signaux TO. La temporisation peut également être fournie comme paramètre au signal. Une valeur zéro indique que la temporisation est illimitée. Un signal TO qui est défaillant après avoir été activé, mais avant d'avoir produit un événement "opération achevée", produira un événement "échec de l'opération" qui comportera le ou les noms du ou des signaux qui sont arrivés à expiration[8].
- `Brief (BR)` (bref) – La durée de ces signaux est tellement courte qu'ils s'arrêtent d'eux-mêmes. Si un événement tel que l'arrêt d'un signal se produit, ou qu'un nouveau paramètre SignalRequests soit défini, un signal BR activé ne s'arrêtera pas. Toutefois, tout signal BR en attente non encore activé sera annulé.

---

6 Voir à l'Annexe A pour une liste exhaustive des signaux.

7 La commande "garder le ou les signaux activés" peut l'emporter sur ce comportement.

8 Si les paramètres ont été transmis au signal, il n'en sera pas fait état.

Les signaux sont appliqués par défaut aux extrémités. Si un signal appliqué à une extrémité entraîne la production d'un flux de média (audio, vidéo, etc.), ce flux de média NE DOIT PAS être transmis dans une connexion reliée à cette extrémité, quel que soit le mode de connexion. P. ex. lorsqu'une tonalité est appliquée à une extrémité qui est impliquée dans une communication active, seule l'entité utilisant l'extrémité concernée entendra la tonalité. Des signaux différents peuvent toutefois définir un comportement différent.

Lorsqu'un signal est appliqué à une connexion qui a reçu un paramètre RemoteConnectionDescriptor (voir § 7.3.3), le flux de média produit par ce signal DOIT être transmis par l'intermédiaire de la connexion *quel que soit* le mode effectif de connexion. Si un paramètre RemoteConnectionDescriptor n'a pas été reçu, la passerelle DOIT renvoyer une erreur (code d'erreur 527 – Paramètre RemoteConnectionDescriptor manquant).

Lorsqu'une liste (éventuellement vide) de signaux est fournie, elle remplace entièrement la liste en cours des signaux TO activés. Les signaux TO effectivement activés qui ne sont pas fournis dans la nouvelle liste DOIVENT être arrêtés et le ou les nouveaux signaux fournis seront alors activés. Les signaux TO effectivement activés qui sont fournis dans la nouvelle liste DOIVENT rester activés sans interruption, et le temporisateur pour ces signaux TO ne sera donc pas modifié. Il n'y a en conséquence aucun moyen de relancer le temporisateur pour un signal TO effectivement activé sans désactiver le signal d'abord. Si le signal TO est paramétré, l'ensemble initial de paramètres DOIT rester valable, quelles que soient les valeurs qui sont fournies par la suite. Un signal donné NE DOIT PAS figurer plus d'une fois dans un paramètre SignalRequests. L'omission du paramètre SignalRequests est interprétée comme une liste vide de demandes de signal.

On trouvera à l'Annexe A les signaux qui sont actuellement définis.

Le paramètre **RequestedEvents** est une liste d'événements que la passerelle est invitée à détecter à l'extrémité. Sauf indication contraire, les événements sont détectés à l'extrémité. Toutefois, certains événements peuvent être détectés dans une connexion. Des exemples d'événements sont donnés ci-après:

- prise de ligne;
- tonalités de télécopie;
- opération achevée;
- appel multifréquence entrant.

On trouvera à l'Annexe A les événements qui sont actuellement définis.

A chaque événement sont associées une ou plusieurs **mesures** qui définissent celle que la passerelle doit prendre lorsque l'événement en question se produit. Les mesures possibles sont les suivantes:

- notifier l'événement immédiatement, en même temps que la liste des événements observés jusque-là;
- recueillir l'événement;
- ne pas tenir compte de l'événement;
- garder le ou les signaux activés;
- insérer la demande NotificationRequest;
- insérer la demande ModifyConnection.

Les types d'événements que l'extrémité sera invitée à détecter sont au nombre de deux: les événements durables et les événements non durables.

Les événements durables sont toujours détectés à une extrémité. Si un événement durable ne fait pas partie de la liste des événements RequestedEvents et qu'il se produise, il sera détecté de toute manière et traité comme tous les autres événements, comme s'il avait été demandé au moyen d'une commande Notify[9]. Donc, sans que ce soit officiel, les événements durables peuvent être considérés comme faisant implicitement toujours partie de la liste des événements RequestedEvents avec effet sur la commande Notify, même si aucune détection manifeste, etc. n'aura lieu[10]. Les événements durables sont identifiés comme tels au moyen de leur définition – voir l'Annexe A.

Les événements non durables sont des événements qui doivent explicitement faire partie de la liste des événements RequestedEvents. La liste (éventuellement vide) d'événements demandés remplace entièrement la précédente liste d'événements demandés. L'extrémité ne détectera, outre les événements durables, que les événements qui figurent dans la liste des événements demandés. Si un événement durable fait partie de la liste des événements RequestedEvents, la mesure spécifiée remplacera la mesure par défaut associée à cet événement pendant la durée de vie de la liste, suite à quoi la mesure par défaut sera rétablie. Un événement donné NE DOIT PAS figurer plus d'une fois dans la liste des événements RequestedEvents. L'omission du paramètre RequestedEvents est interprétée comme une liste vide d'événements demandés. On peut spécifier plusieurs mesures pour un événement, même si une mesure donnée ne peut figurer plus d'une fois pour un événement donné. Le Tableau 2 spécifie les combinaisons admises de mesures:

Tableau 2/J.171.1 – Mesures associées aux événements

Evénement	Notifier	Recueillir	Ne pas tenir compte	Garder le ou les signaux activés	Insérer la demande NotificationRequest	Insérer la demande ModifyConnection
Notifier	–	–	–	√	–	√
Recueillir	–	–	–	√	√	√
Ne pas tenir compte	–	–	–	√	–	√
Garder le ou les signaux activés	√	√	√	–	√	√
Insérer la demande NotificationRequest	–	√	–	√	–	√
Insérer la demande ModifyConnection	√	√	√	√	√	–

Si un client reçoit une demande comportant une mesure non valable ou une combinaison non admise de mesures, il DOIT renvoyer une erreur au contrôleur MGC (code d'erreur 523 – Mesure inconnue ou combinaison non admise de mesures).

Lorsque de nombreuses mesures sont spécifiées, p. ex. "garder le ou les signaux activés" et "notifier", les différentes mesures sont supposées prises simultanément.

Un contrôleur MGC peut envoyer à la passerelle une commande NotificationRequest dont la liste des événements RequestedEvents est vide. Les événements durables seront toutefois encore détectés et notifiés.

---

9 Le paramètre RequestIdentifier sera donc le paramètre RequestIdentifier de la commande effective NotificationRequest.

10 Normalement, lorsqu'une demande d'observation, p. ex. de décrochage, est faite, elle ne peut aboutir que si le combiné n'est pas déjà décroché.

Quand un stimulus qui déclenche de multiples événements demandés est détecté (p. ex. une tonalité de télécopie est le stimulus pour à la fois FXR/gwfax(start) et L/ft), la passerelle DOIT produire seulement un de ces événements (à savoir l'événement préféré en premier parmi de multiples événements demandés qui ont été déclenchés) selon les règles de priorité suivantes:

1) les événements inclus dans une liste d'événements demandés sont ordonnés de gauche à droite, l'événement préféré en premier étant énuméré à gauche.

2) les événements durables qui ne sont pas inclus dans une liste d'événements demandés sont moins préférés que les événements (durables ou non) qui sont inclus dans une liste RequestedEvents. Il n'y a aucun ordre de préférence entre les événements durables qui ne sont pas inclus dans une liste RequestedEvents.

Les signaux appliqués à la suite de la commande SignalRequests sont synchronisés avec l'ensemble des événements spécifiés dans le paramètre RequestedEvents ou découlant de lui, sauf si la mesure "garder le ou les signaux activés" l'emporte. La définition formelle stipule que la production de tous les signaux TO DOIT s'arrêter dès qu'un des événements demandés est détecté, à moins que la mesure "garder le ou les signaux activés" ne soit associée à l'événement spécifié.

Si l'on souhaite qu'un ou plusieurs signaux TO restent activés lorsqu'un événement recherché se produit, la mesure "garder le ou les signaux activés" peut être utilisée. Cette mesure a pour effet de conserver l'activité de tous les signaux TO effectivement activés, en refusant l'arrêt par défaut des signaux TO lorsqu'un événement se produit.

Si l'on souhaite qu'un ou des signaux débutent lorsqu'un événement recherché se produit, la mesure "insérer la demande NotificationRequest " peut être utilisée. La demande imbriquée NotificationRequest peut comporter une nouvelle liste d'événements RequestedEvents et une nouvelle commande SignalRequests. Elle ne peut toutefois pas inclure une autre commande "insérer la demande NotificationRequest". Lorsqu'elle est activée, la liste des événements observés et le tampon de quarantaine resteront inchangés (voir § 7.4.3.1).

La mesure relative à la demande imbriquée NotificationRequest permet au contrôleur MGC d'élaborer un "miniscript" devant être traité par la passerelle immédiatement après la détection de l'événement associé. Toute commande SignalRequests qui est spécifiée dans la demande imbriquée NotificationRequest sera déclenchée immédiatement. Des soins particuliers doivent être pris afin de veiller à ce que les divergences entre le contrôleur MGC et la passerelle soient évitées. Des désaccords à long terme ne devraient toutefois pas avoir lieu puisque de nouvelles commandes SignalRequests remplacent complètement l'ancienne liste de signaux TO activés, et que les signaux de type BR s'arrêtent toujours d'eux-mêmes. Il est recommandé de limiter le nombre de signaux de type OO. Il est souhaitable que le contrôleur MGC n'active qu'occasionnellement tous les signaux OO qui devraient être activés, et ne désactive qu'occasionnellement aussi tous ceux qui devraient être désactivés.

Si l'on désire changer les modes de connexion lorsqu'un événement recherché se produit, la mesure "insérer la demande ModifyConnection" peut être utilisée. La demande imbriquée ModifyConnection peut comporter une liste des changements de mode de connexion, chacun de ceux-ci incluant le changement de mode et l'identificateur de connexion concerné. Le caractère de remplacement "$" peut être utilisé pour indiquer la "connexion actuelle", mais cette notation NE DOIT PAS être employée en dehors d'une commande de prise en charge de connexion – car le caractère de remplacement se rapporte à la connexion concernée pour la commande de traitement de connexion.

La mesure relative à la demande imbriquée ModifyConnection permet au contrôleur MGC d'ordonner à l'extrémité de modifier le mode de connexion d'une ou de plusieurs connexions immédiatement après la détection de l'événement associé. Chaque changement de mode de connexion fonctionne d'une manière semblable à la commande ModifyConnection correspondante. Lorsqu'une liste des changements de mode de connexion est fournie, les changements de mode de

connexion DOIVENT être appliqués l'un après l'autre en allant de gauche à droite. Lorsque tous les changements de mode de connexion ont été effectués, un événement "opération achevée" paramétré au moyen du nom de la mesure achevée se produira (voir l'Annexe A pour plus de détails). Si l'un des changements de mode de connexion devait échouer, un événement "échec de l'opération" paramétré au moyen du nom de la mesure et du changement de mode de connexion qui ont échoué se produira (voir l'Annexe A pour plus de détails) – les autres changements de mode de connexion NE DOIVENT PAS être tentés et les précédents changements réussis de mode de connexion dans la liste NE DOIVENT PAS être modifiés non plus.

Finalement, la mesure "Ne pas tenir compte" peut être employée pour ne pas tenir compte d'un événement, à savoir pour éviter qu'un événement durable soit notifié. Toutefois, la synchronisation entre l'événement et le signal activé se fera encore par défaut.

NOTE – Le paragraphe 7.4.3.1 contient des détails supplémentaires sur la sémantique de la détection et du rapport des événements. Le lecteur est invité à examiner cette question avec soin.

La définition particulière des mesures qui sont nécessaires pour répondre aux commandes SignalRequests sort du cadre de la présente Recommandation de base relative au protocole TGCP. Cette définition peut varier d'un endroit à l'autre, et donc d'une passerelle à l'autre. En conséquence, les définitions qui sont fournies dans des paquetages d'événements peuvent être données en dehors de la présente Recommandation de base. On trouvera une liste initiale des paquetages d'événements à l'Annexe A.

Les commandes RequestedEvents et SignalRequests se réfèrent généralement aux mêmes événements. Dans un cas, la passerelle est invitée à détecter les occurrences d'un événement, tandis que dans l'autre cas, elle est invitée au produire. Il n'y a que peu d'exceptions à cette règle, notamment en ce qui concerne les tonalités de télécopie et de modem, qui peuvent être détectées mais ne peuvent être signalées. Toutefois, nous ne pouvons forcément pas nous attendre que toutes les extrémités détectent tous les événements. Les événements et les signaux particuliers qu'une extrémité donnée peut détecter ou produire sont déterminés en fonction de la liste des paquetages d'événements qui sont pris en charge par cette extrémité. Chaque paquetage spécifie une liste d'événements et de signaux qui peuvent être détectés ou appliqués. Une passerelle qui est invitée à détecter ou à appliquer un événement appartenant à un paquetage qui n'est pas pris en charge par l'extrémité spécifiée DOIT renvoyer une erreur (code d'erreur 512 ou 513 – Non équipé pour la détection d'un événement ou la production d'un signal). Lorsque le nom d'un événement n'est pas qualifié par un nom de paquetage, on suppose que le nom de paquetage pour l'extrémité est le nom par défaut. Si le nom de l'événement n'est pas enregistré dans ce paquetage par défaut, la passerelle DOIT renvoyer une erreur (code d'erreur 522 – Evénement ou signal inexistant).

Le contrôleur MGC peut envoyer une demande NotificationRequest dont la liste des signaux demandés est vide. Cela a pour effet d'arrêter tous les signaux TO activés. Il peut agir de la sorte, p. ex. lorsque l'émission d'une tonalité, à savoir un retour d'appel sonore, devrait s'arrêter.

Le paramètre **QuarantineHandling** est un paramètre facultatif qui spécifie les options de traitement pour le tampon de quarantaine (voir § 7.4.3.1) c'est-à-dire les événements qui ont été détectés par la passerelle avant l'arrivée de cette NotificationRequest mais qui n'ont pas encore été notifiés au contrôleur MGC. Ce paramètre fournit un ensemble d'options de traitement:

– si les événements mis en quarantaine devraient être traités ou rejetés (l'action par défaut consiste à les traiter);
– si la passerelle est censée produire au plus une seule une notification (perpétuelle), ou de multiples notifications (réitérées en boucle), en réponse à cette demande (l'action par défaut consiste à en produire au plus une seule).

Quand ce paramètre est absent, les événements mis en quarantaine DOIVENT être traités. La prise en charge du mode "notification perpétuelle" (au moyen d'une valeur par défaut) et du mode "notification en boucle"est obligatoire. Une extrémité qui reçoit une demande de notification avec

une valeur paramétrique QuarantineHandling non prise en charge DEVRAIT répondre par un code d'erreur 508 (valeur non prise en charge du paramètre QuarantineHandling).

Noter que le paramètre QuarantineHandling régit également le traitement des événements qui ont été détectés et traités mais pas encore notifiés quand la commande est reçue.

Le paramètre **DetectEvents** est un paramètre facultatif qui spécifie une liste minimale d'événements que la passerelle est invitée à détecter dans l'état "notification" et dans l'état "perpétuel". La liste est valable jusqu'à ce qu'une nouvelle valeur soit spécifiée. On trouvera d'autres explications relatives à ce paramètre au § 7.4.3.1.

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.2 Notifications

Les notifications sont envoyées par la passerelle à l'aide de la commande Notify lorsqu'un événement observé doit être notifié:

```
ReturnCode
     ← Notify(EndpointId
               [, NotifiedEntity]
               , RequestIdentifier
               , ObservedEvents)
```

Le paramètre **EndpointId** est le nom de l'extrémité de la passerelle, provenant de la commande Notify, comme défini au § 7.1.1. L'identificateur DOIT être un nom d'extrémité complètement spécifié, comprenant le nom de domaine de la passerelle. La partie locale du nom NE DOIT PAS faire appel à la convention concernant les caractères de remplacement. Un contrôleur MGC qui reçoit une notification avec convention générique DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

Le paramètre **NotifiedEntity** est un paramètre facultatif qui identifie l'entité à laquelle la notification est envoyée. Ce paramètre est le même que celui qui figure dans la commande NotificationRequest qui a déclenché cette notification. Noter que la passerelle MG ne PEUT inclure le nom de domaine de son entité notifiée que si ce nom de domaine n'a été reçu que dans la requête NotificationRequest déclencheuse. Dans ce cas, le contrôleur MGC DEVRAIT accepter la valeur. Ce paramètre est absent si la demande ayant déclenché la notification ne le contenait pas. Quelle que soit la valeur du paramètre NotifiedEntity, la notification DOIT être envoyée à "l'entité notifiée" actuelle pour cette extrémité.

Le paramètre **RequestIdentifier** est un paramètre qui est le même que celui qui figure dans la commande NotificationRequest ayant déclenché cette notification. Il est employé pour corréler cette notification avec la demande de notification qui a déclenchée cette dernière. Les événements durables seront considérés ici comme ayant été inclus dans la dernière commande NotificationRequest (ce qui inclut une demande de notification imbriquée dans des primitives de traitement de connexion). Lorsque aucune commande NotificationRequest n'a été reçue, la valeur de l'identificateur RequestIdentifier utilisé est nulle ("0").

Le paramètre **ObservedEvents** est une liste d'événements que la passerelle a détectés et recueillis, soit par des mesures de "recueil", soit par des mesures de "notification". Une seule notification peut faire état d'une liste d'événements qui seront rapportés dans l'ordre de leur détection. La liste ne peut contenir que des événements durables et des événements qui ont été demandés dans le paramètre RequestedEvents de la commande NotificationRequest ayant déclenché la notification. Les événements détectés dans une connexion comprendront le nom de cette connexion. La liste contiendra les événements qui ont été recueillis (mais non notifiés) et l'événement final qui a déclenché la notification.

Le paramètre **ReturnCode** est un paramètre renvoyé par le contrôleur MGC. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.3 Commande CreateConnection

Cette commande sert à établir une connexion.

```
ReturnCode
[, ConnectionId]
[, SpecificEndPointId]
[, LocalConnectionDescriptor]
       ← CreateConnection(CallId
       , EndpointId
              [, NotifiedEntity]
              [, LocalConnectionOptions]
              , Mode
              [, RemoteConnectionDescriptor]
              [, RequestedEvents]
              [, RequestIdentifier]
              [, SignalRequests]
              [, QuarantineHandling]
              [, DetectEvents])
```

Cette fonction est utilisée lors de l'établissement d'une connexion entre deux extrémités. Une connexion est définie par ses attributs et les extrémités qu'elle associe. Les paramètres d'entrée dans la commande CreateConnection fournissent les informations nécessaires pour construire l'une des deux "vues" des extrémités d'une connexion.

Le paramètre **CallId** est un paramètre qui identifie l'appel (ou la session) auquel cette connexion appartient. Au minimum, ce paramètre est unique auprès de l'ensemble des contrôleurs MGC qui commandent les mêmes passerelles; les connexions qui font partie du même appel ont le même identificateur d'appel. Celui-ci peut être utilisé pour identifier les appels aux fins de rapport et de comptabilité.

Le paramètre **EndpointId** est l'identificateur de l'extrémité de la passerelle où la commande CreateConnection est exécutée. Cet identificateur peut entièrement être spécifié si le paramètre EndpointId de la fonction d'appel a une valeur qui n'est pas un caractère de remplacement ou peut être sous-spécifié lorsqu'il est fait appel à la convention concernant le caractère de remplacement "un quelconque". Si l'extrémité est sous-spécifiée, son identificateur sera attribué par la passerelle et sa valeur complète ne DOIT être renvoyée dans le paramètre SpecificEndPointId de la réponse que si la commande a été suivie d'effet. Dans ce cas, l'extrémité assignée DOIT être en service et NE DOIT PAS avoir déjà de quelconque connexion rattachée. La convention concernant le caractère de remplacement "tous" NE DOIT PAS être employée. Une passerelle qui reçoit une commande CreateConnection avec la convention générique "tous" DOIT renvoyer une erreur (qui DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

Le paramètre **NotifiedEntity** est un paramètre facultatif qui spécifie une nouvelle "entité notifiée" pour l'extrémité.

Le paramètre **LocalConnectionOptions** est une structure décrivant les caractéristiques de la connexion média du point de vue de la passerelle qui exécute la commande CreateConnection. Il informe l'extrémité des caractéristiques relatives à l'envoi et à la réception de la connexion média. Les champs fondamentaux contenus dans le paramètre LocalConnectionOptions sont les suivants:

- **méthode de codage**: une liste des noms littéraux pour l'algorithme de compression (méthode de codage/décodage) employé pour envoyer et recevoir des données de média par l'intermédiaire de la connexion DOIT être spécifiée avec au moins une valeur. Les entrées de cette liste sont classées par ordre de préférence. L'extrémité DOIT choisir au moins un des codecs, et ce choix DEVRAIT se faire selon la préférence indiquée. Si l'extrémité reçoit

par l'intermédiaire de la connexion des données de média codées à l'aide d'une méthode de codage différente, elle PEUT les ignorer. L'extrémité DOIT en outre indiquer quels autres algorithmes de compression elle est prête à prendre en charge en tant qu'algorithmes de remplacement – voir § 8.4.1 pour plus de détails. Une liste des méthodes de codage admissibles est spécifiée dans la Rec. UIT-T J.161. Les noms littéraux définis dans le § 7.5 (Tableau 3/J.161) DOIVENT être utilisés. Les algorithmes de compression inconnus DEVRAIENT être ignorés s'ils sont reçus. Voir § 7.7 pour plus de détails sur le processus de sélection des codecs.

NOTE – La "méthode de codage" inclut les codages audio, photo et vidéo.

- **période de mise en paquets**: une unique période de mise en paquets PEUT être spécifiée avec une seule valeur décimale, exprimée en millisecondes. Si ce spécificateur est utilisé, alors la même période de mise en paquets DOIT être utilisée pour toutes les méthodes de codage autorisée par les options LCO (LocalConnectionOptions). Noter que si aucun champ de méthode de codage n'est spécifié dans les options LCO, la passerelle MG NE DOIT PAS choisir une méthode de codage ayant une période de mise en paquets qui diffère de celle qui est spécifiée ici. Si différentes périodes de mise en paquets pour différents codages sont recherchées, alors ce champ NE DOIT PAS être utilisé. La valeur se rapporte aux médias aussi bien envoyés que reçus. Noter que seule la période valide de mise en paquets, conjointement avec la méthode associée de codage, est à utiliser par la passerelle MG. Une liste de périodes admissibles de mise en paquets est spécifiée dans la spécification IPCablecom relative aux codecs audio/vidéo (Rec. UIT-T J.161). Ce spécificateur NE DOIT PAS être fourni dans les mêmes options LCO que le champ de périodes multiples de mise en paquets. Une passerelle MG DOIT renvoyer une erreur (code d'erreur 524 – incohérence dans les options LocalConnectionOptions) quand il reçoit un paramètre LCO avec les deux champs de Période de mise en paquets et de Périodes multiples de mise en paquets;
- **périodes multiples de mise en paquets**: une liste de périodes de mise en paquets en millisecondes PEUT être spécifiée si et seulement si le champ de Méthode de codage est inclus. Quand il est spécifié, le champ de périodes multiples de mise en paquets en millisecondes DOIT contenir exactement au moins une valeur décimale ou un trait d'union pour chaque entrée dans le champ de Méthode de codage inclus dans les options LCO (LocalConnectionOptions). Cela s'applique même si plusieurs méthodes de codage ont la même valeur. La première entrée dans la liste DOIT être un nombre décimal. Quand un trait d'union est utilisé, le codec en question DOIT utiliser la même période de mise en paquets qu'une des autres entrées dans la liste qui contient effectivement un nombre décimal et par ailleurs le codec NE DOIT PAS consommer plus de largeur de bande que l'autre entrée. Cette règle peut p. ex. être utilisée pour les codecs non vocaux (p. ex. événement téléphonique ou bruit de confort) qui utilisent la même période de mise en paquets que le codec vocal avec lequel ils sont utilisés. Des entrées successives dans la liste de périodes de mise en paquets DOIVENT être ordonnées comme dans les méthodes de codage correspondantes. Ces valeurs se rapportent aux médias aussi bien envoyés que reçus. Noter que la passerelle MG NE DOIT PAS choisir un codec avec une période de mise en paquets qui diffère de celle qui est spécifiée ici. Noter que seule la période valide de mise en paquets, conjointement avec la méthode associée de codage, est à utiliser par la passerelle MG. Une liste de périodes admissibles de mise en paquets est spécifiée dans la spécification IPCablecom relative aux codecs audio/vidéo (Rec. UIT-T J.161). Ce spécificateur NE DOIT PAS être fourni dans les mêmes options LCO comme le champ de Période de mise en paquets. Une passerelle MG DOIT renvoyer une erreur (code d'erreur 524 – incohérence dans les options LocalConnectionOptions) dans les conditions suivantes:
  - quand elle reçoit un paramètre LCO avec à la fois le champ de Période de mise en paquets et le champ de Périodes multiples de mise en paquets;

– quand elle reçoit un paramètre LCO où le nombre de codecs spécifiés dans le champ de Méthode de codage est différent du nombre d'éléments dans le champ de périodes multiples de mise en paquets;

- **compensation d'écho**: ce paramètre indique si la compensation d'écho devrait être employée du côté jonction ou pas[11]. Il peut prendre la valeur "activé" (lorsque la compensation d'écho est demandée) ou "désactivé" (lorsqu'elle est désactivée). Ce paramètre est facultatif. Lorsqu'il est omis, la passerelle de jonction DOIT employer la compensation d'écho initialement. La passerelle média DEVRAIT ultérieurement activer ou désactiver la compensation d'écho conformément à la Rec. UIT-T V.8 quand des données en bande vocale sont détectées. Pour la réactivation de la compensation d'écho, voir la Rec. UIT-T G.168. Après la fin des données en bande vocale, le traitement de compensation d'écho DOIT revenir à la valeur actuelle du paramètre de compensation d'écho. Il est RECOMMANDE que le traitement de compensation d'écho soit laissé à la passerelle plutôt que de faire spécifier ce paramètre par le contrôleur MGC;
- **type de service**: ce paramètre spécifie la classe de service qui sera utilisée pour l'envoi de données de média par l'intermédiaire de la connexion en codant le paramètre de l'en-tête IP sur 8 bits correspondant au type de service sous la forme d'une valeur de deux chiffres hexadécimaux. Ce paramètre est facultatif. Lorsqu'il est omis, une valeur par défaut de 0x00 DOIT être utilisée (sauf configuration différente). Quand ce paramètre est présent et valide, l'extrémité DOIT utiliser la valeur fournie afin de remplir les champs le paramètre de séquence codée de services différenciés (DSCP) dans l'en-tête IP (voir RFC 2474 pour de plus amples informations sur le paramètre DSCP). La valeur du paramètre DOIT être 0x00, ou DOIT être un multiple de quatre dans l'étendue de de 0x01 à 0xFF (bits 6 et 7, ou bits de notation ECN, sont réservés et donc doivent être réglés à "00"). Une extrémité DOIT renvoyer une erreur (code d'erreur 532 – Valeur(s) non prise(s) en charge pour ce paramètre dans les options LocalConnectionOptions) quand elle reçoit une valeur non valide. Le "bit" de gauche dans le paramètre correspond au bit de poids fort dans l'en-tête IP;
- **suppression des silences**: les passerelles de téléphonie peuvent exécuter une détection d'activité vocale et éviter d'envoyer des paquets pendant les périodes de silence. Il est cependant nécessaire, pour certain types de communication (p. ex. les communications par modem) de désactiver la suppression des silences. Ce paramètre peut avoir la valeur "activé" (quand le silence doit être supprimé) ou "désactivé" (quand le silence ne doit pas être supprimé). Ce paramètre est facultatif. Quand ce paramètre est omis, la valeur par défaut est "désactivé". Si la valeur est "activé", dès détection de données en bande vocale, l'extrémité DEVRAIT désactiver la suppression des silences. Après la fin des données en bande vocale, le traitement de la suppression des silences DOIT revenir à la valeur actuelle du paramètre de suppression des silences.

En outre, les champs suivants du paramètre LocalConnectionOptions sont utilisés pour la prise en charge des services de sécurité IPCablecom (une des deux ou les deux suites cryptographiques PEUVENT être présentes:

- **suite cryptographique pour le protocole RTP**: ce paramètre consiste en une liste de suites cryptographiques pour la sécurité du protocole RTP, par ordre de préférence. Les entrées de cette liste sont classées par ordre de préférence, la première suite correspondant au choix préféré. L'extrémité DOIT choisir exactement une des suites cryptographiques conformément aux règles décrites dans la Spécification relative à la sécurité IPCablecom (Rec. UIT-T J.170). Elle DEVRAIT en outre indiquer quelles autres suites cryptographiques elle est prête à prendre en charge en tant que suites de remplacement (voir

[11] La compensation d'écho du côté paquet n'est pas prise en charge.

§ 8.4.1 pour plus de détails). Chaque suite cryptographique est représentée par des chaînes ASCII composées de deux sous-chaînes (éventuellement vides) séparées par une barre oblique ("/"), la première sous-chaîne identifiant l'algorithme d'authentification tandis que la seconde sous-chaîne identifie l'algorithme de chiffrement. Une liste des suites cryptographiques admissibles est spécifiée dans la Rec. UIT-T J.170. Le paramètre de suite cryptographique RTP ne s'applique qu'aux flux de média en protocole RTP. Si le contrôleur MGC contient un paramètre LocalConnectionOptions qui impose de n'utiliser que des médias non RTP (p. ex. les procédures T.38 de relais de télécopie utilisant le protocole UDPTL), le paramètre de suite cryptographique RTP NE DOIT PAS être inclus. Si un paramètre LCO permet des médias aussi bien RTP que non RTP et si un paramètre de suite cryptographique RTP est inclus, il ne s'applique qu'au média RTP. Dans tous les cas, si le flux de média résultant sur la connexion n'est pas un flux de média RTP (p. ex. des procédures T.38 de relais de télécopie utilisant le protocole UDPTL), le paramètre de suite cryptographique RTP DOIT être ignoré, c'est-à-dire que la sécurité RTP ne va pas être utilisée et les paramètres de sécurité RTP ne sont pas inclus dans le descripteur LocalConnectionDescriptor;

- **suite cryptographique pour le protocole RTCP**: ce paramètre consiste en une liste de suites cryptographiques pour la sécurité du protocole RTCP, par ordre de préférence. Les entrées de cette liste sont classées par ordre de préférence, la première suite correspondant au choix préféré. L'extrémité DOIT choisir exactement une des suites cryptographiques, conformément aux règles décrites dans la Spécification relative à la sécurité IPCablecom (Rec. UIT-T J.170). Elle DEVRAIT en outre indiquer quelles autres suites cryptographiques elle est prête à prendre en charge en tant que suites de remplacement (voir § 8.4.1 pour plus de détails). Chaque suite cryptographique est représentée par des chaînes ASCII composées de deux sous-chaînes (éventuellement vides) séparées par une barre oblique ("/"), la première sous-chaîne identifiant l'algorithme d'authentification tandis que la seconde sous-chaîne identifie l'algorithme de chiffrement. Une liste des suites cryptographiques admissibles est spécifiée dans la Rec. UIT-T J.170 relative à la sécurité dans l'architecture IPCablecom. Le paramètre de suite cryptographique RTCP ne s'applique qu'aux flux de média en protocole RTCP. Si le contrôleur MGC contient un paramètre LocalConnectionOptions qui impose de n'utiliser que des médias non RTP (p. ex. les procédures T.38 de relais de télécopie utilisant le protocole UDPTL), le paramètre de suite cryptographique RTCP NE DOIT PAS être inclus. Si un paramètre LCO permet des médias aussi bien RTP que non RTP et si un paramètre de suite cryptographique RTCP est inclus, il ne s'applique qu'au média RTP. Dans tous les cas, si le flux de média résultant sur la connexion n'est pas un flux de média RTP (p. ex. des procédures T.38 de relais de télécopie utilisant le protocole UDPTL), le paramètre de suite cryptographique RTCP DOIT être ignoré, c'est-à-dire que la sécurité RTCP ne va pas être utilisée et les paramètres de sécurité RTCP ne sont pas inclus dans le descripteur LocalConnectionDescriptor.

La passerelle de jonction DOIT répondre par une erreur (code d'erreur 524 – Incohérence du paramètre LCO) si l'une quelconque des règles ci-dessus est violée. Toutes les valeurs par défaut susmentionnées peuvent être modifiées par le processus de préconfiguration.

Le protocole TGCP permet en outre de prendre en charge la surveillance électronique IPCablecom. Lorsqu'une connexion est soumise à une surveillance électronique, tous les paquets de média valables reçus par l'intermédiaire de cette connexion et tous les paquets de média envoyés par l'intermédiaire de cette connexion seront dupliqués et transmis à une fonction d'acheminement sous surveillance électronique[12], après insertion d'un identificateur de connexion d'archivage du contenu

[12] Il convient de noter que la duplication se fait au niveau du réseau – Voir le document PKT-SP-ESP-I01-991229 pour plus de détails.

d'appel. Cette duplication se fera suivant le mode de la connexion, sauf pour les données de média produites par des signaux appliqués à la connexion, qui seront dupliquées sans tenir compte du mode de connexion. P. ex. une connexion en mode "inactif" ne produira pas de données de média interceptées[13], tandis qu'une connexion en mode "sendonly" ne produira que des données interceptées dans le sens de l'envoi. Il ne sera pas tenu compte des paquets dupliqués dans les statistiques relatives à la connexion. Les champs suivants du paramètre LocalConnectionOptions sont utilisés pour la prise en charge de la surveillance électronique IPCablecom:

- **identificateur de connexion d'archivage du contenu d'appel**: l'identificateur de connexion d'archivage du contenu d'appel (CCC, *call content connection*) est une valeur à 32 bits qui spécifie l'identificateur à utiliser pour une connexion soumise à surveillance électronique. Il sera ajouté à l'en-tête des paquets vocaux qui seront interceptés;
- **destination d'archivage du contenu d'appel**: la destination d'archivage du contenu d'appel spécifie une adresse IPv4 suivie d'un point-virgule et d'un numéro de port UDP. Elle spécifie l'adresse IP de destination et le port pour le contenu d'appel intercepté.

Le paramètre **RemoteConnectionDescriptor** est un descripteur de connexion pour le côté distant d'une connexion, à l'autre bout du réseau IP. Il comporte les mêmes champs que le paramètre LocalConnectionDescriptor (à ne pas confondre avec le paramètre LocalConnectionOptions), à savoir les champs qui décrivent une session conformément à la norme du protocole SDP. Le paragraphe 8.4 donne des détails sur l'utilisation prise en charge du protocole SDP dans le profil TGCP. Ce paramètre peut avoir une valeur nulle lorsque l'information pour l'extrémité distante n'est pas connue. Cela peut se produire parce que l'entité qui établit une connexion commence par envoyer une commande CreateConnection à l'une des deux passerelles impliquées. Lorsque la première commande CreateConnection est lancée, aucune information n'est disponible au sujet de l'autre côté de la connexion. Cette information peut être fournie plus tard au moyen d'un messasge ModifyConnection.

Lorsque des codecs sont modifiés pendant une communication, il se peut que, pendant de courtes périodes les extrémités utilisent des codes différents. Comme mentionné ci-dessus, les passerelles de jonction PEUVENT ignorer toute donnée de média reçue qui est codée au moyen d'un codec différent de celui qui est spécifié dans le paramètre LocalConnectionOptions de la connexion.

Le paramètre **Mode** indique le mode de fonctionnement du côté considéré de la connexion. Les options sont "envoi seulement", "réception seulement", "envoi/réception", "inactif", "bouclage en réseau" ou "essai de continuité en réseau". Le traitement de ces modes est spécifié au début du § 7.3. Certaines extrémités peuvent ne pas être en mesure de prendre en charge tous les modes – Voir l'Appendice V.1. Si la commande spécifie un mode que l'extrémité ne prend pas en charge, une erreur DOIT être renvoyée (code d'erreur 517 – Mode non pris en charge). Par ailleurs, lorsqu'une connexion n'a pas encore reçu le paramètre RemoteConnectionDescriptor (RCD), une erreur DOIT être renvoyée si on a tenté de placer la connexion dans l'un des modes "envoi seulement", "envoi/réception", "bouclage en réseau", "essai de continuité dans le réseau" ou si un signal (par opposition à la détection d'un événement) doit être appliqué à la connexion (code d'erreur 527 – Paramètre RemoteConnectionDescriptor manquant RECOMMANDE).

Le paramètre **ConnectionId** est un paramètre renvoyé par la passerelle qui identifie de manière univoque la connexion dans le contexte de l'extrémité considérée. Ce paramètre DOIT être inclus avec toutes réponse provisoire ou définitive à une commande CreateConnection. Ce paramètre NE DOIT PAS être inclus quand une quelconque réponse d'erreur est renvoyée et que la connexion n'a pas été créée.

Le paramètre **LocalConnectionDescriptor** est un paramètre renvoyé par la passerelle, qui fournit une description de session contenant des informations, p. ex. sur des adresses et des ports RTP pour

[13] On suppose qu'aucun signal produisant des données de média n'a été appliqué à la connexion.

les connexions "IN" telles qu'elles sont définies dans le protocole SDP. Il est semblable au paramètre RemoteConnectionDescriptor, sauf qu'il spécifie ce côté de la connexion. Le paragraphe 8.4 donne des détails sur l'utilisation prise en charge du protocole SDP dans le profil TGCP. Le paramètre LocalConnectionDescriptor (LCD) DOIT être inclus avec toute réponse provisoire ou définitive à une commande CreateConnection. Le paramètre LocalConnectionDescriptor NE DOIT PAS être inclus quand une quelconque réponse d'erreur est renvoyée et que la connexion n'a pas été créée.

Lorsqu'une passerelle reçoit une commande "CreateConnection" qui ne comprend pas de paramètre RemoteConnectionDescriptor, elle est dans une situation ambiguë en ce qui concerne la connexion en question. Comme elle a envoyé un paramètre LocalConnectionDescriptor, elle pourrait recevoir des paquets par l'intermédiaire de cette connexion. Mais comme elle n'a pas encore reçu le paramètre RemoteConnectionDescriptor de l'autre passerelle, elle ne sait pas si les paquets qu'elle reçoit ont été autorisés par le contrôleur MGC. Elle doit donc naviguer entre deux risques, à savoir la coupe de certaines annonces importantes et l'écoute de données insensées. Le comportement de la passerelle est déterminé par la valeur du paramètre Mode (sous réserve de la sécurité):

- si le mode est mis à "réception seulement", la passerelle DOIT accepter les signaux vocaux reçus par l'intermédiaire de la connexion et les transmettre vers l'extrémité;
- si le mode est "inactif", "bouclage" ou "essai de continuité", la passerelle DOIT (comme toujours) ignorer les signaux vocaux reçus par l'intermédiaire de la connexion;
- si le mode est "bouclage en réseau" ou "essai de continuité en réseau", la passerelle DOIT renvoyer en écho ou répondre, comme attendu. Les données de média renvoyées en écho ou produites DOIVENT être envoyées à la source de média dont elles proviennent.

Il convient de noter que, lorsque l'extrémité ne dispose pas du paramètre RemoteConnectionDescriptor pour la connexion, celle-ci peut par définition ne pas être dans l'un des modes "envoi seulement" ou "envoi/réception".

Les paramètres **RequestedEvents, RequestIdentifier, SignalRequests, QuarantineHandling** et **DetectEvents** sont tous facultatifs. Ils peuvent être utilisés par le contrôleur MGC pour insérer effectivement une demande de notification qui soit exécutée simultanément avec l'établissement de la connexion. Si un ou plusieurs paramètres sont présents, le paramètre RequestIdentifier DOIT faire partie de ceux-là. L'insertion d'une demande de notification peut donc être confirmée par la présence du paramètre RequestIdentifier. Les autres paramètres peuvent être présents ou pas. Si l'un des paramètres fait défaut, la demande DOIT être traitée comme si elle était une demande NotificationRequest normale, le paramètre en question ayant été omis. Cela peut avoir pour effet d'annuler des signaux et d'arrêter la recherche des événements. Noter que l'absence des paramètres RequestedEvents et SignalRequests n'est interprétée comme une liste vide que si un paramètre RequestIdentifier est inclus.

Considérons, en guise d'exemple d'utilisation, un contrôleur MGC qui souhaiterait placer un appel à destination d'un système de services d'opérateur par l'intermédiaire d'une passerelle de jonction à multifréquence. Ce contrôleur MGC pourrait:

- demander à la passerelle de jonction d'établir une connexion afin de s'assurer qu'elle dispose des ressources nécessaires à l'appel;
- demander à la passerelle de jonction de prendre un circuit multifréquence de services d'opérateur et de lancer l'appel;
- demander à la passerelle de jonction de notifier le contrôleur MGC lorsque l'appel a été placé.

On peut effectuer toutes les opérations qui sont décrites ci-dessus à l'aide d'une seule commande CreateConnection, en incluant une demande de notification avec le paramètre RequestedEvents pour l'événement de réponse et le paramètre SignalRequests pour le signal d'établissement.

Lorsque ces paramètres sont présents, l'établissement de la connexion et la demande de notification DOIVENT être synchronisés, c'est-à-dire qu'ils sont tous deux soit acceptés soit refusés. Dans notre exemple, la commande CreateConnection doit être refusée si la passerelle ne dispose pas de ressources suffisantes ou ne peut obtenir de l'accès au réseau local les ressources appropriées. La demande de notification de lancement d'appel doit être refusée en situation de double-prise, lorsque le circuit est déjà pris. Dans cet exemple, l'appel ne doit pas être placé si la connexion ne peut être établie, et la connexion ne doit pas être établie si le circuit est déjà pris. Au lieu de cela, une erreur devrait être renvoyée (code d'erreur 401 – Circuit déjà pris), qui informerait le contrôleur MGC de la situation de double-prise.

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.4 Commande ModifyConnection

Cette commande sert à modifier les caractéristiques de la "vue" d'une connexion par une passerelle. Cette "vue" de l'appel comprend aussi bien le descripteur de connexion locale que le descripteur de connexion distant.

```
ReturnCode
[, LocalConnectionDescriptor]
        ← ModifyConnection(CallId
                , EndpointId
                , ConnectionId
                [, NotifiedEntity]
                [, LocalConnectionOptions]
                [, Mode]
                [, RemoteConnectionDescriptor]
                [, RequestedEvents]
                [, RequestIdentifier]
                [, SignalRequests]
                [, QuarantineHandling]
                [, DetectEvents])
```

Les paramètres utilisés sont les mêmes que ceux de la commande CreateConnection, avec en outre un paramètre **ConnectionId** qui identifie de manière univoque la connexion dans l'extrémité. Ce paramètre est renvoyé par la commande CreateConnection en même temps que le descripteur de connexion local. Il identifie de manière univoque la connexion dans le contexte de l'extrémité.

Le paramètre **EndpointId** DOIT être un nom d'extrémité entièrement spécifié. Le nom local NE DOIT PAS faire appel à la convention concernant le caractère de remplacement. Une passerelle qui reçoit une commande ModifyConnection avec une convention générique DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

La commande ModifyConnection peut être employée pour attribuer des paramètres de connexion, soumis aux mêmes règles et contraintes que celles qui ont été spécifiées pour la commande CreateConnection:

- fournir des informations sur l'autre bout de la connexion par l'intermédiaire du paramètre **RemoteConnectionDescriptor**;
- activer ou désactiver la connexion en modifiant la valeur du paramètre **mode**. Cela peut être fait à tout moment pendant la connexion, les valeurs des paramètres étant arbitraires. La valeur du mode pour une activation peut p. ex. être "réception seulement";
- modifier les paramètres de la connexion par l'intermédiaire du paramètre **LocalConnectionOptions**, p. ex. en passant à un système de codage différent, en modifiant la période de mise en paquets ou en changeant le traitement de la compensation d'écho.

