

Unión Internacional de Telecomunicaciones

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

J.171.1

(11/2005)

SERIE J: REDES DE CABLE Y TRANSMISIÓN DE
PROGRAMAS RADIOFÓNICOS Y TELEVISIVOS, Y DE
OTRAS SEÑALES MULTIMEDIOS

IPCablecom

**Protocolo de control de pasarela de circuitos
troncales IPCablecom: Perfil 1**

Recomendación UIT-T J.171.1

UIT-T



Recomendación UIT-T J.171.1

Protocolo de control de pasarela de circuitos troncales IPCablecom: Perfil 1

Resumen

Esta Recomendación describe un perfil IPCablecom de una interfaz de programación de aplicación (API), denominada interfaz de controlador de pasarela de medios (MGCI), y el protocolo correspondiente (MGCP) para el control de pasarelas de voz por IP (VoIP) en la RTPC, desde elementos de control de llamada externos. Éste es uno de los dos perfiles que se mencionan en la Rec. UIT-T J.171.0; el otro se especifica en la Rec. UIT-T J.171.2.

En el MGCP se supone que hay una arquitectura de control de llamada en la que la "inteligencia" de control está fuera de las pasarelas y la gestionan elementos de control de llamada externos. El perfil IPCablecom que se describe en la presente Recomendación se denomina protocolo de control de pasarela troncal (TGCP) IPCablecom.

Orígenes

La Recomendación UIT-T J.171.1 fue aprobada el 29 de noviembre de 2005 por la Comisión de Estudio 9 (2005-2008) del UIT-T por el procedimiento de la Recomendación UIT-T A.8.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la UIT. Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Asamblea Mundial de Normalización de las Telecomunicaciones (AMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución 1 de la AMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI.

NOTA

En esta Recomendación, la expresión "Administración" se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

La observancia de esta Recomendación es voluntaria. Ahora bien, la Recomendación puede contener ciertas disposiciones obligatorias (para asegurar, por ejemplo, la aplicabilidad o la interoperabilidad), por lo que la observancia se consigue con el cumplimiento exacto y puntual de todas las disposiciones obligatorias. La obligatoriedad de un elemento preceptivo o requisito se expresa mediante las frases "tener que, haber de, hay que + infinitivo" o el verbo principal en tiempo futuro simple de mandato, en modo afirmativo o negativo. El hecho de que se utilice esta formulación no entraña que la observancia se imponga a ninguna de las partes.

PROPIEDAD INTELECTUAL

La UIT señala a la atención la posibilidad de que la utilización o aplicación de la presente Recomendación suponga el empleo de un derecho de propiedad intelectual reivindicado. La UIT no adopta ninguna posición en cuanto a la demostración, validez o aplicabilidad de los derechos de propiedad intelectual reivindicados, ya sea por los miembros de la UIT o por terceros ajenos al proceso de elaboración de Recomendaciones.

En la fecha de aprobación de la presente Recomendación, la UIT no ha recibido notificación de propiedad intelectual, protegida por patente, que puede ser necesaria para aplicar esta Recomendación. Sin embargo, debe señalarse a los usuarios que puede que esta información no se encuentre totalmente actualizada al respecto, por lo que se les insta encarecidamente a consultar la base de datos sobre patentes de la TSB.

© UIT 2006

Reservados todos los derechos. Ninguna parte de esta publicación puede reproducirse por ningún procedimiento sin previa autorización escrita por parte de la UIT.

ÍNDICE

	Página
1 Alcance	1
2 Referencias	1
2.1 Referencias normativas	1
2.2 Referencias informativas	1
3 Definiciones.....	2
4 Abreviaturas, siglas y acrónimos.....	2
5 Convenios	3
5.1 Bases.....	3
6 Introducción.....	4
6.1 Relación con otras normas IPCablecom.....	5
6.2 Relación con la RFC 3435 y la gramática ABNF	6
7 Interfaz de control de pasarela de medios (MGCI)	6
7.1 Modelo y convenios de denominación	6
7.2 Utilización de SDP	14
7.3 Funciones de control de pasarela.....	14
7.4 Estados, condiciones de cambio-por-fallo y situación de competencia	39
7.5 Códigos de retorno y códigos de error	56
7.6 Códigos de motivo.....	58
7.7 Utilización de los descriptores de opciones de conexión local y de conexión	58
8 Protocolo de control de pasarela de medios	61
8.1 Descripción general	62
8.2 Encabezamiento de instrucción	62
8.3 Formatos del encabezamiento	75
8.4 Codificación de descripción de sesión	79
8.5 Transmisión utilizando el protocolo UDP	90
8.6 Remolque o envío de instrucciones adjuntas.....	92
8.7 Identificadores de transacción y triple toma de contacto	93
8.8 Respuestas provisionales	94
9 Seguridad.....	95
Anexo A – Lotes de eventos	96
A.1 Lote de circuitos troncales ISUP	96
Apéndice I – Interacciones de modos	100
Apéndice II – Ejemplos de codificación de instrucciones	102
II.1 NotificationRequest	102
II.2 Notify.....	102
II.3 CreateConnection	102

	Página
II.4 ModifyConnection.....	104
II.5 DeleteConnection (procedente del controlador de pasarela de medios (MGC)).....	105
II.6 DeleteConnection (procedente de la pasarela de circuitos troncales).....	105
II.7 DeleteConnection (procedente del controlador de pasarela de medios, en caso de múltiples conexiones).....	106
II.8 AuditEndpoint	106
II.9 AuditConnection.....	107
II.10 RestartInProgress.....	108
Apéndice III – Ejemplo de flujo de llamada.....	109
Apéndice IV – Requisitos de los puntos extremos	113
Apéndice V – Información sobre compatibilidad.....	114
V.1 Compatibilidad con NCS.....	114
V.2 Compatibilidad con MGCP	114
Apéndice VI – Gramática ABNF para TGCP.....	116
Apéndice VII – Vigilancia electrónica.....	123
VII.1 MGC.....	123
VII.2 La MG	123
Apéndice VIII – Ejemplos de lotes de eventos.....	125
VIII.1 Lote de servicios de operador MF FGD	125
VIII.2 Lote de protocolo de terminación MF	128
BIBLIOGRAFÍA	131

Recomendación UIT-T J.171.1

Protocolo de control de pasarela de circuitos troncales IPCablecom: Perfil 1

1 Alcance

Esta Recomendación describe un perfil IPCablecom de una interfaz de programación de aplicación (API, *application programming interface*), denominada interfaz de controlador de pasarela de medios (MGCI, *media gateway control interface*), y el protocolo correspondiente (MGCP, *media gateway control protocol*) para el control de pasarelas de voz por IP (VoIP, *voice-over-IP*) en la RTPC, desde elementos de control de llamada externos. Éste es uno de los dos perfiles que se mencionan en la Rec. UIT-T J.171.0; el otro se especifica en la Rec. UIT-T J.171.2.

En el MGCP se supone que hay una arquitectura de control de llamada en la que la "inteligencia" de control de llamada está fuera de las pasarelas y la gestionan elementos de control de llamada externos. El perfil IPCablecom que se describe en la presente Recomendación se denomina protocolo de control de pasarela troncal (TGCP, *trunking gateway control protocol*) IPCablecom.

Esta Recomendación se basa en la Recomendación sobre señalización de llamada basada en red IPCablecom (Rec. UIT-T J.162) y en el IETF RFC 2705. Esta Recomendación, en la que se define el protocolo TGCP IPCablecom, es una especificación independiente del MGCP. El perfil TGCP de MGCP está definido única y exclusivamente por el contenido de la presente Recomendación.

2 Referencias

2.1 Referencias normativas

Las siguientes Recomendaciones del UIT-T y otras referencias contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones y otras referencias son objeto de revisiones, por lo que se preconiza que los usuarios de esta Recomendación investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones y otras referencias citadas a continuación. Se publica periódicamente una lista de las Recomendaciones UIT-T actualmente vigentes. En esta Recomendación, la referencia a un documento, en tanto que autónomo, no le otorga el rango de una Recomendación.

- Recomendación UIT-T J.161 (Proyecto, versión 2), *Requisitos de los códecs de audio para la prestación de servicios de audio bidireccionales por redes de televisión por cable que utilizan módems de cable.*
- Recomendación UIT-T J.162 (2005), *Protocolo de señalización de llamada de red para la prestación de servicios dependientes del tiempo por redes de televisión por cable que utilizan módems de cable.*
- Recomendación UIT-T J.170 (2005), *Especificación de seguridad de IPCablecom.*
- IETF RFC 2327 (1998), *SDP: Session Description Protocol.*

2.2 Referencias informativas

- Recomendación UIT-T E.180/Q.35 (1998), *Características técnicas de los tonos para el servicio telefónico.*
- Recomendación UIT-T J.163 (2005), *Calidad de servicio dinámica para la prestación de servicios en tiempo real por las redes de televisión por cable que utilizan módems de cable.*

- Recomendación UIT-T J.171.0 (2005), *Protocolo de control de pasarela de circuitos troncales IP-Cablecom: Aspectos generales de los perfiles*.
- 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 Definiciones

En esta Recomendación se definen los términos siguientes.

3.1 módem de cable: Dispositivo que proporciona el acceso de datos de alta velocidad a instalaciones de cliente que utilizan equipo construido de acuerdo con las Recs. UIT-T J.83 y J.112.