La commande ne renverra un paramètre **LocalConnectionDescriptor** que si les paramètres de connexion locaux, tels que les ports RTP, etc., sont modifiés. Donc, si p. ex. seul le mode de connexion est changé, un paramètre LocalConnectionDescriptor ne sera pas renvoyé. Le paramètre **LocalConnectionDescriptor** NE DOIT PAS être inclus quand une quelconque réponse d'erreur est renvoyée et que la connexion n'a pas été modifiée. Si un paramètre de connexion est omis, p. ex. mode ou suppression des silences, l'ancienne valeur de ce paramètre sera conservée si possible. Si un changement de paramètre nécessite un changement dans un ou plusieurs paramètres *non spécifiés*, la passerelle est libre de choisir des valeurs appropriées pour les paramètres non spécifiés qui doivent changer[14].

Les informations concernant les adresses RTP fournies dans le paramètre RemoteConnectionDescriptor spécifient l'adresse RTP distante du récepteur de données de média pour la connexion. Ces informations d'adresse RTP peuvent avoir été modifiées par le contrôleur MGC[15]. Lorsqu'elles sont fournies à une passerelle de jonction pour une connexion, cette passerelle ne DEVRAIT accepter que les flux de média (et les flux RTCP) en provenance des adresses RTP spécifiées également. Tout flux de média reçu et provenant d'autres adresses DEVRAIT être ignoré. La Rec. UIT-T J.170 devrait être consultée au sujet desnormes de sécurité supplémentaires.

Les paramètres **RequestedEvents, RequestIdentifier, SignalRequests, QuarantineHandling** et **DetectEvents** sont facultatifs. Ils peuvent être utilisés par le contrôleur MGC pour insérer une demande de notification qui soit liée à la modification de la connexion et exécutée simultanément avec elle. Si un ou plusieurs de ces paramètres sont fournis, le paramètre RequestIdentifier DOIT faire partie de ceux-là. Noter que l'absence des paramètres RequestedEvents et SignalRequests n'est interprétée comme une liste vide que si un paramètre RequestIdentifier est inclus.

Lorsque ces paramètres sont présents, la modification de la connexion et la demande de notification DOIVENT être synchronisées, c'est-à-dire qu'elles sont toutes deux soit acceptées soit refusées.

NotifiedEntity est un paramètre facultatif qui spécifie une nouvelle "entité notifiée" pour l'extrémité.

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.5 Commande DeleteConnection (issue du contrôleur de passerelle média)

Cette commande sert à mettre fin à une connexion. Une conséquence indirecte de cette commande est la collecte de statistiques relatives à l'exécution de la connexion.

```
ReturnCode
, Connection-parameters
     ← DeleteConnection(CallId
              , EndpointId
              , ConnectionId
              [, NotifiedEntity]
              [, RequestedEvents]
              [, RequestIdentifier]
              [, SignalRequests]
              [, QuarantineHandling]
              [, DetectEvents])
```

[14] Cela peut p. ex. se produire si un changement de codec est spécifié et si l'ancien codec utilisait la suppression des silences, mais que le nouveau codec ne le prenne pas en charge. Si p. ex. la période de mise en paquets n'avait pas non plus été spécifiée et si le nouveau codec aurait pris en charge l'ancienne période de mise en paquets, la valeur de ce paramètre ne changerait pas car un changement ne serait pas nécessaire.

[15] P. ex. si le média a besoin de traverser un pare-feu.

Dans cette forme de la commande DeleteConnection, l'identificateur de l'extrémité DOIT être entièrement spécifié. La convention concernant les caractères de remplacement NE DOIT PAS être employée. Une passerelle qui reçoit cette forme de la commande DeleteConnection avec une convention générique DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

Dans le cas général où une connexion a deux extrémités, cette commande doit être envoyée aux deux passerelles impliquées dans la connexion. Après la suppression de la connexion, les flux de média du réseau en mode paquet précédemment pris en charge par la connexion ne sont plus disponibles. Tout paquet de média reçu pour l'ancienne connexion est simplement ignoré, et aucun nouveau paquet de média pour le flux n'est envoyé. Egalement après que la connexion a été supprimée, tout bouclage qui a été demandé pour la connexion DOIT être annulé (à moins que l'extrémité n'ait une autre connexion demandant un bouclage).

En réponse à la commande DeleteConnection, la passerelle renvoie une liste de paramètres qui décrivent l'état de la connexion[16]. Les paramètres de la connexion ne DOIVENT être renvoyés que si la commande est suivie d'effet et que la connexion soit supprimée. Ces paramètres sont les suivants:

- **nombre de paquets envoyés**: nombre total de paquets de données conformes au protocole RTP transmis par l'expéditeur depuis le début de la transmission par l'intermédiaire de la connexion. Le comptage n'est pas réinitialisé lorsque l'expéditeur modifie son identificateur de source de synchronisation (SSRC, *synchronization source*) telle qu'elle est définie dans le protocole RTP, p. ex. suite à une commande Modify. Cette valeur DOIT être fondée sur les informations déjà fournies par le mécanisme RTP;
- **nombre d'octets envoyés**: nombre total d'octets de charge utile (à savoir, à l'exclusion de ceux de l'en-tête ou du bourrage) transmis dans les paquets de données conformes au protocole RTP par l'expéditeur depuis le début de la transmission par l'intermédiaire de la connexion. Le comptage n'est pas réinitialisé lorsque l'expéditeur modifie son identificateur SSRC, p. ex. suite à une commande ModifyConnection. Cette valeur DOIT être fondée sur les informations déjà fournies par le mécanisme RTP;
- **nombre de paquets reçus**: nombre total de paquets de données conformes au protocole RTP reçus par l'expéditeur depuis le début de la réception par l'intermédiaire de la connexion. Le comptage inclut les paquets reçus des différentes sources SSRC si l'expéditeur a utilisé plusieurs valeurs. Tous les paquets reçus DOIVENT être comptés indépendamment du mode de la connexion ou de tout type d'erreur de traitement, p. ex. échec d'authentification;
- **nombre d'octets reçus**: nombre total d'octets de charge utile (à savoir à l'exclusion de ceux de l'en-tête ou du bourrage) reçus par l'expéditeur depuis le début de la réception par l'intermédiaire de la connexion. Le comptage inclut les paquets reçus des différentes sources SSRC si l'expéditeur a utilisé plusieurs valeurs. Tous les paquets reçus DOIVENT être comptés indépendamment du mode de la connexion ou de tout type d'erreur de traitement, p. ex. échec d'authentification;
- **nombre de paquets perdus**: nombre total de paquets de données conformes au protocole RTP qui ont été perdus depuis le début de la réception. Ce nombre est défini comme étant le nombre de paquets attendus moins le nombre de paquets réellement reçus, celui-ci incluant tout paquet tardif ou double. Le comptage inclut les paquets reçus des différentes sources SSRC si l'expéditeur a utilisé plusieurs valeurs. Donc, les paquets qui arrivent en retard ne sont pas considérés comme étant perdus, et la perte peut être négative lorsqu'ils arrivent en double. Le nombre de paquets attendus est défini comme étant le numéro étendu

[16] Les valeurs calculées ne comprendront pas les paquets qui proviennent de la surveillance électronique.

de la toute dernière séquence reçue moins le numéro de séquence initial qui a été reçu. La valeur est égale à zéro si, p. ex. aucun paquet n'a été reçu par cette connexion;

- **gigue entre arrivées**: estimation de la variance statistique du temps entre arrivées des paquets de données conformes au protocole RTP, mesurée en millisecondes et exprimée comme un nombre entier sans signe. La gigue entre arrivées "J" est définie comme étant l'écart moyen (valeur absolue lissée) de la différence "D" de l'espacement des paquets au niveau du récepteur par rapport à celui qui est observé au niveau de l'expéditeur pour une paire de paquets. On trouvera des algorithmes de calcul détaillés dans le document IETF RFC 1889. Le comptage inclut les paquets reçus des différentes sources SSRC si l'expéditeur a utilisé plusieurs valeurs. La valeur est égale à zéro si, p. ex. aucun paquet n'a été reçu par la connexion;
- **délai moyen de transmission**: estimation du temps d'attente dans le réseau, exprimé en millisecondes. Il s'agit de la valeur moyenne de la différence entre les marqueurs temporels NTP des expéditeurs des messages RTCP et les marqueurs temporels NTP des récepteurs, mesurée à la réception des messages. La moyenne est obtenue en faisant la somme de toutes les estimations puis en la divisant par le nombre de messages reçus conformément au protocole RTCP. Il convient de noter que le calcul correct de ce paramètre suppose des horloges synchronisées. Les dispositifs de passerelles de jonction PEUVENT estimer autrement le délai moyen de transmission en divisant par deux le temps mesuré d'un aller-retour.

Voir le document IETF RFC 1889 pour une définition plus détaillée de ces variables.

Les paramètres **RequestedEvents, RequestIdentifier, SignalRequests, QuarantineHandling** et **DetectEvents** sont facultatifs. Ils peuvent être utilisés par le contrôleur MGC pour transmettre une demande de notification imbriquée qui est liée à la suppression de la connexion et exécutée simultanément avec elle. Toutefois, si un ou plusieurs paramètres sont présents, le paramètre RequestIdentifier DOIT faire partie de ceux-là. P. ex. lorsqu'un circuit est déconnecté, la passerelle pourrait être invitée à supprimer la connexion et à commencer à rechercher un événement de prise de ligne. Cela peut se faire au moyen d'une seule commande DeleteConnection en transmettant également le paramètre RequestedEvents pour l'événement de prise de ligne ainsi qu'un paramètre SignalRequests ne contenant pas de valeur. Noter que l'absence des paramètres RequestedEvents et SignalRequests n'est interprétée comme une liste vide que si un paramètre RequestIdentifier est inclus.

Lorsque ces paramètres sont présents, les commandes de suppression de la connexion et de demande de notification DOIVENT être synchronisées, c'est-à-dire qu'elles sont toutes deux soit acceptées soit refusées.

Le paramètre **NotifiedEntity** est un paramètre facultatif qui spécifie une nouvelle "entité notifiée" pour l'extrémité.

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.6 Commande DeleteConnection (issue de la passerelle de jonction)

Dans certaines circonstances, une passerelle peut être amenée à libérer une connexion, p. ex. parce qu'elle a perdu la ressource associée à la connexion. Elle peut mettre fin à la connexion en utilisant une variante de la commande DeleteConnection:

```
ReturnCode
     ← DeleteConnection(CallId,
              EndpointId,
              ConnectionId,
              Reason-code,
              Connection-parameters)
```

Dans cette forme de la commande DeleteConnection, le paramètre **EndpointId** DOIT être entièrement spécifié. La convention concernant les caractères de remplacement NE DOIT PAS être employée. Un contrôleur MGC qui reçoit une commande DeleteConnection avec une convention générique DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

Le paramètre **Reason-code** est une chaîne de texte commençant par un code de cause suivi en option d'une chaîne de texte descriptif. On trouvera une liste des paramètres de code de cause au § 7.6.

Outre les paramètres **CallId, EndpointId** et **ConnectionId**, la passerelle de jonction enverra aussi les paramètres de la connexion qui auraient été renvoyés au contrôleur MGC en réponse à une commande DeleteConnection provenant de ce contrôleur. Le code de cause indique la cause de la commande DeleteConnection.

Le paramètre **ReturnCode** est un paramètre renvoyé par le contrôleur MGC. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.7 Commande DeleteConnection (issue du contrôleur de passerelle média, dans le cas de connexions multiples)

Une variante de la fonction DeleteConnection peut être utilisée par le contrôleur MGC pour supprimer plusieurs connexions en même temps. La commande peut être utilisée pour supprimer toutes les connexions qui se rapportent à un appel pour une extrémité:

```
ReturnCode
     ← DeleteConnection(CallId,
              EndpointId)
```

Dans cette forme de la commande DeleteConnection, le paramètre **EndpointId** NE DOIT PAS utiliser le caractère de remplacement "un quelconque". Toutes les connexions pour l'extrémité ou les extrémités dont le paramètre CallId est spécifié seront supprimées. La commande ne renvoie pas de statistiques individuelles ou de paramètres d'appel. Une passerelle qui reçoit une commande DeleteConnection (de connexions multiples à partir du contrôleur de passerelle média) avec la convention générique "un quelconque" DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

La commande DeleteConnection peut également être utilisée par le contrôleur MGC pour supprimer toutes les connexions qui aboutissent à une extrémité donnée:

```
ReturnCode
     ← DeleteConnection(EndpointId)
```

Dans cette forme de la commande DeleteConnection, le contrôleur MGC peut profiter de la structure hiérarchique des noms d'extrémité pour supprimer toutes les connexions qui appartiennent à un groupe d'extrémités. Dans ce cas, une partie de la composante "nom d'extrémité local" du paramètre EndpointId peut être spécifiée au moyen de la convention concernant le caractère de remplacement "tous", comme stipulé au § 7.1.1. La convention concernant le caractère de remplacement "un quelconque" NE DOIT PAS être utilisée. La commande ne renvoie pas de statistiques individuelles ni de paramètres d'appel.

Après la suppression de la connexion, les flux de médias du réseau en mode paquet précédemment pris en charge par la connexion ne sont plus disponibles. Tout paquet de média reçu pour l'ancienne connexion est simplement ignoré, et aucun nouveau paquet de média pour le flux n'est envoyé. De même, après que la connexion a été supprimée, tout bouclage qui a été demandé pour la connexion DOIT être annulé (à moins que l'extrémité n'ait une autre connexion demandant un bouclage).

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle qui indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

7.3.8 Audit

Le protocole MGCP est fondé sur une architecture de commande d'appel centralisée où un contrôleur MGC joue le rôle de contrôleur à distance des dispositifs clients qui fournissent des interfaces de communication vocale aux utilisateurs et aux réseaux. Afin d'arriver à un degré de disponibilité égal ou supérieur à celui du RTC actuel, certains protocoles font appel aux mécanismes de sondage par écho périodique des abonnés afin de réduire le temps nécessaire à la détection des différentes pannes. A cette fin, le modèle fournit un mécanisme d'audit propre au protocole MGCP entre les passerelles de jonction et le contrôleur MGC d'un système IPCablecom, afin de permettre à celui-ci de vérifier l'état des extrémités et des connexions et d'extraire d'une extrémité des capacités propres au protocole.

Deux commandes permettant d'assurer l'audit sont définies pour les passerelles de jonction:

AuditEndPoint: utilisée par le contrôleur MGC pour déterminer l'état d'une extrémité.

AuditConnection: utilisée par le contrôleur MGC pour obtenir des informations sur une connexion.

Une gestion du réseau, plus poussée que celle qui est obtenue au moyen de ces commandes, est généralement souhaitable, p. ex. en ce qui concerne des informations sur l'état de la passerelle de jonction comparé à celui des différentes extrémités. On prévoit que ces capacités seront prises en charge lors de l'utilisation du protocole simple de gestion de réseau (SNMP) et de la définition d'une base MIB pour la passerelle de jonction, sujets qui sortent du cadre de la présente Recommandation.

7.3.8.1 Commande AuditEndPoint

La commande AuditEndPoint peut être utilisée par le contrôleur MGC pour déterminer l'état d'une extrémité donnée.

```
{ ReturnCode
  [, EndPointIdList]
  [, NumEndPoints] } |
{ ReturnCode
  [, RequestedEvents]
  [, SignalRequests]
  [, RequestIdentifier]
  [, NotifiedEntity]
  [, ConnectionIdentifiers]
  [, DetectEvents]
  [, ObservedEvents]
  [, EventStates]
  [, MaxMGCPDatagram]              [, Capabilities] }
     ← AuditEndPoint(EndpointId
          [, RequestedInfo] |
          { [, SpecificEndpointID]
            [, MaxEndpointIDs] } )
```

Le paramètre **EndpointId** identifie l'extrémité qui fait l'objet d'un audit. La convention concernant le caractère de remplacement "un quelconque" NE DOIT PAS être employée. Une passerelle qui reçoit une commande AuditEndPoint avec la convention générique "un quelconque" DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

La convention concernant le caractère de remplacement "tous" peut être utilisée pour vérifier un groupe d'extrémités. Si cette convention est employée, la passerelle DOIT renvoyer une liste des identificateurs d'extrémité qui concordent avec le caractère de remplacement contenu dans le paramètre **EndPointIdList**. C'est simplement la liste des paramètres SpecificEndpointId – le paramètre RequestedInfo NE DOIT PAS être inclus dans ce cas. Le paramètre **MaxEndPointID** consiste en une valeur numérique qui indique le nombre maximal de paramètres EndpointId à renvoyer. Lorsque des extrémités supplémentaires existent, le paramètre de renvoi **NumEndPoints** DOIT être présent et indiquer le nombre total d'extrémités qui concordent avec le paramètre EndpointID spécifié. Afin d'extraire le bloc suivant de paramètres EndpointID, le paramètre **SpecificEndPointID** est fixé à la valeur de la dernière extrémité renvoyée du précédent paramètre EndpointIDList, et la commande est lancée.

Lorsque la convention concernant les caractères de remplacement n'est pas employée, le paramètre **RequestedInfo** (ne contenant éventuellement pas de valeur) décrit les informations qui sont demandées pour le paramètre EndpointId spécifié – les paramètres SpecificEnpointID et MaxEndpointID NE DOIVENT PAS être utilisés dans ce cas. Les informations suivantes propres aux extrémités peuvent alors faire l'objet d'un audit au moyen de cette commande:

> RequestedEvents, SignalRequests, RequestIdentifier, NotifiedEntity, ConnectionIdentifiers, DetectEvents, ObservedEvents, EventStates, VersionSupported, MaxMGCPDatagram et Capabilities.

Si une extrémité est interrogée sur un paramètre qu'elle ne prend pas en charge, l'extrémité NE DOIT PAS produire d'erreur; en revanche, le paramètre DOIT être omis de la réponse.

Si une extrémité est interrogée sur un paramètre qu'elle prend effectivement en charge mais pour lequel elle ne possède aucune valeur, cette extrémité NE DOIT PAS produire d'erreur; en revanche, le paramètre DOIT être inclus dans la réponse avec une valeur paramétrique vide.

Ce n'est qu'en cas de succès que la réponse AuditEndPoint DOIT, à son tour, comprendre des informations sur chacun des éléments pour lesquels des informations d'audit ont été demandées. Noter que les paramètres qui sont explicitement marqués "facultatif" ci-dessous et non pris en charge par l'extrémité sont omis dans la réponse:

- **RequestedEvents**: valeur effective du paramètre RequestedEvents que l'extrémité emploie, y compris la mesure associée à chaque événement. Les événements durables sont inclus dans la liste.
- **SignalRequests**: liste de signaux TO qui sont effectivement activés, de signaux OO qui sont effectivement "activés" pour l'extrémité (avec ou sans paramètre) et de signaux brefs en attente[17]. Les signaux TO qui sont arrivés à expiration et les signaux brefs effectivement produits ne sont pas compris. Il est fait état des signaux paramétrés avec les paramètres qu'ils ont utilisés.
- **RequestIdentifier**: identificateur de requête pour le dernier paramètre NotificationRequest reçu par l'extrémité (y compris la demande de notification imbriquée dans les primitives traitant la connexion). Si aucune demande de notification n'a été reçue, la valeur zéro sera renvoyée.
- **NotifiedEntity**: "entité notifiée" effective pour l'extrémité. Noter que la passerelle MG ne PEUT comprendre le nom de domaine de son Entité notifiée que si seul ce nom de domaine lui a été fourni via le paramètre NotifiedEntity d'un message ou acquittement TGCP. Le contrôleur MGC DEVRAIT accepter la valeur dans ce cas.
- **ConnectionIdentifiers**: liste des paramètres ConnectionIdentifiers séparés par des virgules concernant toutes les connexions qui existent effectivement pour l'extrémité spécifiée.

---

[17] Actuellement, il ne devrait pas y avoir de signaux brefs en attente.

- **DetectEvents**: valeur effective du paramètre DetectEvents que l'extrémité utilise. Les événements durables sont compris dans la liste.
- **ObservedEvents**: liste effective des événements observés pour l'extrémité.
- **EventStates**: pour les événements dont les états peuvent faire l'objet d'un audit, l'événement correspondant à l'état de l'extrémité est, p. ex. une prise de ligne lorsque le circuit multifréquence pour l'extrémité est effectivement pris. La définition des différents événements stipulera si l'événement concerné est dans un état pouvant faire l'objet d'un audit.
- **VersionSupported**: liste des versions de protocole prises en charge par l'extrémité.
- **MaxMGCPDatagram**: longueur maximale d'un datagramme MGCP en octets pris en charge par l'extrémité (voir § 8.5.3). La valeur exclut tout surdébit de couche inférieure. La prise en charge de ce paramètre est facultative. La longueur maximale par défaut du datagramme MGCP est présumée si une valeur n'est pas renvoyée.
- **Capabilities**: capacités de l'extrémité semblables à celles du paramètre LocalConnectionOptions et comprenant des paquetages d'événements et des modes de connexion. S'il est nécessaire de spécifier que certains paramètres, tels que la suppression des silences, ne sont compatibles qu'avec certains codecs, la passerelle renverra plusieurs ensembles de capacités.
  - **algorithme de compression** liste des codecs pris en charge. Les noms litéraux définis au § 7.5 (Tableau 3) de la spécification des codecs audio et vidéo IPCablecom (J.161) DOIVENT être utilisés. Les algorithmes de compression inconnus DEVRAIENT être ignorés s'ils sont reçus. Le reste des paramètres s'appliquera à tous les codecs spécifiés dans cette liste;
  - **période de mise en paquets** spécification d'une valeur unique ou d'une gamme de valeurs;
  - **largeur de bande** spécification d'une valeur unique ou d'une gamme de valeurs correspondant à la gamme des périodes de mise en paquets (dans l'hypothèse qu'il n'y ait pas de suppression des silences);
  - **compensation d'écho** indication de la prise en charge ou pas[18] de la compensation d'écho;
  - **suppression des silences** indication de la prise en charge ou pas de la suppression des silences;
  - **type de service** indication de la prise en charge ou pas du type de service;
  - **paquetages d'événements** liste des paquetages d'événements pris en charge. Le premier paquetage d'événements de la liste sera le paquetage par défaut;
  - **modes** liste des modes de connexion pris en charge;
  - **suite cryptographique RTP** liste d'algorithmes d'authentification et de chiffrement pris en charge pour le protocole RTP;

[18] Actuellement toutes les extrémités conformes au protocole TGCP doivent prendre en charge la compensation d'écho.

– **suite cryptographique RTCP** liste d'algorithmes d'authentification et de chiffrement pris en charge pour le protocole RTCP;
– **surveillance électronique** indication de la prise en charge ou pas de la surveillance électronique IPCablecom.

Le contrôleur MGC peut alors décider d'utiliser la commande AuditConnection pour obtenir plus d'informations sur les connexions.

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

Si aucune information n'a été demandée et que le paramètre EndpointId renvoie à un paramètre EndpointId valable entièrement spécifié, la passerelle renvoie simplement une réponse de réussite (code de renvoi 200 – Transaction exécutée normalement).

Il convient de noter que toutes les informations renvoyées correspondent simplement à un instantané. Les nouvelles commandes reçues, l'activité locale, etc. peuvent influer sur la plupart des informations susmentionnées. P. ex. l'état de prise de ligne peut changer avant que le contrôleur MGC reçoive ces informations.

7.3.8.2 Commande AuditConnection

L'audit des différentes connexions à une extrémité peut être réalisé au moyen de la commande AuditConnection.

```
ReturnCode
[, CallId]
[, NotifiedEntity]
[, LocalConnectionOptions]
[, Mode]
[, RemoteConnectionDescriptor]
[, LocalConnectionDescriptor]
[, ConnectionParameters]
  ← AuditConnection(EndpointId
         , ConnectionId
         [, RequestedInfo])
```

Le paramètre **EndpointId** identifie l'extrémité qui fait l'objet d'un audit. Les caractères de remplacement NE DOIVENT PAS être utilisés. Une passerelle qui reçoit une commande AuditConnection avec une convention générique DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse. Le paramètre **RequestedInfo** (ne contenant éventuellement pas de valeur) décrit les informations qui sont demandées pour le paramètre **ConnectionId** dans le paramètre EndpointId spécifié. Les informations de connexion suivantes peuvent faire l'objet d'un audit au moyen de cette commande:

CallId, NotifiedEntity, LocalConnectionOptions, Mode, ConnectionParameters, RemoteConnectionDescriptor, LocalConnectionDescriptor.

Si une extrémité est interrogée sur un paramètre de connexion qu'elle ne prend pas en charge, cette extrémité NE DOIT PAS produire d'erreur; en revanche, le paramètre DOIT être omis de la réponse.

Si une extrémité est interrogée sur un paramètre de connexion qu'elle prend effectivement en charge, mais pour lequel elle ne possède aucune valeur, cette extrémité NE DOIT PAS produire d'erreur; en revanche, le paramètre DOIT être inclus dans la réponse avec une valeur paramétrique vide.

Ce n'est qu'en cas de succès que la réponse AuditConnection DOIT, à son tour, comprendre des informations sur chacun des éléments pour lesquels des informations d'audit ont été demandées.

Noter que les paramètres qui sont explicitement marqués "facultatif" ci-dessous et non pris en charge par l'extrémité sont omis dans la réponse:

- **CallId** identificateur de l'appel auquel la connexion appartient;
- **NotifiedEntity** "entité notifiée" pour l'extrémité;
- **LocalConnectionOptions** paramètre LCO fourni pour la connexion;
- **Mode** mode de connexion effectif;
- **ConnectionParameters** paramètres de connexion effectifs pour la connexion;
- **LocalConnectionDescriptor** paramètre LCD que la passerelle a fourni pour la connexion dans une commande précédente;
- **RemoteConnectionDescriptor** dernier paramètre RCD fourni à la passerelle pour cette connexion dans une commande précédente CreateConnection ou ModifyConnection.

Le paramètre **ReturnCode** est un paramètre renvoyé par la passerelle. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

Lorsque aucune information n'est demandée, et que le paramètre EndpointId renvoie à une extrémité valable, la passerelle vérifie simplement que la connexion spécifiée existe et, si tel est le cas, renvoie une réponse favorable (code de renvoi 200 – transaction exécutée).

7.3.9 Redémarrage en cours

La commande RestartInProgress est utilisée par la passerelle pour signaler qu'une extrémité ou un groupe d'extrémités est mis hors service ou est remis en service.

```
ReturnCode
[, NotifiedEntity]
[, VersionSupported]
     ← RestartInProgress(EndpointId
           , RestartMethod
           [, RestartDelay])
```

Le paramètre **EndpointId** identifie les extrémités qui sont mises en service ou hors service. La convention concernant le caractère de remplacement "tous" peut être employée pour appliquer la commande à un groupe d'extrémités, p. ex. toutes les extrémités qui sont reliées à une interface spécifiée, ou même toutes les extrémités qui sont reliées à une passerelle donnée. La convention concernant le caractère de remplacement "un quelconque" NE DOIT PAS être utilisée. Un contrôleur MGC qui reçoit une commande Redémarrage en cours avec la convention générique "un quelconque" DOIT renvoyer une erreur (l'erreur renvoyée DEVRAIT être le code d'erreur 500 – la transaction n'a pas pu être exécutée parce que l'extrémité est inconnue) en réponse.

Le paramètre RestartMethod spécifie la méthode de redémarrage:

- la méthode de redémarrage "progressif" spécifie que l'extrémité ou les extrémités spécifiées seront mises hors service après le "délai de redémarrage" spécifié. Les connexions établies ne sont pas encore touchées, mais le contrôleur MGC devrait s'abstenir d'établir de nouvelles connexions et devrait essayer de réduire progressivement et assez rapidement le nombre de connexions existantes. A l'expiration du délai de redémarrage, la passerelle MG devrait envoyer un nouveau message RSIP avec une méthode de redémarrage "forcé". Ce message va explicitement indiquer au contrôleur MGC que les extrémités sont maintenant hors service;

- la méthode de redémarrage "progressif annulé" spécifie qu'une passerelle procède à l'annulation de la méthode de redémarrage "progressif" précédemment mise en œuvre aux extrémités considérées. Sitôt cette commande émise, la passerelle autorisera immédiatement l'établissement de nouvelles connexions à ces extrémités;
- la méthode de redémarrage "forcé" spécifie que les extrémités spécifiées sont mises hors services de manière brusque. Les connexions établies, s'il y en a, sont perdues;
- la méthode de "redémarrage" spécifie que le service sera rétabli aux extrémités après le "délai de redémarrage" spécifié. Aucune connexion n'est effectivement établie aux extrémités;
- la méthode de "déconnexion" spécifie que l'extrémité a été déconnectée et tente maintenant d'établir la connexion. Le "délai de redémarrage" spécifie le nombre de secondes pendant lesquelles l'extrémité a été déconnectée. Les connexions établies ne sont pas touchées.

Le paramètre facultatif "délai de redémarrage" s'exprime sous la forme d'un nombre de secondes. Si ce nombre fait défaut, la valeur du délai devrait être considérée comme étant nulle. Dans le cas de la méthode "progressive", un délai nul indique que le contrôleur MGC devrait simplement attendre que les connexions existantes prennent fin naturellement, sans établir de nouvelles connexions. Le délai de redémarrage est toujours considéré comme étant nul dans le cas des méthodes "forcée" et "progressif annulé" et donc le paramètre "délai de redémarrage" NE DOIT PAS être utilisé dans le cas des méthodes "forcé" et "progressif annulé". Un délai de redémarrage nul pour la méthode de "redémarrage" indique que le service a déjà été rétabli. Cela se produira généralement après le démarrage ou le redémarrage de la passerelle. Afin d'atténuer les effets causés par le changement d'adresse IP d'une passerelle, le contrôleur MGC PEUT vouloir trancher la question du nom de domaine de la passerelle en interrogeant le système DNS sans tenir compte de la durée de vie de l'enregistrement effectif des ressources pour la passerelle ayant fait l'objet d'un redémarrage.

Les passerelles de jonction DEVRAIENT envoyer, par courtoisie envers le contrôleur MGC, un message RestartInProgress "progressif" ou "forcé" lorsqu'elles sont mises hors service, p. ex. lors de leur fermeture ou de leur mise hors service par un système de gestion de réseau, même si le contrôleur MGC ne peut pas compter toujours recevoir de tels messages. Une passerelle de jonction DOIT cependant envoyer un message de redémarrage en cours "forcé" ou "progressif" lorsqu'une extrémité est mise hors service au moyen du processus de préconfiguration ou sur détection d'une défaillance de la ressource de transport de l'extrémité (p. ex. une perte de signal, la réception d'une indication d'alarme distante, etc.). Les passerelles de jonction DOIVENT envoyer à leur contrôleur MGC un message RestartInProgress de "redémarrage" avec un délai nul lorsqu'elles sont remises en service, conformément à la procédure de redémarrage spécifiée au § 7.4.3.5 – les contrôleurs MGC peuvent compter recevoir ce message. En outre, les passerelles de jonction DOIVENT envoyer à leur "entité notifiée" effective un message RestartInProgress de "déconnexion", conformément à la procédure de "déconnexion" spécifiée au § 7.4.3.6. Le paramètre "délai de redémarrage" NE DOIT PAS être utilisé avec les méthodes de redémarrage "forcé" et "progressif annulé".

Le message RestartInProgress sera envoyé à "l'entité notifiée" effective pour le paramètre EndpointId concerné. Il est escompté qu'un contrôleur MGC par défaut, à savoir "l'entité notifiée", a été configuré pour chacune des extrémités de manière qu'après un redémarrage ce contrôleur MGC par défaut puisse être "l'entité notifiée" pour chacune de ces extrémités. Les passerelles de jonction DOIVENT pleinement mettre à profit les caractères de remplacement de manière à minimiser le nombre de messages RestartInProgress produits lorsque des extrémités multiples d'une passerelle redémarrent et que ces extrémités sont gérées par le même contrôleur MGC.

Le paramètre **ReturnCode** est un paramètre renvoyé par le contrôleur MGC. Il indique le résultat de la commande et comporte un nombre entier (voir § 7.5) suivi en option d'un commentaire.

Un paramètre **NotifiedEntity** PEUT en outre être renvoyé avec la réponse du contrôleur MGC à la commande RestartInProgress. Ce renvoi ne devrait normalement avoir lieu qu'en réponse à un message "redémarrage" ou "déconnecté" (voir également les § 7.4.3.5 et 7.4.3.6). Si un paramètre NotifiedEntity a été inclus dans la réponse renvoyée, il spécifie une nouvelle "entité notifiée" pour l'extrémité (les extrémités) – cette opération ne DEVRAIT être effectuée qu'avec le code d'erreur en réponse 521 (réacheminement d'extrémité). Noter que le comportement précédent de renvoi d'un paramètre NotifiedEntity dans la réponse n'est défini que pour les réponses de type RestartInProgress et ne DEVRAIT ne pas avoir lieu pour les réponses à d'autres commandes. Tout autre comportement est indéfini:

- si la réponse indiquait la réussite (code de renvoi 200 – transaction exécutée), la procédure de redémarrage est achevée et le paramètre NotifiedEntity renvoyé est la nouvelle "entité notifiée" pour l'extrémité ou les extrémités;
- si la réponse provenant du contrôleur MGC a signalé une erreur, la procédure de redémarrage n'est pas encore achevée. Si la réponse était un code 521 (réacheminement d'extrémité), la réponse DOIT inclure un paramètre NotifiedEntity qui spécifie la nouvelle "entité notifiée" pour l'extrémité ou les extrémités, et qui DOIT en conséquence être utilisé lorsque la procédure de redémarrage est réessayée (en tant que nouvelle transaction).

Dans le cas des méthode "redémarrage" et "déconnecté", le redémarrage en question DOIT faire l'objet d'un nouvel essai chaque fois que le Contrôleur de passerelle média renvoie un code d'erreur transitoire (4xx), tandis qu'il DEVRAIT faire l'objet d'un nouvel essai pour toute autre méthode de redémarrage. Il est RECOMMANDE que tout type de redémarrage soit terminé si code d'erreur permanente (5xx) est renvoyé, sauf pour le code d'erreur 521, comme spécifié ci-dessus.

Finalement, un paramètre **VersionSupported** comprenant une liste des versions prises en charge peut être renvoyé si la réponse a indiqué une incompatibilité de version (code d'erreur 528).

7.4 Etats, situations de reprise sur défaillance et situations de concurrence

Afin de mettre en œuvre une signalisation d'appel appropriée, le contrôleur MGC doit suivre l'état de l'extrémité et la passerelle doit s'assurer que les événements sont correctement notifiés au contrôleur MGC. Des situations particulières peuvent exister lorsque la passerelle ou le contrôleur MGC font l'objet d'un redémarrage: la passerelle peut devoir être réacheminée vers un nouveau contrôleur MGC au cours des procédures de reprise sur défaillance; de même, le contrôleur MGC peut devoir prendre des mesures spéciales lorsque la passerelle est mise hors ligne ou fait l'objet d'un redémarrage.

7.4.1 Récapitulatif des points essentiels

Comme il est mentionné au § 7.1.4, les contrôleurs MGC sont identifiés par leur nom de domaine et chaque extrémité possède, à tout moment donné, une et une seule "entité notifiée" qui lui est associée. Dans le présent paragraphe, les points essentiels qui sont d'une importance particulière en ce qui concerne la fiabilité et la reprise sur défaillance dans le cadre du protocole MGCP sont récapitulés:

- un contrôleur MGC est identifié par son nom de domaine et non par ses adresses dans le réseau; plusieurs adresses dans le réseau peuvent être associées à un nom de domaine;
- une extrémité dispose à tout moment donné d'un et d'un seul contrôleur MGC qui lui est associé. Le contrôleur MGC associé à une extrémité est la valeur effective de "l'entité notifiée";
- la valeur attribuée initialement à "l'entité notifiée" est une valeur qui est fixée par la préconfiguration. Lorsqu'une commande comprenant un paramètre NotifiedEntity est reçue à une extrémité, ainsi que des noms d'extrémité comportant des caractères de remplacement, la valeur attribuée à "l'entité notifiée" est celle qui est spécifiée. Si aucune valeur n'est attribuée à "l'entité notifiée" pour une extrémité ou que cette valeur n'ait pas été

fixée explicitement[19], l'adresse par défaut de "l'entité notifiée" est l'adresse de l'expéditeur de la dernière commande de traitement de la connexion ou demande de notification reçue pour l'extrémité. Dans ce cas, le contrôleur MGC sera donc identifié par son adresse dans le réseau, ce qui ne DEVRAIT se faire qu'exceptionnellement;

- les réponses aux commandes sont toujours envoyées à l'adresses de leur émetteur, sans qu'il soit tenu compte de "l'entité notifiée" effective. Lorsqu'un message Notify doit accompagner la réponse, le datagramme est encore envoyé à l'adresse de l'expéditeur de la nouvelle commande reçue, sans qu'il soit tenu compte du paramètre NotifiedEntity pour l'une quelconque des commandes;
- lorsque "l'entité notifiée" renvoie à un nom de domaine qui se décompose en plusieurs adresses IP, les extrémités sont en mesure de passer d'une adresse à l'autre, sans toutefois pouvoir modifier de leur propre initiative le nom de domaine de "l'entité notifiée". Un contrôleur MGC peut toutefois leur ordonner d'effectuer le changement en leur attribuant une nouvelle "entité notifiée";
- si un contrôleur MGC n'est plus disponible, les extrémités qu'il gère seront éventuellement "déconnectées". La seule façon pour que ces extrémités soient connectées à nouveau est que soit le contrôleur MGC défaillant redevienne disponible soit qu'un autre contrôleur MGC (de secours) prenne contact avec les extrémités touchées au moyen d'une nouvelle "entité notifiée";
- lorsqu'un autre contrôleur MGC (de secours) a repris la commande d'un groupe d'extrémités, on suppose que le contrôleur MGC défaillant communiquera avec le contrôleur MGC de secours et se synchronisera avec lui afin que la commande des extrémités touchées puisse lui être repassée, si cela est souhaité. Autrement, le contrôleur MGC défaillant pourrait maintenant simplement devenir le contrôleur MGC de secours.

Il convient de noter que la résolution des différends en ce qui concerne la passation entre différents contrôleurs MGC n'est pas configurée – cela étant strictement fondé sur le fait que les contrôleurs MGC savent ce qu'ils font et qu'ils communiquent entre eux (même si le paramètre AuditEndpoint peut être employé pour connaître "l'entité notifiée" effective).

7.4.2 Retransmission et détection d'associations perdues

Le protocole MGCP est structuré comme un ensemble de transactions, chacune d'elles étant constituée d'une commande et d'une réponse. Les messages relatifs à ce protocole, transportés à l'aide du protocole UDP, peuvent subir des pertes. En l'absence de réponse dans les délais (voir § 8.5), les commandes sont répétées. Les passerelles DOIVENT garder en mémoire une liste des réponses qu'elles ont envoyées au cours des récentes transactions et une liste des transactions qui sont en cours d'exécution. Le terme "Récentes transactions" est défini ici par la valeur Tt_{hist} qui spécifie le nombre de secondes pendant lesquelles les réponses à d'anciennes transactions doivent être conservées. La valeur par défaut de Tt_{hist} est égale à 30 s.

Les identificateurs de transaction des commandes entrantes sont d'abord comparés aux identificateurs de transaction des réponses récentes. Lorsqu'une concordance est trouvée, la passerelle n'effectue pas la transaction, mais reproduit simplement l'ancienne réponse. Si aucune concordance avec une transaction à laquelle il a été répondu précédemment n'est trouvée, l'identificateur de transaction de la commande entrante est comparé à la liste des transactions dont l'exécution n'est pas encore terminée. Si une concordance est trouvée, la passerelle n'effectue pas la transaction. La suite du traitement dépend de la commande en question. S'il s'agit d'une commande CreateConnection ou ModifyConnection, la passerelle DOIT envoyer une réponse provisoire. S'il

[19] Cela pourrait p. ex. se produire lorsque aucune valeur n'est spécifiée pour le paramètre NotifiedEntry.

s'agit de toute autre commande, celle-ci est simplement négligée. Dans un cas comme dans l'autre, une réponse est fournie lorsque l'exécution de la commande est achevée.

Ce mécanisme de répétition est utilisé pour prévenir quatre types d'erreur possibles:

- erreurs de transmission, p. ex. lorsqu'un paquet est perdu en raison du bruit sur la ligne ou de l'encombrement dans une file d'attente;
- défaillance d'un composant, p. ex. lorsqu'une interface de contrôleur MGC n'est plus disponible;
- défaillance d'un contrôleur MGC, p. ex. lorsque toutes ses interfaces deviennent indisponibles;
- reprise sur défaillance, lorsqu'un nouveau contrôleur MGC "prend le relais" de manière transparente.

Les éléments devraient être en mesure d'évaluer à partir de l'historique le taux de perte de paquets. Dans un système correctement configuré, ce taux de perte devrait être très faible, généralement moins de 1% en moyenne. Si un contrôleur MGC ou une passerelle doit reproduire un message plus de quelques fois, il est tout à fait légitime de penser qu'il s'est produit autre chose qu'une erreur de transmission. P. ex. si le taux de perte est uniformément réparti et s'élève à 1%, la probabilité que cinq tentatives consécutives de transmission échouent est de 1 sur 100 milliards, événement qui devrait se produire moins d'une fois tous les dix jours dans le cas d'un contrôleur MGC qui effectue 1000 transactions par seconde. (En effet, le nombre de répétitions qui est considéré comme étant excessif devrait être fonction du taux habituel de perte de paquets.) Lorsque les erreurs ne sont pas réparties uniformément, la probabilité d'échecs consécutifs peut être un peu plus grande. Il convient de noter que le "seuil de suspicion", que nous nommerons "Max1", est normalement inférieur au "seuil de déconnexion", que nous nommerons "Max2", et qu'on DOIT lui attribuer une valeur plus élevée.

L'algorithme de retransmission MGCP est illustré dans la Figure 2 et expliqué plus en détail ci-après.