3.2 IP-Cablecom: Proyecto del UIT-T que incluye una arquitectura y una serie de Recomendaciones que permiten el suministro de servicios en tiempo real a través de redes de televisión por cable que utilizan módems de cable.

4 Abreviaturas, siglas y acrónimos

En esta Recomendación se utilizan las siguientes abreviaturas, siglas y acrónimos.

DNS	Sistema de nombres de dominio (<i>domain name system</i>)
IP	Protocolo Internet (<i>Internet protocol</i>)
IPSec	Seguridad del protocolo Internet (<i>Internet protocol security</i>)
MGC	Controlador de pasarela de medios (<i>media gateway controller</i>)
MGCP	Protocolo de control de pasarela de medios (<i>media gateway control protocol</i>)
MIB	Base de información de gestión (<i>management information base</i>)
MTA	Adaptador de terminal de medios (<i>media terminal adapter</i>)
MWD	Periodo de espera máximo (<i>maximum waiting delay</i>)
NCS	Señalización de llamada basada en la red (<i>network-based call signaling</i>)
NTP	Protocolo de tiempo de red (<i>network time protocol</i>)
PU-RDSI	Parte usuario de la red digital de servicios integrados
QoS	Calidad de servicio (<i>quality of service</i>)
RTCP	Protocolo de control en tiempo real (<i>real time control protocol</i>)
RTO	Temporizador de retransmisión (<i>retransmission timeout</i>)
RTP	Protocolo en tiempo real (<i>real-time protocol</i>)
SDP	Protocolo de descripción de sesión (<i>session description protocol</i>)
SG	Pasarela de señalización (<i>signalling gateway</i>)
SPI	Índice de parámetros de seguridad (<i>security parameters index</i>)

5 Convenios

Si esta Recomendación es implementada, las palabras clave "DEBE" (MUST o SHALL en inglés) Y OBLIGATORIO deben interpretarse como un aspecto obligatorio de esta Recomendación.

Las palabras clave que indican un cierto nivel de importancia de requisitos particulares que se utilizan en toda esta Recomendación se resumen a continuación.

"DEBE(N)"	Esta palabra, o el adjetivo "REQUERIDO", significa que el elemento en cuestión es un requisito absoluto de esta Recomendación.
"NO DEBE(N)"	Esta expresión significa que el elemento es una prohibición absoluta de esta Recomendación.
"DEBERÍA(N)"	Esta palabra, o el adjetivo "RECOMENDADO", significa que, en determinadas circunstancias pueden existir motivos válidos para hacer caso omiso del elemento de que se trate, pero que deberían tenerse en cuenta todas las implicaciones y ponderar cuidadosamente el caso antes de decidir optar por una vía diferente.
"NO DEBERÍA(N)"	Esta expresión significa que pueden existir motivos válidos en determinadas circunstancias en las que el comportamiento indicado sea aceptable o incluso de utilidad, pero que deberían tenerse en cuenta todas las implicaciones y ponderar cuidadosamente el caso antes de implementar cualquier comportamiento descrito con esta etiqueta.
"PUEDE(N)"	Esta palabra, o el adjetivo "OPCIONAL" o "FACULTATIVO", significa que el elemento es verdaderamente facultativo. Un vendedor puede optar por incluir el elemento porque así se exige en un determinado mercado o porque mejora el producto, por ejemplo; otro vendedor puede omitir el mismo elemento.

5.1 Bases

Esta Recomendación describe un perfil IPCablecom de una API, denominada MGCI, y el protocolo correspondiente (MGCP) para el control de pasarelas de voz por IP (VoIP) en la RTPC, desde elementos de control de llamada externos. En el MGCP se supone que hay una arquitectura de control de llamada en la que la "inteligencia" de control de llamada está fuera de las pasarelas y la gestionan elementos de control de llamada externos. El perfil IPCablecom que se describe en la presente Recomendación se denomina Perfil 1 del TGCP.

Esta Recomendación se basa en la Rec. UIT-T J.162 sobre señalización de llamada basada en red (NCS, *network-based call signalling*) IPCablecom, en el MGCP 1.0 de IETF RFC 3435 (que es el resultado de la fusión de los proyectos del IETF sobre el protocolo simple de control de pasarela (SGCP, *simple gateway control protocol*) y sobre la familia de protocolos para el control de dispositivos de protocolo Internet (IPDC, *IP device control*)) y en el resultado del trabajo del Grupo Especial sobre pasarelas RTPC IPCablecom.

Esta Recomendación, en la que se define el protocolo TGCP IPCablecom, es una especificación independiente del MGCP que sirve de referencia estable y responde a los actuales requisitos para la transposición efectiva de esa referencia. En la medida de lo posible, se pretende que sea compatible con la NCS y con el MGCP 1.0 en el entorno IPCablecom, para evitar tener que desarrollar varios protocolos diferentes que resuelvan el mismo problema. Con este fin, se sigue colaborando con los autores de las Recomendaciones relativas a la NCS y al MGCP. No obstante, el perfil TGCP de MGCP está definido única y exclusivamente por el contenido de esta Recomendación.

Este perfil TGCP del MGCP, que en adelante se denominará Protocolo 1.0 de señalización de llamada en la pasarela troncal de la RTPC, TGCP 1.0, perfil TGCP o simplemente TGCP, es una variante del protocolo MGCP 1.0 de la IETF RFC 2435, con las siguientes modificaciones:

- *El protocolo TGCP sólo está destinado a soportar pasarelas de medios de la RTPC de VoIP IPCablecom.* El protocolo TGCP soporta las pasarelas de medios de la RTPC de VoIP, conforme a la definición dada en IPCablecom. Se suprimió la funcionalidad que existía en el MGCP 1.0 y que era irrelevante para el TGCP.
- *El protocolo TGCP contiene ampliaciones y modificaciones del MGCP.* En el TGCP se consideran requisitos específicos de IPCablecom. Sin embargo, se han conservado la arquitectura del MGCP y todas las funciones pertinentes a las pasarelas RTPC.
- *El protocolo TGCP contiene pequeñas simplificaciones del MGCP 1.0.* Cuando había varias posibilidades, no necesariamente indispensables en una pasarela RTPC en el entorno IPCablecom, se efectuaron algunas simplificaciones para el caso de pasarelas troncales.

Si bien el MGCP no es el TGCP, y viceversa, al basarse esta Recomendación en el MGCP se utilizarán indistintamente ambos nombres. A menos que se indique o se pueda deducir del contexto otra cosa, en esta Recomendación la sigla MGCP indica el perfil TGCP del MGCP.

El TGCP ha sido diseñado para satisfacer los requisitos de protocolo para la interfaz entre el controlador de pasarela de medios y la pasarela de medios, en la arquitectura IPCablecom.

6 Introducción

En esta Recomendación se describe un perfil TGCP de una MGCI y su protocolo correspondiente (MGCP), para el control de pasarelas troncales desde elementos de control de llamada externos. Una pasarela troncal es un elemento de red que proporciona acceso a un circuito troncal de señalización analógico, con emulación analógica, o de portador digital y asociado al canal, en redes VoIP.

Estas pasarelas sirven como interfaces de la RTPC y, por tanto, han de seguir las normas eléctricas, de funcionamiento y de señalización pertinentes que correspondan al tipo de troncal que se utilice. Esta Recomendación que define el TGCP está destinada a los operadores de centrales locales en condiciones de competencia (CLEC, *competitor local exchange carrier*). En su primera versión se aplicará solamente a los siguientes tipos de circuitos troncales:

- PU-RDSI con el SS7:
 - Troncal portadora normalizada (*standard bearer trunk*).
- Circuitos troncales MF que combinan centrales locales de acceso uniforme y sistemas de conmutación de acceso (EAEO/AT):
 - Troncales para servicios de operador¹.
 - Acceso de abonado a operadores CLEC a través de un sistema de conmutación de operador local (LEC).
 - Acceso de operador a un EAEO/AT para la verificación de línea ocupada (*busy line verification*) e intervención (*barge in*).
 - Troncales de servicio de emergencia para el acceso a sistemas de conmutación de ese servicio.

En el MGCP se supone que hay una arquitectura de control de llamada en la que la "inteligencia" de control de llamada está fuera de las pasarelas y la gestionan elementos de control de llamada

¹ Se considera que estos servicios los presta un operador de central local (LEC) o que se pueda acceder a ellos a través de él.

externos, denominados MGC. Se supone que estos elementos se sincronizarán entre ellos para enviar instrucciones coherentes a las pasarelas que controlan. El MGCP que se define en esta Recomendación no incluye un mecanismo de sincronización de los MGC, aunque es posible que en futuras Recomendaciones de IPCablecom se especifiquen dichos mecanismos.

El MGCP parte de un modelo de conexión en el que los elementos básicos son puntos extremos y conexiones. Una pasarela contiene un grupo de puntos extremos que son fuentes o sumideros de datos, y que pueden ser físicos o virtuales.

Un ejemplo de punto extremo físico es un circuito troncal en una pasarela troncal que termina una troncal portadora de la PU-RDSI en un conmutador. Otro ejemplo es un cliente incorporado o una pasarela residencial que terminan líneas POTS residenciales (en los teléfonos), aunque esta Recomendación no trata dichos dispositivos.