Un algorithme classique de retransmission consisterait simplement en un comptage du nombre de répétitions successives et en la conclusion que l'association est rompue après que le paquet a été retransmis un nombre excessif (généralement compris entre 7 et 11) de fois. Afin de tenir compte de la possibilité de l'existence d'un "reprise sur défaillance" non détecté ou en cours, nous modifions comme suit l'algorithme classique:

- la passerelle DOIT toujours vérifier la présence d'un nouveau contrôleur MGC. Elle peut en être informée:
  - en recevant une commande où le paramètre NotifiedEntity indique un nouveau contrôleur MGC;
  - ou en recevant une réponse de réacheminement vers un nouveau contrôleur MGC;
- si un nouveau contrôleur MGC est détecté, la passerelle DOIT acheminer les retransmissions des commandes en suspens pour l'extrémité ou les extrémités vers ce nouveau contrôleur MGC. Les réponses aux nouvelles ou anciennes commandes sont toujours transmises à l'adresse de l'expéditeur de la commande;
- avant toute retransmission, on vérifie que le temps qui s'est écoulé depuis l'envoi du datagramme initial n'est pas supérieur à Ts_{max}. Si plus de temps que cette valeur s'est écoulé, les retransmissions DOIVENT cesser. Si une durée supérieure à $2*Tt_{hist}$ s'est écoulée, alors l'extrémité se déconnecte;
- si le nombre de retransmissions vers ce contrôleur MGC est égal à "Max1", la passerelle PEUT activement interroger le serveur de noms afin de détecter les éventuels changements d'interfaces des contrôleurs MGC, sans qu'il soit tenu compte de la durée de vie (TTL) associée à l'enregistrement dans le système DNS;

- la passerelle peut avoir pris connaissance de plusieurs adresses IP pour le contrôleur MGC. Si le nombre de retransmissions pour une adresse IP est supérieur à "Max1" et inférieur à "Max2", et qu'il y ait d'autres adresses IP qui n'ont pas été essayées, la passerelle DOIT réacheminer les retransmissions vers les autres adresses restantes de sa liste locale. De même, la réception de notifications de réseau explicites comme réseau en protocole ICMP, serveur local, protocole, ou port inatteignable DEVRAIT conduire la passerelle à essayer des adresses de remplacement (compte dûment tenu d'éventuelles question de sécurité);
- s'il ne reste plus d'autres interfaces à essayer, et que le nombre de retransmissions soit égal à "Max2", la passerelle DEVRAIT consulter le système DNS encore une fois pour voir si d'autres interfaces sont devenues disponibles. Si ce n'est toujours pas le cas, les retransmissions DOIVENT cesser. Si une durée supérieure à $2*Tt_{hist}$ s'est écoulée, l'extrémité devient déconnectée;
- une fois qu'une extrémité se déconnecte, la suite du traitement dépend de la question de savoir si la perte d'association a été détectée par la passerelle ou par le contrôleur de passerelle média, comme suit:
  - la passerelle DOIT lancer la procédure "déconnecté" comme spécifié dans le § 7.4.3.6;
  - le contrôleur de passerelle média NE DOIT PAS essayer d'utiliser l'extrémité pour d'éventuels nouveaux appels avant que la connexité ait été restaurée. Par ailleurs, le contrôleur de passerelle média DOIT implémenter un algorithme afin de détecter le moment où la connexité avec l'extrémité est ultérieurement restaurée (p. ex. dès réception d'une réponse à une commande périodique AuditEndPoint). Quand la connexité jusqu'à l'extrémité est restaurée et si aucune autre condition n'existe qui empêche l'extrémité de prendre en charge des appels, le contrôleur MGC DOIT veiller à ce que l'extrémité puisse être utilisée pour de nouveaux appels sans nécessiter de quelconque intervention manuelle.

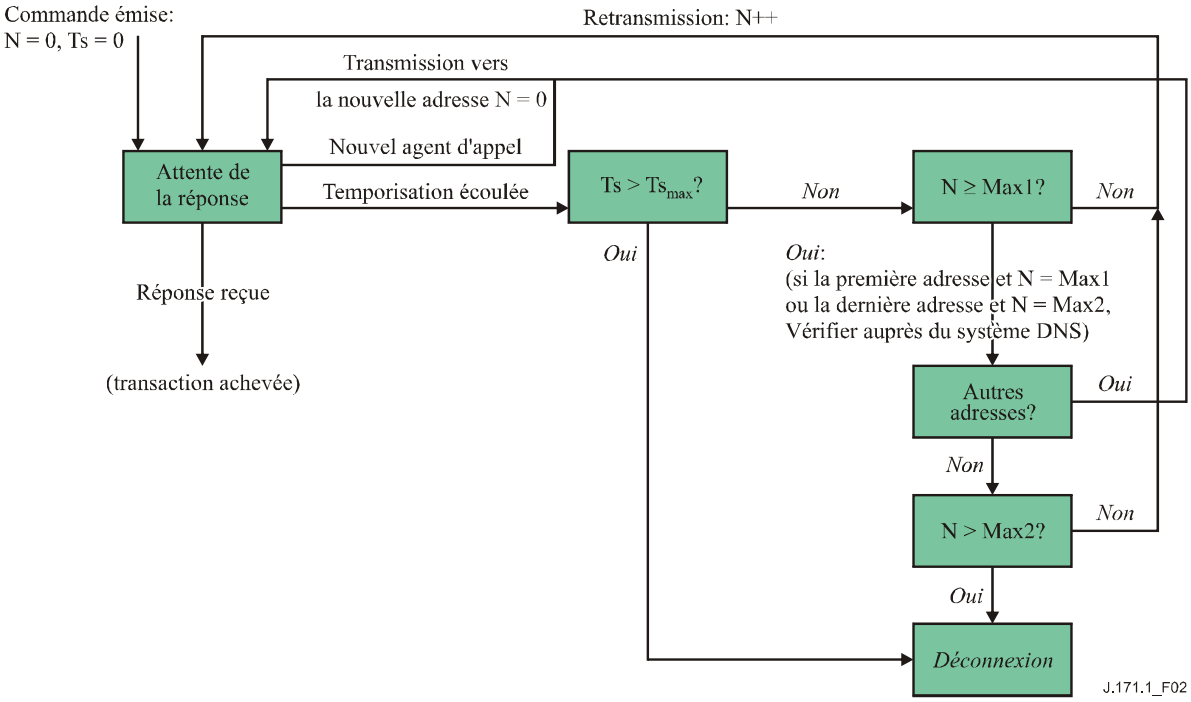


Figure 2/J.171.1 – Algorithme de retransmission

Afin que l'adaptation automatique à la charge du réseau puisse se faire, le protocole MGCP spécifie des temporisations qui augmentent exponentiellement (voir § 8.5.2). Si la temporisation est fixée initialement à 200 ms, la perte d'une cinquième retransmission sera détectée après environ 6 s. Cela constitue sans doute un temps d'attente acceptable pour détecter un reprise sur défaillance. Les retransmissions devraient être poursuivies après ce délai, non seulement pour surmonter peut-être des problèmes transitoires de connexité, mais également afin d'accorder un peu plus de temps à la reprise sur défaillance – Un temps d'attente total de 30 s est sans doute acceptable.

Il est cependant important que le délai de retransmission maximal soit borné. Avant toute retransmission, on vérifie que le temps (Ts) qui s'est écoulé depuis l'envoi du datagramme initial n'est pas plus grand que Ts_{max}. Si une durée supérieure à Ts_{max} s'est écoulée, les retransmissions DOIVENT cesser. Quand Ts_{max} a expiré, ou que toutes les retransmissions à toutes les adresses IP connues ont été envoyées, il y a une pause avant de déclarer l'extrémité déconnectée. Cette pause représente une période pendant laquelle la seule action consiste à attendre une réponse à partir de l'une quelconque des récentes retransmissions. La période en mode passif dure pendant ce qui reste de deux fois l'espérance de vie de la transaction originale ($2*Tt_{hist}$). Ce laps de stabilisation permet à toutes les transactions actives de se terminer ou d'arriver à expiration avant que l'extrémité soit déclarée déconnectée. Cela contribue à garantir que chaque redémarrage de l'extrémité intervient à partir d'un état correct et initial. Si une durée supérieure à $2 * Tt_{hist}$ s'est écoulée, l'extrémité devient déconnectée. La valeur Ts_{max} est associée à la valeur Tt_{hist}, laquelle DOIT être supérieure ou égale à Ts_{max} plus le temps de propagation maximal dans le réseau, Tp_{max}.

En d'autres termes, la relation ci-après DOIT être satisfaite pour empêcher que les commandes retransmises soient exécutées plus d'une seule fois: $Tt_{hist} \geq Ts_{max} + Tp_{max}$.

La valeur par défaut de Ts_{max} est égale à 20 s. Donc, si on suppose que le délai de propagation maximal est de 10 s, les réponses aux anciennes transactions doivent être conservées pendant une durée d'au moins 30 s. L'importance d'un accord entre l'expéditeur et le récepteur concernant ces valeurs ne peut pas être surestimée.

La valeur par défaut de Max1 est égale à 5 retransmissions tandis que celle de Max2 s'élève à 7 retransmissions. Ces deux valeurs peuvent être modifiées au cours du processus de préconfiguration.

En outre, le processus de préconfiguration DOIT être en mesure de désactiver une requête ou les deux requêtes Max1 et Max2 auprès du système DNS.

7.4.3 Situations de concurrence

Le présent paragraphe décrit comment les situations de concurrence sont traitées dans le protocole MGCP.

Premièrement, le protocole MGCP traite les situations de concurrence au moyen de la notion de "liste de quarantaine" où des événements sont mis en quarantaine et au moyen de la détection explicite de la désynchronisation, p. ex. pour des états de prise de ligne non concordants à cause d'une double-prise à une extrémité.

Deuxièmement, le protocole MGCP ne suppose pas que le mécanisme de transport conservera l'ordre des commandes et des réponses. Cela peut conduire à des situations de concurrence auxquelles il peut être remédié en adaptant le comportement du contrôleur MGC de manière à ordonner correctement les commandes.

Finalement, dans certains cas, des passerelles en grand nombre peuvent décider de recommencer à fonctionner en même temps. Cela peut se produire, p. ex. lorsque le courant est interrompu dans une zone ou que la transmission ne peut plus se faire pendant un tremblement de terre ou une tempête de neige. Lorsque le courant et la capacité de transmission sont rétablis, de nombreuses passerelles peuvent décider d'envoyer simultanément des commandes RestartInProgress, ce qui pourrait conduire à un fonctionnement très instable s'il n'est pas bien contrôlé.

7.4.3.1 Liste de quarantaine

Les passerelles commandées par le protocole MGCP recevront des demandes de notification leur enjoignant de surveiller une liste d'événements. Les éléments du protocole qui déterminent comment ces événements seront traités sont les listes "d'événements demandés" et "d'événements détectés".

Lorsque l'extrémité est initialisée, la liste des événements demandés ne comporte que des événements durables pour cette extrémité. Après la réception d'une commande, la passerelle commence à observer au niveau de cette extrémité les occurrences des événements qui sont mentionnés dans la liste, y compris celles des événements durables.

Les événements sont examinés au fur et à mesure de leur apparition. La mesure qui en découle est déterminée par le paramètre "mesure" qui est associé à l'événement de la liste des événements demandés. Les événements qui sont définis comme étant "recueillis" sont recueillis dans une liste d'événements observés. Cela sera poursuivi jusqu'à ce qu'un événement produit déclenche une commande Notify qui sera envoyée à "l'entité notifiée".

A ce moment, la passerelle transmettra la commande Notify et placera l'extrémité dans un "état de notification". Aussi longtemps que cette extrémité est dans cet "état de notification", les événements qui sont détectés à cette extrémité sont entreposés dans un tampon de "quarantaine" pour être traités ultérieurement. Les événements sont, dans un certain sens, mis en "quarantaine". Les événements détectés sont des événements spécifiés par la réunion logique du paramètre RequestedEvents avec le paramètre DetectEvents reçu le plus récemment ou, lorsque celui-ci n'a pas été reçu, par les événements auxquels renvoie le paramètre RequestedEvents. Les événements durables sont également détectés.

L'extrémité quitte "l'état de notification" lors de la réception d'une réponse à la commande Notify[20]. La commande Notify peut être retransmise dans "l'état de notification", comme spécifié au § 7.4.2.

Si l'extrémité est ou devient déconnecté (voir § 7.4.2) pendant cette opération, il ne sera jamais reçu de réponse à la commande Notify, qui est alors perdue et n'est donc plus considérée comme étant en cours, alors que l'extrémité est toujours dans "l'état de notification". Si cela devait arriver, l'achèvement de la procédure de déconnexion spécifiée au § 7.4.3.6 conduirait alors l'extrémité à sortir de "l'état de notification".

Lorsque l'extrémité quitte "l'état de notification", elle remet à zéro la valeur attribuée à l'ensemble des événements qui ont été observés à son niveau. A partir de ce moment, le comportement de la passerelle dépend de la valeur du paramètre de Traitement de quarantaine dans la commande Demande de notification de déclenchement.

Si l'agent d'appel/le contrôleur de passerelle média a spécifié un paramètre QuarantineHandling de valeur "step" ou si aucune valeur n'a été spécifiée, la passerelle utilisera le "mode perpétuel" qui implique que la passerelle DOIT recevoir une nouvelle commande NotificationRequest après avoir envoyé une commande Notify. Jusqu'à ce que cela arrive, l'extrémité est dans un "état perpétuel", et les événements qui surviennent et vont être détectés sont simplement emmagasinés dans la mémoire tampon de quarantaine. Les événements à mettre en quarantaine sont les mêmes que dans "l'état de notification". Une fois que la nouvelle Demande de notification est reçue et exécutée avec succès, l'extrémité sort de "l'état perpétuel".

Si toutefois l'agent d'appel/le contrôleur de passerelle média a spécifié un paramètre QuarantineHandling de valeur "loop" (c'est-à-dire le mode "boucle"), on procédera comme suit. Lorsque la passerelle sort de "l'état de notification", elle remet à zéro la liste des événements observés de l'extrémité et commence à traiter la liste des événements mis en quarantaine, en

[20] Il convient de noter que la mesure Notify ne peut pas être combinée avec une demande NotificationRequest incorporée.

utilisant la liste déjà reçue des événements demandés. Lors du traitement de ces événements, la passerelle peut en rencontrer un qui déclenche une commande Notify à envoyer. Si c'est le cas, la passerelle peut adopter un des deux comportements suivants:

– elle peut immédiatement transmettre une commande Notify qui va rapporter tous les événements qui étaient accumulés dans la liste des événements observés jusqu'à l'événement de déclenchement inclus, en laissant les événements non traités dans la mémoire tampon de quarantaine;

– elle peut tenter de vider la mémoire tampon de quarantaine et transmettre une commande Notify unique rapportant plusieurs ensembles d'événements. Les événements qui suivent le dernier événement de déclenchement DOIVENT être laissés dans la mémoire tampon de quarantaine.

Si la passerelle transmet une commande Notify, l'extrémité va ré-entrer et rester dans "l'état de notification" jusqu'à réception de l'accusé de réception (comme décrit ci-dessus). Si la passerelle ne trouve pas d'événement de quarantaine qui déclenche une commande Notify, elle place l'extrémité en état normal. Les événements sont alors traités comme ils viennent, exactement de la même façon que si l'on venait de recevoir une commande NotificationRequest.

Une passerelle peut recevoir à tout moment une nouvelle commande NotificationRequest pour l'extrémité, y compris dans le cas où celle-ci est déconnectée, ce qui aura également pour effet de sortir l'extrémité de "l'état de notification" en supposant que la commande NotificationRequest s'exécute avec succès. L'activation d'une commande NotificationRequest intégrée est vue ici aussi comme la réception d'une nouvelle commande NotificationRequest, sauf que la liste courante des événements observés reste inchangée plutôt que d'être traitée de nouveau.

Lorsqu'une nouvelle demande NotificationRequest est reçue dans "l'état de notification", la passerelle DEVRAIT essayer de remettre la commande Notify en suspens (noter qu'un message Notify qui avait été perdu à cause d'une déconnexion n'est plus considéré comme étant en suspens) avant une réponse de réussite à la nouvelle demande NotificationRequest. La passerelle fait cela en utilisant la fonctionnalité "de portage" du protocole et en classant les messages (commandes et réponses) à envoyer par ordre d'arrivée en commençant par le plus ancien. Les messages seront ensuite envoyés dans un paquet unique à l'expéditeur de la commande NotificationRequest, quelles que soient l'expéditeur et "l'entité notifiée" pour l'ancienne et la nouvelle commande. Les étapes mises en jeu sont les suivantes:

1) la passerelle construit un message qui transporte dans un paquet unique une répétition de l'ancienne commande Notify en suspens et la réponse à la nouvelle commande NotificationRequest;

2) l'extrémité est ensuite sortie de "l'état de notification" sans attendre la réponse à la commande Notify;

3) une copie de la commande Notify en suspens est conservée jusqu'à ce qu'une réponse soit reçue. Si une temporisation a lieu, la commande Notify sera répétée dans un paquet qui transportera également une répétition de la réponse à la commande NotificationRequest:

   • si le paquet transportant la réponse à la commande NotificationRequest est perdu, l'agent d'appel/contrôleur MGC retransmettra la commande NotificationRequest. La passerelle répondra à cette répétition en retransmettant dans un unique paquet la commande Notify en suspens et la réponse à la commande NotificationRequest – Ce datagramme sera envoyé à l'expéditeur de la commande NotificationRequest;

- les commandes Notify pour une extrémité donnée DOIVENT être remis dans l'ordre de leur envoi. Si la passerelle doit transmettre une nouvelle commande Notify avant qu'une réponse à la précédente commande Notify soit reçue, elle construit un paquet qui transporte par portage une répétition de l'ancienne commande Notify, une répétition de la réponse à la dernière commande NotificationRequest, et une nouvelle commande Notify – Ce datagramme sera envoyé à "l'entité notifiée" effective.

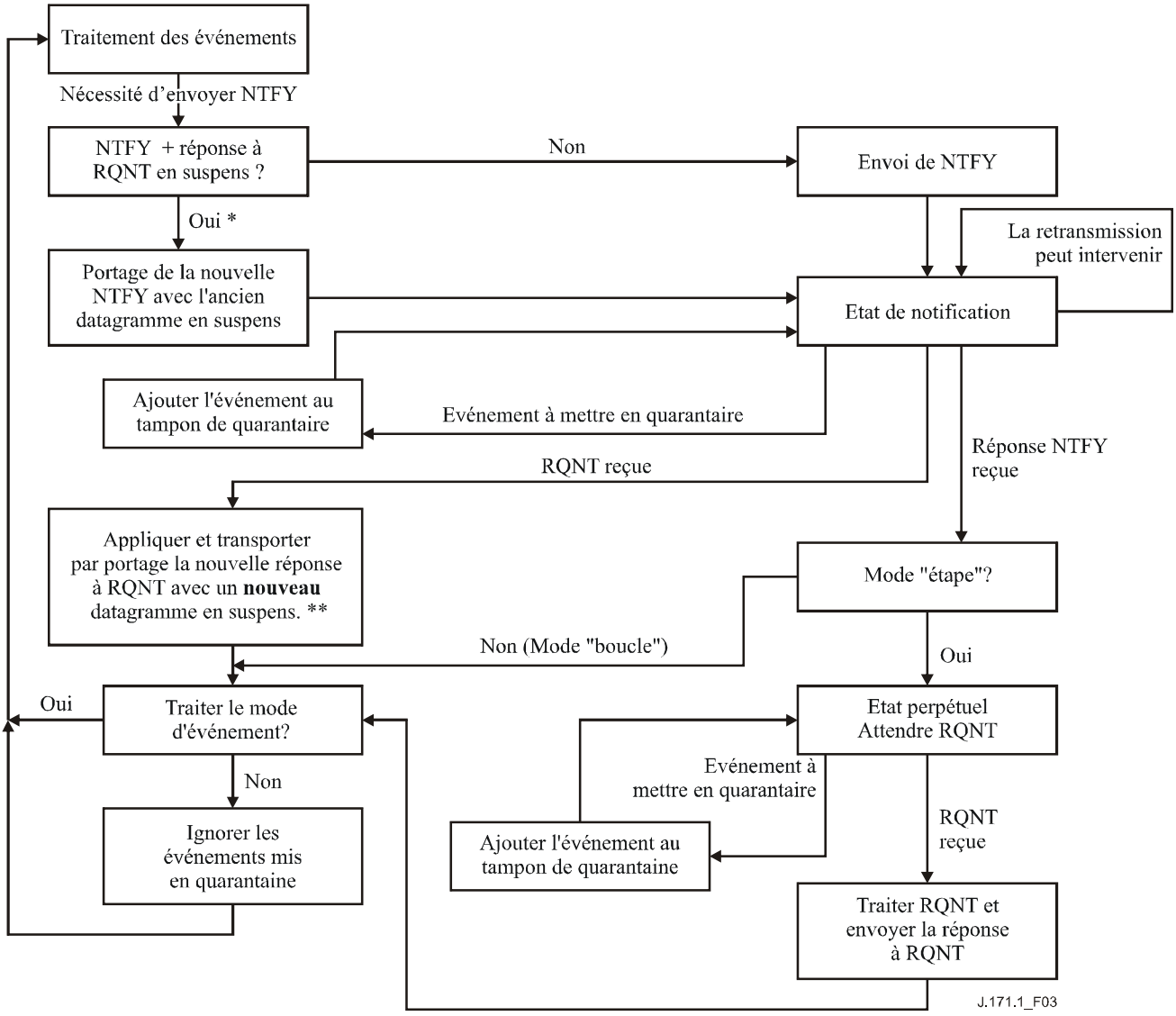
Après la réception d'une commande NotificationRequest, la liste des "événements demandés" est remplacée par les paramètres nouvellement reçus. De plus, lorsque la commande NotificationRequest a été reçue dans "l'état de notification", la liste des "événements observés" est réinitialisée par remise à zéro. Le comportement ultérieur est alors conditionné par la valeur du paramètre QuarantineHandling. Ce paramètre peut spécifier que des événements mis en quarantaine et observés (ce qui dans ce cas correspond à une liste vide) doivent être ignorés, auquel cas tous les événements mis en quarantaine et observés sont rejetés. Si, au contraire, ce paramètre stipule que les événements en quarantaine et observés devraient être traités, la passerelle commencera à traiter la liste des événements en quarantaine et observés en employant la liste nouvellement reçue des "événements demandés". Au cours du traitement de ces événements, la passerelle peut rencontrer un événement qui déclenche l'envoi d'une commande Notify. Si tel est le cas, la passerelle transmettra immédiatement une commande Notify qui signalera tous les événements ayant été recueillis dans la liste des "événements observés" jusqu'à l'événement déclencheur et y compris celui-ci, en reléguant les événements non traités dans le tampon de quarantaine. L'extrémité retourne ensuite dans "l'état de notification".

Une nouvelle demande de notification peut être reçue pendant que la passerelle a accumulé des événements conformément aux précédentes demandes de notification, mais n'a pas encore détecté d'événement déclencheur de notification. Le traitement d'événements non encore notifiés est déterminé, comme pour les événements mis en quarantaine, par les paramètres de traitement de quarantaine:

- si le paramètre de traitement de quarantaine spécifie que les événements de quarantaine doivent être ignorés, la liste des événements observés est simplement remise à zéro;
- si le paramètre de traitement de quarantaine spécifie que les événements de quarantaine doivent être traités, la liste des événements observés est transférée à la liste des événements mis en quarantaine. La liste des événements observés est alors remise à zéro et la liste des événements mis en quarantaine est traitée. La seule exception est l'activation d'une Demande de notification intégrée. Dans ce cas, la liste des événements observés reste inchangée plutôt que d'être traitée de nouveau.

La procédure décrite ci-dessus s'applique à toutes les formes de demandes de notification, qu'elles fassent partie d'une commande de traitement d'une connexion ou qu'elles soient fournies en tant que commande NotificationRequest. Les commandes de traitement de connexion qui ne comportent pas de demande de notification ne sont pas affectées par la procédure susmentionnée et n'affectent pas non plus cette procédure.

La Figure 3 illustre la procédure spécifiée ci-dessus, toutes les transactions étant supposées être exécutées avec succès:



* Cette branche décisionnelle est suivie si la passerelle a besoin d'envoyer un nouveau message Notify pendant qu'elle attend une réponse à un précédent message Notify sur la même extrémité. Cela pourrait se produire à la suite de la réception d'une nouvelle demande RQNT dans "l'état de notification", comme décrit dans le texte accompagnant le diagramme.

** Le "nouveau diagramme en attente" se rapporte au datagramme contenant le message Notify en attente, éventuellement porté par des messages Notify et par des réponses au message RQNT supplémentaires, qui est en cours de retransmission dans "l'état de notification" quand le nouveau message RQNT a été reçu. La remise ordonnée de la réponse au message RQNT avec message(s) Notify en attente est facultative; la passerelle peut choisir également d'envoyer la réponse au message RQNT dans un datagramme distinct. L'exigence visant à garantir la remise ordonnée des messages Notify est par ailleurs obligatoire.

Figure 3/J.171.1 – Algorithme de quarantaine

Les contrôleurs MGC DEVRAIENT fournir dans le même datagramme la réponse à un message Notify de réussite et la nouvelle demande NotificationRequest, en utilisant le mécanisme de portage[21].

[21] Les vendeurs qui choisissent de ne pas appliquer la présente Recommandation devraient examiner attentivement les scénarios d'échec pour les contrôleurs de passerelle média.

7.4.3.2 Détection explicite

Un élément fondamental pour l'état des différentes extrémités est l'état de prise (par décrochage du combiné) d'un circuit de ligne. Bien que les événements modifiant l'état de prise soient durables dans le protocole TGCP, les situations de concurrence et de discordance des états peuvent quand même exister, p. ex. lorsqu'un circuit est pris tandis que le contrôleur MGC s'emploie à demander à la passerelle de rechercher une ligne à prendre (la situation "de double-prise" bien connue en téléphonie – Toutefois, cette question concerne principalement les jonctions bilatérales à signalisation voie par voie, qui ne sont pas prises en charge dans la présente version de la Recommandation).

Afin d'éviter cette situation de concurrence, la passerelle DOIT vérifier la situation au niveau de l'extrémité avant de répondre à une demande NotificationRequest. En particulier, elle DOIT renvoyer une erreur:

1) lorsqu'elle est invitée à notifier une transition d'état de "prise de ligne"[22] tandis que le circuit est déjà pris (code d'erreur 401 – circuit pris);

2) lorsqu'elle est invitée à notifier une situation de "non-prise de ligne"[23] tandis que le circuit n'est pas pris (code d'erreur 402 – circuit non pris).

En outre, des définitions de signaux différentes peuvent spécifier qu'un signal ne fonctionnera que dans certaines conditions, p. ex. un retour d'appel sonore de l'opérateur multifréquence n'est possible que si le circuit est déjà pris. Si de telles conditions sont prérequises pour un signal donné, la passerelle DOIT renvoyer l'erreur spécifiée dans la définition du signal lorsque ces conditions ne sont pas remplies.

Il convient de noter que la vérification des conditions est effectuée au moment de la réception de la demande de notification, tandis que l'événement réel qui a provoqué la situation actuelle peut soit avoir été signalé, avoir été ignoré antérieurement, ou être actuellement en quarantaine.

Les autres variables d'état de la passerelle, telles que la liste des événements demandés ou celle des signaux demandés, sont complètement remplacées après chaque demande NotificationRequest accomplie, ce qui évite les éventuels désaccords de longue durée entre le contrôleur MGC et la passerelle.

Lorsqu'une demande NotificationRequest n'aboutit pas, qu'elle fasse ou non partie d'une commande destinée à prendre en charge les connexions, la passerelle continuera simplement comme si la commande n'avait jamais été reçue, bien qu'une erreur soit renvoyée. Comme toutes les autres transactions, la demande NotificationRequest DOIT fonctionner comme une transaction atomique; tout changement entamé suite à la commande DOIT donc être annulé.

Quand le contrôleur de passerelle média reçoit une réponse d'erreur indiquant qu'une demande de notification a été inefficace, il DOIT commencer à faire en sorte que tous les événements mis en quarantaine par l'extrémité soient traités ou rejetés et en sorte que l'extrémité soit remise en mode de fonctionnement normal, dans lequel les événements récemment demandés sont signalés au fur et à mesure de leur apparition. Par exemple, si le contrôleur de passerelle média reçoit un code d'erreur "401 – extrémité déjà saisie" en réponse à une demande de notification réclamant la détection du signal de "décrochage MT/sup". A ce point, le contrôleur de passerelle média devrait partir du principe que la demande de notification n'a eu aucun effet sur l'extrémité et que celle-ci est dans le même état qu'avant la réception de la commande. Si l'extrémité était en train de mettre en quarantaine des événements dans l'état "perpétuel" avant la réception de la demande de notification,

---

[22] P. ex. en demandant l'événement "sup" dans un circuit BLV/OI multifréquence de terminaison avec un appel déjà en cours.

[23] P. ex. en demandant l'événement "rel" dans un circuit multifréquence de services d'opérateur sans aucun appel en cours.

alors cette extrémité va continuer à mettre en quarantaine des événements dans l'état "perpétuel" après avoir envoyé le code d'erreur "401" en réponse. Afin de faire en sorte que l'extrémité ne soit pas laissée dans un état où elle est de façon permanente en train de mettre en quarantaine des événements, le contrôleur de passerelle média devrait envoyer une nouvelle demande de notification avec un ensemble différent (éventuellement vide) d'événements demandés, de façon à faire passer l'extrémité hors de l'état "perpétuel" et à la faire entrer dans le mode normal où elle pourra signaler de nouveaux événements.

Une autre situation de concurrence peut se produire lorsqu'une commande Notify est émise peu de temps avant la réception par la passerelle d'une demande NotificationRequest. L'identificateur RequestIdentifier est employé pour corréler les commandes Notify avec les commandes NotificationRequest, le contrôleur MGC étant ainsi en mesure de déterminer si la commande Notify a été produite avant ou après que la passerelle a reçu la nouvelle demande NotificationRequest.

7.4.3.3 Sémantique transactionnelle

Alors que le temps d'achèvement des transactions éventuelles augmente, p. ex. en raison des réserves de ressources extérieures, il est de plus en plus important de définir précisément la sémantique transactionnelle. En particulier, la question des situations de concurrence, dans la mesure où elle se rapporte à l'état de prise de ligne, doit faire l'objet d'une définition précise.

Un point important qu'il faut examiner est le fait que l'état de prise peut en fait être modifié entre le moment où la transaction débute et celui où elle s'achève. Plus généralement, il peut être affirmé que la réussite de l'achèvement d'une transaction dépend d'une ou de plusieurs conditions préexistantes, une ou plusieurs d'entre elles pouvant changer dynamiquement au cours de l'exécution de la transaction.

La sémantique la plus simple pour cela consiste à prescrire simplement que toutes les conditions préexistantes DOIVENT s'appliquer à partir du moment où la transaction débute jusqu'à celui où elle s'achève. Donc, si l'une des conditions préexistantes change au cours de l'exécution de la transaction, celle-ci DOIT échouer. En outre, dès le début de la transaction, tous les nouveaux événements sont mis en quarantaine. Lorsque le résultat de la transaction est connu, tous les événements mis en quarantaine sont traités.

Considérons, à titre d'exemple, une transaction qui comporte une demande relative à un événement de "prise de ligne". Lorsque la transaction débute, le circuit est en état de "non-prise" et c'est cette condition préexistante qui prévaut. Si cet état de prise est modifié et devient un état de "prise de ligne" avant que la transaction s'achève, la condition préexistante n'est plus remplie et la transaction échoue rapidement. L'événement de "prise de ligne" sera donc mémorisé dans le tampon de "quarantaine" qui sera traité ultérieurement.

7.4.3.4 Classement des commandes et traitement du désordre

Dans le cadre du protocole MGCP, il n'est pas demandé que le protocole de transport sous-jacent assure le classement séquentiel des commandes envoyées à une passerelle ou à une extrémité. Cette propriété vise à maximiser la ponctualité des actions, mais elle a quelques défauts. P. ex.:

- les commandes Notify peuvent être retardées et aboutir au contrôleur MGC après la transmission d'une nouvelle commande NotificationRequest;
- si une nouvelle commande NotificationRequest est transmise avant qu'une réponse à une commande précédente soit reçue, rien ne garantit que la commande précédente ne sera pas reçue en deuxième position.

Les contrôleurs MGC et les passerelles qui désirent assurer un fonctionnement cohérent des extrémités peuvent utiliser les règles suivantes:

1) lorsqu'une passerelle prend en charge plusieurs extrémités, les commandes se rapportant aux différentes extrémités peuvent être envoyées en parallèle, p. ex. en suivant un modèle

où chaque extrémité est commandée au moyen d'un processus ou d'un chemin qui lui est propre;

2) lorsque plusieurs connexions sont établies à la même extrémité, les commandes se rapportant aux différentes connexions peuvent être envoyées en parallèle;

3) lors d'une connexion donnée, il ne devrait normalement y avoir qu'une seule commande en suspens (de création ou de modification). Toutefois, une commande DeleteConnection peut être lancée à tout moment. En conséquence, une passerelle peut parfois recevoir une commande ModifyConnection qui s'applique à une connexion précédemment supprimée. On ne DOIT pas tenir compte de telles commandes, et une erreur doit être renvoyée (code d'erreur 515 – Identificateur de connexion incorrect);

4) à une extrémité donnée, il ne devrait normalement y avoir à tout moment qu'une seule commande NotificationRequest en suspens. Le paramètre RequestId est utilisé pour corréler les commandes Notify avec la commande NotificationRequest qui les a déclenchées;

5) dans certains cas, une commande DeleteConnection avec caractère de remplacement implicite ou explicite qui s'applique à un groupe d'extrémités peut précéder une commande CreateConnection en suspens. Le contrôleur MGC devrait supprimer individuellement toutes les connexions dont l'achèvement était en suspens au moment de la commande globale DeleteConnection. Par ailleurs, de nouvelles commandes CreateConnection pour les extrémités désignées au moyen d'un caractère de remplacement ne devraient pas être envoyées avant la réception d'une commande DeleteConnection avec caractère de remplacement;

6) lorsque des commandes sont incorporées les unes dans les autres, des spécifications de classement pour toutes les commandes DOIVENT être respectées. P. ex. une commande CreateConnection contenant une demande de notification doit respecter simultanément les spécifications de classement pour la commande CreateConnection et pour la commande NotificationRequest;

7) les commandes AuditEnpoint et AuditConnection ne sont soumises à aucun classement;

8) la commande RestartInProgress doit toujours être la première commande envoyée par une extrémité, comme spécifié dans la procédure de redémarrage (voir § 7.4.3.5). Toute autre commande ou réponse doit être fournie après cette commande RestartInProgress (portage admis);

9) lorsque plusieurs messages accompagnent en portage un unique paquet, ils sont toujours traités dans l'ordre.

Parmi les règles susmentionnées, celles qui spécifient le comportement des passerelles DOIVENT être respectées par les passerelles de jonction, qui toutefois NE DOIVENT PAS faire d'hypothèse quant à l'application ou non des règles par les contrôleurs MGC. En conséquence, les passerelles DOIVENT toujours répondre aux commandes, qu'elles appliquent ou non les règles susmentionnées.

Afin de garantir un fonctionnement cohérent, les passerelles de jonction DEVRAIENT se comporter comme spécifié ci-dessous quand une ou plusieurs des règles ci-dessus ne sont pas suivies:

- lorsqu'une unique commande en instance est attendue (ModifyConnection, NotificationRequest), mais que la même commande est reçue dans une nouvelle transaction avant que l'ancienne ait terminé son exécution, la passerelle DEVRAIT interrompre la commande précédente. Cela inclut le cas où une ou plusieurs commandes ont été encapsulées. L'utilisation du code d'erreur 407 (transaction interrompue) est RECOMMANDÉE;
- si une commande ModifyConnection est reçue pour une commande en instance CreateConnection, la commande ModifyConnection DEVRAIT simplement être rejetée.

L'utilisation du code d'erreur 400 (erreur transitoire) est RECOMMANDÉE. Noter que cette situation constitue une erreur de programmation du contrôleur de passerelle média.

Noter que, lorsque la réception d'une nouvelle commande conduit à interrompre une ancienne commande, celle-ci DEVRAIT être interrompue, que la nouvelle ait ou non réussi. Par exemple, si une commande ModifyConnection est interrompue par une commande DeleteConnection qui elle-même échoue en raison d'une NotificationRequest encapsulée, la commande ModifyConnection est quand même interrompue.

7.4.3.5 Lutte contre l'avalanche de redémarrages

Supposons qu'un grand nombre de passerelles soient activées simultanément. Si elles devaient toutes entamer une transaction RestartInProgress, le contrôleur MGC serait très certainement submergé, et il en résulterait des pertes de messages et un encombrement du réseau pendant la période critique de rétablissement du service. Afin d'éviter ces avalanches, les comportements suivants DOIVENT être adoptés:

1) lorsqu'une passerelle est activée, elle attribue à un temporisateur de redémarrage une valeur aléatoire, répartie uniformément entre zéro et un temps d'attente maximal configurable (MWD, *maximum waiting delay*), p. ex. 360 secondes (voir ci-dessous). On DOIT prendre soin d'éviter que les nombres aléatoires produits aux différentes passerelles qui utiliseraient le même algorithme soient synchrones;
2) la passerelle attend ensuite la fin de la temporisation, la réception d'une commande provenant du contrôleur MGC ou la détection d'une activité locale du circuit, telle que, p. ex. une transition d'état du crochet commutateur (prise de ligne) dans une passerelle de jonction. Une situation de prise préexistante conduit à la production d'un événement de prise de ligne;
3) lorsque la temporisation de redémarrage est écoulée à la réception d'une commande ou à la détection d'une activité ou d'une situation de prise préexistante, la passerelle entame la procédure de redémarrage.

La procédure de redémarrage stipule simplement que l'extrémité DOIT envoyer au contrôleur MGC une commande RestartInProgress l'informant du redémarrage et l'assurant en outre que le premier message (commande ou réponse) qu'il observera en provenance de cette extrémité DOIT être cette commande RestartInProgress. Pendant chaque lancement de procédure "déconnecté", la commande DOIT respecter les exigences de retransmission normale et des identificateurs de transaction (voir § 7.4.2).

Pour y arriver, l'extrémité DOIT profiter pleinement du portage. P. ex. si une activité de prise de circuit a lieu avant l'expiration de la temporisation de redémarrage, un paquet contenant la commande RestartInProgress accompagné d'une commande Notify pour l'événement de prise de ligne sera produit. Dans le cas où la temporisation de redémarrage expire sans qu'une autre activité ait eu lieu, la passerelle envoie simplement un message RestartInProgress.

La procédure de redémarrage est achevée une fois qu'une réponse de succès a été reçue. Si c'est un message d'erreur qui est reçu en réponse, le comportement à suivre dépend du code d'erreur en question:

- si le code d'erreur indique une erreur temporaire (4xx), la procédure de redémarrage DOIT alors être relancée (comme nouvelle transaction);
- si le code d'erreur est 521, l'extrémité est alors réacheminée et la procédure de redémarrage DOIT être relancée (comme nouvelle transaction). La réponse 521 devrait avoir inclus une NotifiedEntity qui sera alors "l'entité notifiée" vers laquelle le redémarrage est lancé;
- si l'erreur est toute autre erreur permanente (5xx), il est alors RECOMMANDÉ que l'extrémité ne lance plus la procédure de redémarrage de son propre chef (jusqu'à ce qu'elle

soit réinitialisée) à moins qu'il n'en soit spécifié autrement. Si une commande est reçue, l'extrémité DOIT relancer la procédure de redémarrage.

Noter que si le redémarrage en cours est superposé par portage avec la réponse (R) à une commande reçue pendant le redémarrage, la retransmission de ce redémarrage en cours n'exige pas le portage de la réponse R. Cependant, alors que l'extrémité est en train de redémarrer, un nouvel envoi de la réponse R exige effectivement le portage du redémarrage en cours afin de garantir la bonne livraison des deux commandes.

Si l'extrémité entre dans l'état "déconnecté" pendant l'exécution de la procédure de redémarrage, la procédure "déconnecté" spécifiée au § 7.4.3.6 DOIT être exécutée et un message "déconnecté" est envoyé pendant la procédure.

Il est prévu que chaque extrémité d'une passerelle disposera d'un contrôleur MGC préconfigurable (à savoir une "entité notifiée") vers lequel le message initial de redémarrage sera envoyé. Lorsque l'ensemble des extrémités d'une passerelle est géré par plus d'un contrôleur MGC, la procédure ci-dessus doit être exécutée pour chaque ensemble d'extrémités géré par un contrôleur MGC donné. La passerelle DOIT profiter pleinement des caractères de remplacement pour minimiser le nombre de messages RestartInProgress produits lorsque plusieurs extrémités d'une passerelle redémarrent et que ces extrémités sont gérées par le même contrôleur MGC.

La valeur du temps MWD est donnée par un paramètre de configuration qui dépend du type de la passerelle. Le raisonnement suivant peut être utilisé pour déterminer la valeur de ce temps dans une passerelle.

Les contrôleurs MGC sont généralement dimensionnés pour prendre en charge le trafic aux heures de pointe au cours desquelles, en moyenne, 60% des circuits de jonction seront occupés à servir des appels dont la durée moyenne est généralement de 3 minutes. Le traitement d'un appel comporte généralement 5 à 6 transactions entre chaque extrémité et le contrôleur MGC. Ce simple calcul montre que le contrôleur MGC devrait traiter 5 à 6 transactions par extrémité, toutes les 5 minutes en moyenne; ou, formulé différemment, environ une transaction par extrémité et par minute. Cela suggère qu'une valeur raisonnable du temps MWD pourrait être de 2 minutes par extrémité. Lorsque la valeur du temps est fixée pour la passerelle, cette valeur devrait être inversement proportionnelle au nombre d'extrémités qui font l'objet d'un redémarrage. P. ex. le temps pourrait être fixé à 5 secondes pour une passerelle qui prend en charge une ligne T1, ou à 180 ms pour une passerelle qui prend en charge une ligne au débit T3.

7.4.3.6 Extrémités déconnectées

Outre la procédure de redémarrage, les passerelles de jonction disposent d'une procédure de "déconnexion" qui est entamée lorsqu'une extrémité se "déconnecte", comme décrit au § 7.4.2. Il convient de noter ici que les extrémités ne peuvent être déconnectées que lorsqu'elles tentent de communiquer avec le contrôleur MGC. Les étapes suivantes sont suivies par une extrémité qui se "déconnecte":

1) un temporisateur de "déconnexion" est initialisé au moyen d'une valeur aléatoire, répartie uniformément entre zéro et un temps d'attente initial de "déconnexion" préconfigurable (Td_{init}), p. ex. 15 secondes. On DOIT prendre soin d'éviter que les nombres aléatoires produits aux différentes passerelles et extrémités, qui utiliseraient le même algorithme, soient synchrones;
2) la passerelle attend ensuite la fin de la temporisation, la réception d'une commande provenant du contrôleur MGC ou la détection d'une activité locale du circuit pour l'extrémité, p. ex. une transition d'état de prise;
3) lorsque la temporisation de "déconnexion" est écoulée, à la réception d'une commande, ou à la détection d'une activité locale du circuit, la passerelle DOIT entamer la procédure de "déconnexion" à l'extrémité avec un nouvel identificateur de transaction. Dans le cas d'une

activité locale du circuit, un temps d'attente minimal de "déconnexion" préconfigurable (Td_{min}) doit en outre s'être écoulé après la déconnexion de la passerelle ou après qu'elle a entamé pour la dernière fois la procédure de "déconnexion", afin de limiter le nombre d'applications de cette procédure;

4) si l'extrémité est encore déconnectée après l'application de la procédure de déconnexion, la temporisation de "déconnexion" est alors doublée, sous réserve d'un temps d'attente maximal de "déconnexion" préconfigurable (Td_{max}), p. ex. 600 s, et la passerelle reprend à l'étape 2).

La procédure de "déconnexion" est semblable à celle du redémarrage en raison du fait qu'elle stipule maintenant simplement que l'extrémité DOIT envoyer au contrôleur MGC une commande RestartInProgress l'informant de la déconnexion de l'extrémité et l'assurant en outre que le premier message (commande ou réponse) qu'il observera en provenance de cette extrémité DOIT être cette commande RestartInProgress. Pendant chaque lancement de procédure "déconnecté", la commande DOIT respecter les exigences de retransmission normale et des identifiants de transaction. (Voir § 7.4.2.) Ce faisant, l'extrémité DOIT tirer pleinement profit du portage.

Dès réception d'un message RestartInProgress contenant une valeur de méthode de redémarrage "déconnecté", le contrôleur de passerelle média DOIT commencer à faire en sorte que tous les événements mis en quarantaine par l'extrémité soient traités ou rejetés et que l'extrémité soit remise à un mode de fonctionnement normal où les événements récemment demandés sont signalés au fur et à mesure de leur apparition. Le contrôleur de passerelle média DEVRAIT envoyer une demande de notification contenant un paramètre de QuarantineHandling réglé à "rejeter" dans ce cas. Le contrôleur de passerelle média peut également décider d'effectuer une ou plusieurs des opérations suivantes: auditer l'extrémité, supprimer toutes les connexions pour l'extrémité ou envoyer une NotificationRequest demandant à ce point d'extrémité de traiter les événements mis en quarantaine (voir § 7.4.3.7).

Une extrémité déconnectée peut souhaiter envoyer une commande (en dehors de RestartInProgress) alors qu'elle est déconnectée. Cette opération ne peut réussir que lorsque le contrôleur de passerelle média est de nouveau joignable, ce qui soulève la question de savoir ce qu'il convient de faire d'une telle commande dans l'intervalle. A la limite, l'extrémité pourrait laisser tomber la commande, cependant cela poserait un problème dans les cas où le contrôleur de passerelle média était en fait disponible mais que l'extrémité n'avait pas encore terminé la procédure "déconnecté" (considérons p. ex. le cas où une NotificationRequest vient juste d'être reçue ce qui a immédiatement eu pour résultat la création d'un message Notify). Pour empêcher de tels scénarios, les extrémités déconnectées NE DOIVENT PAS laisser tomber aveuglément les nouvelles commandes à envoyer pendant une période de Ts_{max} secondes après qu'elles ont reçu une commande de non-audit.

Une façon de satisfaire cette exigence est d'utiliser une mémoire tampon temporaire des commandes à envoyer. Ce faisant, l'extrémité doit cependant s'assurer que:

- elle ne bâtit pas une longue file d'attente de commandes à envoyer;
- elle ne submerge pas le contrôleur de passerelle média en lui envoyant rapidement de trop nombreuses commandes une fois qu'elle est reconnectée.

La mise en mémoire tampon des commandes pendant une durée de Ts_{max} secondes et, une fois que l'extrémité est reconnectée, la limitation du rythme auquel les commandes mémorisées sont envoyées à une commande en attente par extrémité est considérée comme sûre. Si l'extrémité n'est pas connectée dans les Ts_{max} secondes mais qu'une procédure "déconnecté" est lancée dans les Ts_{max} secondes, l'extrémité PEUT superposer en portage la ou les commandes mémorisées sur la commande Redémarrage en cours. Noter qu'une fois une commande envoyée, qu'elle ait été mémorisée ou non au départ ou portée antérieurement, la retransmission de cette commande DOIT cesser Ts_{max} secondes après l'envoi initial, comme décrit au § 7.4.2.

La procédure "déconnecté" est achevée une fois reçue une réponse de succès. Les réponses d'erreur sont traitées comme dans la procédure de redémarrage (§ 7.4.3.5). Si la procédure "déconnecté" doit être relancée à la suite d'une réponse d'erreur, les considérations sur le temporisateur de limitation du rythme de ce relancement s'appliquent aussi.

On notera aussi que si le RestartInProgress est porté sur une réponse (R) à une commande reçue pendant que la déconnexion était en cours, la retransmission du RestartInProgress n'exige pas le portage de la réponse R. Cependant, alors que l'extrémité est déconnectée, le renvoi de la réponse R exige que le RestartInProgress soit quand même superposé afin de garantir l'ordre de livraison des deux commandes.

Noter que si une procédure déconnectée est déjà en cours lorsque la commande est reçue, la procédure de déconnexion existante DOIT être terminée, et une nouvelle procédure doit commencer. Ceci sert à supporter une redirection possible du contrôleur de passerelle média.

Noter également que, pour une extrémité, le fait d'être déconnectée ne signifie pas qu'elle soit dans un état "hors service". La déconnexion d'une extrémité n'indique pas un état de disponibilité de son service, mais indique plutôt l'incapacité de la passerelle à communiquer avec son contrôleur de passerelle média.

La présente Recommandation ne spécifie pas, à dessein, d'autre comportement pour un extrémité déconnectée. Les vendeurs PEUVENT p. ex. choisir de conserver les silences, de reproduire la tonalité de recomposition ou même de permettre qu'un fichier audio importé puisse être joué aux extrémités concernées.

Les valeurs par défaut sont égales à 15 secondes pour Td_{init}, à 15 secondes pour Td_{min} et à 600 secondes pour Td_{max}.

7.4.3.7 Traitement par agent d'appel des extrémités déconnectées

Quand une extrémité est dans l'état "déconnecté", elle peut accumuler un grand nombre d'événements dans le tampon de mise en quarantaine. De même, une extrémité "déconnectée" peut supprimer de façon autonome des connexions établies (si la passerelle se réamorce). Donc, quand la connexité entre une extrémité "déconnectée" et son Agent d'appel finit par être restaurée, l'agent d'appel DOIT être disposé à s'occuper des problèmes suivants:

- le grand nombre de messages Notify qui peuvent être produits par l'extrémité si tous les événements figurant sur la liste de quarantaine sont traités;
- la réception d'événements anciens/altérés, que l'extrémité signale comme n'ayant plus aucune pertinence. Le tampon de mise en quarantaine est une file d'attente de type premier-entré-premier-sorti (FIFO, first-in-first-out) dans laquelle les plus anciens événements sont traités en premier puis notifiés (si demandé) à l'agent d'appel. La mesure prise par l'agent d'appel dès réception d'un ancien événement peut ne pas être significative si l'ancien événement a été remplacé par des événements plus récents (p. ex. un événement de "décrochage MT/sup" ne sera plus pertinent si l'extrémité a déjà passé à l'état de raccrochage);
- une discordance entre les états de connexion de l'agent d'appel et de l'extrémité, où l'agent d'appel estime que l'extrémité possède une ou plusieurs connexions alors qu'en fait elle n'en a aucune.

Les agents d'appel sont libres d'utiliser tout mécanisme pris en charge par le protocole afin de résoudre les problèmes ci-dessus. Une façon d'y parvenir consiste à effectuer les opérations suivantes:

1) nous définissons une nouvelle variable booléenne appelée "synchro-événement-déconnexion", qui est tenue à jour par l'agent d'appel pour chacune de ses extrémités. Quand elle est mise à la valeur "Vrai", cette variable indique que la connexité avec une

extrémité "déconnectée" a récemment été restaurée, mais que la synchronisation événement-signal n'a pas encore été réalisée. (Noter que cette variable est introduite ici afin de décrire le comportement d'un agent d'appel et qu'elle n'est pas destinée à impliquer une quelconque implémentation particulière. Cette variable n'est pas visible de l'extérieur.)

a) Dès que l'agent d'appel est informé du fait qu'une extrémité est déconnectée, il met l'indicateur "synchro-événement-déconnexion" à "Vrai". La procédure "déconnecté" garantit que l'agent d'appel sera informé de l'état de déconnexion de l'extrémité par la réception d'un message RestartInProgress avec la valeur "déconnecté". Quand une extrémité reçoit un accusé de réception positif du message RSIP "déconnecté", elle effectue la procédure "déconnecté". A ce point, l'extrémité pourrait produire immédiatement un message Notify pour deux raisons: envoyer une commande Notify qui a été mise en mémoire tampon pendant que l'extrémité était déconnectée; ou, si l'extrémité se trouvait dans l'état "notification" et en mode "boucle" pendant sa déconnexion, signaler un événement de déclenchement de notification sur la liste de mise en quarantaine.

   - Si l'extrémité fonctionne en mode "étape", le fait de répondre au message Notify ne permettra pas, par là même, de produire d'éventuels autres messages Notify (une NotificationRequest additionnelle serait requise pour cela);
   - Cependant, si l'extrémité fonctionne en mode "boucle", alors une réponse au message Notify va permettre de produire de nouveaux messages Notify. Comme expliqué ci-dessus, cela est parfois indésirable car les événements qui sont signalés peuvent être anciens et un lot d'événements peut avoir été mis en quarantaine, ce qui à son tour va se traduire par un grand nombre de messages Notify et de messages subséquents de NotificationRequest fondés sur des informations altérées.

b) Aussi longtemps qu'une extrémité a l'indicateur "synchro-événement-déconnexion" réglé à "Vrai", l'agent d'appel devrait faire en sorte que le nombre d'événements figurant sur la liste de quarantaine soit rejeté, ou traité de façon contrôlée et ordonnée. Cela peut être réalisé par un certain nombre de méthodes:

   - l'agent d'appel peut envoyer une unique NotificationRequest spécifiant que tous les événements mis en quarantaine doivent être rejetés. Dès réception d'un accusé de réception favorable à cette commande, ou dès réception d'un message Notify avec le même RequestIdentifier, l'agent d'appel devrait mettre l'indicateur "synchro-événement-déconnexion" à la valeur "Faux", point à partir duquel le traitement normal des événements est repris pour l'extrémité. L'inconvénient de cette approche est qu'elle va effacer tous les événements qui ont été accumulés, quel que soit le nombre d'événements accumulés. Dans certains cas, cela peut se traduire par une interruption de service inutile. Afin de résoudre cela, des extensions du protocole seront nécessaires;
   - l'agent d'appel peut envoyer une NotificationRequest spécifiant que les événements mis en quarantaine doivent être traités. Si elle est en fonctionnement en mode "étape", alors l'extrémité va signaler un unique événement de déclenchement de notification pour chaque NotificationRequest reçue, tandis qu'en mode "boucle" elle peut signaler de multiples événements avec une unique NotificationRequest.

     Comme les informations acheminées par les événements notifiés peuvent ne plus être pertinentes, l'agent d'appel ne devrait pas traiter aveuglément ces événements (p. ex., lorsqu'il reçoit notification d'un événement de "décrochage MT/sup" en départ, l'agent d'appel ne devrait pas automatiquement envoyer une NotificationRequest afin d'appliquer la tonalité de numérotation et demander une notification de "raccrochage MT/inf"). Au contraire, l'agent d'appel doit synchroniser ses données d'état interne avec l'état réel de l'extrémité. Etant donné

qu'une extrémité exécute le traitement de la simultanéité signal-événement en fonction de l'état actuel de prise du circuit, l'agent d'appel peut découvrir l'état actuel de prise de circuit sur la base de la réponse à une NotificationRequest. Par exemple, une réponse par code "402 – circuit non saisi" à une NotificationRequest demandant la détection de l'état de "raccrochage MT/inf" implique que la jonction MT (à terminaison multifréquence) est actuellement non saisie. L'agent d'appel peut choisir d'ignorer les événements notifiés qui sont jugés non pertinents sur la base de l'état actuel du crochet commutateur (ignorer par exemple l'événement de numérotation à l'état "MT/inf" si la jonction est actuellement non saisie (à l'état de raccrochage)).

Une fois que tous les événements figurant sur la liste de quarantaine ont été traités, l'agent d'appel devrait mettre l'indicateur "synchro-événement-déconnexion" à la valeur "Faux". L'agent d'appel peut en toute sécurité partir du principe que tous les événements mis en quarantaine ont été traités si un délai de durée égale à Tt_{hist} a expiré depuis la dernière fois que l'agent d'appel a incité l'extrémité à traiter l'événement suivant (c'est -à-dire si un délai de durée égale à Tt_{hist} a expiré depuis que l'agent d'appel a envoyé la réponse au précédent message Notify dans le mode "boucle", ou depuis que l'agent d'appel a reçu la dernière réponse favorable à une NotificationRequest dans le mode "étape").

2) Quand des extrémités passent à l'état déconnecté, les connexions créées dans cette extrémité devraient être désaffectées. Cependant, il est toujours possible qu'une connexion ne puisse plus être conservée par l'extrémité et donc qu'elle soit supprimée; cela va se traduire par une commande DeleteConnection envoyée à l'agent d'appel. Quand l'extrémité est déconnectée, une telle commande peut ne jamais parvenir à l'agent d'appel, auquel cas celui-ci ne sera pas informé de la connexion supprimée. Par conséquent, chaque fois qu'un Agent d'appel est informé du fait qu'une extrémité est déconnectée, il devrait vérifier dans cette extrémité la liste des connexions qui y sont présentes.

7.5 Codes de renvoi et codes d'erreur

Toutes les commandes MGCP donnent lieu à une réponse. La réponse contient un code de renvoi qui indique l'état de la commande. Les codes de renvoi sont des nombres entiers qui ont été répartis en cinq gammes:

- la valeur 000 indique un accusé de réception de réponse[24];
- les valeurs comprises entre 100 et 199 indiquent une réponse provisoire;
- les valeurs comprises entre 200 et 299 indiquent un établissement réussi;
- les valeurs comprises entre 400 et 499 indiquent une erreur transitoire;
- les valeurs comprises entre 500 et 599 indiquent une erreur durable.

La liste des valeurs qui ont été définies est donnée dans le Tableau 3:

Tableau 3/J.171.1 – Codes de renvoi

Code	Signification
000	Accusé de réception de réponse
100	La transaction est en cours d'exécution. Un message d'établissement proprement dit suivra plus tard

[24] Un accusé de réception de réponse est employé pour les réponses provisoires (voir § 8.8).

Tableau 3/J.171.1 – Codes de renvoi

Code	Signification
200	La transaction demandée a été exécutée normalement
250	La ou les connexions ont été supprimées
400	La transaction n'a pu être exécutée à cause d'une erreur transitoire
401	Le combiné est déjà décroché ou le circuit est déjà pris
402	Le combiné est déjà raccroché ou le circuit n'est pas pris
407	Transaction interrompue. La transaction a été interrompue par une certaine action extérieure, p. ex. une commande ModifyConnection interrompue par une commande DeleteConnection
500	La transaction n'a pu être exécutée parce que l'extrémité est inconnue
501	La transaction n'a pu être exécutée parce que l'extrémité n'est pas prête
502	La transaction n'a pu être exécutée parce que l'extrémité ne dispose pas de ressources suffisantes
503	Caractère générique "tous" non pris complètement en charge. La transaction contenait un caractère générique "tous", mais la passerelle ne les prend pas entièrement en charge. Noter que ce code n'est admissible que pour les NotificationRequest non vides
505	Descripteur de connexion distante non pris en charge. Ce code DEVRAIT être utilisé lorsqu'un ou plusieurs paramètres ou valeurs obligatoires ne sont pas pris en charge dans le RemoteConnectionDescriptor
506	Impossible de satisfaire à la fois aux LocalConnectionOptions et au RemoteConnectionDescriptor. Ce code DEVRAIT être utilisé lorsque les Options de connexion locale et le Descripteur de connexion distante contiennent un ou plusieurs paramètres ou valeurs obligatoires qui se contredisent ou ne peuvent être pris en charge en même temps (sauf pour les échecs de négociations de codec, voir le code d'erreur 534)
510	La transaction n'a pu être exécutée parce qu'une erreur de protocole a été détectée
511	La transaction n'a pu être exécutée parce que la commande contenait une extension non reconnue
512	La transaction n'a pu être exécutée parce que la passerelle n'est pas équipée pour détecter l'un des événements demandés
513	La transaction n'a pu être exécutée parce que la passerelle n'est pas équipée pour produire l'un des signaux demandés
514	La transaction n'a pu être exécutée parce que la passerelle ne peut envoyer l'annonce spécifiée
515	La transaction renvoie à une identificateur de connexion erroné (pouvant déjà avoir été supprimé)
516	La transaction renvoie à un identificateur d'appel inconnu
517	Mode non pris en charge ou non valable
518	Paquetage non pris en charge ou inconnu
519	Extrémité n'ayant pas de table numérique de mappages
520	La transaction n'a pu être exécutée parce que l'extrémité est en cours de "redémarrage"
521	Extrémité réacheminée vers un autre contrôleur MGC
522	Evénement ou signal faisant défaut
523	Mesure inconnue ou combinaison interdite de mesures
524	Incohérence interne dans le paramètre LocalConnectionOptions
525	Extension inconnue dans le paramètre LocalConnectionOptions
526	Largeur de bande insuffisante
527	Paramètre RemoteConnectionDescriptor manquant

Tableau 3/J.171.1 – Codes de renvoi

Code	Signification
528	Version de protocole incompatible
529	Défaillance matérielle interne
532	Valeur(s) non prise(s) en charge dans le paramètre LocalConnectionOptions
533	Réponse trop longue
534	Echec de négociation de codec
538	Erreur de paramètre d'événement /de signal (par exemple, manquant, erroné, non pris en charge, inconnu, etc.)

7.6 Codes de cause

Les codes de cause sont employés par la passerelle lors de la suppression d'une connexion pour informer le contrôleur MGC sur la cause qui a conduit à supprimer cette connexion. Le code de cause est un nombre entier, et les valeurs ont été définies dans le Tableau 4:

Tableau 4/J.171.1 – Codes de cause

Code	Signification
900	Dysfonctionnement de l'extrémité
901	Extrémité mise hors service
902	Perte de la connexité de couche inférieure (p. ex. synchronisation en aval)

7.7 Utilisation des options de connexion locale et des descripteurs de connexion

La séquence normale dans l'établissement d'une connexion bilatérale implique au moins trois étapes:

1) le contrôleur de passerelle média/l'agent d'appel demande à la première passerelle de "créer une connexion" sur une extrémité. La passerelle alloue des ressources à cette connexion, et répond à la commande en fournissant une "description de session" (désignée comme son LocalConnectionDescriptor). La description de session contient les informations nécessaires pour que d'autres entités puissent envoyer des paquets vers la connexion nouvellement créée;
2) le contrôleur de passerelle média/l'agent d'appel demande alors à la seconde passerelle de "créer une connexion" sur une extrémité. La commande transporte une "description de session" fournie par la première passerelle (désignée maintenant comme le RemoteConnectionDescriptor). La passerelle alloue des ressources à cette connexion et répond à la commande en fournissant sa propre "description de session" (LocalConnectionDescriptor);
3) le contrôleur de passerelle média/l'agent d'appel utilise une commande "modifier la connexion" pour fournir cette seconde "description de session" (maintenant désignée comme RemoteConnectionDescriptor) à la première extrémité. Une fois que cela est fait, les communications peuvent avoir lieu dans les deux sens.

Lorsque le contrôleur de passerelle média/l'agent d'appel produit une commande Créer ou ModifyConnection, il y a donc trois paramètres qui déterminent le média pris en charge par cette connexion:

- LocalConnectionOptions: paramètre fourni par le contrôleur de passerelle média/l'agent d'appel pour commander les paramètres de média utilisés par la passerelle pour la connexion. Lorsque ce paramètre est fourni, la passerelle doit se conformer à ces paramètres de média jusqu'à ce que la connexion soit supprimée ou qu'une commande ModifyConnection soit reçue;
- RemoteConnectionDescriptor: paramètre fourni par le contrôleur de passerelle média/l'agent d'appel pour convoyer les paramètres de média pris en charge par l'autre côté de la connexion. Lorsque ce paramètre est fourni, la passerelle doit se conformer à ces paramètres de média jusqu'à ce que la connexion soit supprimée ou qu'une commande ModifyConnection soit reçue;
- LocalConnectionDescriptor: paramètre fourni par la passerelle au contrôleur de passerelle média/l'agent d'appel pour convoyer les paramètres de média qu'elle prend en charge pour la connexion. Lorsque ce paramètre est fourni, la passerelle doit respecter ces paramètres de média jusqu'à ce que la connexion soit supprimée ou que la passerelle produise un nouveau LocalConnectionDescriptor. En plus des paramètres de média attribués à la connexion, la passerelle peut signaler des capacités additionnelles prises en charge dans le LocalConnectionDescriptor. Noter que de telles capacités DOIVENT être fournies à l'extérieur de la ligne "m=" dans le profil SDP. La passerelle est libre de signaler la totalité de ses capacités prises en charge, indépendamment des paramètres d'options LCO ou de descripteur RCD reçus du contrôleur de passerelle média/de l'agent d'appel, indépendamment des paramètres de média associés à la connexion.

Les choix du codec et de la période de mise en paquets ne DOIVENT être effectués, comme décrit dans le présent paragraphe, que si:

a) la passerelle reçoit une commande CRCX; ou si

b) la passerelle reçoit une commande MDCX et que l'un des paramètres suivants soit présent:
   - méthode de codage (a: dans LocalConnectionOptions);
   - période de mise en paquets (p: dans LocalConnectionOptions);
   - période de mise en paquets multiple (mp: dans LocalConnectionOptions);
   - RemoteConnectionDescriptor.

De plus, ce processus de choix du codec et de la période de mise en paquets doit seulement utiliser les informations présentes dans la demande de connexion et ne retenir aucune des valeurs qui pourraient avoir été reçues dans des demandes de connexion précédentes. Par exemple, si une passerelle a reçu une commande MDCX avec tous les paramètres LCO nécessaires mais qu'il manquât un RemoteConnectionDescriptor, elle négociera comme si aucun RemoteConnectionDescriptor n'avait été reçu pour cette connexion. De même, si tous les paramètres ci-dessus sont omis dans une commande MDCX, les codecs et les périodes de mise en paquet négociés existants resteront inchangés.

Pour déterminer quels codecs et périodes de mise en paquets fournir dans le LocalConnectionDescriptor, il y a trois listes de codecs et de périodes de mise en paquets qui doivent être prises en compte par la passerelle:

- une liste des codecs et des périodes de mise en paquets permis par les LocalConnectionOptions. Un codec est autorisé par les LocalConnectionOptions s'il satisfait aux contraintes spécifiées par les champs de méthode de codage, de période de mise en paquets et de période de mise en paquets multiple. Si un ou plusieurs de ces champs sont omis, le champ omis ne crée aucune contrainte sur les codecs autorisés;
- une liste des codecs et des périodes de mise en paquets dans le RemoteConnectionDescriptor;

- une liste interne des codecs et des périodes de mise en paquets que prend en charge la passerelle pour la connexion. Une passerelle peut accepter un ou plusieurs codecs et périodes de mise en paquets pour une connexion donnée.

Le choix du codec (y compris tous les paramètres de média pertinents) peut alors être décrit par les étapes suivantes:

1) une liste approuvée des codecs/périodes de mise en paquets est formée par l'intersection de la liste des codecs/périodes de mise en paquets et des codecs/périodes de mise en paquets autorisés par les LocalConnectionOptions. Si celles-ci n'ont pas été fournies, la liste des codecs/périodes de mise en paquets approuvée contient donc la liste interne. Si les LocalConnectionOptions ont été fournies mais que les paramètres de codec aient été omis, les LocalConnectionOptions permettent implicitement tous les codecs de la liste interne, pourvu qu'ils ne soient pas incompatibles avec une des périodes de mise en paquets spécifiées. De même, si les LocalConnectionOptions ont été fournies mais que la ou les périodes de mise en paquets aient été omises, les LocalConnectionOptions contiennent implicitement l'ensemble des périodes de mise en paquets acceptées par la liste interne;
2) si la liste approuvée des codecs/périodes de mise en paquets est vide, il s'est produit un échec de négociation de codec et une réponse d'erreur est produite (le code d'erreur 534 – échec de négociation de codec – est recommandé);
3) autrement, une liste négociée de codecs/périodes de mise en paquets est formée en prenant l'intersection de la liste approuvée des codecs/périodes de mise en paquets et des codecs/périodes de mise en paquets permises par le RemoteConnectionDescriptor. S'il n'a pas été fourni de RemoteConnectionDescriptor, la liste négociée des codecs/périodes de mise en paquets contient donc la liste approuvée des codecs/périodes de mise en paquets. Si le RemoteConnectionDescriptor ne contient aucune ligne de flux de média, il est survenu un échec de négociation de codec et une réponse d'erreur est produite (le code d'erreur 534 – échec de négociation de codec – est recommandé). Si le RemoteConnectionDescriptor contient plusieurs flux de média, l'adaptateur MTA DEVRAIT n'accepter que l'un d'entre eux et rejeter les autres en mettant leur port à zéro dans le LocalConnectionDescriptor. Si le RemoteConnectionDescriptor a été fourni, mais que la ou les périodes de mise en paquets aient été omises, la liste négociée des périodes de mise en paquets contient l'ensemble des périodes de mise en paquets tirée de la liste approuvée. L'adaptateur MTA DOIT choisir des valeurs par défaut raisonnables d'après le document RFC 2327 si la période de mise en paquets est explicitement omise à la fois dans les LocalConnectionOptions et dans le RemoteConnectionDescriptor;
4) si la liste négociée des codecs/périodes de mise en paquets est vide, un échec de négociation de codec est survenu est une réponse d'erreur est produite (le code d'erreur 534 – échec de négociation de codec – est recommandé);
5) autrement, la négociation de codec a réussi, et la liste négociée des codecs/périodes de mise en paquets est retournée dans le LocalConnectionDescriptor.

Noter que l'intervalle de mise en paquets pour les procédures T.38 est sélectionné au moyen de la même procédure que pour les codecs audio, comme décrit ci-dessus.

Dans le cas où une passerelle n'accepte pas plus d'un codec par point d'extrémité, il y a deux options que peut utiliser la passerelle pour décider combien de codecs elle veut prendre en charge pour cette connexion:

1) la passerelle accepte plusieurs codecs et peut passer d'un codec à l'autre en temps réel. La passerelle retourne tous les codecs négociés dans la ligne SDP de flux de média. Plusieurs codecs dans la ligne m= signifient que l'appareil doit être prêt à recevoir des paquets de média de tout codec négocié. De même, la passerelle peut envoyer des paquets de média à partir de tout codec négocié et passer de l'un à l'autre en tant que de besoin;

2) la passerelle prend en charge un ou plusieurs codecs mais ne peut pas commuter entre différents codecs en temps réel. La passerelle négocie donc et retourne seulement un codec sur la ligne du flux de média SDP (en option, la passerelle met aussi les codecs acceptés supplémentaires dans l'attribut de protocole SDP "X-pc-codecs"). Avec cette méthode, un changement de codec doit être initialisé par le serveur CMS afin de changer de codec.

7.7.1 Négociation selon RFC 2833

La liste interne des codecs pris en charge DOIT comprendre le codeur-décodeur d'événement téléphonique, avec les événements 0 à 15. Ce message fera en sorte que le relais des tonalités DTMF selon la norme RFC 2833 soit utilisé pour la connexion quand cela est autorisé par le paramètre d'options LCO (par inclusion ou au moyen d'un paramètre a: vide) et quand cela est autorisé par le descripteur RCD.

Un exemple de paramètre LCO qui autorise le relais des tonalités DTMF selon la norme RFC 2833 est le suivant:

```
L: a:PCMU;PCMA;telephone-event, mp:10;20;-
```

Le codec d'événement téléphonique NE DOIT PAS être le seul codec fourni dans le paramètre d'options LCO à partir du contrôleur de passerelle média. Si l'extrémité reçoit un paramètre d'options LCO contenant seulement le codec d'événement téléphonique, alors elle DOIT renvoyer le code d'erreur 524 – incohérence interne dans les options de connexion locale. Si la liste approuvée de codecs, telle que décrite dans le § 7.7, contient seulement le codec d'événement téléphonique, l'extrémité DOIT renvoyer le code d'erreur 534 – échec de négociation de codec. De même, si la liste négociée des codecs, telle que décrite dans le § 7.7, contient seulement le codec d'événement téléphonique, l'extrémité DOIT renvoyer le code d'erreur 534.

Si le champ de période de mise en paquets est utilisé dans le paramètre d'options LCO pour la connexion, l'extrémité DOIT utiliser ce rythme de mise en paquets pour les paquets de relais DTMF. Si le champ de période de mise en paquets multiple est utilisé dans le paramètre d'options LCO, le contrôleur de passerelle média DOIT utiliser un trait d'union pour désigner le rythme de mise en paquets du codec d'événement téléphonique. Si l'extrémité reçoit un paramètre d'options LCO avec le champ de période de mise en paquets multiple contenant un rythme de mise en paquets pour codec d'événement téléphonique non réglé à un trait d'union, l'extrémité DOIT renvoyer le code d'erreur 524 – Paramètre d'options LCO incohérent. Quand une extrémité renvoie un descripteur LCD qui contient la capacité de recevoir le codec d'événement téléphonique, cette extrémité DOIT utiliser un trait d'union afin de désigner le rythme de mise en paquets dans l'attribut "mptime" du protocole SDP.

Un exemple de descripteur LCD qui signale la prise en charge de la numérotation par codage DTMF selon la norme RFC 2833 est le suivant:

```
v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1296 RTP/AVP 0 8 105
a=mptime:10 20 -
a=rtpmap:105 telephone-event/8000/1
```

Pour de plus amples informations sur l'utilisation de la numérotation par codage DTMF selon la norme RFC 2833, voir la spécification des codecs audio/vidéo IPCablecom, Rec. UIT-T J.161.

7.7.2 Négociation de l'adresse IP distante et du port distant

L'adresse IP distante et le port distant sont fournis par le Descripteur de connexion distante. Une fois ces informations obtenues dans une commande efficace de traitement de connexion (par

exemple une commande Modifier Connexion), l'extrémité DOIT continuer à utiliser ces informations jusqu'à ce qu'un nouveau Descripteur de connexion distante soit fourni, qui spécifie une nouvelle adresse IP distante ou un nouveau port distant, ou jusqu'à ce que la connexion soit supprimée. Noter que la réception d'une commande Modifier Connexion sans Descripteur de connexion distante (qui peut faire échouer la négociation du codec) n'invalide pas les informations actuelles d'adresse IP distante et de port distant, même si le média est modifié d'audio à image ou vice versa.

8 Protocole de commande de passerelle média

Le protocole MGCP implémente l'interface de commande de passerelle média comme un ensemble de transactions. Celles-ci sont composées d'une commande et d'une réponse obligatoire. Les types de commande sont au nombre de huit:

- CreateConnection;
- ModifyConnection;
- DeleteConnection;
- NotificationRequest;
- Notify;
- AuditEndpoint;
- AuditConnection;
- RestartInProgress.

Les quatre premières commandes sont envoyées par le contrôleur MGC à une passerelle. La commande Notify est envoyée par la passerelle au contrôleur MGC. La passerelle peut également envoyer une commande DeleteConnection, comme défini au § 7.3.6. Le contrôleur MGC peut envoyer une quelconque commande d'audit à la passerelle et, finalement, celle-ci peut envoyer une commande RestartInProgress au contrôleur MGC.

8.1 Description générale

Toutes les commandes sont composées d'un en-tête de commande qui, pour certaines d'entre elles, peut être suivi d'une description de session.

Toutes les réponses sont composées d'un en-tête de réponse qui, pour certaines d'entre elles, peut être suivi d'une description de session.

Les en-têtes et les descriptions de session sont codés au moyen d'un ensemble de lignes de texte, séparées par un caractère de retour à la ligne et d'avancement (ou, éventuellement de retour à la ligne seulement). Les en-têtes sont séparés des descriptions de session par un interligne.

Le protocole MGCP emploie un identificateur de transaction dont la valeur est comprise entre 1 et 999999999 pour corréler les commandes et les réponses. L'identificateur de la transaction est codé au moyen d'un composant de l'en-tête de commande et est répété au moyen d'un composant de l'en-tête de réponse.

8.2 En-tête de commande

L'en-tête de commande est composé des éléments suivants:

- une ligne de commande identifiant la mesure ou l'action demandée, l'identificateur de la transaction, l'extrémité pour laquelle la mesure est demandée, et la version du protocole MGCP;
- un ensemble de lignes de paramètres composées d'un nom de paramètre suivi de sa valeur.

Sauf indication ou stipulation contraires par d'autres normes en référence, tous les composants de l'en-tête de commande sont insensibles à la hauteur de casse. Cela vaut pour les actions ainsi que pour les paramètres et leurs valeurs, et toutes les comparaisons DOIVENT traiter les majuscules et les minuscules ainsi que leurs combinaisons comme étant égales.

8.2.1 Ligne de commande

Une ligne de commande est composée des éléments suivants:

- le nom de l'action demandée;
- l'identification de la transaction;
- le nom de l'extrémité ou des extrémités qui devraient exécuter la commande (dans les notifications ou les redémarrages, le nom de l'extrémité ou des extrémités qui émettent la commande);
- la version du protocole.

Ces quatre éléments sont codés au moyen de chaînes de caractères d'imprimerie ASCII séparés par des blancs, c'est-à-dire, par les caractères d'espace ASCII (0x20) ou de tabulation (0x09). Les passerelles de jonction DEVRAIENT employer un seul séparateur d'espace ASCII, mais elles DOIVENT être en mesure d'analyser des messages contenant des blancs supplémentaires.

8.2.1.1 Codage de l'action demandée

Les actions demandées sont codées au moyen de codes ASCII à quatre lettres majuscules ou minuscules (les comparaisons DOIVENT être insensibles à la hauteur de casse), comme défini dans le Tableau 5:

Tableau 5/J.171.1 – Codes de l'action demandée

Action	Code
CreateConnection	CRCX
ModifyConnection	MDCX
DeleteConnection	DLCX
NotificationRequest	RQNT
Notify	NTFY
AuditEndpoint	AUEP
AuditConnection	AUCX
RestartInProgress	RSIP

De nouvelles actions pourront être définies dans des versions ultérieures du protocole. Il pourrait être nécessaire, à des fins expérimentales, d'employer de nouvelles actions avant que celles-ci soient confirmées dans une version publiée du présent protocole. Les actions expérimentales devraient être identifiées par un code à quatre lettres commençant par la lettre X (p. ex. XPER).

Une passerelle qui reçoit une commande contenant une action expérimentale qu'elle ne prend pas en charge DOIT renvoyer une erreur (code d'erreur 511 – Extension non reconnue).

8.2.1.2 Identificateurs de transaction

Les identificateurs de transaction sont utilisés pour corréler les commandes et les réponses.

Une passerelle de jonction prend en charge les deux espaces distincts suivants de nom d'identificateur de transaction:

- un espace de nom d'identificateur de transaction pour l'envoi de transactions;
- un espace de nom d'identificateur de transaction pour la réception de transactions.

Les identificateurs de transaction pour les commandes qui sont envoyées à une passerelle de jonction donnée DOIVENT au minimum être uniques pendant toute la durée des transactions qui sont effectuées par l'ensemble des contrôleurs MGC commandant cette passerelle de jonction (voir § 8.5). Donc, indépendamment du contrôleur MGC émetteur, les passerelles de jonction peuvent toujours détecter les transactions reproduites en examinant simplement l'identificateur de transaction. La coordination entre les contrôleurs MGC en ce qui concerne ces identificateurs de transaction sort toutefois du cadre de la présente Recommandation.

Les identificateurs de transaction pour toutes les commandes envoyées en provenance d'une passerelle de jonction donnée DOIVENT être uniques pendant toute la durée des transactions (voir § 8.5), indépendamment du contrôleur MGC auquel la commande est envoyée. Donc, un contrôleur MGC peut toujours détecter une transaction reproduite provenant d'une passerelle de jonction à partir de la combinaison du nom de domaine de l'extrémité et de l'identificateur de transaction. La passerelle, pour sa part, peut toujours détecter un accusé de réception de réponse reproduit en examinant l'identificateur ou les identificateurs de transaction.

L'identificateur de transaction est codé au moyen d'une chaîne à neuf chiffres décimaux au plus. Dans les lignes de commande, il suit immédiatement le code de l'action.

Les valeurs des identificateurs de transaction sont comprises entre 1 et 999999999. Les identificateurs de transaction NE DEVRAIENT PAS utiliser de zéros à gauche (non signifiants). L'égalité est basée sur la valeur numérique, les zéros à gauche étant ignorés. Une entité de protocole MGCP NE DOIT PAS réutiliser un identificateur de transaction moins de trois minutes après l'achèvement d'une commande précédente où cet identificateur était employé.

8.2.1.3 Codage des extrémités, des contrôleurs de passerelle média et des noms de l'entité notifiée

Les noms des extrémités et des contrôleurs MGC sont codés au moyen d'adresses de courrier électronique, comme défini dans le document IETF RFC 2821. Dans ces adresses, le nom de domaine identifie le système auquel l'extrémité est rattachée, tandis que la partie de gauche identifie une extrémité particulière de ce système. Les deux composantes DOIVENT être insensibles à la hauteur de casse.

Des exemples de ces noms sont donnés ci-après:

ds/ds1-3/2@TGCP2.whatever.net	Deuxième circuit de la troisième interface DS1 dans la passerelle de jonction TGCP2 du réseau "quelconque".
MGC@mgc.whatever.net	Contrôleur de passerelle média du réseau "quelconque".

Le nom des entités notifiées s'exprime à l'aide de la même syntaxe, le numéro du port pouvant éventuellement être ajouté, comme dans l'adresse suivante:

```
MGC@mgc.whatever.net:5234
```

Dans le cas où le numéro du port est omis, le port par défaut pour le protocole MGCP (2727 sauf préconfiguration contraire) sera utilisé. On trouvera des précisions supplémentaires sur les noms d'extrémité au § 7.1.1.

8.2.1.4 Codage de la version du protocole

La version du protocole est codée au moyen du mot clé "MGCP" suivi d'un blanc et du numéro de la version, qui précède lui-même le nom de profil "TGCP" et un numéro de version de celui-ci. Les

numéros de version comportent un numéro de version principal, un point et un numéro de version secondaire. Les numéros principaux et secondaires sont codés au moyen de nombres décimaux. Le numéro de version du profil défini par la présente Recommandation est 1.0.

La version du protocole pour la présente Recommandation DOIT être codée comme suit:

```
MGCP 1.0 TGCP 1.0
```

La partie "TGCP 1.0" indique qu'il s'agit du profil TGCP 1.0 du protocole MGCP 1.0.

Une entité qui reçoit une commande contenant une version de protocole qu'elle ne prend pas en charge DOIT répondre par une erreur (code d'erreur 528 – version de protocole incompatible).

8.2.2 Lignes de paramètre

Les lignes de paramètre sont composées d'un nom de paramètre, qui dans la plupart des cas comporte un seul caractère en majuscule, suivi d'un double point, d'un blanc et d'une valeur pour le paramètre. Les noms et les valeurs de paramètre sont toutefois encore insensibles à la hauteur de casse. Les paramètres qui peuvent figurer dans les commandes sont définis dans le Tableau 6.

Tableau 6/J.171.1 – Paramètres de commande

Nom du paramètre	Code	Valeur du paramètre
ResponseAck (Note)	K	Voir la description
CallId	C	Chaîne hexadécimale à 32 caractères au plus
ConnectionId	I	Chaîne hexadécimale à 32 caractères au plus
NotifiedEntity	N	Identificateur, de format IETF RFC 821, composé d'une chaîne arbitraire et du nom de domaine de l'entité demandeuse, éventuellement complétés par un numéro de port, comme dans l'adresse suivante: Call-agent@ca.whatever.net:5234
RequestIdentifier	X	Chaîne hexadécimale du paramètre RequestIdentifier, la longueur NE DOIT PAS dépasser 32 caractères
LocalConnectionOptions	L	Voir la description
Connection Mode	M	Voir la description
RequestedEvents	R	Voir la description
SignalRequests	S	Voir la description
ObservedEvents	O	Voir la description
ConnectionParameters	P	Voir la description
ReasonCode	E	Voir la description
SpecificEndPointId	Z	Identificateur, de format IETF RFC 821, composé d'une chaîne arbitraire, suivie éventuellement du signe "@", lui-même précédant le nom de domaine de la passerelle de jonction à laquelle cette extrémité est rattachée
MaxEndPointIds	ZM	Chaîne décimale à 16 caractères au plus
NumEndPoints	ZN	Chaîne décimale à 16 caractères au plus
RequestedInfo	F	Voir la description
QuarantineHandling	Q	Voir la description
DetectEvents	T	Voir la description
EventStates	ES	Voir la description
RestartMethod	RM	Voir la description

Tableau 6/J.171.1 – Paramètres de commande

Nom du paramètre	Code	Valeur du paramètre
RestartDelay	RD	Nombre de secondes codé au moyen d'un nombre décimal
Capabilities	A	Voir la description
VersionSupported	VS	Voir la description
MaxMGCPDatagram	MD	Voir la description
NOTE – Le paramètre ResponseAck ne figure pas au § 7.3 parce que les identificateurs de transaction n'apparaissent pas dans notre exemple d'interface API. Les responsables chargés de l'implémentation peuvent choisir une approche différente.		

Les paramètres ne sont pas nécessairement présents dans toutes les commandes. Le Tableau 7 donne la relation entre les paramètres et les commandes. La lettre M signifie "obligatoire", O "facultatif" et F "interdit":

Tableau 7/J.171.1 – Association entre paramètres et commandes par requête

Nom du paramètre	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck (Note)	O	O	O	O	O	O	O	O
CallId	M	M	O	F	F	F	F	F
ConnectionId	F	M	O	F	F	F	M	F
RequestIdentifier	O	O	O	M	M	F	F	F
LocalConnectionOptions	O	O	F	F	F	F	F	F
ConnectionMode	M	O	F	F	F	F	F	F
RequestedEvents	O[a)]	O[a)]	O[a)]	O[a)]	F	F	F	F
SignalRequests	O[a)]	O[a)]	O[a)]	O[a)]	F	F	F	F
NotifiedEntity	O	O	O	O	O	F	F	F
ReasonCode	F	F	O	F	F	F	F	F
ObservedEvents	F	F	F	F	M	F	F	F
Connection parameters	F	F	O	F	F	F	F	F
SpecificEndpointId	F	F	F	F	F	O	F	F
MaxEndPointIds	F	F	F	F	F	O	F	F
NumEndPoints	F	F	F	F	F	F	F	F
RequestedInfo	F	F	F	F	F	O	O	F
QuarantineHandling	O	O	O	O	F	F	F	F
DetectEvents	O	O	O	O	F	F	F	F
EventStates	F	F	F	F	F	F	F	F
RestartMethod	F	F	F	F	F	F	F	M

Tableau 7/J.171.1 – Association entre paramètres et commandes par requête

Nom du paramètre	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
RestartDelay	F	F	F	F	F	F	F	O
Capabilities	F	F	F	F	F	F	F	F
VersionSupported	F	F	F	F	F	F	F	F
MaxMGCPDatagram	F	F	F	F	F	F	F	F
RemoteConnectionDescriptor	O	O	F	F	F	F	F	F

[a] Les paramètres RequestedEvents et SignalRequests sont facultatifs dans la demande NotificationRequest. S'ils sont omis, les listes correspondantes seront considérées comme vides. Pour les commandes de traitement des connexions, cela s'applique aussi lorsqu'un identificateur RequestIdentifier est présent.

NOTE – Le paramètre ResponseAck ne figure pas au § 7.3 parce que les identificateurs de transaction n'apparaissent pas dans notre exemple d'interface API. Les responsables chargés de l'implémentation peuvent choisir une approche différente.

Les passerelles de jonction et les contrôleurs MGC DEVRAIENT toujours fournir les paramètres obligatoires avant les paramètres facultatifs; cependant les passerelles de jonction NE DOIVENT PAS échouer lorsque la présente Recommandation n'est pas appliquée.

Si les responsables chargés de l'implémentation ont besoin d'expérimenter de nouveaux paramètres, p. ex. lors de l'élaboration d'une nouvelle application dans le cadre du protocole MGCP, ils devraient identifier ces paramètres au moyen de noms qui commencent par les chaînes "X-" ou "X+", comme dans l'exemple suivant:

```
X-FlowerOfTheDay: Daisy
```

Les noms de paramètre qui commencent par "X+" sont des extensions de paramètre obligatoires. Une passerelle qui reçoit une extension de paramètre obligatoire qu'elle ne comprend pas DOIT répondre par une erreur (code d'erreur 511 – extension non reconnue).

Les noms de paramètre qui commencent par "X-" sont des extensions de paramètre non critiques. Une passerelle qui reçoit une extension de paramètre non critique qu'elle ne comprend pas peut en toute sécurité ne pas tenir compte de ce paramètre.

Il convient de noter que les actions expérimentales sont de la forme X*ABC*, tandis que les paramètres expérimentaux sont de la forme X-ABC.

Lorsqu'une ligne de paramètre est reçue avec un paramètre interdit, ou une quelconque erreur de format, l'entité qui la reçoit devrait répondre par le code d'erreur le plus précis possible pour l'erreur en question. Le code d'erreur le moins précis est le code 510 – erreur de protocole. Un texte de commentaires peut toujours être fourni.

8.2.2.1 Accusé de réception de réponse

Le paramètre d'accusé de réception de réponse est utilisé pour prendre en charge le dialogue à trois décrit au § 8.7. Il contient une liste de "domaines d'identificateurs de transaction confirmée", séparés par des virgules.