Un ejemplo de punto extremo virtual es una fuente de audio en un servidor de contenido de audio. La creación de puntos extremos físicos requiere la instalación de equipos, mientras que la de los virtuales puede hacerse utilizando software. Sin embargo, el perfil TGCP de MGCP sólo se refiere a puntos extremos físicos.

Las conexiones son punto a punto, es decir, asociaciones entre puntos extremos a los efectos de transmitir datos entre ellos. Una vez se ha establecido dicha asociación para ambos puntos extremos, puede efectuarse la transferencia de datos. La asociación se lleva a cabo creando dos mitades que componen la conexión, una en el punto extremo de origen y otra en el de terminación.

Los MGC ordenan a las pasarelas crear conexiones entre puntos extremos, detectar determinados eventos, por ejemplo pruebas de continuidad, y generar ciertas señales, por ejemplo la de tono de llamada. Es potestad exclusiva del MGC especificar cuándo y cómo se establecen las conexiones, y entre cuáles puntos extremos, así como los eventos y señales que se han de detectar en cada uno de esos puntos. De este modo, la pasarela es un simple dispositivo que no tiene ningún estado de llamada, que recibe instrucciones generales del MGC sin que para ello necesite conocer, e incluso entender, conceptos tales como llamada, estados de llamada, características o interacciones de características. Cuando se introducen nuevos servicios, se cambian los perfiles de abonado, etc., estas modificaciones son transparentes para la pasarela. El MGC implementa los cambios y genera las nuevas instrucciones para las pasarelas, correspondientes a las modificaciones efectuadas. Cuando la pasarela se reinicie, se encontrará en un estado vacío y llevará simplemente a cabo las instrucciones del MGC a medida que las recibe.

6.1 Relación con otras normas IPCablecom

Una pasarela RTPC IPCablecom consta de tres componentes funcionales:

- El controlador de pasarela de medios (MGC, *media gateway controller*), que contiene la inteligencia de llamada y termina la señalización de llamada. También conocido como agente de llamada.
- La pasarela de medios (MG, *media gateway*), que termina, por orden y bajo el control del MGC, el canal portador. También se denomina pasarela troncal (TGW, *trunking gateway*),
- La pasarela de señalización (SG, *signalling gateway*), que conecta la señalización de llamada a la RTPC y suministra la función de traducción de señalización.

Además de los protocolos de pasarela RTPC, IPCablecom también incluye un protocolo de señalización de llamada basada en la red (NCS), que es un perfil del MGCP. La norma SDP es fundamental en esta arquitectura. Tanto la NCS como el TGCP utilizan el SDP para la descripción de sesiones.

Los sistemas NCS y TGCP de IPCablecom tramitan los estados de llamadas a través de agentes de llamada CMS y MGC. La figura 1 muestra la relación entre estos componentes:

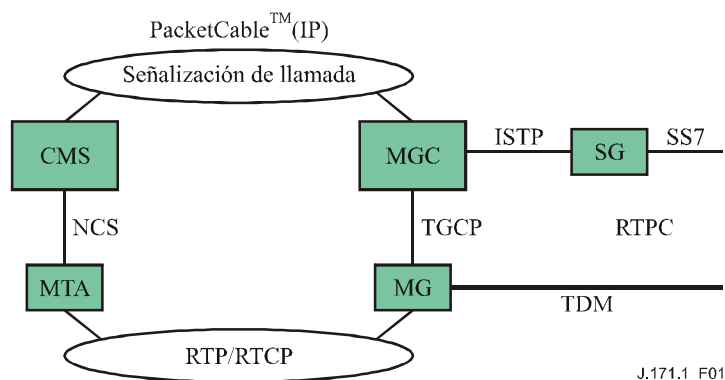


Figura 1/J.171.1 – Relación entre los componentes NCS y TGCP

6.2 Relación con la RFC 3435 y la gramática ABNF

La RFC 3435 incluye una descripción formal de la sintaxis del protocolo MGCP conforme a la "forma aumentada de Backus-Naur para especificaciones de sintaxis" (ABNF). Los creadores de sistemas utilizan esta descripción formal para producir dispositivos compatibles. En el apéndice VI se suministra una copia de la sintaxis del protocolo MGCP, comentada y editada con el fin de indicar su aplicabilidad a las Recomendaciones IPCablecom. La conformidad con estas directrices puede mejorar la interoperabilidad, minimizando los fallos causados por diferentes interpretaciones de la sintaxis y la gramática.

7 Interfaz de control de pasarela de medios (MGCI)

Las funciones de la interfaz MGCI incluyen el control de la conexión, control de punto extremo, auditoría e informes de calificación. Estas funciones utilizan el mismo modelo de sistema y los mismos convenios de denominación.

7.1 Modelo y convenios de denominación

El MGCP presupone un modelo de conexión en el que los constructivos básicos son puntos extremos y conexiones. Las conexiones se agrupan en llamadas. Una o más conexiones pueden pertenecer a una misma llamada. Las conexiones y llamadas se establecen por iniciativa de uno o varios MGC. No obstante, hay que precisar que en ninguno de estos casos se establece una "conexión" dentro de una red IPCablecom, en el sentido en que se entiende el término "conexión" en el contexto de la red telefónica pública con conmutación de circuitos. Los términos "llamada" y "conexión" en este contexto (y en la presente Recomendación) se utilizan para facilitar la exposición y no para indicar una similitud técnica o de cualquier otro tipo entre la red IPCablecom y la RTPC.

7.1.1 Nombres de punto extremo

Los nombres de punto extremo, llamados también identificadores de punto extremo, tienen dos componentes que, por definición, no distinguen entre mayúsculas o minúsculas:

- el nombre de dominio de la pasarela que gestiona el punto extremo;
- un nombre de punto extremo local dentro de esa pasarela.

Los nombres de punto extremo serán de la forma:

local-endpoint-name@domain-name

donde domain-name es un nombre de dominio absoluto tal como se define en IETF RFC 1034 e incluye uno para el sistema principal (host); ejemplo de nombre de dominio:

MyTrunkingGateway.cablelabs.com

Asimismo, domain-name puede ser una dirección IPv4 en forma de expresión con puntos decimales, representada como una cadena de texto y encerrada entre un corchete izquierdo y un corchete derecho ("[" y "]") como en "[128.96.41.1]" – para una información detallada, consúltese IETF RFC 821. Ahora bien, en general no se aconseja utilizar direcciones IP.

Las pasarelas de circuitos troncales tienen asociados uno o más puntos extremos (por ejemplo, uno para cada circuito troncal) y cada uno de los puntos extremos está identificado por un nombre de punto extremo local distinto. Al igual que el nombre de dominio, el nombre de punto extremo local es insensible a la escritura en mayúscula o minúscula. El nombre de punto extremo local tiene asociado un determinado tipo de punto extremo, por ejemplo, DS-0, o una línea de acceso analógica. El tipo puede obtenerse del nombre de punto extremo local. El nombre de punto extremo local es un nombre jerárquico, en el que el componente menos específico del nombre es el término más a la izquierda y el componente más específico es el término más a la derecha. Expresado esto de una manera más formal, el nombre de punto extremo debe satisfacer las siguientes reglas de denominación:

- Los distintos términos del nombre de punto extremo local deben estar separados por una barra oblicua ("/", carácter ASCII 2F hex).
- Los distintos términos son cadenas de caracteres ASCII compuestos de letras, dígitos u otros caracteres imprimibles, con excepción de los caracteres utilizados como delimitadores en nombres de punto extremo ("'", "@"), los caracteres utilizados como comodín ("*", "\$") y los caracteres de espacio blanco.
- Los comodines son un asterisco ("*") o un signo de dólar ("\$\$") para los términos del trayecto de denominación que van a ser abarcados por el comodín. Por tanto, el nombre completo de punto extremo local aparece así:

term1/term2/term3

y si para uno de los términos del nombre de punto extremo local se usa un comodín, el nombre de punto extremo local aparece así:

term1/term2/* si term3 es el comodín.

term1/*/* si term2 y term3 son los comodines.

En cada uno de estos ejemplos se hubiera podido utilizar un signo de dólar en lugar del asterisco.

- El uso de comodín sólo está permitido desde la derecha; por tanto, si se usa un comodín para un término, el comodín abarca los términos a la derecha de éste.
- En aquellos casos en que se utilizan en forma mixta los comodines signo de dólar y asterisco, los signos de dólar sólo están permitidos desde la derecha; por tanto, si un término tiene un comodín signo de dólar, todos los términos a la derecha de ese término deben también contener comodines signo de dólar.
- Un término representado por un asterisco debe interpretarse como: "utilizar *todos* los valores de este término conocidos dentro del alcance de la pasarela de circuitos troncales en cuestión". A menos que se especifique lo contrario, se refiere a todos los puntos extremos configurados para el servicio, sin importar cuál sea su estado real de servicio, es decir, si están en servicio o fuera de él.
- Un término representado por un signo de dólar debe interpretarse como: "utilizar *cualquier* valor de este término que sea conocido dentro del alcance de la pasarela de circuitos

troncales en cuestión". A menos que se especifique lo contrario, se refiere solamente a los puntos extremos que están en servicio.

- Cada punto extremo puede especificar detalles adicionales en las reglas de denominación para ese tipo de punto extremo; sin embargo, dichas reglas no podrán estar en contradicción con lo anteriormente expuesto.

Si se utilizan tipos de punto extremo diferentes, o incluso términos secundarios diferentes, por ejemplo "líneas", dentro del mismo tipo de punto extremo, el resultado será dos nombres de punto extremo local diferentes. En consecuencia, cada "línea" se tratará como un punto extremo distinto. Como la porción de nombre de dominio es parte del identificador de punto extremo, no es posible intercambiar libremente diferentes formas o valores que hagan referencia a la misma entidad. Tras un rearranque, siempre se UTILIZARÁ la forma y el valor suministrados más recientemente.

7.1.1.1 Nombres de punto extremo en las pasarelas de circuitos troncales

Para los puntos extremos de pasarelas de circuitos troncales se utilizarán los convenios de denominación adicionales especificados en esta cláusula.

Las pasarelas de circuitos troncales soportarán el siguiente tipo de punto extremo básico:

- `ds` Un circuito troncal DS-0
Normalmente el tipo de punto extremo básico se configura con información adicional sobre el tipo de señalización soportado en el circuito troncal y el papel que desempeña en el sistema de conmutación.

7.1.1.1.1 Puntos extremos de circuito troncal

Además de los convenios de denominación especificados anteriormente, los nombres de punto extremo local para puntos extremos de pasarela de circuitos troncales RTPC de tipo "ds" DEBEN ser conformes a las siguientes reglas:

- Los nombre de punto extremo local consistirán en una serie de términos separados entre sí por una barra oblicua ("/"), que describen la jerarquía física dentro de la pasarela:

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

- El primer término (`ds`) identifica el esquema de denominación de puntos extremos utilizado y el tipo de punto extremo básico.
- El último término es un número decimal que indica el número de *canal*² en el nivel más bajo de la jerarquía.
- Los términos intermedios entre el primero (`ds`) y el último (número de canal) representan niveles intermedios de la jerarquía y consisten en `<unit-type>` y `<unit #>` separados por un guión ("-") donde:
 - el `<unit-type>` identifica el nivel jerárquico particular. Son valores de `<unit-type>` actualmente definidos: "s", "su", "oc3", "ds3", "e3", "ds2", "e2", "ds1", "e1" donde "s" indica número de intervalo de tiempo y "su" indica una subunidad dentro de un intervalo de tiempo. También se permitirán otros valores que representan niveles jerárquicos físicos que no han sido incluidos en esta lista, pero que siguen las mismas reglas de denominación.
 - el `<unit #>` es un número decimal que se utiliza para hacer referencia a un ejemplar particular de un `<unit-type>` en ese nivel de la jerarquía.

² Obsérvese la utilización del término "channel" (canal) en lugar de "timeslot" (intervalo de tiempo).

- El número de niveles y la denominación de esos niveles se basa en la jerarquía física dentro de la pasarela de medios, como se muestra en los siguientes ejemplos:
 - Una pasarela de medios que tiene cierto número de interfaces DS1:

`ds/ds1-#/`

- Una pasarela de medios que tiene cierto número de interfaces OC3, que contienen jerarquías DS3 y DS1 canalizadas:

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

- Una pasarela de medios que contiene cierto número de intervalos de tiempo, cada uno de los cuales tiene cierto número de interfaces DS3:

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

- Es posible que algunos puntos extremos no contengan todos los niveles posibles de una jerarquía; sin embargo, todos los niveles soportados por un punto extremo dado están contenidos en el esquema de denominación de puntos extremos. Por ejemplo, un DS3 sin entramado DS1 podría representarse por el siguiente esquema de denominación:

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

En cambio, un DS3 *con* entramado DS1 no podría representarse por ese esquema de denominación.

- La denominación mediante uso de comodines sigue los convenios indicados en 7.1.1, donde el carácter asterisco ("*") hace referencia a "all" (todos), y el carácter signo de dólar (\$) hace referencia a "any" (cualquiera). También es posible utilizar comodines en forma de una gama "[N-M]" que representa una "gama" ("range") de canales desde el canal N hasta el canal M inclusive:
 - Debe señalarse que la utilización del comodín "all" para el primer término (`ds`) hace referencia a todos los tipos de punto extremo en la pasarela de medios, cualquiera que sea su tipo. Esto se utilizará normalmente para fines administrativos, por ejemplo, auditoría o rearranque.
 - Un nombre de punto extremo local puede aparecer subespecificado, suministrando un número de términos menor que el normal, comenzando por la izquierda del nombre de punto extremo. En ese caso, se supone que los términos que faltan a la derecha del último término especificado son el carácter comodín "*", (todos). Ahora bien, si los términos especificados contienen el carácter comodín "any of" (cualquiera de), se supone que los términos que faltan a la derecha del último término especificado son el carácter comodín "any of".
 - Dondequiera que se permita el uso del comodín "all", el comodín gama de canales "[N-M]" puede utilizarse en el último término (es decir, `<channel-#>`) del nombre de punto extremo local, en vez de aquél. El comodín de gama hará entonces referencia a todos los canales de N a M. Las reglas y restricciones aplicables a la utilización del comodín "all" también son aplicables a la utilización del comodín de gama ("range").

Los siguientes ejemplos ilustran la utilización de comodines:

- | | |
|--------------------------|---|
| <code>ds/ds1-3/*</code> | Todos los canales en ds1 número 3 de la pasarela de medios en cuestión. |
| <code>ds/ds1-3/\$</code> | Cualquier canal en ds1 número 3 de la pasarela de medios en cuestión. |
| <code>ds/*</code> | Todos los puntos extremos de circuitos troncales de la pasarela de medios en cuestión. |
| <code>*</code> | Todos los puntos extremos (cualquiera que sea el tipo de punto extremo) de la pasarela de medios en cuestión. |

ds/ds1-3/[1-24] Canales 1 a 24 en ds1 número 3 de la pasarela de medios en cuestión.

De esta forma se definen los nombres canónicos para puntos extremos de una pasarela de circuitos troncales. Se espera que en una futura versión de esta Recomendación se soporte la utilización de alias, por ejemplo para el soporte de la reunión de múltiples circuitos troncales DS-0 para llamadas de vídeo, por ejemplo en la forma "s/ds1-1/H0-1".

7.1.2 Nombres de llamada

Las llamadas se identifican mediante identificadores únicos, independientes de las plataformas o agentes subyacentes. Los identificadores de llamadas son cadenas hexadecimales creadas por el MGC. El sistema DEBE soportar identificadores de llamada cuya longitud máxima sea 32 caracteres.

Los identificadores de llamada DEBEN, al menos, ser únicos dentro del conjunto de MGC que controlan las mismas pasarelas. Sin embargo, la coordinación de estos identificadores de llamada entre los MGC está fuera del ámbito de esta Recomendación. Cuando un MGC crea varias conexiones pertenecientes a una misma llamada, sea en la misma pasarela, sea en pasarelas diferentes, todas estas conexiones serán vinculadas a la misma llamada por medio del identificador de llamada. Este identificador lo pueden utilizar los procedimientos de contabilidad o gestión, los cuales están fuera del ámbito del MGCP.

7.1.3 Nombres de conexión

La pasarela crea identificadores cuando recibe instrucciones para crear una conexión. Estos identificadores identifican la conexión en el contexto de un punto extremo. Los identificadores de conexión se tratan en MGCP como cadenas hexadecimales. La pasarela DEBE asegurarse de que transcurre un periodo de espera adecuado, de al menos tres minutos, entre el momento en que termina una conexión en que se utilizó un identificador y el momento en que se utiliza el mismo identificador en una nueva conexión para el mismo punto extremo. El sistema tiene que soportar nombres de conexión cuya longitud máxima sea 32 caracteres.

7.1.4 Nombres de controladores de pasarela de medios y de otras entidades

El protocolo de control de pasarela de medios permite la implementación de MGC redundantes para una mayor fiabilidad de la red. Esto significa que no existe una vinculación fija entre entidades y plataformas de equipo físico o interfaces de red.

Al igual que los nombres de punto extremo, los nombres de MGC constan de dos partes. La porción local del nombre no tiene una estructura interna. Ejemplo de nombre de MGC:

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

Precauciones que garantizan la fiabilidad:

- Las pasarelas de circuitos troncales, los MGC y otras entidades similares se identifican por sus nombres de dominio, no por sus direcciones. Se pueden asociar varias direcciones a un nombre de dominio. Si una instrucción no puede enviarse a una de las direcciones de red, las implementaciones DEBEN reintentar la transmisión utilizando otra dirección.
- Las entidades pueden pasar a otra plataforma. Las asociaciones entre nombre lógico (nombre de dominio) y plataforma real se mantienen en el sistema de nombres de dominio (DNS, *domain name system*). Los MGC y las pasarelas DEBEN llevar la cuenta del tiempo de validez de cada registro, que se indica en el DNS. DEBEN interrogar al DNS para renovar la información, si el tiempo de validez ha expirado.