Chaque "domaine d'identificateurs de transaction confirmée" est composé soit d'un nombre décimal, lorsque le domaine ne comprend qu'une transaction, soit de deux nombres décimaux séparés par un trait d'union unique, donnant les identificateurs inférieur et supérieur de transaction du domaine.

Un exemple d'accusé de réception de réponse est donné ci-après:

```
K: 6234-6255, 6257, 19030-19044
```

8.2.2.2 Identificateur de demande

Le paramètre RequestIdentifier établit une corrélation entre une commande Notify et la demande NotificationRequest qui l'a déclenchée (y compris une NotificationRequest imbriquée dans des primitives de traitement de connexion). Il a la forme d'une chaîne hexadécimale dont la longueur NE DOIT PAS dépasser 32 caractères. Les identificateurs de demande sont comparés en tant que chaînes plutôt qu'en tant que valeurs numériques. La chaîne "0" est réservée pour la signalisation d'événements durables lorsque aucune demande NotificationRequest n'a encore été reçue (voir § 7.3.2).

8.2.2.3 Options locales de connexion

Les options locales de connexion décrivent les paramètres opérationnels que les contrôleurs MGC ordonnent à la passerelle d'employer pour une connexion. Ces paramètres sont les suivants:

- la période de mise en paquets en millisecondes, codée au moyen du mot clé "p" suivi d'un double point et d'un nombre décimal;
- la période de mise en paquets multiple en millisecondes pour chaque codec des options LCO de méthode de codage, codée sous la forme du mot clé "mp" suivi de deux points et d'une liste de nombres décimaux ou de traits d'union, avec une entrée pour chaque entrée dans le champ de Méthode de codage. Chaque valeur de période de mise en paquets est séparée de son successeur par un seul point-virgule. La première entrée dans la liste DOIT être un nombre décimal. Les entrées suivantes dans la liste DOIVENT être soit un nombre décimal, soit un trait d'union;
- le nom littéral de l'algorithme de compression tel que spécifié dans la Rec. UIT-T J.161, codé sous la forme du mot clé "a" suivi de deux points et d'une chaîne de caractères. Si le contrôleur de passerelle média (MGC) spécifie une liste de valeurs, celles-ci seront séparées par un point-virgule. Dans le protocole RTP, les codecs audio DOIVENT être spécifiés au moyen de noms de codage définis dans le Profil AV du protocole RTP RFC 1890, au moyen de noms de codage enregistrés auprès de l'autorité IANA, ou au moyen de noms de codage cités en référence ou définis dans la Spécification des codecs audio/vidéo IPCablecom. Les médias non audio enregistrés en tant que type d'extension MIME DOIVENT utiliser le formalisme "<type d'extension MIME>/<sous-type d'extension MIME>", comme dans l'expression "image/t38". Il est RECOMMANDÉ de prendre en charge également les autres variantes usuelles de noms de codec littéraux;
- le paramètre de compensation d'écho, codé au moyen du mot clé "e" suivi d'un double point et de la valeur "on" ou "off";
- le paramètre de type de service, codé au moyen du mot clé "t" suivi d'un double point et de la valeur codée par deux chiffres hexadécimaux;
- le paramètre de suppression des silences, codé au moyen du mot clé "s" suivi d'un double point et de la valeur "on" ou "off".

Les paramètres LocalConnectionOptions utilisés pour la sécurité sont codés de la manière suivante:

- la suite cryptographique RTP est codée au moyen du mot clé "sc-rtp" suivi d'un double point et d'une chaîne de suite cryptographique RTP comme défini ci-dessous. Une liste de valeurs peut être spécifiée, auquel cas ces valeurs DOIVENT être séparées par un unique point-virgule;
- la suite cryptographique RTCP est codée au moyen du mot clé "sc-rtcp" suivi d'un double point et d'une chaîne de suite cryptographique RTCP, comme défini ci-dessous. Une liste de valeurs peut être spécifiée, auquel cas ces valeurs DOIVENT être séparées par un unique point-virgule.

Les chaînes relatives aux suites cryptographiques pour les protocoles RTP et RTCP suivent le formalisme suivant:

ciphersuite =	[AuthenticationAlgorithm] "/" [EncryptionAlgorithm]
AuthenticationAlgorithm =	1*(ALPHA / DIGIT / "-" / "_")
EncryptionAlgorithm =	1*(ALPHA / DIGIT | "-" / "_")

où ALPHA et DIGIT sont définis dans le document IETF RFC 2234. Les blancs ne sont pas admis dans une suite cryptographique ni entre suites cryptographiques adjacentes lorsque de multiples suites cryptographiques sont fournies. L'exemple suivant illustre l'emploi d'une suite cryptographique et d'une liste de suites cryptographiques:

```
sc-rtp 62/51;64/51;60/50
```

La liste en vigueur des suites cryptographiques prises en charge dans les réseaux IPCablecom est fournie dans la Rec. UIT-T J.170.

Lorsque plusieurs paramètres sont présents, les valeurs sont séparées par des virgules. L'introduction d'un paramètre sans valeur DOIT être considérée comme une erreur (code d'erreur 524 – incohérence dans le paramètre LocalConnectionOptions).

Des exemples d'options locales de connexion sont donnés ci-après:

```
L: p:10, a:PCMU
L: p:10, a:PCMU, e:off, t:20, s:on
L: p:30, a:G729, e:on, t:A0, s:off
```

Le type de service à valeur hexadécimale "20" implique une priorité IP égale à 1 tandis qu'un type "A0" implique une priorité IP égale à 5.

Cet ensemble d'attributs peut être élargi au moyen d'attributs d'extension. Ces attributs d'extension sont composés d'un nom d'attribut, suivi d'un double point, et d'une liste de valeurs d'attribut séparées par des points-virgules. Le nom d'attribut DOIT commencer par les deux caractères "x+" pour une extension obligatoire ou "x-" pour une extension non obligatoire. Si une passerelle reçoit un attribut d'extension obligatoire qu'elle ne reconnaît pas, elle DOIT rejeter la commande avec une erreur (code d'erreur 525 – extension inconnue dans le paramètre LocalConnectionOptions).

8.2.2.4 Capacités

Les capacités informent le contrôleur MGC sur ses capacités lorsqu'il est soumis à un audit. Le codage des capacités est fondé sur le codage des paramètres LocalConnectionOptions pour les paramètres qui sont communs aux deux. En outre, les capacités peuvent aussi contenir une liste des paquetages pris en charge et une liste des modes pris en charge.

Les paramètres utilisés sont les suivants:

- la période de mise en paquets exprimée en millisecondes, codée au moyen du mot clé "p" suivi d'un double point et d'un nombre décimal. Un domaine peut être spécifié au moyen de deux nombres décimaux séparés par un trait d'union;
- le nom litéral de l'algorithme de compression, codé au moyen du mot clé "a" suivi d'un double point et d'une chaîne de caractères. Les noms littéraux définis dans le Tableau 3 de la spécification des codecs audio/vidéo IPCablecom (J.161) DOIVENT être utilisés. Une liste de valeurs peut être spécifiée, auquel cas ces valeurs seront séparées par des points-virgules;
- la largeur de bande en kilobits par seconde (1000 bit/s), codée au moyen du mot clé "b" suivi d'un double point et d'un nombre décimal. Un domaine peut être spécifié au moyen de deux nombres décimaux séparés par un trait d'union;

- le paramètre de compensation d'écho, codé au moyen du mot clé "e" suivi d'un double point et de la valeur "on" si la compensation d'écho est effectuée, et "off" dans les autres cas;
- le paramètre de type de service, codé au moyen du mot clé "t" suivi d'un double point et d'une valeur "0" si le type de service n'est pas pris en charge, les autres valeurs indiquant la prise en charge du type de service;
- le paramètre de suppression des silences, codé au moyen du mot clé "s" suivi d'un double point et de la valeur "on" lorsque la suppression des silences est prise en charge, et "off" dans les autres cas;
- les paquetages d'événements pris en charge par cette extrémité, codés au moyen du mot clé "v" suivi d'un double point et d'une liste des noms de paquetages pris en charge, séparés par des points-virgules. La première valeur spécifiée correspondra au paquetage par défaut pour l'extrémité;
- les modes de connexion pris en charge par cette extrémité, codés au moyen du mot clé "m" suivi d'un double point et d'une liste des modes de connexion pris en charge, séparés par des points-virgules, comme défini au § 8.2.2.7;
- le mot clé "sc-rtp" suivi d'un double point et d'une liste de suites cryptographiques pour le protocole RTP séparées par des points-virgules, utilisant le même codage que dans le paramètre LocalConnectionOptions;
- le mot clé "sc-rtcp" suivi d'un double point et d'une liste de suites cryptographiques pour le protocole RTCP séparées par des points-virgules, utilisant le même codage que dans le paramètre LocalConnectionOptions.

Lorsque plusieurs paramètres sont présents, les valeurs sont séparées par des virgules.

Des exemples de capacité sont donnés ci-après:

```
A: a:PCMU; p:10-30, e:on, s:off, v:IT,
              m:sendonly;recvonly;sendrecv;inactive
A: a:G729; p:30-90, e:on, s:on, v:IT,
              m:sendonly;recvonly;sendrecv;inactive,
              sc-rtp: 64/51;60/51, sc-rtcp:71/81
```

Il convient de noter que les codecs et les algorithmes de sécurité ne sont donnés qu'à titre d'exemple – des Recommandations IPCablecom distinctes donnent des précisions concernant les codecs et les algorithmes concrètement pris en charge, ainsi que le codage utilisé (voir les Recommandations UIT-T J.170, J.162 et J.161). Noter que les codecs et les algorithmes de sécurité sont seulement des exemples. Des spécifications IPCablecom distinctes précisent les codecs et algorithmes réels pris en charge ainsi que le codage utilisé (voir RFC 1827 et RFC 1034). Noter aussi que chaque ensemble de capacités est fourni sur une seule ligne. Les exemples ci-dessus montrent chaque ensemble sur plusieurs lignes à cause des seules contraintes de formatage de la présente Recommandation.

8.2.2.5 Paramètres de connexion

Les paramètres de connexion sont codés au moyen d'une chaîne de paires formées du type et de la valeur, où le type est un des codes du tableau ci-dessous et où la valeur est un entier décimal. Les types sont séparés des valeurs par le signe "=". Les paramètres sont séparés les uns des autres par une virgule.

Les types des paramètres de connexion sont spécifiés dans le Tableau 8.

Tableau 8/J.171.1 – Types de paramètres de connexion

Nom du paramètre de connexion	Code	Valeur du paramètre de connexion
Paquets envoyés	PS	Nombre de paquets qui ont été envoyés par l'intermédiaire de la connexion
Octets envoyés	OS	Nombre d'octets qui ont été envoyés par l'intermédiaire de la connexion
Paquets reçus	PR	Nombre de paquets qui ont été reçus par l'intermédiaire de la connexion
Octets reçus	OR	Nombre d'octets qui ont été reçus par l'intermédiaire de la connexion
Paquets perdus	PL	Nombre de paquets qui n'ont pas été reçus par l'intermédiaire de la connexion, obtenu à partir des lacunes parmi les numéros de séquence
Gigue	JI	Gigue moyenne entre l'arrivée des paquets, exprimée comme un nombre entier de millisecondes
Latence	LA	Latence moyenne, exprimée comme un nombre entier de millisecondes
Paquets distants envoyés	PC/RPS	Nombre de paquets envoyés sur la connexion du point de vue de l'extrémité distante
Octets distants envoyés	PC/ROS	Nombre d'octets envoyés sur la connexion du point de vue de l'extrémité distante
Paquets distants perdus	PC/RPL	Nombre de paquets qui n'ont pas été reçus sur la connexion, déduit des lacunes dans les numéros de séquence, du point de vue de l'extrémité distante
Gigue distante	PC/RJI	Gigue moyenne du délai entre arrivées de paquets, en millisecondes, exprimée en nombre entier, du point de vue de l'extrémité distante

Les noms des paramètres d'extension de connexion sont composés de la chaîne "X-" suivie d'un nom de paramètre d'extension à deux lettres. Les contrôleurs MGC qui reçoivent des extensions non reconnues DOIVENT, sans notification, ne pas tenir compte de ces extensions. Si une extrémité reçoit des paquets RTCP avec ces statistiques, elle DOIT retourner les paramètres distants (Rxx ci-dessus) dans sa réponse aux commandes Supprimer Connexion et Audit de connexion.

Un exemple du codage d'un paramètre de connexion est donné ci-après:

```
P: PS=1245, OS=62345, PR=0, OR=0, PL=0, JI=0, LA=48, PC/RPS=0, PC/ROS=0,
PC/RPL=0, PC/RJI=0
```

8.2.2.6 Codes de cause

Les codes de cause sont des valeurs numériques à trois chiffres. Il sont suivis en option d'un blanc et de commentaires, p. ex.:

```
E:   900 Mauvais fonctionnement de l'extrémité
```

On trouvera une liste des codes de cause au § 7.6.

8.2.2.7 Mode de connexion

Le mode de connexion décrit la façon dont fonctionne la connexion. Les valeurs possibles sont indiquées dans le Tableau 9.

Tableau 9/J.171.1 – Valeurs du mode de connexion

Mode	Signification
M: sendonly	La passerelle devrait seulement envoyer des paquets
M: recvonly	La passerelle devrait seulement recevoir des paquets
M: sendrecv	La passerelle devrait envoyer et recevoir des paquets
M: inactive	La passerelle ne devrait ni envoyer ni recevoir des paquets
M: loopback	La passerelle devrait placer l'extrémité en mode de bouclage
M: conttest	La passerelle devrait placer l'extrémité en mode d'essai de continuité
M: netwloop	La passerelle devrait placer l'extrémité en mode de bouclage en réseau
M: netwtest	La passerelle devrait placer l'extrémité en mode d'essai de continuité en réseau

8.2.2.8 Codage du nom d'événement ou de signal

Les noms d'événement ou de signal sont composés d'un nom de paquetage facultatif, séparé par une barre oblique (/) du nom de l'événement effectif. Le nom d'événement peut éventuellement être suivi du signe arobase (@) et de l'identificateur de la connexion où l'événement devrait être observé. Les noms d'événement sont utilisés dans les paramètres RequestedEvents, SignalRequests, DetectEvents, ObservedEvents et EventStates. Chaque événement est identifié par un code d'événement. Ces codages ASCII ne sont pas sensibles à la hauteur de casse. Des valeurs telles que "co", "Co", "CO" ou "cO" devraient être considérées comme étant égales.

Les noms d'événement suivants constituent des exemples valables:

`IT/co1`	Essai de continuité à l'émission dans le paquetage de jonction ISUP
`MT/oc`	Opération complète dans le paquetage du protocole de terminaison multifréquence
`co1`	Essai de continuité à l'émission dans le paquetage de jonction ISUP, en supposant que ce paquetage est prescrit par défaut pour l'extrémité
`IT/rt@0A3F58`	Retour d'appel sonore pour la connexion "0A3F58"

On peut en outre désigner les événements au moyen de caractères de remplacement, au lieu de les nommer individuellement, dans les paramètres RequestedEvents et DetectEvents (mais pas dans les paramètres SignalRequests, ObservedEvents ou EventStates):

`IT/all`	Tous les événements du paquetage de jonction ISUP

Enfin, le signe astérisque peut être employé pour désigner "toutes les connexions" et le signe dollar pour désigner la "connexion actuelle". Les notations suivantes constituent des exemples valides:

`IT/ma@*`	Evénement de démarrage de média RTP dans toutes les connexions pour l'extrémité
`IT/rt@$`	Retour d'appel sonore pour la connexion actuelle

On trouvera un ensemble initial de paquetages d'événements pour les passerelles de jonction à l'Annexe A.

8.2.2.9 Paramètre RequestedEvents

Le paramètre RequestedEvents fournit la liste des événements qui ont été demandés. Les codes d'événement actuellement définis sont décrits à l'Annexe A. Chaque événement peut être spécifié par une mesure demandée, ou par une liste de mesures. Toutes les mesures ne peuvent pas être combinées – Voir le § 7.3.1 pour les combinaisons valables. Ces mesures, lorsqu'elles sont spécifiées, sont codées au moyen d'une liste de mots clés compris entre des parenthèses et séparés par des virgules. Ces codes, pour les différentes mesures, sont indiqués dans le Tableau 10 ci-après.

Tableau 10/J.171.1 – Codes de mesures

Mesure	Code
Notifier immédiatement	N
Recueillir	A
Ne pas tenir compte	I
Garder le ou les signaux activés	K
Insérer la demande NotificationRequest	E
Insérer la demande ModifyConnection	C

Lorsque aucune mesure n'est spécifiée, la mesure par défaut est la notification de l'événement. Cela signifie que "ft" et "ft(N)" sont p. ex. équivalents. Des événements qui ne font pas partie d'une liste sont ignorés, sauf lorsqu'il s'agit d'événements durables.

La liste des événements demandés est codée sur une seule ligne, les groupes d'événements ou de mesures étant séparés par des virgules. Un exemple de codage du paramètre RequestedEvents est donné ci-après:

`R: oc(N), of(N)` Notifier l'achèvement de l'opération, notifier l'échec de l'opération.

Le format de la demande NotificationRequest imbriquée est le suivant:

```
E ( R( <RequestedEvents>), S( <SignalRequests>) )
```

chacune des grandeurs R et S étant facultative et éventuellement fournie dans un ordre différent.

Le format de la mesure imbriquée ModifyConnection est le suivant:

```
C(M(<ConnectionMode1>( <ConnectionID1> )) , ... ,
                    M(<ConnectionModen>(ConnectionIDn )))
```

L'exemple suivant illustre l'emploi de la demande imbriquée ModifyConnection:

```
R: ma@23B34D(A, C(M(sendrecv($)))), oc(N), of(N)
```

Lors du démarrage d'un média pour la connexion "23B34D", changer le mode de connexion et passer de "connection effective" à "envoyer et recevoir". Notifier les événements en ce qui concerne "l'opération achevée" et "l'opération échouée".

8.2.2.10 Paramètre SignalRequests

Le paramètre SignalRequests fournit le nom des signaux qui ont été demandés. On trouvera les signaux actuellement définis à l'Annexe A. Un signal donné ne peut figurer qu'une fois dans la liste, et tous les signaux seront appliqués, par définition, en même temps. La passerelle de média DOIT prendre en charge, au minimum, un signal unique sur chaque extrémité et prendre en charge simultanément la création d'un signal sur chaque connexion pour une extrémité donnée. Des paquetages spécifiques PEUVENT définir des exigences allant au-delà de ces capacités minimales.

Pour les combinaisons de signaux allant au-delà des exigences minimales que la passerelle de média ne prend pas en charge, celle-ci DEVRAIT retourner le code d'erreur 502.

Certains signaux peuvent être qualifiés par des paramètres de signal. Lorsqu'un signal est qualifié par plusieurs paramètres de signal, ceux-ci sont séparés par des virgules. Chaque paramètre de signal DOIT posséder le format spécifié ci-après (des blancs étant admis):

signal-parameter = signal-parameter-value /

signal-parameter-name "="signal-parameter-value /

signal-parameter-name "(" signal-parameter-list ")"

signal-parameter-list = signal-parameter-value 0*("," signal-parameter-value)

où la production signal-parameter-value peut être soit une chaîne soit une chaîne entre apostrophes doubles. Deux caractères d'apostrophes doubles consécutif dans une chaîne entre apostrophes doubles produira par échappement un seul caractère d'apostrophes doubles. Par exemple, la chaîne `"ab""c"` produira la chaîne `ab"c`.

Chaque signal est d'un des types de signal suivants, qui lui sont associés (voir § 7.3.1):

- On/Off (OO) (activé/désactivé)
- Time-out (TO) (temporisé)
- Brief (BR) (bref)

Les signaux OO peuvent être paramétrés au moyen d'un signe "+" pour activer le signal, ou d'un signe "-" pour le désactiver. Lorsqu'un signal OO n'est pas paramétré, il est activé. Les deux commandes suivantes activeront le signal "mysignal":

```
mysignal(+), mysignal
```

Les signaux TO peuvent être paramétrés au moyen du paramètre de signal "TO" et d'une temporisation qui l'emporte sur la valeur par défaut. Lorsqu'un signal TO n'est pas paramétré au moyen d'une temporisation, la valeur de temporisation par défaut sera utilisée. Les deux commandes suivantes se traduiront par un signal de tonalité de retour d'appel sonore d'une durée de 6 secondes:

```
rt(to=6000)
rt(to(6000))
```

Des signaux individuels peuvent définir des paramètres de signal supplémentaires.

Les paramètres de signal seront placés entre parenthèses, comme dans l'exemple hypothétique suivant:

```
S: display(10/14/17/26, "555 1212", CableLabs)
```

Lorsque plusieurs signaux sont demandés, leurs codes sont séparés par des virgules, comme indiqué ci-après:

```
S: signal1, signal2
```

8.2.2.11 Paramètre ObservedEvents

Le paramètre ObservedEvents fournit la liste des événements qui ont été observés. Les codes d'événement sont les mêmes que ceux qui sont utilisés dans la demande NotificationRequest. Lorsqu'un événement est détecté et observé dans une connexion, il permet d'identifier la connexion dans laquelle il a été détecté au moyen de la syntaxe "@<connection>". Des exemples d'événements observés sont donnés ci-après:

```
O: ma@A43B81
O: ft
O: IT/ft
O: IT/ft, IT/mt
```

8.2.2.12 Paramètre RequestedInfo

Le paramètre RequestedInfo contient une liste de codes de paramètre séparés par des virgules, comme défini dans le § 8.2.2 sur les lignes de paramètre, le § 7.3.8 donne une liste des paramètres qui peuvent faire l'objet d'un audit. Les valeurs du Tableau 11 suivant sont également prises en charge.

Tableau 11/J.171.1 – Valeurs du paramètre RequestedInfo prises en charge

Paramètre RequestedInfo	Code
LocalConnectionDescriptor	LC
RemoteConnectionDescriptor	RC

Si l'on souhaite, p. ex. soumettre à un audit les valeurs des paramètres NotifiedEntity, RequestIdentifier, RequestedEvents, SignalRequests, DetectEvents, EventStates, LocalConnectionDescriptor et RemoteConnectionDescriptor, les valeurs du paramètre RequestedInfo seront les suivantes:

```
F: N, X, R, S, T, ES, LC, RC
```

La demande de capacités, pour la commande AuditEndpoint, est codée au moyen du code de paramètre "A" comme dans l'expression suivante:

```
F: A
```

8.2.2.13 Paramètre QuarantineHandling

Le paramètre de traitement de quarantaine contient une liste de mots clé séparés par une virgule:

- le mot clé "process" (*traiter*) ou "discard" (*rejeter*) pour indiquer le traitement des événements mis en quarantaine et observés. Si ni traiter ni rejeter n'est présent, on suppose traiter;
- le mot clé "step" (*étape*) ou "loop" (*boucle*) pour indiquer si au plus une modification est attendue, ou si plusieurs modifications sont autorisées. Si ni "étape" ni "boucle" n'est présent, on suppose "étape". La prise en charge de ces deux mots clés est obligatoire.

Les valeurs suivantes sont des exemples valides:

```
Q: loop
Q: process
Q: loop discard
```

8.2.2.14 Paramètre DetectEvents

Le paramètre DetectEvents est codé au moyen d'une liste d'événements séparés par des virgules, p. ex.:

```
T: ft, mt
```

Il convient de noter qu'aucune mesure ne peut être associée aux événements.

8.2.2.15 Paramètre EventStates

Le paramètre EventStates est codé au moyen d'une liste d'événements séparés par des virgules, p. ex.:

```
ES: MO/rlc
```

Il convient de noter qu'aucune mesure ne peut être associée aux événements.

8.2.2.16 Paramètre RestartMethod

Le paramètre RestartMethod est codé au moyen de l'un des mots clés "graceful", "cancel-graceful", "forced", "restart" ou "disconnected", p. ex.:

```
RM: restart
```

8.2.2.17 Paramètre VersionSupported

Le paramètre VersionSupported est codé au moyen d'une liste de versions prises en charge, séparées par des virgules, p. ex.:

```
VS: MGCP 1.0, MGCP 1.0 TGCP 1.0
```

8.2.2.18 Paramètre CallIdentifier

Le paramètre CallIdentifier est codé comme une chaîne hexadécimale, d'au plus 32 caractères. Les identificateurs d'appel sont comparés comme chaînes plutôt que comme valeurs numériques.

8.2.2.19 Paramètre ConnectionIdentifier

Le paramètre ConnectionIdentifier est codé comme une chaîne hexadécimale, d'au plus 32 caractères. Les identificateurs de connexion sont comparés comme chaînes plutôt que comme valeurs numériques.

8.2.2.20 Paramètre MaxMGCPDatagram

Le paramètre MaxMGCPDatagram est codé comme une chaîne d'un maximum de neuf chiffres décimaux – des zéros initiaux ne sont pas autorisés. L'exemple ci-après illustre l'utilisation de ce paramètre:

```
MD: 8100
```

8.3 Formats d'en-tête de réponse

L'en-tête de réponse est composé d'une ligne de réponse suivie en option d'en-têtes qui codent les paramètres de réponse.

La ligne de réponse commence par un code de réponse qui est une valeur numérique à trois chiffres. Ce code est suivi d'un blanc, d'un identificateur de transaction et d'un commentaire facultatif, précédé d'un blanc, p. ex. dans l'expression suivante:

```
200 1201 OK
```

Le Tableau 12 résume les paramètres de réponse qui doivent obligatoirement ou facultativement figurer dans un en-tête de réponse, en fonction de la commande qui a déclenché la réponse, en supposant que la commande a réussi. Le lecteur est toutefois prié d'examiner les différentes définitions des commandes parce que le présent tableau ne donne qu'un résumé des informations. La lettre M signifie obligatoire, O facultatif et F interdit.

Tableau 12/J.171.1 – Association des paramètres avec les réponses de commande

Nom du paramètre	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck (Note 5)	O (Note 1)	O (Note 1)	O (Note 1)	O (Note 1)	O (Note 1)	O (Note 1)	O (Note 1)	O (Note 1)
CallId	F	F	F	F	F	F	O	F
ConnectionId	O (Note 2)	F	F	F	F	O	F	F
RequestIdentifier	F	F	F	F	F	O	F	F
LocalConnectionOptions	F	F	F	F	F	O	O	F
ConnectionMode	F	F	F	F	F	F	O	F
RequestedEvents	F	F	F	F	F	O	F	F
SignalRequests	F	F	F	F	F	O	F	F
NotifiedEntity	F	F	F	F	F	O	O	O
ReasonCode	F	F	F	F	F	F	F	F
ObservedEvents	F	F	F	F	F	O	F	F
ConnectionParameters	F	F	O (Note 3)	F	F	F	O	F
Specific Endpoint ID	O	F	F	F	F	O	F	F
MaxEndPointIds	F	F	F	F	F	F	F	F
NumEndPoints	F	F	F	F	F	O	F	F
RequestedInfo	F	F	F	F	F	F	F	F
QuarantineHandling	F	F	F	F	F	F	F	F
DetectEvents	F	F	F	F	F	O	F	F
EventStates	F	F	F	F	F	O	F	F
RestartMethod	F	F	F	F	F	F	F	F
RestartDelay	F	F	F	F	F	F	F	F
Capabilities	F	F	F	F	F	O	F	F
VersionSupported	F	F	F	F	F	O	F	O
LocalConnection Descriptor	O (Note 4)	O (Note 4)	F	F	F	F	O	F
MaxMGCPDatagram	F	F	F	F	F	O	F	F
RemoteConnection Descriptor	F	F	F	F	F	F	O	F

NOTE 1 – Le paramètre ResponseAck NE DOIT PAS être utilisé avec des réponses autres qu'une réponse finale émise après une réponse provisoire en ce qui concerne la transaction en question. Dans ce cas, la présence du paramètre ResponseAck DOIT déclencher un message d'accusé de réception de réponse – Il ne sera pas tenu compte des valeurs ResponseAck fournies.

NOTE 2 – Dans le cas d'un message CreateConnection, la ligne de réponse est suivie par un paramètre d'identificateur de connexion et par un Descripteur de connexion locale. Cette ligne peut également être suivie par un paramètre d'identificateur d'extrémité spécifique, si la demande de création a été envoyée à un identificateur d'extrémité remplacé par une structure générique. Les paramètres Identificateur de connexion et Descripteur de connexion locale sont marqués comme étant facultatifs dans le tableau. En fait, ils sont obligatoires avec toutes les réponses favorables quand une connexion a été créée, et sont interdits quand la réponse est défavorable et qu'aucune connexion n'a été créée.

Tableau 12/J.171.1 – Association des paramètres avec les réponses de commande

NOTE 3 – Les paramètres de connexion ne sont valides que dans une réponse efficace à une commande Supprimer Connexion non remplacée par une structure générique et envoyée par l'agent d'appel.

NOTE 4 – Un paramètre Descripteur de connexion locale DOIT être transmis avec une réponse favorable (code 200) à une commande Créer connexion. Il DOIT également être transmis en réponse à une commande Modifier Connexion si cette modification se traduit par un changement du Descripteur de connexion locale. Celui-ci est codé comme une "description de session", comme défini dans le § 8.4. Il est séparé de l'en-tête de réponse par une ligne vide.

NOTE 5 – Le paramètre ResponseAck ne figure pas au § 7.3 parce que les identificateurs de transaction n'apparaissent pas dans notre exemple d'interface API. Les responsables chargés de l'implémentation peuvent choisir une approche différente.

Les paramètres de réponse sont décrits pour chaque commande dans les paragraphes qui suivent.

8.3.1 Commande CreateConnection

Dans le cas d'un message CreateConnection, la ligne de réponse est suivie d'un paramètre identificateur de connexion contenant une réponse de réussite (code 200). Un paramètre LocalConnectionDescriptor est en outre transmis avec une réponse favorable. Il est codé au moyen d'une "description de session", telle qu'elle est définie au § 8.4. Il est séparé de l'en-tête de réponse par un interligne, à savoir:

```
200 1204 OK
I: FDE234C8

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 18 96 97 0
a=rtpmap:96 G726-32/8000
a=rtpmap:97 telephone-event/8000
a=mptime:20 10 – 10
```

Lorsqu'une réponse provisoire a été donnée précédemment, la réponse finale peut en outre contenir le paramètre accusé de réception de réponse, comme dans ce qui suit:

```
200 1204 OK
K:
I: FDE234C8

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 18 96 97 0
a=rtpmap:96 G726-32/8000
a=rtpmap:97 telephone-event/8000
a=mptime:20 10 – 10
```

Il est accusé réception de la réponse finale au moyen du paramètre accusé de réception de réponse:

```
000 1204
```

8.3.2 Commande ModifyConnection

Dans le cas d'un message ModifyConnection de réussite, la ligne de réponse est suivie d'un paramètre LocalConnectionDescriptor, lorsque la modification a conduit à modifier les paramètres de session (le seul changement du mode de connexion ne modifie pas les paramètres de session, p. ex.). Ce paramètre est codé au moyen d'une "description de session", telle qu'elle est définie au § 8.4. Il est séparé de l'en-tête de réponse par un interligne.

```
200 1207 OK

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime: 20
```

Lorsqu'une réponse provisoire a été donnée précédemment, la réponse finale peut en outre contenir le paramètre accusé de réception de réponse, comme dans ce qui suit:

```
526 1207 Aucune largeur de bande
K:
```

Il est accusé réception de la réponse finale au moyen du paramètre accusé de réception de réponse:

```
000 1207 OK
```

8.3.3 Commande DeleteConnection

En fonction de la version du message DeleteConnection, la ligne de réponse peut être suivie par une ligne de paramètre Paramètres de connexion, comme défini au 8.2.2.5.

```
250 1210 OK
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

8.3.4 Commande NotificationRequest

La réponse à une commande NotificationRequest ne contient aucun paramètre de réponse supplémentaire.

8.3.5 Commande Notify

La réponse à une commande Notify ne contient aucun paramètre de réponse supplémentaire.

8.3.6 Commande AuditEndpoint

Dans le cas d'une commande AuditEndpoint, la ligne de réponse peut être suivie des informations sur chacun des paramètres demandés – Chaque paramètre figurera sur une ligne distincte. Les paramètres pour lesquels aucune valeur concrète n'existe seront quand même fournis. Tout nom local d'extrémité "prolongé" par un caractère de remplacement figurera sur une ligne distincte au moyen du code de paramètre "SpecificEndpointId", p. ex.:

```
200 1200 OK
Z: ds/ds1-1/1@tgw.whatever.net
Z: ds/ds1-1/2@tgw.whatever.net
ZN: 24
```

Un exemple de réponse à un message AuditEndPoint contenant un nom d'extrémité sans caractère générique est donné ci-après. Noter que SpecificEndPointId n'est pas fourni dans ce cas. Noter aussi que chaque ensemble de capacités est fourni sur une seule ligne. L'exemple ci-dessous montre chaque ensemble sur plusieurs lignes du fait des contraintes de format de la présente Recommandation.

```
200 1200 OK
A: a:PCMU, p:10, e:on, s:off, t:1, v:X,
     m:sendonly;recvonly;sendrecv;inactive
A: a:G728, p:20, e:on, s:off, t:1, v:L,
     m:sendonly;recvonly;sendrecv;inactive
A: a:G729, p:30, e:on, s:on, t:1, v:X,
     m:sendonly;recvonly;sendrecv;inactive;confrnce
```

8.3.7 Commande AuditConnection

Dans le cas d'une commande AuditConnection, la réponse peut être suivie des informations sur chacun des paramètres demandés. Les paramètres pour lesquels aucune valeur concrète n'existe seront quand même fournis. Les descripteurs de connexion figureront toujours en dernière position et chacun sera précédé d'un interligne, p. ex.:

```
2200 1203 OK
C: A3C47F21456789F0
N: [128.96.41.12]
L: mp:20;10, a:PCMU;G728
M: sendrecv
P: PS=622, OS=31172, PR=390, OR=22561, PL=5, JI=29, LA=50

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1296 RTP/AVP 96
a=rtpmap:96 G728/8000
a=mptime: 10
```

Si aussi bien un descripteur de connexion locale qu'un descripteur de connexion distante sont fournis, le descripteur de connexion locale sera le premier des deux. Si un descripteur de connexion est demandé, mais qu'il n'existe pas pour la connexion faisant l'objet d'un audit, ce descripteur figurera seulement dans le champ destiné à la version du protocole SDP.

8.3.8 Commande RestartInProgress

La réponse à une commande RestartInProgress peut comprendre le nom d'un autre contrôleur MGC à consulter, p. ex. lorsque le contrôleur MGC réachemine l'extrémité vers un autre contrôleur MGC comme dans ce qui suit:

```
521 1204 Redirect
N: MGC-1@whatever.net
```

8.4 Codage de la description de session

La description de session est codée conformément au protocole de description de session (SDP, *session description protocol*); toutefois, les passerelles de jonction peuvent faire certaines hypothèses simplificatrices concernant la description de session, comme spécifié dans ce qui suit. Il convient de noter que les descriptions de session sont sensibles à la hauteur de casse, suivant le document IETF RFC 2327.

L'utilisation du protocole SDP dépend du type de session, tel qu'il est spécifié dans le paramètre "media".

- si le paramètre de média est mis à "audio", la description de session vise un service audio;
- si le paramètre de média est mis à "image", la description de session vise un service d'image comme la télécopie.

8.4.1 Utilisation du service audio selon le protocole SDP

Dans une passerelle de jonction, ne doivent être décrites que les sessions qui, à un instant quelconque, emploient exactement un seul type de média à extension MIME, soit le média "audio" (pour la téléphonie ou les données en bande vocale) ou le média "image" (pour les communications de télécopie conformes aux procédures T.38). Les paramètres du protocole SDP qui sont à prendre en considération pour les types de média à base aussi bien "audio" que "image" sont spécifiés dans le § 8.4.2. Les paramètres spécifiques du média "audio" sont spécifiés dans le § 8.4.3. Les paramètres spécifiques du média "image" (utilisé pour les procédures T.38) sont spécifiés dans le § 8.4.4. La passerelle de jonction DOIT prendre en charge les descriptions de session qui sont conformes à ces règles et respectent l'ordre suivant:

1) le profil SDP présenté ci-après;
2) le document IETF RFC 2327 (SDP): *Session Description Protocol*.

Le contrôleur MGC devrait prendre des précautions s'il considère qu'il est nécessaire d'altérer le profil SDP reçu d'une extrémité. Le profil SDP permet de communiquer les capacités d'une extrémité à un autre extrémité. Si le contrôleur MGC décide de modifier le profil SDP, il NE DOIT PAS altérer le profil SDP au point qu'il viole les règles définies dans le présent paragraphe.

Le profil SDP qui est fourni décrit l'emploi du protocole de description de session dans le cadre du protocole TGCP. On trouvera dans le document IETF RFC 2327 la description générale et l'explication des différents paramètres, ainsi que la plupart des paramètres utilisés pour l'audio seulement. Les paramètres spécifiques des procédures T.38 pour les images peuvent être trouvés dans la Rec. UIT-T T.38. Ci-dessous sont détaillées les valeurs que les extrémités conformes au protocole TGCP doivent fournir pour ces champs (envoi) et ce que ces extrémités doivent faire avec les valeurs fournies ou non fournies pour ces champs (réception). Il convient de noter que le profil SDP utilisé ici n'est pas conforme au modèle d'offre/réponse défini dans le document RFC 3264. Si un contrôleur MGC a besoin d'interagir avec une autre entité qui utilise le modèle d'offre/réponse, ce contrôleur MGC peut donc avoir besoin d'éditer le profil SDP qu'il reçoit de l'extrémité.

8.4.2 Paramètres SDP communs à l'utilisation du service audio et à celle du service d'image

8.4.2.1 Paramètre de version du protocole (v=)

```
v=<version>
v=0
```

Envoi: ce champ DOIT être fourni conformément au document IETF RFC 2327 (à savoir v=0).

Réception: ce champ DOIT être fourni conformément au document IETF RFC 2327.

8.4.2.2 Paramètre d'origine (o=)

Le champ correspondant à l'origine (o=) est composé de 6 sous-champs dans le document IETF RFC 2327:

```
o=<username> <session-ID> <version> <network-type> <address-type> <address>
o=-  2987933615    2987933615 IN IP4       126.16.64.4
```

Elément <Username> (nom d'utilisateur):

Envoi: le trait d'union DOIT être employé comme nom d'utilisateur lorsque la confidentialité est demandée. Le trait d'union DEVRAIT être utilisé dans les autres cas.[25]

Réception: ce champ DEVRAIT être ignoré.

Session-ID (identificateur de session):

Envoi: ce champ DOIT être conforme au document IETF RFC 2327 afin de permettre l'interfonctionnement avec des clients sur réseaux non IPCablecom.

Réception: ce champ DEVRAIT être ignoré.

Version:

Envoi: ce champ est conforme au document IETF RFC 2327.

Réception: ce champ DEVRAIT être ignoré.

Network Type (type de réseau):

Envoi: le type "IN" DOIT être utilisé.

Réception: ce champ DEVRAIT être ignoré.

Address Type (type d'adresse):

Envoi: le type "IP4" DOIT être utilisé.

Réception: ce champ DEVRAIT être ignoré.

Address (adresse):

Envoi: ce champ DOIT être conforme au document IETF RFC 2327 afin de permettre l'interfonctionnement avec des clients sur réseaux non IPCablecom.

Réception: ce champ DOIT être ignoré.

8.4.2.3 Paramètre de nom de session (s=)

```
s=<session-name>
s=-
```

Envoi: le trait d'union DOIT être employé comme nom de session.

Réception: ce champ DOIT être ignoré.

8.4.2.4 Paramètre d'informations relatives à la session et au média (i=)

```
i=<session-description>
```

Envoi: pour le protocole TGCP, ce champ NE DOIT PAS être employé.

Réception: ce champ DOIT être ignoré.

8.4.2.5 Paramètre URI (u=)

```
u= <URI>
```

Envoi: pour le protocole TGCP, ce champ NE DOIT PAS être employé.

Réception: ce champ DOIT être ignoré.

---

[25] Puisque les extrémités conformes au protocole TGCP ne savent pas quand la confidentialité est demandée, elles DEVRAIENT toujours employer un trait d'union.

8.4.2.6 Paramètre d'adresse du courrier électronique et de numéro de téléphone (e=, p=)

```
e=<e-mail-address>
p=<phone-number>
```

Envoi: pour le protocole TGCP, ce champ NE DOIT PAS être employé.

Réception: ce champ DOIT être ignoré.

8.4.2.7 Paramètre de données de connexion (c=)

Le paramètre de données de connexion est composé de 3 sous-champs:

```
c=<network-type>   <address-type>     <connection-address>
c=IN               IP4                10.10.111.11
```

Network Type (type de réseau):

Envoi: le type "IN" DOIT être employé.

Réception: le type "IN" DOIT être présent.

Address Type (type d'adresse):

Envoi: le type "IP4" DOIT être employé.

Réception: le type "IP4" DOIT être présent.

Connection Address (adresse de connexion):

Envoi: ce champ DOIT être rempli au moyen d'une adresse IP destinée à la transmission de point à point où l'application recevra le flux de média. En conséquence, la durée de vie NE DOIT PAS être présente et le "nombre d'adresses" NE DOIT PAS être présent non plus. Ce champ NE DOIT PAS être rempli au moyen d'un nom de domaine entièrement qualifié au lieu d'une adresse IP. Une adresse non nulle spécifie aussi bien l'adresse de l'expéditeur que celle du receveur pour le ou les flux de média sur lesquels elle porte.

Réception: une adresse IP destinée à la transmission de point à point ou un nom de domaine entièrement qualifié DOIT être présent. Une adresse non nulle spécifie aussi bien l'adresse de l'expéditeur que celle du receveur pour le ou les flux de média sur lesquels elle porte.

8.4.2.8 Paramètre de largeur de bande (b=)

```
b=<modifier> : <bandwidth-value>
b=AS : 64
```

Envoi: les informations concernant la largeur de bande sont facultatives dans le protocole SDP, mais elles DEVRAIENT toujours être présentes[26]. Lorsqu'un paramètre rtpmap ou qu'un codec mal connu[27] est employé, les informations concernant la largeur de bande DOIVENT être employées.

Réception: les informations concernant la largeur de bande DEVRAIENT être présentes. Si un modificateur de largeur de bande n'est pas présent, le récepteur DOIT attribuer aux codecs bien connus des valeurs de largeur de bande par défaut raisonnables.

Modifier (modificateur):

Envoi: le type "AS" DOIT être employé.

---

[26] Si ce champ n'est pas employé, le portier pourrait interdire la largeur de bande appropriée.

[27] Un codec mal connu est un codec qui n'est pas défini dans la Rec. UIT-T J.161 relative au codec IPCablecom.

Réception: le type "AS" DOIT être présent.

Bandwith Value (largeur de bande):

Envoi: le champ DOIT être rempli au moyen de la spécification relative à la largeur de bande maximale du flux de média en kilobits par seconde. Voir le § 7.5 de la spécification des codecs IPCablecom (J.161) pour la manière de calculer la valeur de largeur de bande.

Réception: la spécification relative à la largeur de bande maximale du flux de média en kilobits par seconde DOIT être présente.

8.4.2.9 Paramètres de temps, intervalles de répétition et fuseaux horaires (t=, r=, z=)

```
t=<start-time><stop-time>
t=36124033         0
r=<repeat-interval> <active-duration> <list-of-offsets-from-start-time>
z=<adjustment-time> <offset>
```

Envoi: le temps DOIT être présent; le temps de démarrage PEUT être zéro, mais DEVRAIT être le temps effectif, et le temps d'arrêt DEVRAIT être zéro. Les intervalles de répétition et les fuseaux horaires NE DEVRAIENT PAS être employés; s'ils le sont, il faudrait que cela soit en conformité avec le document IETF RFC 2327.

Réception: si l'un de ces champs est présent, il DEVRAIT être ignoré.

8.4.2.10 Paramètre d'attributs (a=)