Además de la referenciación proporcionada por la utilización de nombres de dominio y del DNS, el concepto de "entidad notificada" es esencial a los efectos de la fiabilidad y del cambio-por-fallo en el MGCP. La "entidad notificada" para un punto extremo es el MGC que controla ese punto extremo en ese momento. En cualquier instante dado, un punto extremo tiene asociada una, y sólo

una, "entidad notificada" y, cuando necesita enviar una instrucción al MGC, DEBE enviarla a la "parte notificada" a que pertenece el punto extremo o los puntos extremos en ese momento. Tras el arranque inicial, se DEBE fijar la "entidad notificada" a un valor suministrado. La mayor parte de las instrucciones enviadas por el MGC permiten denominar explícitamente la "entidad notificada" mediante la utilización de un parámetro "NotifiedEntity". La "entidad notificada" permanecerá en el mismo estado hasta que se reciba el nuevo parámetro "NotifiedEntity" o hasta que el punto extremo rearranque. Si la "entidad notificada" para un punto extremo está vacía, o no se ha fijado explícitamente un valor³, la opción por defecto será la dirección fuente de la última conexión que ha tratado la instrucción o petición de notificación recibida para el punto extremo. Por tanto, la auditoría no modificará la "entidad notificada".

En la cláusula 7.4 se presenta una descripción más detallada de la fiabilidad y el cambio-por-fallo.

7.1.5 Mapas de dígitos

En MGCP, el MGC puede pedir a la pasarela que registre los dígitos marcados por el usuario. Este procedimiento suele utilizarse en líneas de acceso analógicas con pasarelas residenciales para la toma de los números marcados por el usuario; también puede utilizarse para interfaces CAS PBX. En lugar de enviar los dígitos al MGC uno a uno, a medida que se van detectando, el MGC puede proporcionar una gramática que indica cuántos dígitos deben acumularse antes de notificar al MGC. Esta gramática se conoce por *mapa de dígitos*.

Ninguno de los tipos de circuito troncal soportados por la actual versión de la Recomendación sobre TGCP necesita mapa de dígitos. Por eso no se han incluido mapas de dígitos en la presente Recomendación.

7.1.6 Eventos y señales

El concepto de eventos y señales es esencial en MGCP. Un MGC puede pedir que se le notifiquen ciertos eventos que se producen en un punto extremo, por ejemplo, eventos de descolgar. Un MGC puede también pedir que se apliquen a un punto extremo ciertas señales, por ejemplo, el tono de llamada.

Los eventos y señales se agrupan en lotes dentro de los cuales comparten el mismo espacio de nombre y que en lo sucesivo se designarán por nombres de evento. Un lote es una serie de eventos y señales soportados por un determinado tipo de punto extremo. Por ejemplo, un lote puede soportar ciertos eventos y señales para circuitos troncales de la PU-RDSI, y otro lote puede soportar otro grupo de eventos y señales para circuitos troncales MF. Puede haber uno o más lotes para un determinado tipo de punto extremo y cada tipo de punto extremo tiene un lote por defecto con el que está asociado.

Los nombres de eventos consisten en un nombre de lote y un código de evento; como cada lote define un espacio de nombre distinto, pueden utilizarse los mismos códigos de evento en lotes diferentes. Los nombres de lote y los códigos de eventos consisten en cadenas de letras, dígitos y guiones, que pueden estar escritas indiferentemente en mayúsculas o minúsculas, con la restricción de que un guión NO PODRÁ ser ni el primer carácter ni el último carácter de un nombre. Algunos códigos de evento pueden tener que ser parametrizados con datos adicionales, lo que se efectúa insertando los parámetros entre paréntesis. El nombre de lote se separa del código de evento por un carácter barra oblicua ("/"). El nombre de lote puede dejar de incluirse en el nombre de evento, en cuyo caso se supone que es el nombre de lote por defecto para el tipo de punto extremo en cuestión. Por ejemplo, para un circuito troncal PU-RDSI que tiene por defecto el lote PU-RDSI (nombre "IT"), se considera que los dos siguientes nombres de eventos son iguales:

IT/OC Operación completa en el lote PU-RDSI para un circuito troncal PU-RDSI.

³ Esto podría suceder como resultado de la especificación de un parámetro NotifiedEntity vacío.

oc Operación completa en el lote PU-RDSI (por defecto) para un circuito troncal PU-RDSI.

En el cuadro 1 se muestran los tipos de punto extremo de pasarela troncal y los lotes definidos para ellos en este documento. El MGC DEBE soportar todos los que figuran en el cuadro 1. El MG DEBE soportar "IT", "FXR" y "XRM", y DEBERÍA soportar los lotes "MO" y "MT".

Cuadro 1/J.171.1 – Lotes correspondientes a los tipos de punto extremo

Tipo de punto extremo	Lote	Nombre de lote	Lote por defecto	1 ^{ra} versión en la que se incluyó el requisito de soporte del lote
DS-0	Troncal PU-RDSI	IT	Sí	1,0
DS-0	OSS MF	MO	No	1,5
DS-0	Terminación MF	MT	No	1,5
DS-0	FAX	FXR	No	1,5
DS-0	Métrica VoIP	XRM	No	1,5

En IPCablecom pueden definirse y/o registrarse nombres de lotes y códigos de evento adicionales. Un cambio en los lotes definidos en esta Recomendación DEBE tener por consecuencia un cambio del nombre del lote, o un cambio del número de versión del perfil de TGCP, o ambos.

Cada lote DEBE tener una definición que DEBE definir el nombre del lote y la definición de cada evento perteneciente al lote. La definición de evento DEBE incluir el nombre preciso del evento, es decir, el código de evento, una definición en texto simple del evento y, cuando proceda, la definición precisa de las señales correspondientes, por ejemplo las frecuencias exactas de las señales de audio, como el tono de llamada de retorno o el tono de facsímil. Las definiciones de evento deben especificar si los eventos son persistentes (véase 7.3.1) y si hay estados de evento que se pueden conocer en una auditoría (véase 7.3.8.1). También hay que definir el tipo de señal (señales de activación/desactivación), señales sometidas a la expiración de un periodo de temporización, o señales breves, y las señales sometidas a la expiración de un periodo de temporización DEBEN tener un valor de periodo de temporización por defecto definido (véase 7.3.1).

Abstracción hecha de los lotes de IPCablecom, los implementadores PUEDEN adquirir experiencia definiendo lotes experimentales. El nombre de los lotes experimentales DEBE comenzar por los dos caracteres "x-" o "X-"; IPCablecom NO REGISTRARÁ nombres de lote que comiencen por estos dos caracteres. Una pasarela que reciba una instrucción referente a un lote no soportado DEBE retornar un error (código de error 518 – Lote no soportado).

Cada nombre de lote y cada código de evento soporta una notación mediante uso de comodín. El carácter comodín "*" (asterisco) puede utilizarse para hacer referencia a todos los lotes soportados por el punto extremo en cuestión y el código de evento "all" para hacer referencia a todos los eventos en el lote en cuestión. Por ejemplo:

IT/all hace referencia a todos los eventos en el lote circuitos troncales PU-RDSI para un circuito troncal PU-RDSI.

*/all para un circuito troncal PU-RDSI; hace referencia a todos los lotes y todos los eventos en esos lotes soportados por el punto extremo en cuestión.

En consecuencia, NO SE ASIGNARÁ el nombre de lote "*" y el código de evento "all" NO SE UTILIZARÁ en ningún lote.

Los eventos y señales se detectan y generan por defecto en puntos extremos; sin embargo, algunos eventos y señales también pueden detectarse y generarse en las conexiones a un punto extremo o sólo en esas conexiones y no en el punto extremo. Por ejemplo, se puede pedir a los puntos extremos que proporcionen un tono de llamada de retorno en una conexión. Para que un evento o

señal pueda ser detectado o generado en una conexión, la definición del evento/señal DEBE especificar explícitamente que el evento/señal puede detectarse o generarse en una conexión.

Cuando se deba aplicar una señal en una conexión, el nombre de la conexión se añade al nombre del evento utilizando un signo de "a comercial" (o signo de "arroba") (@) como delimitador, como por ejemplo en:

IT/rt@0A3F58

Si se llega a suspender la conexión mientras se detecta o aplica un evento o una señal, DEBE interrumpirse dicha detección de evento o generación de señal. Según el tipo de señal, el punto extremo DEBERÍA generar un fallo, es decir, si el tipo de señal es TO, se generará el evento "fallo de operación" puesto que la conexión asociada con la señal fue suprimida antes de la expiración de la señal. Las acciones asociadas con la notificación de fallo han de ser conformes a lo definido en la cláusula 7.3.1 para el procesamiento de peticiones de eventos (NotificationRequest).

El carácter comodín "*" (asterisco) puede utilizarse para designar "todas las conexiones" en el punto extremo o puntos extremos afectados. Cuando se utiliza este convenio, la pasarela DEBE generar o detectar el evento en todas las conexiones ligadas al punto extremo o puntos extremos. Un ejemplo del uso de este convenio es:

IT/ma@*

No obstante, cuando se observa efectivamente el evento, la pasarela DEBE incluir el nombre de la conexión específica en la que se presenta. El carácter comodín "\$" (signo de dólar) puede utilizarse para designar "la conexión actual". Este convenio NO SE UTILIZARÁ a menos que la petición de notificación de evento esté "encapsulada" dentro de una instrucción CreateConnection o ModifyConnection. Cuando se usa este convenio, la pasarela DEBERÁ generar o detectar el evento en la conexión que se está creando o modificando en ese momento. Un ejemplo del uso de este convenio es:

IT/rt@\$

Cuando se procesa una instrucción con el comodín "conexión actual", la pasarela DEBE ampliar el carácter comodín "\$" al valor de la conexión actual. Si una instrucción subsiguiente hace referencia explícita (por ejemplo, por auditoría) o implícita (por ejemplo, por persistencia) a uno de tales eventos, la pasarela DEBE utilizar el valor ampliado. En otras palabras, el comodín "conexión actual" se amplía una vez, al inicio del procesamiento de la instrucción en el que fue incluido explícitamente.

Se puede emplear el id de conexión, o un comodín en su lugar, junto con los convenios "todos los lotes" y " todos los eventos". Por ejemplo, se puede utilizar la notación:

/all@

para designar todos los eventos en todas las conexiones para el o los puntos extremos afectados. Sin embargo, se desaconseja decididamente la utilización de los comodines "todos los lotes" y " todos los eventos". Los agentes de llamada deben poder funcionar en un entorno en el que algunos puntos extremos no soporten todos los lotes. Si un punto extremo recibe una instrucción que hace referencia a un lote que no soporta, responderá con un código de error 518 (Lote no soportado o desconocido). Al recibir esta respuesta de error, el agente de llamada podría ensayar de nuevo la instrucción sin el parámetro del lote, pero si la instrucción original tenía parámetros para varios lotes, no habría forma de saber cuál o cuáles lotes excluir. El agente de llamada también puede emplear la instrucción AuditEndpoint para determinar cuál es el conjunto de lotes soportado por un punto extremo.

7.2 Utilización de SDP

El MGC utiliza el MGCP para proporcionar a las pasarelas la descripción de parámetros de conexión tales como direcciones IP, puerto UDP y perfiles RTP. Salvo que se indique o implique otra cosa en esta Recomendación, las descripciones SDP DEBEN seguir los convenios indicados en el protocolo de descripción de sesión (SDP, *session description protocol*), que actualmente constituye una norma IETF propuesta, documentada en IETF RFC 2327. Además, los MGC y las pasarelas de medios DEBEN hacer caso omiso de todo parámetro, atributo o campo SDP que no sea aceptado por el agente de llamada o la pasarela.

El SDP permite la descripción de conferencias multimedia. El perfil de TGCP sólo soportará el establecimiento de conexiones de audio que utilicen el tipo de medios "audio".

El SDP permite la descripción de fax en tiempo real que utilice el tipo de media "imagen". El perfil de TGCP soportará el establecimiento de conexiones de fax que utilicen el tipo de medios "imagen".

7.3 Funciones de control de pasarela

Esta cláusula describe las instrucciones del MGCP en forma de una llamada de procedimiento a distancia (RPC, *remote procedure call*) como una interfaz de programación de aplicación (API), a la que se hará referencia como la interfaz de control de pasarela de medios (MGCI). Para cada instrucción MGCP se define una función MGCI que toma y retorna los mismos parámetros que la correspondiente instrucción MGCP. Las funciones mostradas en esta cláusula proporcionan una descripción de alto nivel del funcionamiento del MGCP y describen un ejemplo de una API del tipo RPC que PUEDE utilizarse para una implementación de MGCP. Aunque la API de MGCI es simplemente un ejemplo, el comportamiento semántico definido por MGCI es parte integrante de esta Recomendación y todas las implementaciones DEBEN ser conformes con la semántica especificada para MGCI. Los mensajes MGCP efectivamente intercambiados, incluidos los formatos y codificaciones utilizados, se definen en la cláusula 8. Las pasarelas de circuitos troncales y los agentes de llamada DEBEN implementarlos exactamente como están especificados.

El servicio MGCI comprende instrucciones de tratamiento de conexiones y de tratamiento de puntos extremos. A continuación se presenta una visión de conjunto de las instrucciones:

- El MGC puede enviar una instrucción NotificationRequest a una pasarela ordenando que vigile la aparición de determinados eventos tales como tomas, o tonos de facsímil, en un determinado punto extremo.
- La pasarela emite entonces la instrucción Notify para informar al MGC cuándo los eventos solicitados se producen en el punto extremo especificado.
- El MGC puede utilizar la instrucción CreateConnection para crear una conexión que termina en un punto extremo dentro de la pasarela.
- El MGC puede utilizar la instrucción ModifyConnection para modificar los parámetros de conexión anteriormente establecida.
- El MGC puede utilizar la instrucción DeleteConnection para suprimir una conexión existente. En algunas circunstancias, la instrucción DeleteConnection puede ser también utilizada por una pasarela para indicar que ya no se puede sostener una conexión.
- El MGC puede utilizar las instrucciones AuditEndpoint y AuditConnection para conocer el estado de un "punto extremo" y de las conexiones que puedan estar asociadas a él. Generalmente, es conveniente que la gestión de red vaya más allá de las capacidades proporcionadas por estas instrucciones, por ejemplo, que proporcione información sobre el status de la pasarela de circuitos troncales y de cada uno de los circuitos troncales. Se espera que esas capacidades sean soportadas mediante el empleo del protocolo de gestión de red simple (SNMP, *simple network management protocol*) y la definición de una MIB, lo que está fuera del ámbito de esta Recomendación.

- La pasarela puede utilizar la instrucción `RestartInProgress` para notificar al MGC que el punto extremo, o un grupo de puntos extremos gestionados por la pasarela, son retirados del servicio o que se restablece el servicio.

Estos servicios permiten a un controlador (normalmente el MGC) ordenar a una pasarela que cree conexiones que terminen en un punto extremo ligado a la pasarela y que le informe los eventos que se producen en el punto extremo. Actualmente, un punto extremo de pasarela de circuitos troncales está limitado a un circuito troncal concreto de la pasarela de circuitos troncales.

Las conexiones se agrupan en "llamadas". Varias conexiones, que pueden o no pertenecer a una misma llamada, pueden terminar en un mismo punto extremo. Cada conexión es calificada por un parámetro "modo", que puede fijarse a "sólo envío" (`sendonly`), "sólo recepción" (`recvonly`), "envío/recepción" (`sendrecv`), "inactivo" (`inactive`), "conexión en bucle" (`loopback`), "prueba de continuidad" (`conttest`), "conexión en bucle de red" (`netwloop`) o "prueba de continuidad de red" (`netwtest`). El parámetro "modo" determina si pueden enviarse/recibirse paquetes de medios por la conexión; sin embargo, no hay ninguna modificación del RTCP.

Las señales de audio recibidas del punto extremo DEBEN ser enviadas a través de cualquier conexión para ese punto extremo cuyo modo sea "sólo lectura" o "envío/recepción", salvo si el punto extremo tiene una conexión en modo "conexión en bucle" o "prueba de continuidad de red". Sin embargo, el audio generado al aplicar una señal a una conexión DEBE enviarse para todos los modos en la conexión, excepto el "conexión en bucle de red".

El tratamiento de las señales de audio recibidas por estas conexiones viene también determinado por los parámetros de modo:

- Las señales de audio recibidas en paquetes de datos a través de conexiones en los modos "inactivo", "conexión en bucle" o "prueba de continuidad" se DEBEN descartar.
- Las señales de audio recibidas en paquetes de datos a través de conexiones en los modos "sólo recepción", o "envío/ recepción" DEBEN mezclarse y después enviarse al punto extremo⁴, a menos que éste tenga otra conexión en modo "en bucle de red" o "prueba de continuidad".
- Las señales de audio originadas en el punto extremo se transmiten a través de todas las conexiones que funcionen en los modos "sólo envío", o "envío/recepción", salvo si el punto extremo tiene otra conexión en modo "en bucle de red" o "prueba de continuidad".

Para detectar los eventos en una conexión, ésta DEBE estar por defecto en uno de los siguientes modos: "sólo recepción", "envío/ recepción", "en bucle de red" o "prueba de continuidad de red". La detección del evento se aplica solamente al audio entrante. Entonces normalmente las conexiones que estén en modo "sólo envío", "inactivo", "en bucle", o "prueba de continuidad" no detectarán ningún evento, aunque no se considera error una petición para que detecten eventos.

Los modos "conexión en bucle" y "prueba de continuidad" se utilizan durante operaciones de mantenimiento y prueba de continuidad. Un punto extremo puede tener más de una conexión en el modo "conexión en bucle" o "prueba de continuidad". Mientras haya una conexión en dicho modo, y ninguna otra en el punto extremo esté en otro modo de mantenimiento o prueba, la operación de mantenimiento o prueba DEBE proseguir inalterada. La prueba de continuidad (COT, *continuity test*) tiene dos variantes, una especificada por la UIT y otra utilizada en varias redes nacionales. En el primer caso, se trata de una prueba en bucle. El conmutador de origen envía un tono (el tono de ida) por el circuito portador y espera que el conmutador de terminación conecte en bucle el circuito. Si el conmutador de origen comprueba que se ha retornado el mismo tono (el tono de retorno), la prueba de continuidad ha tenido éxito. Si no, la prueba de continuidad ha fracasado. En el segundo caso, los tonos de ida y retorno son diferentes. El conmutador de origen envía un determinado tono

⁴ No obstante, actualmente no se requiere que los puntos extremos TGCP soporten la mezcla.

de ida. El de terminación lo detecta y envía otro tono diferente. Cuando el conmutador de origen detecta el tono de retorno, la COT ha tenido éxito. Si transcurrido un cierto tiempo no ha detectado ningún tono de retorno, la prueba de continuidad ha fracasado.