```
a= <attribute> : <value>
a= mptime: <alternative 1> <alternative 2> …a = X-pc-bridge: <number-ports>
a= <attribute>
a= recvonly
a= sendrecv
a= sendonly
a= ptime
```

Envoi: une ou plusieurs des lignes d'attribut "a" spécifiées ci-après PEUVENT être présentes.

Réception: une ou plusieurs des lignes d'attribut "a" spécifiées ci-après PEUVENT être présentes et DOIVENT dès lors être mises à exécution.

Noter que le protocole SDP exige que les attributs inconnus soient ignorés.

mptime: cet attribut définit une liste de valeurs de période de mise en paquets que l'extrémité est capable d'utiliser (en émission et en réception) pour la connexion considérée.

Envoi: l'attribut "mptime" DOIT être présent. Il DOIT y avoir précisément une seule entrée dans la liste pour chaque entrée <format> fournie dans la ligne "m=". Le numéro d'entrée j dans cette liste définit le champ de période de mise en paquets pour le numéro d'entrée j dans la ligne "m=". La première entrée dans la liste DOIT être un nombre décimal tandis que les entrées subséquentes dans la liste DOIVENT être soit un nombre décimal ou un trait d'union. Pour les formats de média où un rythme de mise en paquets unique n'est pas applicable (par exemple des codecs non vocaux tels qu'un événement téléphonique ou un bruit de confort), un trait d'union ("-") DOIT être codé à l'emplacement correspondant dans la liste des périodes de mise en paquets.

Réception: achemine la liste des périodes de mise en paquets que l'extrémité distante est capable d'utiliser pour la connexion considérée. Il n'y a qu'une seule période pour chaque format de média dans la ligne "m=". Pour les formats de média dont la période de mise en paquets est spécifiée comme étant un trait d'union ("-"),

l'extrémité DOIT utiliser une seule des périodes de mise en paquets qui ont été effectivement spécifiées dans la liste. Si l'attribut "mptime" est absent, alors la valeur de l'attribut "ptime", si présent, DOIT être considérée comme indiquant le champ de période de mise en paquets pour tous les codecs présents dans la ligne "m=". Si ni l'attribut "mptime" ni l'attribut "ptime" n'est présent, alors la passerelle de média doit se replier sur la valeur par défaut pour les codecs bien connus (comme défini dans le document RFC 1890).

X-pc-bridge:

Envoi: les extrémités conformes au protocole TGCP NE DOIVENT PAS employer cet attribut.

Réception: si les extrémités conformes au protocole TGCP reçoivent cet attribut, elles DOIVENT l'ignorer.

recvonly:

Envoi: ce champ ne devrait pas être fourni par une extrémité en TGCP.

Réception: ce champ DOIT être ignoré.

sendrecv:

Envoi: ce champ ne devrait pas être fourni par une extrémité en TGCP.

Réception: ce champ DOIT être ignoré.

sendonly:

Envoi: ce champ ne devrait pas être fourni par une extrémité en TGCP.

Réception: ce champ DOIT être ignoré.

ptime:

Envoi: l'attribut "ptime" DEVRAIT être envoyé s'il a été reçu dans un RemoteConnectionDescriptor ou si le contrôleur MGC a utilisé le champ de période de mise en paquets ('p:') dans les options du paramètre LocalConnectionOption.

Réception: ce champ DOIT être ignoré si le profil SDP contient l'attribut "mptime" (tel que requis dans les dispositifs conformes à la spécification PacketCable). Si l'attribut "mptime" n'est pas présent, alors ce champ sert à définir l'intervalle de mise en paquets pour tous les codecs présents dans la description du profil SDP. Si ni l'attribut "mptime" ni l'attribut "ptime" n'est présent, alors la passerelle de média doit se replier sur la valeur par défaut pour les codecs bien connus (comme défini dans la Rec. UIT-T J.161).

8.4.3 Utilisation du service audio dans le protocole SDP

Les paramètres suivants du protocole SDP sont appliqués au niveau du média et sont propres à l'utilisation du service audio. Les extrémités conformes à la spécification PacketCable NE DOIVENT envoyer AUCUN de ces paramètres dans un descripteur de média image (voir § 8.4.4). Si, cependant, l'extrémité reçoit un profil SDP avec des paramètres d'attribut propres aux images seulement dans un descripteur de média audio, ces paramètres DEVRAIENT être ignorés. Par ailleurs, quand des paramètres de capacité de média doivent être fournis, chaque descripteur de capacité de média (qui contient la ligne descriptive de capacité, a=cdsc, assortie de 0, une ou plusieurs lignes d'attribut de capacité, p. ex. a=cpar) DOIT apparaître après le dernier attribut de média et chaque descripteur de capacité de média DOIT être énuméré séparément.

8.4.3.1 Paramètre de clés de chiffrement

```
k=<method>
k=<method> : <encryption-keys>
```

Les services de sécurité pour les réseaux IPCablecom sont définis dans la Rec. UIT-T J.170. Ceux qui sont spécifiés pour les protocoles RTP et RTCP ne sont pas conformes à ceux des documents IETF RFC 1889, IETF RFC 1890 et IETF RFC 2327. Dans l'intérêt de l'interfonctionnement avec des dispositifs non IPCablecom, le paramètre "k" ne sera donc pas employé pour acheminer des paramètres de sécurité.

Envoi: ce champ NE DOIT PAS être employé.

Réception: ce champ DEVRAIT être ignoré.

8.4.3.2 Attributs (a=)

```
a=<attribute> : <value>
a=rtpmap : <payload type> <encoding name>/<clock rate> [/<encoding parameters>]
a=rtpmap : 0              PCMU / 8000
a=fmtp:<format> <format specific parameters>
a=X-pc-codecs: <alternative 1> <alternative 2> ...
a=X-pc-secret: <method>:<encryption key>[pad]
a =X-pc-csuites-rtp: <alternative 1> <alternative 2> ...
a =X-pc-csuites-rtcp: <alternative 1> <alternative 2> ...
```

Envoi: on PEUT inclure une ou plusieurs des lignes d'attribut "a" spécifiées ci-dessous.

Réception: on PEUT inclure une ou plusieurs des lignes d'attribut "a" spécifiées ci-dessous et on DOIT les manipuler en conséquence.

rtpmap:

Envoi: lorsqu'il est utilisé, ce champ DOIT l'être conformément au document RFC 2327 du groupe IETF. Il PEUT être utilisé tant pour les codecs bien connus que pour ceux qui ne le sont pas. Les noms de codage utilisés sont fournis dans une Recommandation IPCablecom distincte (voir Recommandations UIT-T J.161 et J.170). Sur une connexion donnée, le type de charge utile dynamique pour une méthode de codage donnée DOIT être le même dans le sens envoi et dans le sens réception. Par ailleurs, sur une connexion donnée, une fois qu'un type de charge utile dynamique a été transposé sur une méthode de codage donnée, ce type de charge utile NE DOIT PAS être ensuite transposé sur une autre méthode de codage.

Réception: lorsqu'il est utilisé, ce champ DOIT être utilisé conformément au document RFC 2327 du groupe IETF. Pour une connexion donnée, les implémentations NE DEVRAIENT pas échouer si une méthode de codage donnée est transposée sur des types de charge utile différents dans le sens envoi et dans le sens réception, ou si un type de charge utile donné est retransposé.

fmtp:

Envoi: ce champ PEUT être utilisé pour fournir des paramètres spécifiques d'un format particulier. Par exemple, ce champ pourrait être utilisé pour décrire des événements de téléphonie pris en charge par le format RFC 2833. Lorsqu'il est utilisé, le format DOIT être un des formats spécifiés pour le média. Les paramètres spécifiés sont fournis dans une Recommandation distincte qui précise l'utilisation du format.

Réception: lorsqu'il est utilisé, ce champ DOIT être utilisé conformément au document RFC 2327.

X-pc-codecs:

Envoi: ce champ contient une liste d'autres codecs que l'extrémité est capable d'utiliser pour cette connexion. La liste est ordonnée par degré de préférence décroissant, c'est-à-dire que le codec préféré en variante est le premier de la liste. Un codec est codé de manière similaire à un "nom de codage" dans le champ rtpmap.

Réception: transporte une liste de codecs que l'extrémité distante est capable d'utiliser pour cette connexion. Les codecs NE DOIVENT PAS être utilisés jusqu'au moment de leur signalisation par le biais d'une ligne de média (m=).

X-pc-secret:

Envoi: ce champ contient un secret de bout en bout et (éventuellement) le bourrage (PAD) à utiliser pour la sécurité RTP et RTCP. Le secret et le bourrage sont codés de manière similaire au paramètre de clé de chiffrement (k=) du document RFC 2327 IETF, avec les contraintes suivantes:

la clé de chiffrement NE DOIT PAS contenir de suite cryptographique mais seulement une phrase de passe;

la <method> spécifiant le codage de la phrase de passe DOIT être soit en clair ("clear") ou en codage "base64" comme défini dans le document RFC 2045, à l'exception de la longueur maximale de ligne qui n'est pas spécifiée ici. La méthode "clear" (*en clair*) NE DOIT PAS être utilisée si le secret ou le bourrage contient des caractères interdits dans le protocole SDP.

Les exigences pour le moment de transmission du bourrage sont décrites dans la Recommandation J.170 sur la sécurité. S'il est présent, ce champ DOIT être séparé du secret par au moins un espace. Bourrage et secret DOIVENT utiliser la même méthode de codage.

Réception: ce champ achemine un secret de bout en bout à utiliser pour la sécurité RTP et RTCP. S'il est présent, son utilisation est conforme à la description donnée dans la Rec. UIT-T J.170 sur la sécurité et il DOIT être séparé du secret par au moins un espace. Bourrage et secret DOIVENT utiliser la même méthode de codage.

X-pc-csuites-rtp
X-pc-csuites-rtcp

Envoi: ce champ contient une liste de suites cryptographiques que l'extrémité est capable d'utiliser pour cette connexion (RTP et RTCP respectivement). La première suite cryptographique énumérée est celle que le point d'extrémité prévoit actuellement d'utiliser. Toutes les suites cryptographiques éventuelles restant dans la liste représentent des variantes classées par ordre de préférence décroissante, c'est-à-dire que la suite cryptographique préférée en variante est la deuxième dans la liste. Une suite cryptographique est codée comme spécifié ci-après:

ciphersuite = [AuthenticationAlgorithm] "/" [EncryptionAlgorithm]

AuthenticationAlgorithm = 1*(ALPHA / DIGIT / "-" / "_")

EncryptionAlgorithm = 1*(ALPHA / DIGIT / "-" / "_")

où ALPHA et DIGIT sont définis dans le document RFC 2234. Les espaces ne sont pas autorisés à l'intérieur d'une suite cryptographique. L'exemple ci-après illustre l'utilisation d'une suite cryptographique:

```
62/51
```

La liste effective des suites cryptographiques doit être fournie dans la Rec. UIT-T J.170.

Réception: achemine une liste de suites cryptographiques que le point d'extrémité distant est capable d'utiliser pour cette connexion. Toute autre suite cryptographique que la première de la liste ne peut être utilisée sans avoir été signalisée par le biais d'une nouvelle ligne de suites cryptographiques où la suite cryptographique souhaitée figure en premier.

8.4.3.3 Annonces de média (m=)

Les annonces de médias (m=) consistent en quatre sous-champs comme suit:

```
M=<media> <port>   <transport>   <format>[<format>]
M=audio        3456      RTP/AVP        0    97
```

Média:

Envoi: le type de média "audio" DOIT être utilisé.

Réception: le type reçu DOIT être "audio".

Port:

Envoi: ce champ DOIT être rempli conformément au document RFC 2327 IETF. Le port spécifié est celui de réception, que le flux soit unilatéral ou bilatéral. Le port d'envoi peut être différent.

Réception: ce champ DOIT être utilisé conformément au document RFC 2327 IETF. Le port spécifié est celui de réception. Le port d'envoi peut être différent.

Transport:

Envoi: le protocole de transport "RTP/AVP" DOIT être utilisé.

Réception: le protocole de transport DOIT être "RTP/AVP".

Formats de média:

Envoi: un type de média approprié tel que défini dans le document RFC 2327 IETF DOIT être utilisé. Spécifiquement, ce champ contient une liste d'un ou de plusieurs types de charge utile RTP que cette extrémité est prête à recevoir sur la connexion et avec lesquels elle préférerait effectuer ses envois. Chaque type de charge utile est mappé univoque avec un codec, de façon statique ou dynamique. Le mappage statique DEVRAIT être utilisé s'il est disponible (par exemple, 0 pour PCMU, 8 pour PCMA). Si un mappage dynamique de charge utile est utilisé, un attribut RTPMAP DOIT être aussi présent et DOIT suivre les lignes directrices du § 8.4.3.2.

Réception: conformément à la norme RFC 2327. Spécifiquement, ce champ indique le ou les types de charge utile que l'autre côté de la connexion est prêt à recevoir.

8.4.4 Utilisation du service d'image en appliquant les procédures T.38

Les paramètres SDP suivants sont appliqués au niveau du média et sont propres à l'utilisation du service d'image en appliquant les procédures T.38. Les extrémités conformes à la spécification IPCablecom NE DOIVENT envoyer AUCUN de ces paramètres dans un descripteur de média audio (voir § 8.4.3). Si, cependant, l'extrémité reçoit un profil SDP avec des paramètres d'attribut propres à la seule capacité audio dans un descripteur de média d'image, ces paramètres DEVRAIENT être ignorés. Par ailleurs, quand des paramètres de capacité de média doivent être fournis, chaque descripteur de capacité de média (qui contient la ligne descriptive de capacité, a=cdsc, assortie de 0, une ou plusieurs lignes d'attribut de capacité, p. ex. a=cpar) DOIT apparaître après le dernier attribut de média et chaque descripteur de capacité de média DOIT être énuméré séparément.

8.4.4.1 Clés de chiffrement

```
k= <method>
k= <method> : <encryption-keys>
```

Il n'y a actuellement aucun service de sécurité qui soit défini pour le type de média "image/t38" .

Envoi: ce champ NE DOIT PAS être utilisé.

Réception: ce champ DEVRAIT être ignoré.

8.4.4.2 Attributs (a=)

```
a= <attribute> : <value>
a=T38FaxVersion: <version>
a=T38MaxBitrate: <bitrate>
a=T38FaxRateManagement: <faxratemanagement>
a=T38FaxMaxBuffer: <maxbuffer>
a=T38FaxMaxDatagram: <maxsize>
a=T38FaxUdpEC: <ECmethod>
a=T38FaxFillBitRemoval
a=T38FaxTranscodingMMR
a=T38FaxTranscodingJBIG
```

Envoi: une ou plusieurs des lignes d'attribut "a" spécifiées ci-dessous peuvent être incluses.

Réception: une ou plusieurs des lignes d'attribut "a" spécifiées ci-dessous peuvent être incluses et DOIVENT être manipulées en conséquence. Les valeurs d'attribut sont insensibles à la hauteur de casse. Les implémentations DOIVENT accepter les codages mixtes en minuscules et/ou majuscules de tous les attributs.

Noter que le protocole SDP exige que les attributs inconnus soient ignorés.

T38FaxVersion:

Comme cela est défini dans la Rec. UIT-T T.38: le destinataire de l'offre DOIT accepter cette version ou modifier l'attribut de version de façon qu'il désigne une version égale ou inférieure lors de la transmission d'une réponse à l'offre initiale. Le destinataire d'une offre NE DOIT PAS renvoyer de réponse contenant une version supérieure à celle qui a été offerte.

Comme défini également dans la Rec. UIT-T T.38, des implémentations anciennes d'équipement T.38 peuvent ne pas offrir de numéro de version T.38. À la réception d'un profil SDP sans l'attribut de version, l'extrémité DOIT partir du principe que la version est 0. Cela est appliqué dans le développement ci-dessous lors de l'émission et de la réception de cet attribut:

Envoi: l'extrémité DOIT indiquer la version qu'elle a l'intention d'utiliser avec l'attribut de version de télécopie T.38. Cependant, elle NE DOIT PAS indiquer de version supérieure à celle qui a été reçue dans un RemoteConnectionDescriptor.

Réception: si un RemoteConnectionDescriptor est reçu et si l'attribut de version de télécopie T.38 n'est pas inclus, alors l'extrémité DOIT utiliser la version 0 de la Rec. UIT-T T.38. Si l'attribut est inclus, l'extrémité DOIT utiliser une version de cette spécification égale ou inférieure à la version indiquée.

T38MaxBitrate:

Envoi: l'attribut T38MaxBitrate NE DOIT PAS être inclus.

Réception: l'attribut T38MaxBitrate DEVRAIT être ignoré.

T38FaxRateManagement:

Envoi: l'attribut T38FaxRateManagement DOIT être inclus et DOIT avoir une valeur égale à "transferredTCF" quand le protocole UDPTL est utilisé. Avec la valeur "transferredTCF", la commande de vérification TCF est transmise de bout en bout par opposition à une valeur d'attribut égale à "localTCF" lorsque la commande TCF est produite localement. Noter que la valeur "localTCF" n'est appropriée que quand un protocole de transport fiable tels que TCP est utilisé.

Réception: quand le protocole UDPTL est utilisé, l'attribut T38FaxRateManagement DOIT être présent avec une valeur égale à "transferredTCF" ou DOIT être absent, auquel cas un transfert de commande TCF est présumé. Toutes autres valeurs de cet attribut DOIVENT être rejetées (code d'erreur 505 – descripteur de connexion distante non pris en charge).

T38FaxMaxBuffer:

Envoi: l'attribut T38FaxMaxBuffer NE DOIT PAS être inclus.

Réception: l'attribut T38FaxMaxBuffer DEVRAIT être ignoré.

T38FaxMaxDatagram:

Envoi: l'attribut T38FaxMaxDatagram DOIT être inclus. La valeur indiquée NE DOIT PAS être inférieure à 160 octets. Cette valeur est fondée sur une période de mise en paquets de 40 ms et sur un débit binaire de 14,400 kbit/s. Cet attribut contient le protocole UDPTL sans les en-têtes IP et UDP.

Réception: les extrémités NE DOIVENT PAS envoyer de datagramme plus grand que spécifié dans l'attribut T38FaxMaxDatagram. Avant d'envoyer un quelconque datagramme T.38, l'extrémité DOIT veiller à ce qu'il soit dans les limites définies par cet attribut. Si la valeur spécifiée de l'attribut T38FaxMaxDatagram est trop petite pour prendre en charge la redondance pour un datagramme donné, mais suffisante pour prendre en charge les procédures T.38 sans redondance, alors l'extrémité DOIT envoyer ce datagramme T.38 sans redondance. Si la valeur est trop petite pour permettre que le datagramme soit envoyé sans redondance, l'extrémité NE DOIT PAS envoyer ce datagramme T.38 et l'extrémité DOIT alors produire une indication de défaillance.

T38FaxUdpEC:

La prise en charge de la redondance est obligatoire tandis que celle de la correction d'erreur directe est facultative. L'utilisation de l'un ou l'autre procédé est soumise à négociation.

Envoi: l'attribut T38FaxUdpEC DOIT être inclus. La valeur "t38UDPFEC" PEUT être envoyée si la correction FEC est prise en charge et s'il n'y a eu aucun descripteur RCD fourni avec la commande ou si la valeur de l'attribut reçue dans le descripteur RCD pour cette commande a la valeur "t38UDPFEC". Sinon la valeur "t38UDPRedundancy" DOIT être envoyée.

Réception: la redondance DOIT être utilisée si la valeur de l'attribut T38FaxUdpEC a la valeur "t38UDPRedundancy". Si l'attribut T38FaxUdpEC a la valeur "t38UDPFEC" et si la correction FEC est prise en charge par l'extrémité, alors la correction FEC DEVRAIT être utilisée. Si l'attribut T38FaxUdpEC a la valeur "t38UDPFEC" et si la correction FEC n'est pas prise en charge, alors la redondance DOIT être utilisée. Si cet attribut n'est pas inclus, l'extrémité NE DOIT PAS utiliser la redondance ou la correction FEC.

T38FaxFillBitRemoval:

La prise en charge de la suppression du bit de remplissage est facultative et tout usage de cette fonction doit faire l'objet d'une négociation.

Envoi: si l'insertion et la suppression d'un bit de remplissage sont prises en charge et recherchées et si la commande soit ne comportait pas de descripteur RCD ou comportait un descripteur RCD avec l'attribut T38FaxFillBitRemoval présent, alors cet attribut T38FaxFillBitRemoval DOIT être inclus et l'insertion et la suppression d'un bit de remplissage DOIVENT alors être utilisées. Dans tous les autres cas, l'attribut T38FaxFillBitRemoval NE DOIT PAS être inclus et l'insertion et la suppression d'un bit de remplissage NE DOIVENT PAS être utilisées.

Réception: l'insertion et la suppression d'un bit de remplissage NE DOIVENT PAS être utilisées si l'attribut T38FaxFillBitRemoval est absent.

T38FaxTranscodingMMR:

Le transcodage de modification MMR n'est pas applicable aux procédures T.38 fondées sur le protocole UDPTL.

Envoi: quand le protocole UDPTL est en cours d'utilisation pour les procédures T.38, l'attribut T38FaxTranscodingMMR NE DOIT PAS être inclus.

Réception: si l'attribut T38FaxTranscodingMMR est présent pour les procédures T.38 fondées sur le protocole UDPTL, la commande DOIT être rejetée (code d'erreur 505 – descripteur de connexion distante non pris en charge).

T38FaxTranscodingJBIG:

Le transcodage JBIG n'est pas applicable aux procédures T.38 fondées sur le protocole UDPTL.

Envoi: quand le protocole UDPTL est en cours d'utilisation pour les procédures T.38, l'attribut T38FaxTranscodingJBIG NE DOIT PAS être inclus.

Réception: si l'attribut T38FaxTranscodingJBIG est présent pour les procédures T.38 fondées sur le protocole UDPTL, la commande DOIT être rejetée (code d'erreur 505 – descripteur de connexion distante non pris en charge).

8.4.4.3 Annonces de média (m=)

La ligne d'annonces de média (m=) se compose de 4 sous-champs:

```
m= <media> <port> <transport> <fmt list>
"m= image 3456 udptl t38"
```

Media:

Envoi: le type de média 'image' DOIT être utilisé pour les procédures T.38 fondées sur le protocole UDPTL.

Réception: le type reçu DOIT être 'image' pour les procédures T.38 fondées sur le protocole UDPTL.

Port:

Envoi: ce champ DOIT être rempli conformément à la norme RFC 2327. Le port spécifié est celui de la réception. Le port d'émission peut être différent.

Réception: ce champ DOIT être utilisé conformément à la norme RFC 2327. Le port spécifié est celui de la réception. Le port d'émission peut être différent.

Transport:

Envoi: le protocole de transport 'udptl' DOIT être utilisé pour les procédures T.38 fondées sur le protocole UDPTL.

Réception: le protocole de transport DOIT être 'udptl' pour les procédures T.38 fondées sur le protocole UDPTL. Les implémentations DEVRAIENT également tolérer la forme en majuscules "UDPTL", ainsi que les formes mixtes majuscules et/ou minuscules de la chaîne "udptl".

Format du média:

Envoi: le format du média DOIT être "t38".

Réception: le format du média DOIT être "t38".

8.5 Transmission par l'intermédiaire du protocole datagramme d'utilisateur (UDP)

8.5.1 Livraison fiable de messages

Les messages conformes au protocole MGCP sont transmis par l'intermédiaire du protocole datagramme d'utilisateur (UDP, *user datagram protocol*). Les commandes sont envoyées à l'une des adresses IP qui sont définies dans le système de dénomination de domaine (DNS, *domain name system*) pour l'extrémité ou pour le contrôleur MGC spécifié. Les réponses sont renvoyées à l'adresse émettrice de la commande. Toutefois, il convient de noter que la réponse peut en fait provenir d'une adresse IP autre que celle à laquelle la commande a été envoyée.

Lorsque aucun port n'est mis à la disposition de l'extrémité[28], les commandes DOIVENT être envoyées au port MGCP par défaut, qui a le numéro 2427 pour les commandes envoyées à des passerelles et 2727 pour les commandes envoyées à des contrôleurs de passerelle média. Afin de minimiser les problèmes de compatibilité ascendante, il est RECOMMANDE que le contrôleur de passerelle média indique toujours explicitement le port MGCP à utiliser dans les messages TGCP (et ne compte pas sur la valeur par défaut).

Les messages MGCP, acheminés par l'intermédiaire du protocole UDP, peuvent subir des pertes. En l'absence de réponse en temps utile, les commandes sont répétées. Les entités du protocole MGCP sont censées garder en mémoire une liste des réponses envoyées aux transactions récentes, c'est-à-dire une liste de toutes les réponses envoyées pendant les Tt_{hist} dernières secondes, ainsi qu'une liste des transactions qui sont en cours d'exécution. Les identificateurs de transaction des commandes entrantes sont comparés aux identificateurs de transaction des récentes réponses. Si une concordance est établie, l'entité du protocole MGCP n'exécute pas la transaction, mais répète simplement la réponse. Si aucune concordance n'est établie, l'entité examine la liste des transactions en cours d'exécution. Si une concordance est établie, elle n'exécute pas la transaction, mais la laisse simplement à l'écart.

L'entité demandeuse a la charge de fournir des temporisations appropriées pour toutes les commandes en suspens et de réessayer les commandes lorsque les temporisations ont été dépassées. Une stratégie de retransmission est spécifiée au § 8.5.2.

En outre, lorsque des commandes reproduites ne réussissent pas à obtenir de réponse, l'entité de destination est supposée indisponible. L'entité demandeuse a la charge de rechercher des services excédentaires ou de libérer des connexions existantes ou en suspens, comme spécifié au § 7.4.

8.5.2 Stratégie de retransmission

La présente Recommandation évite de spécifier de quelconques valeurs permanentes pour les temporisateurs de retransmission, parce que ces valeurs sont généralement fonction du réseau.

---

[28] On peut attribuer à chaque extrémité une adresse et un port de contrôleur MGC distincts.

Normalement, les temporisateurs de retransmission devraient évaluer la temporisation en mesurant le temps qui s'écoule entre l'envoi d'une commande et le renvoi d'une réponse. Les passerelles de jonction DOIVENT implémenter une stratégie de retransmission utilisant une temporisation exponentielle avec des valeurs initiale et maximale configurables pour les temporisateurs de retransmission.

Les passerelles de jonction DEVRAIENT employer l'algorithme implémenté dans le protocole TCP-IP, qui utilise deux variables.

- Le temps d'acquittement moyen (AAD, *average acknowledgement delay*), évalué à l'aide d'une moyenne lissée exponentiellement des délais observés,
- l'écart moyen (ADEV, *average deviation*), évalué à l'aide d'une moyenne lissée exponentiellement de la valeur absolue de la différence entre les délais observés et la moyenne actuelle.

Le temporisateur de retransmission (RTO) dans le protocole TCP est fixé à la somme du délai moyen et de N fois l'écart moyen, N étant une constante.

Après une retransmission, l'entité du protocole MGCP devrait agir de la manière suivante:

- multiplier par deux la valeur évaluée du délai moyen, AAD;
- calculer une valeur aléatoire, uniformément répartie entre 0,5 AAD et 1 AAD;
- attribuer au temporisateur de retransmission (RTO) la valeur minimale de:
  - la somme de cette valeur aléatoire et de N fois l'écart moyen,
  - RTO_{max}, dont la valeur par défaut est de 4 secondes.

L'effet de cette procédure est double. Parce qu'elle comporte une composante dont la croissance est exponentielle, elle ralentira automatiquement le flux de messages en cas d'encombrement, en fonction des besoins de communication en temps réel. Parce ce qu'elle comporte une composante aléatoire, elle rompra la synchronisation possible entre les notifications déclenchées par le même événement extérieur.

La valeur initiale qui sert au temporisateur de retransmission est de 200 ms par défaut tandis que la valeur maximale s'élève à 4 secondes par défaut. Ces valeurs par défaut peuvent être modifiées par le processus de préconfiguration.

8.5.3 Taille maximale de datagramme, fragmentation et ré-assemblage

Les messages TGCP qui sont transmis sur le protocole UDP s'appuient sur le protocole IP pour la fragmentation et le ré-assemblage de grands datagrammes. La taille théorique maximale d'un datagramme IP est de 65 535 octets. Avec un en-tête IP de 20 octets et un en-tête de 8 octets, cela laisse une taille maximale théorique de message TGCP de 65 507 octets lorsqu'on utilise UDP.

Cependant, le protocole IP n'exige pas de recevoir d'un serveur local des datagrammes de plus de 576 octets (RFC 1122), ce qui donnerait une taille de message TGCP trop petite pour être acceptable. Par conséquent, le protocole TGCP déclare que les implémentations DOIVENT prendre en charge des datagrammes TGCP jusqu'à 4000 octets, ce qui exige la prise en charge de la fragmentation IP correspondante et le réassemblage. Noter que la limite des 4000 octets s'applique au niveau TGCP. La redondance de couche inférieure exigera la prise en charge de datagrammes IP encore plus longs: la redondance UDP et IP sera au moins de 28 octets, et par exemple, IPsec en ajoutera encore plus.

On devrait noter que ce qui précède s'applique aussi bien aux agents d'appel qu'aux extrémités. Les agents d'appel peuvent auditer les extrémités pour déterminer si elles acceptent de plus grands datagrammes TGCP que ce qui est spécifié ci-dessus. Les extrémités n'ont actuellement pas de capacité similaire pour déterminer si un agent d'appel accepte de plus grandes tailles de datagramme TGCP.

8.6 Portage

Dans certains cas, un contrôleur MGC souhaiterait envoyer plusieurs messages en même temps à une ou plusieurs extrémités d'une passerelle et vice versa. Lorsque plusieurs messages doivent être envoyés dans les mêmes paquets UDP, ils sont séparés par une ligne de texte qui comporte un unique point, comme dans l'exemple suivant:

```
200 2005 OK
.
DLCX 1210 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
```

Les messages joints par portage DOIVENT être traités comme s'ils avaient été reçus un par un dans des datagrammes différents. Chaque message contenu dans le datagramme DOIT être traité en entier et dans l'ordre en commençant par le premier message, et chaque commande DOIT recevoir une réponse. Les erreurs rencontrées dans un message qui a été porté NE DOIVENT PAS affecter l'un quelconque des autres messages reçus dans ce paquet. Chaque message est traité séparément.

Le portage peut être utilisé pour réaliser deux objectifs:

- garantir la livraison et le traitement des messages dans l'ordre;
- partager le sort de la livraison du message.

Lorsque le portage est utilisé pour garantir la livraison des messages dans l'ordre, les entités DOIVENT s'assurer que cette propriété de livraison dans l'ordre est conservée lors des retransmissions des messages individuels. Par exemple, lorsque plusieurs commandes Notify sont envoyées par portage (comme décrit au § 7.4.3.1).

Le partage du sort de la livraison du message garantit que soit tous les messages sont livrés, soit aucun d'eux. Lorsque le portage est utilisé pour garantir ce sort commun, les entités DOIVENT aussi s'assurer que cette propriété est conservée lors des retransmissions. Par exemple, à réception d'une commande Notify provenant d'une extrémité fonctionnant en mode perpétuel, l'agent d'appel peut souhaiter envoyer la réponse et une nouvelle commande NotificationRequest dans un seul datagramme pour s'assurer que les deux partageront le même sort de livraison du message.

8.7 Identificateurs de transaction et dialogue à trois

Les identificateurs de transaction sont des nombres entiers compris entre 1 et 999 999 999. Les contrôleurs MGC peuvent décider d'employer un ensemble de nombres propre à chaque passerelle qu'ils administrent, ou d'employer le même ensemble de nombres pour toutes les passerelles qui font partie d'un certain groupe donné. Les contrôleurs MGC peuvent décider de répartir la charge de la gestion d'une grande passerelle en plusieurs processus indépendants. Ces processus se partageront le même ensemble de nombres pour les identificateurs de transaction. Ce partage peut s'effectuer de plusieurs façons, comme celle qui consiste en l'attribution centralisée d'identificateurs de transaction, ou en la préattribution à différents processus d'ensembles d'identificateurs ne se chevauchant pas. Les réalisations du partage DOIVENT garantir que des identificateurs de transaction uniques soient attribués à toutes les transactions émanant d'un quelconque contrôleur MGC et envoyées pendant une durée de Tt_{hist} secondes à une passerelle donnée. Les passerelles peuvent détecter facilement les transactions doubles au moyen d'un simple examen de l'identificateur de transaction.

Le paramètre d'accusé de réception de réponse peut se retrouver dans toute commande. Il transporte un ensemble "de gammes d'identificateurs de transaction confirmés" pour les réponses finales reçues. Les réponses provisoires NE DOIVENT PAS être confirmées.

Les passerelles conformes au protocole MGCP peuvent choisir de supprimer les copies des réponses aux transactions dont l'identificateur figure dans les "gammes d'identificateurs de

transaction confirmés" reçues dans un message, mais le fait que la transaction ait été exécutée DOIT encore être conservé pendant Tt_{hist} secondes. Par ailleurs, lorsqu'un message d'accusé de réception de réponse[29] est reçu, la réponse dont il accuse réception peut être supprimée. Les passerelles devraient, sans notification, rejeter d'autres commandes provenant de ce contrôleur MGC lorsque l'identificateur de transaction appartient à ces gammes et que la réponse a été émise il y a moins de Tt_{hist} secondes.

Soient $term_{new}$ et $term_{old}$, qui sont les noms d'extrémité respectivement contenus dans une nouvelle commande cmd_{new} et dans une ancienne commande cmd_{old}. Les identificateurs de transaction à confirmer dans la commande cmd_{new} DEVRAIENT alors être déterminés au moyen des éléments suivants:

1) si le nom d'extrémité $term_{new}$ ne contient aucun caractère de remplacement:
   a) réponses non confirmées aux anciennes commandes où le nom d'extrémité $term_{old}$ est le même que le nom d'extrémité $term_{new}$;
   b) en option, une ou plusieurs réponses non confirmées où le nom d'extrémité $term_{old}$ contient le caractère de remplacement "un quelconque", et où le nom d'extrémité renvoyé dans la réponse est $term_{new}$;
   c) en option, une ou plusieurs réponses non confirmées où le nom d'extrémité $term_{old}$ contient le caractère de remplacement "tous", et où le nom d'extrémité $term_{new}$ est couvert par le caractère de remplacement contenu dans $term_{old}$;
   d) en option, une ou plusieurs réponses non confirmées où le nom d'extrémité $term_{old}$ contient le caractère de remplacement "un quelconque", où aucun nom d'extrémité n'a été renvoyé et où le nom d'extrémité $term_{new}$ est couvert par le caractère de remplacement contenu dans $term_{old}$.
2) Si le nom d'extrémité $term_{new}$ contient le caractère de remplacement "tous":
   a) en option, une ou plusieurs réponses non confirmées où le nom d'extrémité $term_{old}$ contient le caractère de remplacement "tous", et où le nom d'extrémité $term_{new}$ est couvert par le caractère de remplacement contenu dans le nom d'extrémité $term_{old}$.
3) Si le nom d'extrémité $term_{new}$ contient le caractère de remplacement "un quelconque":
   a) en option, une ou plusieurs réponses non confirmées où le nom d'extrémité $term_{old}$ contient le caractère de remplacement "tous" et où le nom d'extrémité $term_{new}$ est couvert par le caractère de remplacement contenu dans le nom d'extrémité $term_{old}$ lorsque le caractère de remplacement "un quelconque" dans le nom d'extrémité $term_{new}$ a été remplacé par le caractère de remplacement "tous".

Une réponse donnée NE DEVRAIT PAS être confirmée dans deux messages distincts.

Les exemples suivants illustrent l'emploi de ces règles:

- si le nom d'extrémité $term_{new}$ est "ds/ds1-2/1" et que le nom d'extrémité $term_{old}$ soit "ds/ds1-2/1", l'ancienne réponse peut être confirmée au moyen de la règle 1a;
- si le nom d'extrémité $term_{new}$ est "ds/ds1-1/3" et que le nom d'extrémité $term_{old}$ soit "*", l'ancienne réponse peut être confirmée au moyen de la règle 1c;
- si le nom d'extrémité $term_{new}$ est "ds/ds1-2/*" et que le nom d'extrémité $term_{old}$ soit "*", l'ancienne réponse peut être confirmée au moyen de la règle 2a;

[29] A distinguer d'une commande contenant un paramètre d'accusé de réception de réponse.

- si le nom d'extrémité $term_{new}$ est "ds/ds1-2/$" et que le nom d'extrémité $term_{old}$ soit "ds/ds1-2/*", l'ancienne réponse peut être confirmée au moyen de la règle 3a.

Les valeurs des "gammes d'identificateurs de transaction confirmés" NE DEVRAIENT PAS être employées si plus de Tt_{hist} secondes se sont écoulées depuis que la passerelle a émis sa dernière réponse à destination de ce contrôleur MGC, ou lorsqu'une passerelle reprend l'exploitation. Dans cette situation, les commandes devraient être acceptées et traitées, sans essai d'identificateur de transaction.

En outre, une réponse NE DEVRAIT pas être confirmée lorsqu'elle a été reçue il y a plus de Tt_{hist} secondes.

Les messages qui confirment les réponses peuvent être émis et reçus en désordre. La passerelle gardera l'ensemble des identificateurs de transaction reçus dans les commandes récentes.

8.8 Réponses provisoires

Dans certains cas, les temps d'achèvement des transactions peuvent être bien plus longs que dans d'autres cas. Le protocole TGCP utilise le protocole UDP comme protocole de transport et la fiabilité est assurée au moyen de retransmissions à base de temporisation sélective, celle-ci étant fondée sur une évaluation de l'addition du temps aller-retour dans le réseau et du temps nécessaire à l'achèvement de la transaction. Des variations significatives des temps d'achèvement des transactions sont en conséquence problématiques lorsque l'on désire que la détection de la perte de messages soit rapide et sans surcharge excessive.

Afin de surmonter ce problème, une réponse provisoire DOIT donc être donnée, à condition que le temps nécessaire à l'achèvement de la transaction dépasse un court laps de temps donné (200 ms est RECOMMANDÉ). Cette réponse provisoire accuse réception de la commande, même si le résultat de celle-ci n'est éventuellement pas encore connu, p. ex. en raison d'une réservation de ressources en suspens. A titre d'orientation, une transaction qui nécessite l'achèvement d'une communication externe, p. ex. la réservation des ressources de réseau, devrait émettre une réponse provisoire. En outre, si une double commande CreateConnection ou ModifyConnection est reçue, et que l'exécution de la transaction ne soit pas encore achevée, une réponse provisoire DOIT être renvoyée.

La sémantique transactionnelle pure impliquerait que des réponses provisoires ne renvoient pas d'informations autres que le fait que la transaction est effectivement exécutée, tandis qu'une démarche optimiste permettant le renvoi de certaines informations entraînerait une réduction du délai encouru par le système autrement.

Des réponses provisoires ne DOIVENT être envoyées qu'en réponse à une commande CreateConnection ou ModifyConnection. Afin de réduire le délai encouru par le système, un identificateur de connexion et une description de session DOIVENT faire partie de la réponse provisoire à la commande CreateConnection. Lorsqu'une description de session est renvoyée par la commande ModifyConnection, cette description de session DOIT également faire partie de la réponse provisoire. Si la transaction est achevée avec succès, l'information renvoyée dans la réponse provisoire DOIT être répétée dans la réponse finale. On considère qu'il s'agit d'une erreur de protocole lorsque cette information n'est pas reprise ou que des informations précédemment fournies sont modifiées dans une réponse de réussite. Si la transaction échoue, un code d'erreur est renvoyé – les informations renvoyées précédemment ne sont plus valables.

Une transaction exécutant effectivement une commande CreateConnection ou ModifyConnection DOIT être annulée si une commande DeleteConnection est reçue pour cette extrémité. Dans ce cas, une réponse pour la transaction annulée DEVRAIT encore être renvoyée automatiquement, et une réponse pour la transaction annulée DOIT être renvoyée si une retransmission de la transaction annulée est détectée (le code d'erreur 407 DEVRAIT être utilisé).

Lorsqu'une réponse provisoire est reçue, la valeur de temporisation pour la transaction concernée DOIT être beaucoup plus élevée pour cette transaction ($Tt_{longtran}$). Le but de cette temporisation est en premier lieu de détecter une défaillance de l'extrémité. La valeur par défaut de $Tt_{longtran}$ est de 5 secondes, mais le processus de préconfiguration peut modifier cette valeur.

Lorsque l'exécution de la transaction prend fin, la réponse finale est envoyée et la réponse provisoire maintenant obsolète est supprimée. Afin que la perte d'une réponse finale puisse rapidement être détectée, on DOIT accuser réception des réponses finales émises pour une transaction après les réponses provisoires. L'extrémité DOIT donc inclure un paramètre "ResponseAck" sans valeur dans ces réponses finales et seulement dans ces réponses. La présence de ce paramètre dans la réponse finale déclenchera une réponse "accusé de réception de réponse" qui sera renvoyée à l'extrémité. Cette dernière réponse comportera dans l'en-tête de réponse un identificateur de transaction concernant la réponse dont elle accuse réception. Sa réception est soumise à la même temporisation et aux mêmes stratégies et procédures de retransmission que les réponses aux commandes (voir § 7.4), à savoir que l'expéditeur de la réponse finale la retransmettra si "l'accusé de réception de réponse" n'est pas reçu dans les délais. Il n'est jamais accusé réception de la réponse "accusé de réception de réponse".

9 Sécurité

Si des entités non autorisées pouvaient utiliser le protocole MGCP, elles seraient en mesure d'établir des appels non autorisés ou d'interférer avec des appels autorisés. La sécurité ne fait pas partie intégrante du protocole MGCP. Au lieu de cela, le protocole MGCP prévoit l'existence d'une couche inférieure assurant la sécurité proprement dite.

Des spécifications et des solutions concernant la sécurité pour le protocole TGCP sont données dans la Rec. UIT-T J.170 qu'il conviendrait de consulter pour de plus amples informations.

Annexe A

Paquetages d'événements

La présente annexe définit un ensemble initial de paquetages d'événements pour les divers types d'extrémité actuellement définis par l'architecture IPCablecom pour les passerelles de jonction.

Chaque paquetage définit un nom de paquetage et des codes d'événement, et donne des définitions pour chacun des événements du paquetage. Les tableaux des événements ou des signaux de chaque paquetage comportent les cinq colonnes suivantes:

- **code** le code d'événement unique dans le paquetage employé pour l'événement ou le signal;
- **description** une brève description de l'événement ou du signal;
- **événement** une marque de pointage figure dans cette colonne si l'événement peut être demandé par le contrôleur MGC. Sinon, un ou plusieurs des symboles suivants peuvent figurer dans la colonne:
  - **"P"** indiquant que l'événement est durable;
  - **"S"** indiquant que l'événement est un état d'événement qui peut faire l'objet d'un audit;
  - **"C"** indiquant que l'événement ou le signal peut être détecté dans une connexion ou appliqué à celle-ci;
- **signal** si rien ne figure dans cette colonne pour un événement, celui-ci ne peut être signalé en réponse à une commande par le contrôleur MGC. Sinon, les symboles suivants identifient le type d'événement:
  - **"OO"** signal activé/désactivé. Le signal est activé jusqu'à la commande par le contrôleur MGC de l'arrêter, et vice versa;
  - **"TO"** signal temporisé. Le signal est activé pendant un temps donné à moins d'être supplanté par un nouveau signal. Les temporisations par défaut sont fournies. Une valeur nulle indique que la temporisation est illimitée. Le processus de préconfiguration peut modifier ces valeurs par défaut;
  - **"BR"** signal bref. L'événement a une durée connue et courte;
- **informations supplémentaires** donnent des informations en supplément pour l'événement ou le signal, p. ex. la durée par défaut des signaux TO.

Sauf spécification contraire, tous les événements ou signaux sont détectés ou appliqués aux extrémités et le flux audio produit par celles-ci n'est pas transmis dans les connexions auxquelles l'extrémité peut être reliée. Le flux audio produit par des événements ou signaux détectés dans une connexion ou appliqués à celle-ci sera toutefois transmis dans la connexion associée, quel que soit le mode de connexion.

A.1 Paquetage de jonction de l'ISUP

Nom du paquetage: IT

Les codes suivants sont utilisés pour identifier des événements et des signaux pour le paquetage "IT". Une passerelle média prenant en charge le paquetage "IT" DOIT prendre en charge tous les événements et signaux énumérés dans ce tableau.

Tableau A.1/J.171.1 – Evénements et signaux du paquetage de jonction de l'ISUP

Code	Description	Evénement	Signal	Informations supplémentaires
co1	Tonalité de continuité 1	√	TO	Temporisation = 3 secondes
co2	Tonalité de continuité 2	√	TO	Temporisation = 3 secondes
ft	Tonalité de télécopie	√	–	
ld	Connexion de longue durée	C	–	
ma	Démarrage du média	C	–	
mt	Tonalité de modem	√	–	
oc	Opération achevée	√	–	
of	Echec de l'opération	√	–	
ro	Tonalité de recomposition	–	TO	Temporisation = 30 secondes
rt	Tonalité de retour d'appel sonore	–	C, TO	Temporisation = 180 secondes
TDD	Tonalités des appareils de télécommunication pour les personnes malentendantes (TDD, *telecommunications device for the deaf*)	√		

La définition des différents événements et signaux est donnée ci-après:

tonalité de continuité 1 (co1): tonalité à 2010 Hz suivant la Rec. UIT-T Q.724. Afin de se conformer aux pratiques actuelles relatives aux essais de continuité, l'événement NE DEVRAIT PAS être produit avant la suppression de la tonalité. La tonalité est de type TO – l'essai de continuité ne sera appliqué qu'au cours de périodes précises. Le processus de préconfiguration peut modifier la valeur par défaut.

tonalité de continuité 2 (co2): tonalité à 1780 Hz suivant la Rec. UIT-T Q.724. Afin de se conformer aux pratiques actuelles relatives aux essais de continuité, l'événement NE DEVRAIT PAS être produit avant la suppression de la tonalité. La tonalité est de type TO – l'essai de continuité ne sera appliqué qu'au cours de périodes précises. Le processus de préconfiguration peut modifier la valeur par défaut.

Les tonalités de continuité sont employées lorsque le contrôleur MGC souhaite procéder à un essai de continuité: les types d'essai sont au nombre de deux, à l'aide d'une tonalité simple et d'une tonalité double. L'entité lançant l'essai de continuité signale et détecte les tonalités appropriées pour le circuit concerné. P. ex. on pourrait utiliser les messages suivants pour un essai de continuité de passage d'un circuit à 4 fils à un circuit à 2 fils:

passerelle de départ

```
RQNT 1234 ds/ds3-1/ds1-6/17@tgw1.example.net
X: AB123FE0
S: co2
R: co1
```

passerelle de terminaison

```
CRCX 1234 ds/ds1-4/7@tgw2.example.net
C: A3C47F21456789F0
L: p:10, a:PCMU
M: conttest
```

La passerelle de départ envoie le signal demandé et attend le retour de la tonalité appropriée pour le circuit en question. Lorsqu'elle détecte cette tonalité et juge que l'essai de continuité a réussi, elle produit l'événement "co1" qui, dans l'exemple, sera notifié au contrôleur MGC. Si l'essai ne réussit pas avant l'interruption, un événement "opération achevée" sera produit et envoyé, dans ce cas aussi, au contrôleur MGC. De même, si une erreur se produit avant l'interruption, un événement "échec de l'opération" sera produit. Les événements "oc" et "of" seront paramétrés au moyen du nom de l'événement ou du signal dont ils font état, à savoir "co1" dans ce cas.

Tonalité de télécopie (ft): l'événement de tonalité de télécopie est produit lorsqu'une communication de type télécopie est détectée – Voir p. ex. la Rec. UIT-T T.30 ou V.21.

Connexion de longue durée (ld): la "connexion de longue durée" est détectée lorsqu'une connexion a été établie depuis plus d'une certain temps. La valeur par défaut est 1 h, mais peut être modifiée par le processus de préconfiguration.

L'événement peut être détecté dans une connexion. Lorsque aucune connexion n'est spécifiée, l'événement s'applique à toutes les connexions reliées à l'extrémité, quel que soit l'instant où elles ont été établies.

Démarrage du média (ma): l'événement de démarrage du média se produit dans une connexion lorsque le premier paquet de média valable[30] et conforme au protocole RTP est reçu par l'intermédiaire de la connexion. Cet événement peut être employé pour synchroniser un signal local, p. ex. un retour d'appel sonore, avec l'arrivée d'un flux de média provenant d'une autre entité.

L'événement peut être détecté dans une connexion. Lorsque aucune connexion n'est spécifiée, l'événement s'applique à toutes les connexions reliées à l'extrémité, quel que soit l'instant où elles ont été établies.

Tonalités de modem (mt): l'événement de tonalité de modem est produit lorsqu'une communication de type modem est détectée – voir p. ex. la Rec. UIT-T V.8.

Opération achevée (oc): l'événement d'opération achevée est produit lorsque la passerelle a été invitée à appliquer un ou plusieurs signaux de type TO à l'extrémité, et qu'un ou plusieurs d'entre eux ont pris fin sans avoir été arrêtés par la détection d'un événement demandé tel que "tonalité de continuité 1". Le rapport d'achèvement peut comporter en tant que paramètre le nom du signal dont la durée de vie s'est achevée, comme dans l'expression suivante:

```
O: IT/oc(IT/co1)
```

Lorsque le signal rapporté a été appliqué à une connexion, le paramètre fourni comportera également le nom de la connexion, comme dans l'expression suivante:

```
O: IT/oc(IT/rt@0A3F58)
```

Lorsque l'événement d'opération achevée est demandé, il ne peut être paramétré au moyen de paramètres d'événement quelconques. Lorsque le nom du paquetage est omis, on suppose qu'il s'agit du nom par défaut.

L'événement d'opération achevée peut par ailleurs être produit de la manière qui est définie dans le protocole de base, p. ex. lorsqu'une commande imbriquée ModifyConnection s'achève avec succès, comme dans l'expression suivante[31]:

```
O: IT/oc(B/C)
```

---

[30] Lorsque les services d'authentification et d'intégrité sont employés, un paquet conforme au protocole RTP n'est considéré comme valable qu'une fois qu'il a passé les vérifications de sécurité.

[31] Il convient de noter l'emploi ici de "B" en tant que préfixe pour le paramètre signalé.

Echec de l'opération (of): en général, l'événement échec de l'opération peut être produit lorsque l'extrémité a été invitée à appliquer un ou plusieurs signaux du type TO à l'extrémité, et qu'un ou plusieurs d'entre eux ont échoué avant l'interruption. Le rapport d'achèvement peut comporter en tant que paramètre le nom du signal qui a échoué, comme dans l'expression suivante:

```
O: IT/of(IT/co2)
```

Lorsque le signal rapporté a été appliqué à une connexion, le paramètre fourni comportera également le nom de la connexion, comme dans l'expression suivante:

```
O: IT/of(IT/rt@0A3F58)
```

Lorsque l'événement d'échec de l'opération est demandé, les paramètres d'événement ne peuvent être spécifiés. Lorsque le nom du paquetage est omis, on suppose qu'il s'agit du nom par défaut.

L'événement d'échec de l'opération peut par ailleurs être produit de la manière qui est définie dans le protocole de base, p. ex. lorsqu'une commande imbriquée ModifyConnection échoue, comme dans l'expression suivante:

```
O: IT/of(B/C(M(sendrecv(AB2354))))
```

Il convient de noter l'emploi ici de "B" en tant que préfixe pour le paramètre signalé.

Tonalité de recomposition (ro): la tonalité de recomposition, alias tonalité d'encombrement, est spécifiée dans la Rec. UIT-T E.180/Q.35.

Tonalité de retour d'appel sonore (rt): la tonalité de retour d'appel sonore est spécifiée dans la Rec. UIT-T E.180/Q.35. Sa définition est conforme aux normes nationales relatives à la tonalité de retour d'appel sonore et PEUT être établie lors de la préconfiguration. Le signal de retour d'appel sonore peut être appliqué aussi bien à l'extrémité qu'à la connexion.

Tonalités des appareils de télécommunication pour malentendants (TDD): l'événement TDD est produit lorsqu'une communication de type TDD est détectée – voir la Rec. UIT-T V.18.

Appendice I

Combinaison des modes

Une connexion MGCP peut assurer le passage d'un ou de plusieurs flux de média. Ces flux sont soit entrants (en provenance d'une extrémité distante) soit sortants (produits à l'extrémité du circuit). Le paramètre "mode de connexion" fixe le sens et la production de ces flux. Lorsqu'une seule connexion est établie à une extrémité, le mappage de ces flux est simple; l'extrémité du circuit fait passer le flux entrant à travers le circuit et produit le flux sortant à partir du signal du circuit, en fonction du paramètre de mode.

Toutefois, lorsque plusieurs connexions sont établies à une extrémité, il peut y avoir de nombreux flux entrants et sortants. Suivant le mode de connexion employé, ces flux peuvent interagir différemment les uns avec les autres et avec les flux à destination ou en provenance de l'extrémité.

Tableau I.1/J.171.1 – Regroupement des différentes connexions lorsqu'une ou plusieurs connexions sont simultanément actives

		Mode de connexion A					
		sendonly	**recvonly**	**sendrecv**	**loopback/ conttest**	**inactive**	**netwloop/ netwtest**
Mode de connexion B	**sendonly**	A_{out}= H_{in} B_{out}= H_{in} H_{out}=NA	A_{out}=NA B_{out}=H_{in} H_{out}= A_{in}	A_{out}=H_{in} B_{out}=H_{in} H_{out}=A_{in}	A_{out}=NA B_{out}=NA H_{out}=cot	A_{out}= NA B_{out}= H_{in} H_{out}=NA	A_{out}=A_{in} B_{out}= H_{in} H_{out}=NA
	recvonly		A_{out}= NA B_{out}= NA H_{out}=A_{in}+B_{in}	A_{out}= H_{in} B_{out}= NA H_{out}=A_{in}+B_{in}	A_{out}=NA B_{out}=NA H_{out}=cot	A_{out}=NA B_{out}=NA H_{out}= B_{in}	A_{out}= A_{in} B_{out}= NA H_{out}= B_{in}
	sendrecv			A_{out}= H_{in} B_{out}= H_{in} H_{out}=A_{in}+B_{in}	A_{out}=NA B_{out}=NA H_{out}=cot	A_{out}= NA B_{out}= H_{in} H_{out}= B_{in}	A_{out}= A_{in} B_{out}= H_{in} H_{out}= B_{in}
	loopback/ conttest				A_{out}=NA B_{out}=NA H_{out}=cot	A_{out}= NA B_{out}= NA H_{out}= cot	A_{out}= NA B_{out}= NA H_{out}= cot
	inactive					A_{out}= NA B_{out}= NA H_{out}= NA	A_{out}= A_{in} B_{out}= NA H_{out}=NA
	netwloop/ netwtest						A_{out}= A_{in} B_{out}= B_{in} H_{out}=NA

Le Tableau I.1 décrit comment les différentes connexions devraient être combinées lorsqu'une ou plusieurs connexions sont "actives" en même temps. Une connection active est définie ici comme étant une connexion qui est dans l'un des modes suivants:

- "envoi/réception";
- "envoi seulement";
- "réception seulement".

Les connexions en modes "bouclage en réseau", "essai de continuité en réseau" ou "inactif" ne subissent aucun effet des connexions dans les modes "actif". Les conventions suivantes sont employées dans le Tableau I.1:

- A_{in} est le flux de média entrant en provenance de la connexion A;
- B_{in} est le flux de média entrant en provenance de la connexion B;
- H_{in} est le flux de média entrant en provenance de la jonction;
- A_{out} est le flux de média sortant vers la connexion A;
- B_{out} est le flux de média sortant vers la connexion B;
- H_{out} est le flux de média sortant vers l'extrémité, où "cot" indique l'essai de continuité, que le mode soit "essai de continuité" ou "bouclage";
- NA indique l'absence de flux dans tous les cas.

Appendice II

Exemples de codage des commandes

Le présent appendice donne des exemples de commandes et de réponses et indique le codage effectivement employé. Toutes les commandes y sont traitées. Les commentaires figurant dans les commandes et les réponses sont facultatifs.

II.1 Commande NotificationRequest

Le premier exemple illustre une commande NotificationRequest qui lance un essai de continuité et cherche à le vérifier. "L'entité notifiée" pour l'extrémité est fixée par l'adresse "ca@ca1.whatever.net:5678" et le paramètre RequestIdentifier sera reproduit dans la commande Notify correspondante:

```
RQNT 1201 ds/ds1-1/2@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
N: mgc@mgc1.whatever.net:5678
X: 0123456789AC
R: co1, oc(N), of(N)
S: co1
```

La réponse indique que la transaction a réussi:

```
200 1201 OK
```

II.2 Commande Notify

L'exemple ci-après illustre un message Notify qui notifie la réussite d'un essai de continuité comme l'indiquent les événements observés. Puisqu'une "entité notifiée" a été spécifiée dans la commande NotificationRequest qui a déclenché la notification, elle est répétée ici. En outre, le paramètre RequestIdentifier est également repris, afin d'assurer la corrélation entre cette commande Notify et la commande NotificationRequest qui en est à l'origine:

```
NTFY 2002 ds/ds1-1/2@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
N: mgc@mgc1.whatever.net:5678
X: 0123456789AC
O: co1
```

La réponse à la commande Notify indique que la transaction a réussi:

```
200 2002 OK
```

II.3 Commande CreateConnection

Le premier exemple illustre une commande CreateConnection destinée à établir une connexion à l'extrémité spécifiée. La connexion figure dans l'identificateur CallId spécifié. Le paramètre LocalConnectionOptions spécifie que le codec utilisé est donné par la loi μ de la Rec. UIT-T G.711 et que la période de mise en paquets est égale à 10 ms. Le mode de connexion est le mode "réception seulement":

```
CRCX 1204 ds/ds1-1/17@tgw2.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: recvonly
```

La réponse indique que la transaction a réussi, et un identificateur de connexion pour la connexion nouvellement établie est donc inclus. Une description de session pour la nouvelle connexion est également incluse – Il convient de noter qu'elle est précédée d'un interligne.

```
200 1204 OK
I: FDE234C8

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime:10
```

Le deuxième exemple illustre une commande CreateConnection contenant une demande de notification et un paramètre RemoteConnectionDescriptor:

```
CRCX 1205 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: recvonly
X: 0123456789AD
R: MO/sup(addr(K0, 4,1,1, s2), id(K0,0,0,7,3,2,5,5,5,1,2,3,4,s0))
S: MO/ans

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime:10
```

La réponse indique que la transaction a échoué, parce que le circuit était déjà pris. En conséquence, il n'est renvoyé ni un identificateur de connexion connection-id ni une description de session:

```
401 2005 Circuit déjà saisi
```

Notre troisième exemple illustre l'emploi de la réponse provisoire et du dialogue à trois:

```
CRCX 1206 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
K: 1205
C: A3C47F21456789F0
L: p:10, a:PCMU
M: inactive

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0 18
a=mptime:10
```

Une réponse provisoire est d'abord renvoyée:

```
100 1206 Pending
I: DFE233D1

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime:10
```

A noter que le point terminal a choisi de prendre en charge le seul codec PCMU, c'est-à-dire le numéro de charge utile 0.

Peu après, la réponse finale est reçue:

```
200 1206 OK
K:
I: DFE233D1

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime:10
```

Le contrôleur MGC accuse réception de la réponse finale, comme il en a été prié:

```
000 1206
```

et la transaction est achevée.

II.4 Commande ModifyConnection

Le premier exemple montre une commande ModifyConnection qui attribue simplement au mode de connexion la valeur "envoi/réception" – "L'entité notifiée" est également fixée:

```
MDCX 1209 ds/ds1-1/21@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
N: mgc@mgc1.whatever.net
M: sendrecv
```

La réponse indique que la transaction a réussi:

```
200 1209 OK
```

Dans le deuxième exemple, nous transmettons une description de session et incorporons une demande de notification avec la commande ModifyConnection. L'extrémité commencera à produire des tonalités de retour d'appel sonore à destination du RTPC jusqu'à ce qu'elle détecte un flux audio dans la connexion spécifiée pour le signal de retour d'appel sonore:

```
MDCX 1210 ds/ds1-1/3@abc5.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
M: recvonly
X: 0123456789AE
R: ma@ FDE234C8
S: rt

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime:10
```

La réponse indique que la transaction a réussi:

```
200 1206 OK
```

II.5 Commande DeleteConnection (par le contrôleur de passerelle média)

Dans cet exemple, le contrôleur MGC ordonne simplement à la passerelle de jonction de supprimer la connexion FDE234C8 à l'extrémité spécifiée:

```
DLCX 1210 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
```

La réponse indique la réussite et signale que la connexion a été supprimée. Les paramètres de connexion pour cette connexion sont donc inclus aussi:

```
250 1210 OK
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

II.6 Commande DeleteConnection (par la passerelle de jonction)

Dans cet exemple, la passerelle de jonction envoie une commande DeleteConnection au contrôleur MGC pour lui signaler qu'une connexion à l'extrémité spécifiée a été supprimée. Le code de cause ReasonCode spécifie le motif de suppression et les paramètres de connexion pour cette connexion sont fournis aussi:

```
DLCX 1210 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
E: 900 - Hardware error
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

Le contrôleur MGC envoie à la passerelle une réponse de réussite:

```
200 1210 OK
```

II.7 Commande DeleteConnection (par le contrôleur de passerelle média dans le cas de connexions multiples)

Dans le premier exemple, le contrôleur MGC ordonne à la passerelle de jonction de supprimer toutes les connexions liées à l'appel `"A3C47F21456789F0"` à l'extrémité spécifiée:

```
DLCX 1210 ds/ds1-1/6@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
```

La réponse indique la réussite et signale que la ou les connexions ont été supprimées:

```
250 1210 OK
```

Dans le deuxième exemple, le contrôleur MGC ordonne à la passerelle de jonction de supprimer toutes les connexions reliées à toutes les extrémités qui sont spécifiées:

```
DLCX 1210 ds/ds1-1/*@tgw-2567.quelconque.net MGCP 1.0 TGCP 1.0
```

La réponse indique la réussite:

```
250 1210 OK
```

II.8 Commande AuditEndpoint

Dans le premier exemple, le contrôleur MGC veut savoir quelles extrémités existent au niveau de la passerelle de jonction spécifiée. Il emploie donc le caractère de remplacement "tous" pour la partie locale du nom d'extrémité. Le contrôleur MGC ne veut que deux noms d'extrémité:

```
AUEP 1200 *@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
ZM: 2
```

La passerelle de jonction indique la réussite et inclut une liste de deux noms d'extrémité. Le total des noms d'extrémité concordant avec le nom spécifié par le caractère de remplacement s'élevait à 24:

```
200 1200 OK
Z: ds/ds1-1/1@tgw-2567.whatever.net
Z: ds/ds1-1/2@tgw-2567.whatever.net
ZN: 24
```

Dans le deuxième exemple, les capacités de l'une des extrémités sont demandées:

```
AUEP 1201 ds/ds1-1/1@tgw-2567.quelconque.net MGCP 1.0 TGCP 1.0
F: A
```

La réponse indique la réussite et donne aussi les capacités. Deux codecs sont pris en charge, de capacités différentes toutefois. En conséquence, deux ensembles de capacités sont renvoyés:

```
200 1201 OK
A: a:PCMU, p:10-100, e:on, s:off, v:IT, m:sendonly;recvonly;sendrecv;
                    inactive;loopback;conttest;netwloop;netwtest
A: a:G728, p:30-90, e:on, s:on, v:IT, m: sendonly;recvonly;sendrecv;
                         inactive;loopback;conttest;netwloop
```

Dans le troisième exemple, le contrôleur MGC procède à un audit de toutes les informations possibles concernant l'extrémité:

```
AUEP 2002 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
F: R, S,X,N,I,T,O,ES
```

La réponse indique la réussite:

```
200 2002 OK
R: IT/ft,mt(N)
S:
X: 0123456789B1
N: [128.96.41.12]
I: 32F345E2
T: ft
O:
ES:
```

La liste des événements demandés contient deux événements. Lorsque aucun nom de paquetage n'est spécifié, on suppose qu'il s'agit du nom par défaut. Il en va de même pour les mesures et il faut supposer, pour l'événement "IT/ft", que la mesure est donc la mesure par défaut – Notify. L'omission d'une valeur pour le paramètre "SignalRequests" signifie qu'aucun signal n'est effectivement activé. "L'entité notifiée" effective renvoie à une adresse IP et une seule connexion existe à l'extrémité. La valeur réelle du paramètre DetectEvents est "ft" et la liste pour le paramètre ObservedEvents est vide tout comme le paramètre EventStates.

II.9 Commande AuditConnection

Le premier exemple montre une commande AuditConnection où il est procédé à l'audit des paramètres CallId, NotifiedEntity, LocalConnectionOptions, ConnectionMode, LocalConnectionDescriptor et des paramètres de connexion:

```
AUCX 2003 ds/ds1-1/18@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
I: 32F345E2
F: C,N,L,M,LC,P
```

La réponse indique la réussite et donne des informations sur le paramètre RequestedInfo:

```
200 2003 OK
C: A3C47F21456789F0
N: mgc@mgc1.whatever.net
L: p:10, a:PCMU
M: sendrecv
P: PS=395, OS=22850, PR=615, OR=30937, PL=7, JI=26, LA=47

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1296 RTP/AVP 0
a=mptime:10
```

Dans le deuxième exemple, il est demandé de procéder à l'audit des paramètres RemoteConnectionDescriptor et LocalConnectionDescriptor:

```
AUCX 1203 ds/ds1-1/2@tgw.whatever.net MGCP 1.0 TGCP 1.0
I: FDE234C8
F: RC,LC
```

La réponse indique la réussite et donne des informations sur le paramètre RequestedInfo. Dans ce cas, il n'existe pas de paramètre RemoteConnectionDescriptor; donc, en ce qui concerne celui-ci, seul le champ de la version de protocole est inclus:

```
200 1203 OK

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1296 RTP/AVP 0
a=mptime:10

v=0
```

II.10 Commande RestartInProgress

Le premier exemple illustre un message RestartInProgress envoyé par une passerelle de jonction pour informer le contrôleur MGC que l'extrémité spécifiée sera mise hors service dans 300 secondes:

```
RSIP 1200 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
RM: graceful
RD: 300
```

La réponse du contrôleur MGC indique que la transaction a réussi:

```
200 1200 OK
```

Dans le deuxième exemple, le message RestartInProgress message envoyé par la passerelle de jonction informe le contrôleur MGC que toutes les extrémités de cette passerelle seront remises en service dans 0 seconde, c'est-à-dire qu'elles sont de nouveau en service. Le délai aurait aussi bien pu être omis:

```
RSIP 1204 *@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
RM: restart
RD: 0
```

La réponse du contrôleur MGC indique la réussite et signale en outre aux extrémités concernées la nouvelle "entité notifiée":

```
200 1204 OK
N: MGC-1@whatever.net
```

Il se peut aussi que la commande échoue, la nouvelle "entité notifiée" étant indiquée comme dans l'expression suivante:

```
521 1204 OK
N: MGC-1@whatever.net
```

Dans ce cas, la commande devrait ensuite être réessayée afin de respecter la "procédure de redémarrage" (voir § 7.4.3.5), cette fois allant vers le contrôleur MGC "`MGC-1@whatever.net`".

Appendice III

Exemple de flux d'appel

Dans le présent appendice, un exemple de flux d'appel est donné entre un utilisateur faisant partie d'un réseau employant un adaptateur MTA et un protocole de signalisation[32] non spécifiés, et un utilisateur en dehors du réseau, joignable par l'intermédiaire d'une passerelle de jonction utilisant le protocole TGCP et d'une passerelle de signalisation prenant en charge la signalisation de l'ISUP du système de signalisation n° 7 (SS7). Il convient de noter que ce flux d'appel, bien qu'étant un flux valable, n'est donné qu'à titre d'exemple qui peut ou ne peut pas être employé dans la pratique.

Dans le flux d'appel ci-après dans la Figure III.1, le sigle CMS renvoie au serveur de gestion d'appels (*call management server*), MGC au contrôleur de passerelle média, TGW à la passerelle de jonction (*trunking gateway*) et SG à la passerelle de signalisation.

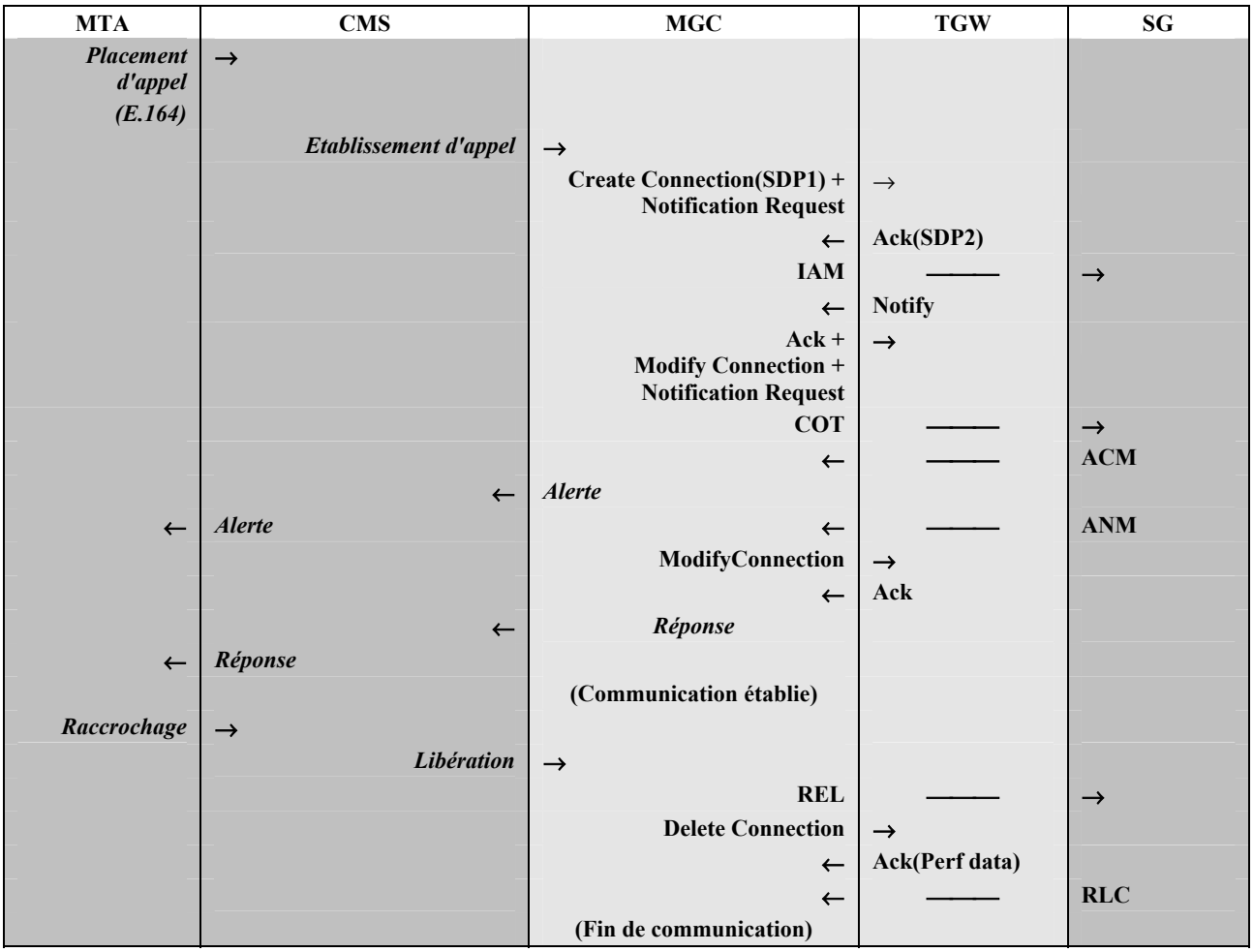


Figure III.1/J.171.1 – Exemple de flux d'appel

[32] Cette signalisation peut être de type NCS ou DCS.

Au cours de ces échanges, le profil TGCP du protocole MGCP est employé par le contrôleur MGC pour commander la passerelle de jonction. On suppose qu'un protocole non spécifié existe entre l'adaptateur MTA, le serveur CMS et le contrôleur MGC.

Nous supposons que l'adaptateur MTA indique (directement ou indirectement) au contrôleur MGC qu'il souhaite établir une communication vocale avec un numéro de téléphone conforme à la Rec. UIT-T E.164 et qu'il joint à sa demande une description de session. Le serveur CMS recherche le numéro demandé, conclut qu'il doit placer un appel en dehors du réseau et approche donc le contrôleur MGC approprié. Celui-ci décide qu'il doit placer l'appel par l'intermédiaire de la passerelle de jonction `tgw.whatever.net`. En outre, il décide qu'un essai de continuité devrait être effectué pour cet appel.

La première commande est une combinaison des commandes CreateConnection et NotificationRequest qui est envoyée à la passerelle de jonction:

```
CRCX 2001 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
L: p:10, a:PCMU
M: inactive
X: 0123456789B0
R: co2, oc, of
S: co1

v=0
o=- 25678 753849 IN IP4 128.96.41.1
s=-
c=IN IP4 128.96.41.1
t=0 0
m=audio 3456 RTP/AVP 0
a=mptime:10
```

A ce stade, la passerelle de jonction reçoit l'ordre d'entamer l'essai de continuité, d'attendre son résultat et de signaler celui-ci. La production du signal de l'essai de continuité et la détection de sa réussite (ou de son échec) au moyen du mécanisme des événements sont synchronisées de manière que, lorsque l'événement "co2" se produit, l'essai "co1" s'arrête. Noter que l'extrémité a choisi de ne prendre en charge que le codec PCMU, c'est-à-dire la charge utile numéro 0. La partie de la commande se rapportant à l'établissement de la connexion ordonne d'établir une connexion inactive à l'extrémité spécifiée conformément à la Rec. UIT-T G.711 avec une période de mise en paquets de 10 ms. La commande incorpore également la description de session reçue de l'adaptateur MTA de départ.

La passerelle de jonction de sortie accusera réception de la commande en envoyant dans la description de session ses propres paramètres tels que l'adresse, les ports et le profil RTP ainsi que l'identificateur de la nouvelle connexion:

```
200 2001 OK
I: 32F345E2

v=0
o=- 4723891 7428910 IN IP4 128.96.63.25
s=-
c=IN IP4 128.96.63.25
t=0 0
m=audio 1297 RTP/AVP 0
a=mptime:10
```

Le contrôleur MGC envoie, au commutateur relié au circuit où l'appel est placé, par l'intermédiaire de la passerelle de signalisation, un message initial d'adresse conforme au système SS7. Ce message comporte une indication précisant que l'essai de continuité doit être effectué.

Par la suite, nous supposons que l'essai de continuité a réussi. En conséquence, l'événement "co2" est produit et notifié au contrôleur MGC:

```
NTFY 3001 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
X: 0123456789B0
O: co2
```

Le contrôleur MGC envoie au commutateur distant un essai COT conforme au système SS7 indiquant "essai de continuité réussi" et accuse réception de la commande Notify reçue. Il joint également la combinaison des commandes ModifyConnection et NotificationRequest ordonnant à la passerelle de placer la connexion en mode "réception seulement" et de commencer à attendre les tonalités de télécopie et de modem:

```
200 3001 OK
.
MDCX 2006 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: 32F345E2
M: recvonly
X: 0123456789B0
R: ft,mt
```

A ce stade, le contrôleur MGC a établi un chemin de transmission à l'alternat (simplex). Le téléphone relié à l'adaptateur MTA d'entrée sera en mesure de recevoir des signaux tels que des tonalités ou des annonces qui peuvent être produites en cas d'erreurs, ainsi que le début de la parole qui sera très probablement produite lorsque l'utilisateur à l'arrivée répondra au téléphone.

Le contrôleur MGC reçoit ensuite un message d'adresse complète conforme au système SS7 indiquant que l'entité appelée est dûment appelée, puis un message de réponse conforme au système SS7 indiquant que l'entité appelée a répondu. Le contrôleur MGC place la connexion en mode bilatéral simultané (duplex) en envoyant à la passerelle de jonction la commande ModifyConnection suivante:

```
MDCX 2007 ds/ds1-1/6@tgw.quelconque.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: 32F345E2
M: sendrecv
```

La passerelle de jonction répond immédiatement à cette commande:

```
200 2007 OK
```

En parallèle, le contrôleur MGC informe l'adaptateur MTA de départ de l'existence de l'événement de réponse à l'appel et enregistre à quel moment la réponse à l'appel a eu lieu.

A ce stade, la communication est entièrement établie.

Un peu plus tard, le téléphone relié à l'adaptateur MTA de départ, dans notre scénario, est raccroché et un événement de raccrochage est transmis au contrôleur MGC (soit directement soit indirectement par l'intermédiaire du serveur CMS comme dans le cas exposé ici) l'instruisant que la communication devrait se terminer.

Le contrôleur MGC vérifie qu'il y aurait bien lieu d'effectuer la déconnexion, p. ex. lorsqu'il n'existe pas de maintien pour complément de service. Il envoie donc un message de libération conforme au système SS7 au commutateur distant, ainsi qu'une commande DeleteConnection à la passerelle de jonction:

```
DLCX 2009 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: 32F345E2
```

Les passerelles de jonction répondront par un accusé de réception qui contient les paramètres de la connexion:

```
250 2009 OK
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

Une confirmation de la terminaison de la communication sous la forme d'un message de fin de libération conforme au système SS7 est également reçue par le contrôleur MGC qui finalement enregistre la fin de la communication.

Appendice IV

Spécifications relatives aux extrémités

Le présent appendice définit un ensemble de spécifications propres aux extrémités dans le cadre du protocole TGCP.

IV.1 Modes de connexion pris en charge

Les modes de connexion qu'une extrémité donnée DOIT prendre en charge dans le cadre du protocole TGCP sont énumérés dans le Tableau IV.1:

Tableau IV.1/J.171.1 – Listes des modes de connexion qui doivent être pris en charge par une extrémité TGCP

Type d'extrémité	Information supplémentaire relative à l'extrémité	sendonly	recvonly	sendrecv	inactive	loopback	conttest	netwloop	netwtest
DS-0	Jonction ISUP	√	√	√	√	√	√	√	√
DS-0	Jonction MF	√	√	√	√	–	–	√	√

Appendice V

Informations relatives à la compatibilité

Le présent appendice donne des informations sur la compatibilité du protocole TGCP.

V.1 Compatibilité avec la signalisation NCS

La présente version du protocole TGCP est fondée, dans la mesure du possible, sur la Rec. UIT-T J.162 et alignée sur celle-ci. Puisque le protocole TGCP et la signalisation NCS portent sur des types de passerelles différents, il existe plusieurs différences, qui sont résumées ci-après:

- **modes de connexion**: la signalisation NCS et le protocole TGCP ont en commun un ensemble de modes de connexion, mais chacun a aussi des modes que l'autre ne prend pas en charge:
  - la signalisation NCS prend en charge les modes de connexion "conférence" et "duplication", contrairement au protocole TGCP;
  - le protocole TGCP prend en charge les modes de connexion "essai de continuité" et "bouclage", contrairement à la signalisation NCS;
- **scripts de numérotation**: le protocole TGCP ne prend pas en charge les scripts de numérotation, contrairement à la signalisation NCS. Cela a les conséquences suivantes:
  - il n'existe aucune commande dans le protocole TGCP qui puisse accepter un script de numérotation en tant que paramètre;
  - la mesure "recueillir conformément au script de numérotation" n'est pas prise en charge dans le protocole TGCP;
  - le "script de numérotation" ne peut pas être soumis à un audit;
- **qualité de service dynamique**: la signalisation NCS prend en charge la dynamique IPCablecom de signalisation de la qualité de service, contrairement au protocole TGCP.

Outre ce qui précède, les différences suivantes, non reliées au protocole, existent entre la signalisation NCS et le protocole TGCP:

- **paquetages d'événements**: les paquetages initiaux d'événements dans le protocole TGCP et dans la signalisation NCS sont différents;
- **système de dénomination des extrémités**: les systèmes de dénomination des extrémités dans le protocole TGCP et dans la signalisation NCS diffèrent un peu.

V.2 Compatibilité avec le protocole MGCP

Le protocole TGCP (tout comme la signalisation NCS) est en outre un profil MGCP 1.0 conforme au document IETF RFC 2705, mais il contient également quelques éléments supplémentaires. Ci-après est donnée une liste des éléments ajoutés au protocole TGCP qui ne figurent pas dans le protocole MGCP:

- **système de dénomination des extrémités**: un système de dénomination des extrémités spécifique a été introduit pour les extrémités DS-0. Les règles relatives au remplacement de caractères, plus restrictives que dans le protocole MGCP, introduisent aussi la notion "d'étendue" pour les extrémités DS-0;
- **commande incorporée ModifyConnection**: une nouvelle mesure relative à la commande incorporée ModifyConnection a été introduite;

- **sécurité**: les services de sécurité IPCablecom sont pris en charge dans le protocole TGCP. Cela influe sur le paramètre LocalConnectionOptions, sur les capacités et sur le protocole SDP;
- **extraction des noms d'extrémité**: la commande AuditEndpoint a été renforcée de manière à permettre le renvoi du nombre d'extrémités qui concordent avec un nom comportant un caractère de remplacement, ainsi que l'emploi d'un mécanisme d'extraction par bloc de ces noms d'extrémité. Cela implique, outre le renforcement de la commande AuditEndpoint, l'introduction de deux nouveaux noms de paramètre, à savoir MaxEndPointIds et NumEndPoints;
- **versions prises en charge**: la réponse RestartInProgress et la commande AuditEndpoint ont été renforcées par l'adjonction d'un paramètre VersionSupported destiné à permettre aux contrôleurs MGC et aux passerelles de déterminer quelles versions de protocole chacun prend en charge;
- **codes d'erreur**: deux nouveaux codes d'erreur ont été introduits, à savoir les codes 532 et 533;
- **utilisation du protocole SDP**: un nouveau profil d'utilisation du protocole SDP a été introduit dans le protocole TGCP. Ce profil et tous les exemples d'utilisation en particulier nécessitent une stricte conformité avec le protocole SDP, quelle que soit l'utilité des champs inclus. Des extensions propres à l'environnement IPCablecom ont également été ajoutées au protocole SDP;
- **réponse provisoire**: des précisions supplémentaires et la recommandation du mécanisme de réponse provisoire ont été incorporées dans le protocole TGCP. Une réponse d'accusé de réception d'une réponse (000) a été introduite, tandis qu'un paramètre ResponseAck sans valeur attribuée a été admis dans les réponses finales qui suivent les réponses provisoires, et qu'une procédure pour ce mécanisme a été spécifiée;
- **paramètres de signal**: la syntaxe des paramètres de signal a été étendue de manière à permettre l'utilisation dans les paramètres de signal de parenthèses s'équilibrant. La temporisation de tous les signaux TO peut être modifiée par un paramètre de signal;
- **paquetages d'événements**: le protocole TGCP introduit un ensemble de nouveaux paquetages d'événements.

Finalement, il convient de noter que le protocole TGCP fournit des interprétations et, dans certains cas, une recommandation ou une clarification supplémentaire du comportement de base du protocole MGCP qui n'est pas toujours en mesure de refléter le comportement voulu du protocole MGCP.

Appendice VI

Formalisme ABNF pour les profils TGCP

La norme RFC 3435 contient une description formelle de la syntaxe du protocole MGCP conformément au "Formalisme BNF augmenté pour Spécifications syntaxiques". Cette description formelle est référencée par les développeurs pour la création de dispositifs interopérables. Une copie de la syntaxe du protocole MGCP, annotée et éditée afin d'indiquer son applicabilité aux spécifications PacketCable, est fournie dans le présent appendice.

Les implémentations DEVRAIENT être conformes aux portions de ce formalisme ABNF qui se rapportent à leurs spécifications respectives, c'est-à-dire les protocoles NCS et TGCP. Noter également qu'il y a quelques codages paramétriques (par exemple, requête imbriquée, scripts de numérotation, noms d'extension de vendeur) où le formalisme de signalisation NCS et/ou le formalisme du protocole TGCP diffère de celui du protocole MGCP.

Cinq annotations sont utilisées afin de distinguer cinq cas différents:

1) Le langage de la norme RFC a été modifié de façon à tenir compte des exigences des protocoles NCS et TGCP.
2) Le langage de la norme RFC est applicable au protocole NCS et éventuellement au protocole MGCP mais non au protocole TGCP.
3) Le langage de la norme RFC est applicable au protocole TGCP et éventuellement au protocole MGCP et non au protocole NCS.
4) Le langage de la norme RFC n'est applicable qu'aux protocoles NCS et TGCP.
5) Le langage de la norme RFC n'est applicable qu'au protocole MGCP.

Dans chaque cas, le langage est indiqué par une police de caractères différente, comme spécifié ci-dessous.