Si el modo es "conexión en bucle", la pasarela TIENE QUE devolver la señal entrante, procedente del punto extremo, en retorno al mismo punto extremo. Éste es el procedimiento general para la prueba de continuidad de circuitos troncales, con arreglo a las especificaciones de la UIT. Si el modo es "prueba de continuidad", se informa a la pasarela que el otro extremo del circuito ha iniciado un procedimiento de prueba de continuidad conforme a los procedimientos especificados para varias redes nacionales. La pasarela pondrá el circuito en el modo de transpondedor requerido para las pruebas de continuidad con dos tonos.

Además, cuando una conexión para un punto extremo está en el modo "conexión en bucle" o "prueba de continuidad":

- las señales de audio recibidas por cualquier conexión para el punto extremo *no* DEBEN enviarse al punto extremo;
- las señales de audio recibidas en el punto extremo *no* DEBEN enviarse a ninguna conexión para el punto extremo.

Si el modo es "conexión en bucle de red", las señales de audio recibidas de la conexión serán devueltas en eco por la misma conexión. El modo "conexión en bucle de red" DEBERÍA funcionar simplemente como un reflector de paquetes RTP. No se transmitirán medios al punto extremo.

El modo "prueba de continuidad de red" se utiliza para comprobar la continuidad a través de la red IP. Se envía a los puntos extremos una señal específica del tipo de punto extremo, través de la red IP, y se supone que el punto extremo devolverá en eco la señal a través de la red IP después de pasarla por el equipo interno de la pasarela, para verificar el funcionamiento correcto. Antes de ser devuelta, la señal DEBE pasar por una decodificación y recodificación internas. Para puntos extremos DS-0, será una señal de audio que NO SE PASARÁ a un circuito conectado al punto extremo, cualquiera que sea el estado de toma de ese circuito en ese momento.

Las conexiones nuevas y las conexiones existentes del punto extremo NO SERÁN afectadas por conexiones que se encuentren en modo "conexión en bucle de red" o "prueba de continuidad de red". No obstante, las restricciones de recursos locales pueden limitar el número de nuevas conexiones que podrán efectuarse.

Véanse los ejemplos de interacciones de modos en el apéndice I.

7.3.1 NotificationRequest

La instrucción NotificationRequest (petición de notificación) se utiliza para pedir a la pasarela que envíe una notificación cuando se produzcan ciertos eventos en un punto extremo. Por ejemplo, se puede pedir que se envíe una notificación cuando en un punto extremo se detectan tonos relacionados con la comunicación de facsímil. La entidad que recibe esta notificación, por lo general el MGC, podrá entonces considerar que se debe utilizar un tipo diferente de codificación en las conexiones vinculadas a este punto extremo y dar a la pasarela las instrucciones correspondientes⁵.

⁵ La nueva instrucción sería una ModifyConnection.

```

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

```

EndpointId es el identificador del punto extremo o de los puntos extremos en la pasarela en la que se ejecuta NotificationRequest. El EndpointId DEBE seguir las reglas para los nombres de punto extremo especificadas en 7.1.1. NO SE UTILIZARÁ el comodín "any of". Una pasarela que reciba una NotificationRequest con el comodín "any of" DEBE retornar un error (que DEBERÍA tener el código de error 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido). El sistema TIENE QUE soportar el comodín "all of" para las NotificationRequests cuando RequestedEvents, SignalsRequest y DetectEvents estén vacías o se hayan omitido. Para simplificar, algunas pasarelas pueden decidir no soportar el comodín "all-of" para la NotificationRequests cuando uno o varios de estos parámetros no estén vacíos o no hayan sido omitidos. Dichas pasarelas responderán con el código de error 503 si reciben una NotificationRequest con comodín "all-of" que no puedan procesar por esta razón.

NotifiedEntity es un parámetro opcional que especifica una nueva "entidad notificada" para el punto extremo. Cuando se utiliza, se DEBE especificar todo el nombre del MGC, lo cual incluye tanto el nombre local como el de dominio, aun si se emplea una dirección IP entre paréntesis para este último. Véanse 7.1.1 y 7.1.4 para más información. Si, a pesar de ello, sólo se suministra el nombre de dominio, el MG DEBERÍA utilizarlo como ID del agente de llamada.

RequestIdentifier se utiliza para correlacionar esta petición con la notificación que pueda ocasionar. Se repetirá en la correspondiente instrucción Notify.

SignalRequests es un parámetro que contiene el conjunto de señales que ha de aplicar la pasarela. A menos que se especifique otra cosa, las señales se aplican al punto extremo; sin embargo, algunas señales pueden aplicarse a una conexión. Son ejemplos de señales⁶:

- Prueba de continuidad;
- Establecer comunicación MF sistema de soporte de operaciones (OSS, *operations support system*).

Atendiendo a su comportamiento, se distinguen tres tipos diferentes de señales:

- On/off (OO) – Estas señales producen efecto desde que son activadas (conmutadas a On) hasta que son desactivadas (conmutadas a Off), lo que puede suceder como resultado de una nueva SignalRequests con la señal desactivada (véase más adelante). Las señales de tipo OO son, por definición, ídems potentes, por lo que múltiples peticiones de activación (o de desactivación) de una determinada señal OO son perfectamente válidas y NO PROVOCARÁN ningún error. Una vez activadas, NO SERÁN desactivadas hasta que el MGC lo ordene explícitamente o el punto extremo re arranque.
- Time-out (TO) – Estas señales producen efecto desde que son activadas (conmutadas a On) hasta que, o bien son suprimidas (por la aparición de un evento o por el hecho de no ser incluida en una subsiguiente lista de señales (que puede ser una lista vacía)), o bien transcurre un determinado periodo de temporización. Una señal que desaparece por el transcurso de un periodo de temporización genera un evento "operación finalizada" (para una definición más elaborada de este evento, véase A.1). Una señal TO podría ser "efectuar llamada MF" con un periodo de temporización de 16 segundos. Si se produce un evento

⁶ En el anexo A se presenta una lista completa de señales.

antes de la expiración del periodo de 16 segundos, la señal será detenida por defecto⁷. Si la señal no es detenida, expirará el periodo de temporización para la señal, se detendrá la señal y se generará un evento "operación finalizada", que el MGC podrá o no haber solicitado que se le notifique. Si el MGC ha solicitado que se le notifique el evento "operación finalizada", el evento "operación finalizada" enviado al MGC incluirá el nombre o nombres de la señal o señales que desaparecieron por temporización⁸. La señal o señales generadas en una conexión incluirán el nombre de esa conexión. Las señales afectadas por un periodo de temporización tienen un valor de temporización por defecto, que puede ser modificado en la configuración. Asimismo, el periodo de temporización puede proporcionarse como un parámetro de la señal. El valor cero indica que el periodo de temporización es infinito. Una señal TO que fracasa después de haber sido iniciada, pero antes de generar un evento "operación finalizada", generará un evento "fallo de operación", que incluirá el nombre o nombres de la señal o señales a que se aplica la temporización⁸.

- Brief (BR) – La duración de estas señales es tan corta que terminan por sí mismas. Si se produce un evento de detención de la señal, o se aplica una nueva instrucción SignalRequests, una señal BR que se encuentre en curso en ese momento no se detendrá. Sin embargo, se suprimirá toda señal BR pendiente que todavía no haya comenzado a aplicarse.

Las señales se aplican por defecto a puntos extremos. Si una señal aplicada a un punto extremo tiene por consecuencia la generación de un tren de medios (de audio, vídeo, etc.), el tren de medios NO SE REENVIARÁ por ninguna conexión asociada con ese punto extremo, cualquiera que sea el modo de conexión. Por ejemplo, si se aplica un tono a un punto extremo que participa en una comunicación activa, sólo la parte que utiliza el punto extremo en cuestión oír el tono. No obstante, cada señal puede definir comportamientos diferentes.

Cuando se aplica una señal a una conexión que ha recibido un RemoteConnectionDescriptor (véase 7.3.3), el tren de medios generado por esa señal se reenviará por la conexión *cualquiera que sea* el modo de la conexión. Si no se ha recibido un RemoteConnectionDescriptor, la pasarela retornará un error (código de error 527 – Falta RemoteConnectionDescriptor).

Cuando se suministra una lista de señales (que puede estar vacía), esta lista sustituye íntegramente a la actual lista de señales sometidas a temporización que están activas. Las señales sometidas a temporización que estén activas y no hayan sido incluidas en la nueva lista DEBEN ser detenidas y la nueva o nuevas señales proporcionadas serán ahora activadas. Las señales sometidas a temporización que estén activas en ese momento y que se hayan incluido en la nueva lista de señales DEBEN permanecer activas sin interrupción, y el temporizador para esas señales no será afectado. Por tanto, actualmente no hay forma de rearrancar el temporizador para una señal sometida a temporización que está activa en un momento dado sin antes desactivar la señal. Si la señal de temporización tiene parámetros, el conjunto inicial de parámetros DEBE seguir vigente cualquiera que sean los valores que se suministren posteriormente. Una determinada señal NO APARECERÁ más de una vez en una SignalRequests. La omisión del parámetro SignalRequests se interpreta como una lista vacía SignalRequests.

Las señales actualmente definidas se indican en el anexo A.

RequestedEvents es una lista de eventos que deben ser detectados por la pasarela en el punto extremo. A menos que se especifique otra cosa, los eventos se detectan en el punto extremo; sin embargo, algunos eventos pueden detectarse en la conexión. Son ejemplos de eventos:

- toma;

⁷ La acción "mantener señal(es) activa(s)" puede prevalecer sobre este comportamiento.