;la norme RFC 3435 est modifiée dans son formalisme de façon à tenir compte des protocoles NCS et TGCP

```
;les caractères gras indiquent: NCS seulement (et éventuellement MGCP)
;les caractères italiques indiquent: TGCP seulement (et éventuellement MGCP)
;les caractères gras italiques indiquent: NCS et TGCP seulement
;le texte en caractères gris indique: MGCP seulement

MGCPMessage = MGCPCommand / MGCPResponse
MGCPCommand = MGCPCommandLine 0*(MGCPParameter) [EOL *SDPinformation]
MGCPCommandLine = MGCPVerb 1*(WSP) transaction-id 1*(WSP)
                        endpointName 1*(WSP) MGCPversion EOL
MGCPVerb = "EPCF" / "CRCX" / "MDCX" / "DLCX" / "RQNT"
         / "NTFY" / "AUEP" / "AUCX" / "RSIP" / extensionVerb
extensionVerb  = ALPHA 3(ALPHA / DIGIT) ; à titre expérimental, commence avec X
transaction-id = 1*9(DIGIT)
endpointName      = LocalEndpointName "@" DomainName
LocalEndpointName = LocalNamePart 0*("/" LocalNamePart)
LocalNamePart     = AnyName / AllName / NameString
AnyName           = "$"
AllName           = "*"
NameString        = 1*(jeu-de-caractères-autorisés)
; VCHAR except "$", "*", "/", "@"
jeu-de-caractères-autorisés  = %x21-23 / %x25-29 / %x2B-2E
                             / %x30-3F / %x41-7E
DomainName = 1*255(ALPHA / DIGIT / "." / "-")    ; comme défini
           / "#" number                          ; dans la norme RFC 821
           / "[" IPv4address / IPv6address "]"   ; voir la norme RFC 2373
```

```
; Réécrit en formalisme ABNF à partir de la norme RFC 821
number =  1*DIGIT
;à partir de la norme RFC 2373
IPv6address = hexpart [ ":" IPv4address ]
IPv4address = 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT
; cette production, bien qu'apparaissant dans la norme RFC 2373, n'est pas
référencée
; IPv6prefix  = hexpart "/" 1*2DIGIT
hexpart = hexseq / hexseq "::" [ hexseq ] / "::" [ hexseq ]
hexseq  = hex4 *( ":" hex4)
hex4    = 1*4HEXDIG
MGCPversion = "MGCP" 1*(WSP) 1*(DIGIT) "." 1*(DIGIT)
                            [1*(WSP) ProfileName]
ProfileName   = "NCS 1.0"    ; pour NCS
              / "TGCP 1.0"   ; pour TGCP
               / VCHAR *( WSP / VCHAR)
MGCPParameter = ParameterValue EOL
; Vérifier le code d'informations si plus de valeurs paramétriques sont définies
; la plupart des valeurs facultatives ne peuvent être omises que lors de la
vérification
ParameterValue = ("K"  ":" 0*(WSP)  [ResponseAck])
               / ("B"  ":" 0*(WSP)  [BearerInformation])
               / ("C"  ":" 0*(WSP)  CallId)
               / ("I"  ":" 0*(WSP)  [ConnectionId])
               / ("N"  ":" 0*(WSP)  [NotifiedEntity])
               / ("X"  ":" 0*(WSP)  [RequestIdentifier])
               / ("L"  ":" 0*(WSP)  [LocalConnectionOptions])
               / ("M"  ":" 0*(WSP)  ConnectionMode)
               / ("R"  ":" 0*(WSP)  [RequestedEvents])
               / ("S"  ":" 0*(WSP)  [SignalRequests])
               / ("D"  ":" 0*(WSP)  [DigitMap])  ; pour NCS (et MGCP)
               / ("O"  ":" 0*(WSP)  [ObservedEvents])
               / ("P"  ":" 0*(WSP)  [ConnectionParameters])
               / ("E"  ":" 0*(WSP)  ReasonCode)
               / ("Z"  ":" 0*(WSP)  [SpecificEndpointID])
               / ("Z2" ":" 0*(WSP)  SecondEndpointID)
               / ("I2" ":" 0*(WSP)  SecondConnectionID)
               / ("F"  ":" 0*(WSP)  [RequestedInfo])
               / ("Q"  ":" 0*(WSP)  QuarantineHandling)
               / ("T"  ":" 0*(WSP)  [DetectEvents])
               / ("RM" ":" 0*(WSP)  RestartMethod)
               / ("RD" ":" 0*(WSP)  RestartDelay)
               / ("A"  ":" 0*(WSP)  [Capabilities])
               / ("ES" ":" 0*(WSP)  [EventStates])
               / ("PL" ":" 0*(WSP)  [PackageList])     ; Vérification seulement
               / ("MD" ":" 0*(WSP)  MaxMGCPDatagram)   ; Vérification seulement
               / (extensionParameter ":" 0*(WSP) [parameterString])
               / VersionSupported  ; NCS et TGCP - réponse seulement
               / MaxEndpointIds    ; NCS et TGCP
               / NumEndpoints      ; NCS et TGCP - réponse seulement
; <extensionParameter> ":" chaîne paramétrique définie par NCS et TGCP
VersionSupported = "VS" ":" MGCPversion *( "," 0*(WSP) MGCPversion)
MaxEndpointIds  = "ZM" ":" 0*(WSP) 1*16(DIGIT)
NumEndpoints    = "ZN" ":" 0*(WSP) 1*16(DIGIT)  ; Réponses seulement
; Une réponse finale peut comprendre un acquittement ResponseAck vide
ResponseAck =  confirmedTransactionIdRange
               *( ","  0*(WSP) confirmedTransactionIdRange )
confirmedTransactionIdRange = transaction-id ["-" transaction-id]
BearerInformation = BearerAttribute 0*("," 0*(WSP) BearerAttribute)
BearerAttribute   = ("e" ":" BearerEncoding)
                  / (BearerExtensionName [":" BearerExtensionValue])
BearerExtensionName  = PackageLCOExtensionName
BearerExtensionValue = LocalOptionExtensionValue
BearerEncoding = "A" / "mu"
```

```
CallId = 1*32(HEXDIG)
; la réponse à la demande d'audit peut comprendre une liste d'identificateurs
ConnectionId = 1*32(HEXDIG) 0*("," 0*(WSP) 1*32(HEXDIG))
SecondConnectionID = ConnectionId
NotifiedEntity = [LocalName "@"] DomainName [":" portNumber]
LocalName  = LocalEndpointName            ; aucune structure interne
portNumber = 1*5(DIGIT)
RequestIdentifier = 1*32(HEXDIG)
LocalConnectionOptions = LocalOptionValue 0*(WSP)
                         0*("," 0*(WSP) LocalOptionValue 0*(WSP))
LocalOptionValue = ("p"  ":" packetizationPeriod)
                 / ("a"  ":" compressionAlgorithm)
                 / ("b"  ":" largeur de bande) ; seulement pour les capacités
                                               dans les protocoles NCS et TGCP
                 / ("e"  ":" echoCancellation)
                 / ("gc" ":" gainControl)
                 / ("s"  ":" silenceSuppression)
                 / ("t"  ":" typeOfService)
                 / ("r"  ":" resourceReservation)
                 / ("k"  ":" encryptiondata)
                 / ("nt" ":" ( typeOfNetwork /
                               supportedTypeOfNetwork))
                 / (LocalOptionExtensionName
                       [":" LocalOptionExtensionValue])
                 / MPacketizationPeriod   ; NCS et TGCP seulement
                 / RTPciphersuite         ; NCS et TGCP seulement
                 / RTCPciphersuite        ; NCS et TGCP seulement
                 / DQoSGateID             ; NCS seulement
                 / DQoSReservation        ; NCS seulement
                 / DQoSResourceID         ; NCS seulement
                 / DQoSReserveDestination ; NCS seulement
                 / CallContentId          ; TGCP seulement
                 / CallContentDestination ; TGCP seulement

Capabilities    =  CapabilityValue 0*(WSP)
                     0*("," 0*(WSP) CapabilityValue 0*(WSP))
CapabilityValue = LocalOptionValue
                / ("v" ":" supportedPackages)
                / ("m" ":" supportedModes)

PackageList     = pkgNameAndVers 0*("," pkgNameAndVers)
pkgNameAndVers  = packageName ":" packageVersion
packageVersion  = 1*(DIGIT)
; pour les protocoles NCS et TGCP, le format de l'étendue n'est autorisé que
; pour les capacités
; et non pour les options de connexion locales.
packetizationPeriod  = 1*4(DIGIT) ["-" 1*4(DIGIT)]
compressionAlgorithm = algorithmName 0*(";" algorithmName)
algorithmName        = 1*(SuitableLCOCharacter)
bandwidth            = 1*4(DIGIT) ["-" 1*4(DIGIT)]
echoCancellation     = "on" / "off"
gainControl          = "auto" / ["-"] 1*4(DIGIT)
silenceSuppression   = "on" / "off"
typeOfService        = 1*2(HEXDIG)     ; 1 hex seulement pour capacités
resourceReservation  = "g" / "cl" / "be"
; les paramètres de chiffrement sont codés comme dans le profil SDP
; (la norme RFC 2327)
; NOTE: Une clé de chiffrement peut contenir un algorithme comme spécifié dans
; la norme RFC 1890
encryptiondata = ( "clear" ":" encryptionKey )
               / ( "base64" ":" encodedEncryptionKey )
               / ( "uri" ":" URItoObtainKey )
```

```
              / ( "prompt" ) ; valeur définie dans le profil SDP,
                               non utilisable dans le protocole MGCP!
encryptionKey = 1*(SuitableLCOCharacter) / quotedString
; Voir la norme RFC 2045
encodedEncryptionKey = 1*(ALPHA / DIGIT / "+" / "/" / "=")
URItoObtainKey = 1*(SuitableLCOCharacter) / quotedString
typeOfNetwork = "IN" / "ATM" / "LOCAL" / OtherTypeOfNetwork
; Registered with IANA - voir la norme RFC 2327
OtherTypeOfNetwork     = 1*(SuitableLCOCharacter)
supportedTypeOfNetwork = typeOfNetwork *(";" typeOfNetwork)
supportedModes    = ConnectionMode 0*(";" ConnectionMode)
supportedPackages = packageName 0*(";" packageName)
packageName = 1*(ALPHA / DIGIT / HYPHEN) ; Trait d'union ni en premier
                                           ni en dernier
LocalOptionExtensionName = VendorLCOExtensionName
                         / PackageLCOExtensionName
                         / OtherLCOExtensionName
VendorLCOExtensionName   = "x" ("+"/"-") 1*32(SuitableExtLCOCharacter)
PackageLCOExtensionName  = packageName "/"
                           1*32(SuitablePkgExtLCOCharacter)
; must not start with "x-" or "x+"
OtherLCOExtensionName    = 1*32(SuitableExtLCOCharacter)
; <LocalOptionExtensionName> ":" <LocalOptionExtensionvalue>
; productions définies par le protocole NCS/TGCP
MPacketizationPeriod = "mp" ":" multiplepacketizationPeriod
multiplepacketizationPeriod = mpPeriod 0*(";" mpPeriod)
mpPeriod             = 1*4(DIGIT) / HYPHEN
RTPciphersuite   = "sc-rtp"  ":" ciphersuite
RTCPciphersuite  = "sc-rtcp" ":" ciphersuite
ciphersuite      = [AuthenticationAlgorithm] "/" [EncryptionAlgorithm]
AuthenticationAlgorithm = 1*( ALPHA / DIGIT / "-" / "_" )
EncryptionAlgorithm     = 1*( ALPHA / DIGIT / "-" / "_" )
; <LocalOptionExtensionName> ":" <LocalOptionExtensionvalue>
; production définie seulement par le protocole NCS
DQoSGateID             = "dq-gi" [":" 1*8(HEXDIG)]  ; n'est vide que pour
                                                    ; les capacités
DQoSReservation        = "dq-rr" ":" DQoSResMode *(";" DQoSResMode)
DQoSResMode            = "sendresv" / "recvresv" / "snrcresv" /
                         "sendcomt" / "recvcomt" / "snrccomt"
DQoSResourceID         = "dq-ri" ":" 1*8(HEXDIG)
DQoSReserveDestination = "dq-rd" ":" IPv4address [":" portNumber]
; <LocalOptionExtensionName> ":" <LocalOptionExtensionvalue>
; défini par TGCP seulement
CallContentId          = "es-cci" ":" 1*8(HEXDIG)
CallContentDestination = "es-ccd" ":" IPv4address ":" portNumber

LocalOptionExtensionValue = (1*(SuitableExtLCOValChar)
                                                    / quotedString)
                            *(";" (1*(SuitableExtLCOValChar)
                                                      / quotedString))
;Note: Aucun mode de "données".
ConnectionMode = "sendonly" / "recvonly" / "sendrecv"
               / "confrnce" / "inactive"
               / "loopback" / "conttest" ; TGCP (et MGCP) seulement
               / "replcate" ; NCS seulement
               / "netwloop" / "netwtest"
               / ExtensionConnectionMode
ExtensionConnectionMode = PkgExtConnectionMode
PkgExtConnectionMode    = packageName "/" 1*(ALPHA / DIGIT)
RequestedEvents = requestedEvent 0*("," 0*(WSP) requestedEvent)
requestedEvent  = (eventName ["(" requestedActions ")"])
                / (eventName "(" requestedActions ")"
                             "(" eventParameters ")" )
```

```
eventName = [(packageName / "*") "/"]
                (eventId / "all" / eventRange
                                        / "*" / "#") ; pour tonalités DTMF
                                  ["@" (ConnectionId / "$" / "*")]
eventId = 1*(ALPHA / DIGIT / HYPHEN)    ; Trait d'union ni en premier
                                          ni en dernier
eventRange = "[" 1*(DigitMapLetter / (DIGIT "-" DIGIT) /
                         (DTMFLetter "-" DTMFLetter)) "]"
DTMFLetter = "A" / "B" / "C" / "D"
requestedActions = requestedAction 0*("," 0*(WSP) requestedAction)
requestedAction  = "N" / "A"
                 / "D" ; pour NCS (et MGCP)
                 / "S" / "I" / "K"
                 / "E" "(" EmbeddedRequest ")"
                 / ExtensionAction
                 / "C" "(" EmbeddedModeChange     ; pour NCS et TGCP
                      0*("," 0*WSP EmbeddedModeChange) ")"   ; seulement
;seuls les protocoles NCS et TGCP définissent l'action intégrée
;de la commande ModifyConnection.
;le formalisme MGCP n'admet pas le format utilisé dans les
;protocoles NCS et TGCP:
EmbeddedModeChange = "M" "(" ConnectionMode "(" EmConnectionId ")" ")"
EmConnectionId     = ConnectionId / "$"
ExtensionAction  = PackageExtAction
PackageExtAction = packageName "/" Action ["(" ActionParameters ")"]
Action           = 1*ALPHA
ActionParameters = eventParameters        ; peut contenir des actions
;NOTE: Devrait tolérer un ordre différent lors de la réception, par exemple,
;pour NCS
EmbeddedRequest = (      "R" "(" EmbeddedRequestList ")"
                    ["," 0*(WSP) "S" "(" EmbeddedSignalRequest ")"]
                    ["," 0*(WSP) "D" "(" EmbeddedDigitMap ")"]      )
                / (      "S" "(" EmbeddedSignalRequest ")"
                    ["," 0*(WSP) "D" "(" EmbeddedDigitMap ")"] )
                / (      "D" "(" EmbeddedDigitMap ")" )
                / NCSTGCPEmbeddedRequest
;le texte ci-dessous concerne seulement les protocoles NCS et TGCP.
;La différence par rapport au protocole MGCP est simplement que l'ordre
;des éléments n'est pas fixé. De même, les scripts de numérotation ne sont
; pas utilisés dans le protocole TGCP.
NCSTGCPEmbeddedRequest = NCSTGCPEmbeddedRequestItem
                          *2("," 0*(WSP) NCSTGCPEmbeddedRequestItem)
NCSTGCPEmbeddedRequestItem  = ("R" "(" EmbeddedRequestList ")" )
                            / ("S" "(" EmbeddedSignalRequest ")" )
                            / ("D" "(" EmbeddedDigitMap ")" )
EmbeddedRequestList   = RequestedEvents
EmbeddedSignalRequest = SignalRequests
EmbeddedDigitMap = DigitMap
SignalRequests   = SignalRequest 0*("," 0*(WSP) SignalRequest )
SignalRequest    = eventName [ "(" eventParameters ")" ]
eventParameters  = eventParameter 0*("," 0*(WSP) eventParameter)
eventParameter   = eventParameterValue
                 / eventParameterName "=" eventParameter
                 / eventParameterName "(" eventParameters ")"
eventParameterString = 1*(SuitableEventParamCharacter)
eventParameterName   = eventParameterString
eventParameterValue  = eventParameterString / quotedString
; pour NCS (et MGCP)
DigitMap           = DigitString  / "(" DigitStringList ")"
DigitStringList    = DigitString 0*( "|" DigitString )
DigitString        = 1*(DigitStringElement)
DigitStringElement = DigitPosition ["."]
DigitPosition      = DigitMapLetter / DigitMapRange
; NOTE "X" est maintenant inclus
```

```
DigitMapLetter     = DIGIT / "#" / "*" / "A" / "B" / "C" / "D" / "T"
                   / "X" / ExtensionDigitMapLetter
ExtensionDigitMapLetter = "E" / "F" / "G" / "H" / "I" / "J" / "K"
                        / "L" / "M" / "N" / "O" / "P" / "Q" / "R"
                        / "S" / "U" / "V" / "W" / "Y" / "Z"
; NOTE: La forme "[x]" est maintenant autorisée dans le protocole MGCP.
; dans le protocole NCS, seule la forme "x" est autorisée
DigitMapRange = "[" 1*DigitLetter "]"
              / "X" ; Ajouté pour NCS seulement
DigitLetter   = *((DIGIT "-" DIGIT) / DigitMapLetter)
ObservedEvents = SignalRequests
EventStates    = SignalRequests
ConnectionParameters = ConnectionParameter
                       0*( "," 0*(WSP) ConnectionParameter )
ConnectionParameter  = ( "PS" "=" packetsSent )
                     / ( "OS" "=" octetsSent )
                     / ( "PR" "=" packetsReceived )
                     / ( "OR" "=" octetsReceived )
                     / ( "PL" "=" packetsLost )
                     / ( "JI" "=" jitter )
                     / ( "LA" "=" averageLatency )
                     / ( ConnectionParameterExtensionName
                           "=" ConnectionParameterExtensionValue )
                     / RemotePacketsSent
                     / RemoteOctetsSent
                     / RemotePacketsLost
                     / RemoteJitter
; les protocoles NCS et TGCP définissent les quatre noms suivants d'extension
; de paramètre de connexion:
RemotePacketsSent  = "PC/RPS" "=" packetsSent
RemoteOctetsSent   = "PC/ROS" "=" octetsSent
RemotePacketsLost  = "PC/RPL" "=" packetsLost
RemoteJitter       = "PC/JI"  "=" jitter
packetsSent     = 1*9(DIGIT)
octetsSent      = 1*9(DIGIT)
packetsReceived = 1*9(DIGIT)
octetsReceived  = 1*9(DIGIT)
packetsLost     = 1*9(DIGIT)
jitter          = 1*9(DIGIT)
averageLatency  = 1*9(DIGIT)
ConnectionParameterExtensionName = VendorCPExtensionName
                                 /    PackageCPExtensionName
VendorCPExtensionName  = "X" "-" 2*ALPHA
                       / NCSTGCPVendorCPExtensionName
;Le texte ci-dessous concerne seulement les protocoles NCS et TGCP.
;La différence par rapport au protocole MGCP est simplement que
;celui-ci exige 2 caractères alphabétiques tandis que NCS et TGCP
;autorisent 2 ou 3 caractères alphabétiques pour le nom VendorCPExtensionName
NCSTGCPVendorCPExtensionName = "X" "-" 2*3ALPHA
PackageCPExtensionName = packageName "/" CPName
CPName = 1*(ALPHA / DIGIT / HYPHEN)
ConnectionParameterExtensionValue = 1*9(DIGIT)
MaxMGCPDatagram = 1*9(DIGIT)
ReasonCode = 3DIGIT
             [1*(WSP) "/" packageName]     ; Seulement pour 8xx
             [WSP 1*(%x20-7E)]

SpecificEndpointID = endpointName
SecondEndpointID   = endpointName
RequestedInfo = infoCode 0*("," 0*(WSP) infoCode)
infoCode = "B" / "C" / "I" / "N" / "X" / "L" / "M" / "R" / "S"
         / "D" ; pour NCS (et MGCP) seulement
    / "O" / "P" / "E" / "Z" / "Q" / "T" / "RC" / "LC"
    / "A" / "ES" / "RM" / "RD" / "PL" / "MD" / extensionParameter
```

```
        / "VS" / "ZM" / "ZN" ; NCS et TGCP définissent ces trois
                             ; paramètres d'extension
;NCS et TGCP permettent de commander le traitement et le bouclage dans un
;ordre quelconque
QuarantineHandling = loopControl / processControl
                   / (loopControl "," 0*(WSP) processControl )
                   / (processControl "," 0*(WSP) loopControl)
loopControl    = "step" / "loop"
processControl = "process" / "discard"
DetectEvents = SignalRequests
RestartMethod = "graceful" / "forced" / "restart" / "disconnected"
              / "cancel-graceful" / extensionRestartMethod
extensionRestartMethod = PackageExtensionRM
PackageExtensionRM     = packageName "/" 1*32(ALPHA / DIGIT / HYPHEN)
RestartDelay = 1*6(DIGIT)
extensionParameter = VendorExtensionParameter
                   / PackageExtensionParameter
                   / OtherExtensionParameter
VendorExtensionParameter  = "X" ("-"/"+") 1*6(ALPHA / DIGIT)
PackageExtensionParameter = packageName "/"
                            1*32(ALPHA / DIGIT / HYPHEN)
; Ne doit jamais commencer par la forme "x-" ou x+"
OtherExtensionParameter   = 1*32(ALPHA / DIGIT / HYPHEN)

;Si le premier caractère est une apostrophe double, alors il s'agit d'une chaîne
;de citation entre apostrophes doubles
parameterString = (%x21 / %x23-7F) *(%x20-7F) ; le premier et le dernier
                                                caractère ne doivent jamais être
                                                un espace vide
                   / quotedString
MGCPResponse = MGCPResponseLine 0*(MGCPParameter)
                                      *2(EOL *SDPinformation)
MGCPResponseLine = responseCode 1*(WSP) transaction-id
                        [1*(WSP) "/" packageName]    ; seulement pour 8xx
                             [WSP responseString] EOL
responseCode = 3DIGIT
responseString = *(%x20-7E)
SuitablePkgExtLCOCharacter =  SuitableLCOCharacter
SuitableExtLCOCharacter = DIGIT / ALPHA / "+" / "-" / "_" / "&"
              / "!" / "'" / "|" / "=" / "#" / "?"
              / "." / "$" / "*" /       "@" / "[" / "]"
              / "^" / "`" / "{" / "}" / "~"
SuitableLCOCharacter   = SuitableExtLCOCharacter / "/"
SuitableExtLCOValChar  = SuitableLCOCharacter / ":"
; VCHAR sauf """, "(", ")", ",", et "="
SuitableEventParamCharacter = %x21 / %x23-27 / %x2A-2B
                            / %x2D-3C / %x3E-7E
; NOTE: Jeu de caractères UTF8 codé
quotedString  = DQUOTE 0*(quoteEscape / quoteChar) DQUOTE
quoteEscape   = DQUOTE DQUOTE
quoteChar = (%x00-21 / %x23-FF)
EOL = CRLF / LF
HYPHEN = "-"
; Voir dans la norme RFC 2327 le formalisme SDP approprié qui peut être
; substitué.
SDPinformation = SDPLine CRLF *(SDPLine CRLF) ; voir dans la norme RFC 2327
                                                les définitions sont appropriées
SDPLine        = 1*(%x01-09 / %x0B / %x0C / %x0E-FF) ; for proper def.
```

Appendice VII

Surveillance électronique

VII.1 Contrôleur MGC

Le format des paramètres de surveillance électronique contenus dans les options LCO d'une commande CRCX ou MDCX est le suivant:

- l'identificateur de connexion d'archivage du contenu d'appel, codé par le mot clé "es-cci" suivi d'un point-virgule et d'une chaîne contenant un maximum de 8 caractères hexadécimaux correspondant aux 32 bits de l'identificateur de connexion d'archivage du contenu d'appel;
- la destination d'archivage du contenu d'appel, codée par le mot clé "es-ccd" suivi d'un point-virgule et d'une adresse IP codée comme une adresse IP pour la partie relative au domaine d'un nom d'extrémité. L'adresse IP est suivie d'un point-virgule et d'un maximum de 5 caractères décimaux représentant le numéro de port UDP à utiliser.

Un contrôleur MGC DOIT inclure les deux paramètres "es-cci" et "es-ccd" dans les options LCO d'une commande CRCX ou MDCX lors de la notification de paramètres de surveillance électronique de passerelle MG.

L'exemple ci-après décrit une commande CRCX avec paramètres de surveillance électronique:

```
CRCX 1204 ds/ds1-1/1@mg.cablelabs.com MGCP 1.0 TGCP 1.0
C: 5678ABCD
L: p:10, a:PCMU, es-cci:123456, es-ccd:[128.96.41.1]:3456
M: sendrecv
X: 1237
```

L'exemple ci-après décrit une commande MDCX avec paramètres de surveillance électronique:

```
MDCX 1206 ds/s-1/ds1-1/1@mg.cablelabs.com MGCP 1.0 TGCP 1.0
C: 5678ABCD
I: 32F345E2
L: p:10, a:PCMU, es-cci:123456, es-ccd:[128.96.41.1]:3456
M: sendrecv
X: 1238
```

VII.2 Passerelle MG

Quand une passerelle MG reçoit une commande CRCX avec un paramètre non vide "es-cci" et un paramètre non vide "es-ccd" dans les options LCO, cette passerelle MG DOIT commencer à dupliquer, à réexpédier et à encapsuler tous les paquets qui sont reçus et transmis sur la connexion. Le processus consistant à dupliquer, à réexpédier et à encapsuler tous les paquets sur une connexion est appelé *surveillance du contenu d'appel*. Une passerelle MG c'est-à-dire qui exécute une surveillance du contenu d'appel DOIT dupliquer, réexpédier et encapsuler tous les paquets qu'elle produit pour la connexion. Une passerelle MG qui exécute une surveillance du contenu d'appel DOIT dupliquer, réexpédier et encapsuler tous les paquets qu'elle reçoit sur une connexion. Les paquets qui sont encapsulés DOIVENT être identiques aux paquets acheminés par la connexion. La passerelle MG DOIT encapsuler tous les paquets dupliqués et réexpédiés avec le paramètre d'identificateur de connexion d'archivage du contenu d'appel contenu dans le champ "es-cci" du paramètre LCO. La passerelle MG DOIT réexpédier tous les paquets dupliqués et encapsulés à l'adresse IP et au port UDP indiqués dans le champ "es-ccd" du paramètre LCO.

Une passerelle MG qui exécute une surveillance du contenu d'appel DOIT mettre fin à la surveillance du contenu d'appel quand soit:

1) la connexion est terminée en raison d'une commande DLCX à partir du contrôleur MGC;
2) la connexion est terminée en raison d'une commande DLCX à partir de la passerelle MG;
3) la connexion est terminée en raison de conditions d'erreur interne conditions telles que:
   – la passerelle MG subit une défaillance de composant;
   – la jonction à débit DS1 utilisée par l'extrémité est mise hors service;
4) la passerelle MG reçoit une commande MDCX avec un champ de paramètre vide "es-cci" ("es-cci:") ou "es-ccd" ("es-ccd:").

Si une passerelle MG qui exécute une surveillance du contenu d'appel reçoit une commande MDCX avec un paramètre LCO qui contient un paramètre "es-cci" ou "es-ccd" valide ou nouveau, la passerelle MG DOIT utiliser ce nouveau paramètre "es-cci" ou "es-ccd" quand elle exécute la surveillance du contenu d'appel.

Si une passerelle MG reçoit une commande CRCX ou une commande MDCX avec les paramètres "es-cci" et "es-ccd", mais ne prend pas en charge la surveillance électronique toute en étant en mesure d'exécuter la commande sauf pour la portion relative à la surveillance électronique, cette passerelle MG DOIT renvoyer le code de réponse 210 – la transaction demandée a été exécutée normalement, mais la passerelle ne prend pas en charge la surveillance électronique.

Si une passerelle MG reçoit une commande CRCX ou une commande MDCX avec les paramètres "es-cci" et "es-ccd", prend effectivement en charge la surveillance électronique et est en mesure d'exécuter la commande sauf pour la portion relative à la surveillance électronique en raison de contraintes en terme de ressource, cette passerelle MG DOIT renvoyer le code de réponse 211 – la transaction demandée a été exécutée normalement, mais la passerelle n'a pas pu exécuter la surveillance électronique parce qu'il ne possède pas de ressources suffisantes.

Si une passerelle MG reçoit une commande CRCX ou une commande MDCX avec un paramètre LCO qui contient seulement un paramètre "es-cci" ou "es-ccd" mais non les deux, et est en mesure d'exécuter la commande sauf pour la portion relative à la surveillance électronique, cette passerelle MG DOIT renvoyer le code de réponse 212 – la transaction demandée a été exécutée normalement, mais tous les paramètres nécessaires à la surveillance électronique paramètres ne se trouvaient pas dans le paramètre LCO.

Si une passerelle MG, qui n'est pas en train d'exécuter une surveillance du contenu d'appel sur une connexion, reçoit une commande MDCX avec le paramètre "es-cci" ou "es-ccd" pour cette connexion, cette passerelle MG DOIT renvoyer le code de retour 213 – la transaction demandée a été exécutée normalement, mais la surveillance électronique n'a pas pu être lancée à mi-trajet.

Si une passerelle MG reçoit une commande CRCX ou une commande MDCX avec un champ "es-cci" ou "es-ccd" inutilisable et si cette passerelle MG est en mesure d'exécuter la commande sauf pour la portion relative à la surveillance électronique, la passerelle MG DOIT renvoyer le code de retour 214 – la transaction demandée a été exécutée normalement, mais les paramètres de surveillance électronique n'ont pas été reconnus. Si la passerelle MG, qui reçoit la commande MDCX avec le paramètre inutilisable "es-cci" ou "es-ccd", est déjà en train d'exécuter la surveillance du contenu d'appel, alors cette passerelle MG DOIT continuer à exécuter la surveillance du contenu d'appel et ignorer les paramètres "es-cci" et "es-ccd" dans la commande MDCX.

Quand une passerelle MG reçoit une commande AuditEndPoint avec le paramètre de capacités, cette passerelle MG DOIT renvoyer le mot clé "es" si la surveillance électronique est prise en charge.

Quand une passerelle MG, qui exécute une surveillance du contenu d'appel, reçoit une commande AuditConnection pour cette connexion avec un paramètre LocalConnectionOptions contenant un des paramètres qui sont audités, cette passerelle MG DOIT renvoyer, dans ce paramètre LCO, les paramètres "es-cci" et "es-ccd" qu'elle est actuellement en train d'utiliser afin d'exécuter la surveillance du contenu d'appel.

Appendice VIII

Exemple de paquetages d'événements

VIII.1 Paquetage de services d'opérateur multifréquences du groupe de fonctions D

Nom du paquetage: MO

Les codes contenus dans le Tableau VIII.1 servent à identifier des événements ou des signaux dans le cadre du paquetage "MO" pour la "signalisation des services d'opérateur" dans les circuits de jonction sortants unilatéraux multifréquences. La signalisation des services d'opérateur multifréquences du groupe de fonctions C est également prise en charge. Ce paquetage sera utilisé pour des jonctions de services généraux d'opérateur ainsi que pour des jonctions réservées aux services d'urgence:

Tableau VIII.1/J.171.1 – Codes utilisés pour identifier les événements et signaux du paquetage MO

Code	Description	Evénement	Signal	Informations supplémentaires
ans	Réponse à un appel	P	–	
ft	Tonalité de télécopie	√	–	
ld	Connexion de longue durée	C	–	
mt	Tonalité de modem	√	–	
orbk	Retour d'appel sonore de l'opérateur	√	–	
rbz	Blocage d'extrémité distante	P	–	
rcl	Rappel de l'opérateur	–	BR	
rel	Libération d'appel	P	BR	
res	Reprise d'appel	–	BR	
rlc	Libération achevée	P, S	BR	
sup(<addr>, <id>)	Etablissement d'un appel	–	TO	Temporisation variable
sus	Suspension d'appel	–	BR	
swk	Autorisation de numérotation	√	–	
TDD	Tonalités des appareils de télécommunication pour les personnes malentendantes (TDD)	√		
oc	Opération achevée	√		
of	Echec de l'opération	√		

La définition des différents événements et signaux est donnée ci-après:

Réponse à un appel (ans): ce signal est produit lorsque l'enregistrement-identification automatique des numéros est demandé par le système OSS, c'est-à-dire que l'appel n'a pas nécessairement été renvoyé en pseudo-transit vers un opérateur. Après la production d'un signal de réponse à un appel, le maintien de ressource est établi, à savoir que seul le système OSS pourra maintenant libérer le circuit de jonction.

Tonalité de télécopie (ft): l'événement de tonalité de télécopie est produit lorsqu'une communication de type télécopie est détectée – Voir p. ex. la Rec. UIT-T T.30 ou V.21.

Connexion de longue durée (ld): la "connexion de longue durée" est détectée lorsqu'une connexion a été établie depuis plus d'un certain temps. La valeur par défaut est 1 h mais peut être modifiée par le processus de préconfiguration.

Cet événement peut être détecté dans une connexion. Lorsque aucune connexion n'est spécifiée, l'événement s'applique à toutes les connexions reliées à l'extrémité, quel que soit l'instant où elles ont été établies.

Tonalité de modem (mt): l'événement de tonalité de modem est produit lorsqu'une communication de type modem est détectée – Voir p. ex. la Rec. UIT-T V.8.

Retour d'appel sonore de l'opérateur (orbk): cet événement se produit lorsque le système OSS demande que l'entité appelante soit alertée[33].

Blocage d'extrémité distante (rbz): cet événement a lieu lorsque le système OSS marque le circuit de jonction. Un événement de libération sera produit lorsque le circuit ne sera plus occupé.

Rappel de l'opérateur (rcl): ce signal peut être employé pour demander le rappel de l'opérateur, p. ex. à la suite d'une impulsion par le crochet commutateur du client afin de reprendre l'opérateur.

Libération d'appel (rel): la libération d'appel peut être signalée à la passerelle média, mais lorsque le maintien de ressource a été établi, il ne peut y avoir de déconnexion de la communication avant que le système OSS libère celle-ci. La passerelle média produit un événement de "libération d'appel" lorsqu'elle considère que le système OSS a libéré le circuit de jonction. Dans ce cas, l'événement peut être paramétré au moyen de l'un des codes du Tableau VIII.2, qui indiquent la cause de la libération:

Tableau VIII.2/J.171.1 – Codes de cause de libération d'appel

Code de cause	Cause
0	Libération normale
3	Pas de chemin vers la destination
8	Préemption
19	Pas de réponse
21	Appel rejeté
27	Destination hors service
28	Format de numéro non valable (p. ex. adresse incomplète)
38	Réseau hors service
111	Erreur de protocole ou de signalisation non spécifiée (p. ex. fin de temporisation)

Reprise d'appel (res): ce signal indique que le correspondant a repris l'appel, à savoir qu'il a décroché.

Libération achevée (rlc): l'extrémité et le contrôleur MGC emploient l'événement ou le signal de libération achevée pour confirmer que la communication a été libérée et que le circuit de jonction est disponible pour un autre appel.

[33] Si l'entité appelante a raccroché, la sonnerie sera généralement employée, tandis qu'une tonalité de recomposition sera en général utilisée dans le cas où l'entité appelante a décroché.

Etablissement d'un appel (sup(<addr>, <id>)): établissement d'un appel au système de service d'opérateur au moyen des informations qui ont été reçues au sujet de l'adresse et de l'identification. Les informations d'adresse seront de la forme suivante:

addr(MF_1, MF_2, …, MF_n)

tandis que celles qui se rapportent à l'identification seront de la forme suivante:

id(MF_1, MF_2, …, MF_n)

chaque chiffre MF_i correspondant à l'un des symboles numériques MF suivants dans le Tableau VIII.3:

Tableau VIII.3/J.171.1 – Symboles numériques MF

Symbole	Chiffre MF	Symbole	Chiffre MF
0	MF 0	K0	MF K0 ou KP
1	MF 1	K1	MF K1
2	MF 2	K2	MF K2
3	MF 3	S0	MF S0 ou ST
4	MF 4	S1	MF S1
5	MF 5	S2	MF S2
6	MF 6	S3	MF S3
7	MF 7	K0	MF K0 ou KP
8	MF 8		
9	MF 9		

Donc, un signal d'établissement d'appel pourrait p. ex. être de la forme suivante:

sup(addr(K0, 5,5,5,1,2,1,2, SO), id(K0, 5,5,5,1,2,3,4, SO))

Suspension d'appel (sus): ce signal indique que le correspondant a mis l'appel en suspens, à savoir qu'il a raccroché.

Autorisation de numérotation (swk): un contrôleur de passerelle média peut demander que la passerelle média lui notifie à quel moment le signal d'autorisation de numérotation est produit.

Tonalités des appareils de télécommunication pour les personnes malentendantes (TDD): l'événement TDD est produit lorsqu'une communication de type TDD est détectée – Voir la Rec. UIT-T V.18.

Opération achevée (oc): l'événement d'opération achevée est produit lorsque la passerelle a été invitée à appliquer un ou plusieurs signaux de type TO à l'extrémité et qu'un ou plusieurs d'entre eux ont pris fin sans avoir été arrêtés par la détection d'un événement demandé tel que passage à l'état décroché ou chiffre composé. Le rapport d'achèvement peut comporter en tant que paramètre le nom du signal dont la durée de vie s'est achevée, comme dans l'expression suivante:

O: MO/oc(MO/sup)

Lorsque l'événement d'opération achevée est demandé, il ne peut être paramétré au moyen de paramètres d'événement quelconques. Lorsque le nom du paquetage est omis, on suppose qu'il s'agit du nom par défaut.

Echec de l'opération (of): en général, l'événement d'échec de l'opération peut être produit lorsque l'extrémité a été invitée à appliquer un ou plusieurs signaux du type TO à l'extrémité et qu'un ou

plusieurs d'entre eux ont échoué avant l'interruption. Le rapport d'achèvement peut comporter en tant que paramètre le nom du signal qui a échoué, comme dans l'expression suivante:

```
O: MO/of(MO/sup)
```

Lorsque l'événement d'échec de l'opération est demandé, les paramètres d'événement ne peuvent être spécifiés. Lorsque le nom du paquetage est omis, on suppose qu'il s'agit du nom par défaut.

VIII.2 Paquetage de protocoles de terminaison multifréquence

Nom du paquetage: MT

Dans la présente version de la Recommandation relative au protocole TGCP, le paquetage ne peut être employé que pour la vérification de l'occupation des lignes (BLV, *busy-line verification*) et pour l'interruption par l'opérateur (OI, *operator interrupt*) dans des circuits de jonction entrants unilatéraux multifréquences de terminaison dédiés à ces tâches[34].

Les codes figurant dans le Tableau VIII.4 sont employés pour identifier les événements et les signaux dans le cas du paquetage "MT" pour les "circuits de jonction multifréquences de terminaison" entrants unilatéraux employés pour les services BLV et OI:

Tableau VIII.4/J.171.1 – Codes employés pour identifier les événements et signaux du paquetage MT

Code	Description	Evénement	Signal	Informations supplémentaires
ans	Réponse à un appel	–	BR	
bz	Tonalité d'occupation	–	TO	Temporisation = 30 s
hf	Impulsion-crochet	–	BR	
inf	Elément numérique d'information	√		
oc	Opération achevée	√	–	
of	Echec de l'opération	√	–	
oi	Interruption par l'opérateur	√	–	
pst	Tonalité continue	–	TO	Temporisation = illimitée
rel	Libération d'appel	P	BR	
res	Reprise d'appel	–	BR	
rlc	Libération achevée	P, S	BR	
ro	Tonalité de recomposition	–	TO	Temporisation = 30 s
sup	Etablissement d'un appel	P	–	
sus	Suspension d'appel	–	BR	

La définition des différents événements et signaux est donnée ci-après:

NOTE – Voir la Rec. UIT-T E.180/Q.35 pour les détails techniques particuliers des tonalités.

Réponse à un appel (ans): le signal de réponse à un appel informe l'extrémité que l'entité ayant fait l'objet d'une vérification a répondu. Cela comprend le cas où cette entité avait déjà décroché. L'extrémité est censée transmettre la supervision de la réponse au système OSS.

---

[34] Noter que lorsque les services d'opérateur sont fournis par un fournisseur extérieur au réseau, le système OSS peut ne pas avoir accès aux bases de données des abonnés pour déterminer si les services BLV et OI devraient être autorisés ou pas.

Tonalité d'occupation (bz): poste occupé.

Impulsion-crochet (hf): ce signal indique que l'entité ayant fait l'objet d'une vérification a effectué une impulsion-crochet.

Eléments numériques d'information (inf (<inf-digits>): ce signal est employé dans un circuit de jonction entrant multifréquence pour indiquer les chiffres reçus. Les valeurs du paramètre <inf-digits> sont tous les chiffres qui ont été recueillis jusqu'au délimiteur et y compris celui-ci, à savoir ST, ST', ST" ou ST"'.

Les valeurs du paramètre <inf-digits> sont données dans une liste de chiffres MF séparés par des virgules:

$MF_1, MF_2, \ldots, MF_n$

chaque chiffre MF_I correspondant à l'un des symboles numériques suivants dans le Tableau VIII.5:

Tableau VIII.5/J.171.1 – Symboles numériques MF

Symbole	Chiffre MF	Symbole	Chiffre MF
0	MF 0	K0	MF K0 ou KP
1	MF 1	K1	MF K1
2	MF 2	K2	MF K2
3	MF 3	S0	MF S0 ou ST
4	MF 4	S1	MF S1
5	MF 5	S2	MF S2
6	MF 6	S3	MF S3
7	MF 7	K0	MF K0 ou KP
8	MF 8		
9	MF 9		

Donc un signal ou un événement pourrait p. ex. être de la forme suivante:

```
inf(k0, 5,5,5,1,2,3,4, s0)
```

Un exemple dans lequel la temporisation entre les chiffres expire après les chiffres 5,5,5 serait le suivant:

```
inf(k0, 5,5,5)
```

Opération achevée (oc): voir la définition correspondante dans le paquetage des passerelles de jonction ISUP.

Echec de l'opération (of): voir la définition correspondante dans le paquetage des passerelles de jonction ISUP.

Interruption par l'opérateur (oi): l'événement d'interruption par l'opérateur apparaît lorsque l'opérateur tente d'interrompre la communication. Il produit une tonalité "interruption par l'opérateur". Puisque aucune tonalité normalisée n'est définie à cette fin, cet événement est défini de manière à se produire lorsqu'un certain niveau énergétique est détecté dans le circuit de jonction, correspondant à une transition du bruit de circuit à la voix ou aux tonalités. Il convient de noter qu'il n'est pas possible de détecter une transition inverse de la voix ou des tonalités au bruit de circuit.

Tonalité continue (pst): **libération d'appel (rel)**: le contrôleur MGC peut utiliser le signal de libération pour libérer la communication[35]. Dans ce cas, le signal de libération peut ne pas être paramétré.

L'extrémité peut en revanche employer cet événement pour informer le contrôleur MGC qu'elle a libéré la communication. Dans ce cas, l'événement peut être paramétré au moyen de l'un des codes figurant dans le Tableau VIII.6, qui indiquent la cause de la libération:

Tableau VIII.6/J.171.1 – Codes de cause de libération de l'appel

Code du cause	Cause
0	Libération normale
3	Pas de route vers la destination
8	Préemption
19	Pas de réponse
21	Appel rejeté
27	Destination hors service
28	Format de numéro non valable (p. ex. adresse incomplète)
38	Réseau hors service
111	Erreur de protocole ou de signalisation non spécifiée (p. ex. fin de temporisation)

Reprise d'appel (res): ce signal indique que l'entité qui a fait l'objet d'une vérification a repris l'appel, à savoir qu'elle a décroché.

Libération achevée (rlc): l'extrémité et le contrôleur MGC emploient l'événement ou le signal de libération achevée pour confirmer que la communication a été libérée et que le circuit de jonction est disponible pour d'autres appels.

Etablissement d'appel (sup): un événement "sup" est utilisé pour indiquer l'arrivée d'un appel entrant (correspondant à l'événement de décrochage entrant). Cet événement est fourni sans paramètre.

Suspension d'appel (sus): ce signal indique que l'entité qui a fait l'objet d'une vérification a mis l'appel en suspens, à savoir qu'elle a raccroché.

[35] Noter que l'opérateur qui effectue la vérification commande normalement la libération de connexions achevées sans essai et que le signal de suspension devrait donc généralement être employé.

BIBLIOGRAPHIE

– *Bellcore Notes on the Networks*, Bellcore, SR-2275.

– *Compatibility Information for Feature Group D Switched Access Service,* Bellcore, TR-NPL-000258, Issue 1, octobre 1985.

– *Interoffice LATA Switching Systems Generic Requirements (LSSGR): Verification Connections (25-05-0903)*, Bellcore, TR-TSY-000531, Issue 2, juillet 1987.

– *Signalling for Analog Interfaces*, Bellcore, LSSGR GR-506-CORE, Issue 1, June 1996.

– *Switching System Generic Requirements for Call Control Using the Integrated Services Digital Network User Part (ISDNUP)*, Bellcore, LSSGR GR-317-CORE, Issue 2, décembre 1997.

– *Custom Call-Handling Features (FSD 80 Series)*, Bellcore, OSSGR GR-1176-CORE, Issue 1, mars 1999.

– IETF RFC 1827 (1995), *IP Encapsulating Security Payload (ESP)*.

– IETF RFC 2974 (Experimental, 2000), *Session Announcement Protocol*.

– *RTP Parameters*, http://www.iana.org/assignments/rtp-parameters.

SÉRIES DES RECOMMANDATIONS UIT-T

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

Imprimé en Suisse
Genève, 2006