⁸ Si se pasaron parámetros a la señal, no se informarán los parámetros.

- tonos de facsímil;
- operación completa;
- llamada MF entrante.

Los eventos actualmente definidos se indican en el anexo A.

A cada evento está asociada una o más **acciones** que una pasarela tiene que ejecutar cuando se produce un determinado evento. Son posibles acciones:

- notificar el evento inmediatamente, junto con la lista acumulativa de eventos observados;
- acumular el evento;
- no tener en cuenta el evento;
- mantener las señales activas;
- NotificationRequest insertada;
- ModifyConnection insertada.

El punto extremo detectará dos clases de eventos solicitados: persistentes y no persistentes.

Los eventos persistentes siempre se detectan en un punto extremo. Si se produce un evento persistente que no esté incluido en la lista de RequestedEvents, el evento será detectado de todas formas, y procesado al igual que todos los demás, como si dicho evento persistente se hubiera solicitado con una acción Notify⁹. En consecuencia, desde un punto de vista informal, puede considerarse que los eventos persistentes siempre han sido incluidos en la lista de RequestedEvents con una acción para Notify, si bien no se realizará ninguna detección de doble toma, ni otra acción similar¹⁰. Los eventos persistentes se identifican como tales por su definición (véase el anexo A).

Eventos no persistentes son aquellos que deben ser incluidos explícitamente en la lista de RequestedEvents. La lista (que puede estar vacía) de eventos solicitados sustituye íntegramente a una anterior lista de eventos solicitados. Además de los eventos persistentes, el punto extremo sólo detectará los eventos que estén especificados en la lista de eventos solicitados. Si un evento persistente está incluido en la lista de RequestedEvents, la acción especificada sustituirá a la acción por defecto asociada con el evento durante el tiempo en que esté vigente la lista de RequestedEvents; una vez transcurrido este periodo de tiempo, se restablece la acción por defecto. Un evento dado NO APARECERÁ más de una vez en una lista de RequestedEvents. La omisión del parámetro RequestedEvents se interpreta como una lista vacía RequestedEvents. Se pueden especificar varias acciones para un evento, pero una acción determinada no puede aparecer más de una vez para un evento dado. La matriz del cuadro 2 especifica las combinaciones válidas de acciones:

⁹ Por tanto, el RequestIdentifier será el RequestIdentifier de la NotificationRequest en curso.

¹⁰ Normalmente, cuando, por ejemplo, se solicita la detección de un evento de descolgar, la petición sólo puede tener éxito si el teléfono no está descolgado en ese momento.

Cuadro 2/J.171.1 – Acciones asociadas con los eventos

Evento	Notificar	Acumular	No tener en cuenta	Mantener señales activas	Notification Request insertada	ModifyConnection insertada
Notificar	–	–	–	√	–	√
Acumular	–	–	–	√	√	√
No tener en cuenta	–	–	–	√	–	√
Mantener señales activas	√	√	√	–	√	√
NotificationRequest insertada	–	√	–	√	–	√
ModifyConnection insertada	√	√	√	√	√	–

Si un cliente recibe una petición con una acción no válida o una combinación no válida de acciones, devolverá al MGC un error (código de error 523 – Combinación desconocida o no válida de acciones).

Cuando se especifican múltiples acciones, por ejemplo, "mantener señales activas" y "notificar", se supone que las distintas acciones se realizan simultáneamente.

El MGC puede enviar a la pasarela una NotificationRequest con una lista de RequestedEvents vacía. No obstante, los eventos persistentes serán detectados y notificados.

Cuando se detecta un estímulo que activa varios eventos solicitados (por ejemplo, un tono de fax es el estímulo tanto para FXR/gwfax(start) como para L/ft), la pasarela generará solamente uno de los eventos (a saber, el prioritario de los eventos solicitados activados) basándose en las siguientes reglas:

- 1) La prioridad de los eventos incluidos en una lista RequestedEvents es decreciente de izquierda a derecha (el evento más a la izquierda es el de mayor prioridad).
- 2) Los eventos persistentes que no están incluidos en una lista RequestedEvents tienen menos prioridad que los eventos (persistentes o no) que están incluidos en una lista RequestedEvents. No existe un orden de preferencia bien definido entre los eventos persistentes que no están incluidos en una lista RequestedEvents.

Las señales aplicadas por las SignalRequests se sincronizan con la toma de eventos especificados o implícitos en el parámetro RequestedEvents, salvo si sobre ellas prevalece la acción "mantener señales activas". Según la definición formal, la generación de todas las señales "sometidas a temporización" DEBE detenerse tan pronto como se detecte uno de los eventos solicitados, a menos que la acción "mantener señales activas" esté asociada al evento especificado.

Si se desea que la señal o señales sometidas a temporización continúen cuando se produce un evento buscado, puede utilizarse la acción "mantener señales activas". Esta acción tiene por efecto mantener activas todas las señales sometidas a temporización que estén activas en ese momento, es decir, anular la interrupción por defecto de esas señales cuando ocurre el evento.

Si se desea que las señales comiencen cuando se produce un evento buscado, puede utilizarse la acción "NotificationRequest insertada". La NotificationRequest insertada puede incluir una nueva lista de RequestedEvents y nuevas SignalRequests. Sin embargo, la "NotificationRequest insertada" no puede incluir otra "NotificationRequest insertada". Cuando se activa la "NotificationRequest insertada", la lista de eventos observados y la memoria intermedia para cuarentena no serán afectadas (véase 7.4.3.1).

La acción NotificationRequest insertada permite al MGC indicar un comportamiento a la pasarela inmediatamente después de la detección del evento asociado. Toda SignalRequests especificada en la NotificationRequest insertada comenzará inmediatamente. Hay que proceder con suma cautela para evitar discrepancias entre el MGC y la pasarela. Sin embargo, no deberían producirse discrepancias de larga duración, pues las nuevas SignalRequests sustituyen totalmente a la antigua lista de señales sometidas a temporización activas y las señales de tipo BR siempre se detienen por sí mismas. Se aconseja limitar el número de las señales de tipo On/Off. Se considera buena práctica que el MGC active ocasionalmente todas las señales de tipo On/Off que deberían estar activadas y desactive todas las que deberían estar desactivadas.

Si se desea que el modo de conexión cambie cuando se produce un evento buscado, puede utilizarse la acción "ModifyConnection insertada". La acción ModifyConnection insertada puede incluir una lista de cambios de modo de conexión, cada uno de los cuales consiste en el cambio de modo y el identificador de la conexión afectada. El comodín "\$" puede utilizarse para designar "la conexión actual"; sin embargo, esta notación NO SE UTILIZARÁ fuera de una instrucción de tratamiento de conexión. El comodín hace referencia a la conexión en cuestión para la instrucción de tratamiento de conexión.

La acción ModifyConnection insertada permite al MGC ordenar al punto extremo que cambie el modo de conexión de una o más conexiones inmediatamente después de la detección del evento asociado. Cada cambio de modo de conexión funciona de manera similar a una instrucción ModifyConnection correspondiente. Cuando se suministra una lista de cambios de modo de conexión, los distintos cambios SE HARÁN sucesivamente de izquierda a derecha. Cuando todos los cambios de modo de conexión han finalizado, se genera un evento "operación finalizada" parametrizado con el nombre de la acción completada (véase el anexo A). Si fracasa cualquiera de los modos de conexión, se genera un evento "fallo de operación" parametrizado con el nombre de la acción fracasada y del cambio de modo de conexión (véase el anexo A). Los restantes cambios del modo de conexión NO SE INTENTARÁN, y se mantendrán los anteriores cambios de modo de conexión realizados satisfactoriamente.

Por último, la acción Ignore puede utilizarse para no tener en cuenta un evento, por ejemplo con el fin de evitar que se notifique un evento persistente. No obstante, la sincronización entre el evento y la señal activa se producirá por defecto, de todas formas.

NOTA – En la cláusula 7.4.3.1 hay más información sobre la semántica de la detección e informes de eventos. Se invita al lector a examinarla detenidamente.

La definición específica de acciones que se solicitan mediante estas SignalRequests está fuera del ámbito del cuerpo principal de la Recomendación sobre TGCP. Esta definición puede variar de un lugar a otro y, por consiguiente, de una pasarela a otra. En consecuencia, las definiciones se proporcionan en lotes de eventos, los cuales pueden proporcionarse fuera del cuerpo principal de la Recomendación. En el anexo A se presenta una lista inicial de lotes de eventos.

Las instrucciones RequestedEvents y SignalRequests generalmente hacen referencia a los mismos eventos. En un caso se pide a la pasarela que detecte la aparición del evento y en el otro caso que genere el evento. Esta regla tiene pocas excepciones, entre las cuales debe señalarse especialmente los tonos de facsímil y de módem, que pueden ser detectados, pero no señalizados. Sin embargo, no debe esperarse que todos los puntos extremos detecten todos los eventos. Los eventos y señales concretos que un determinado punto extremo puede detectar o generar vienen determinados por la lista de lotes de eventos soportados por el punto extremo en cuestión. Cada lote especifica una lista de eventos y señales que pueden ser detectados o aplicados. Una pasarela a la que se pide que detecte o aplique un evento perteneciente a un lote que no está soportado por el punto extremo especificado DEBE retornar un error (código de error 512 ó 513 – No equipado para detectar evento o generar señal). Cuando el nombre de evento no está calificado por un nombre de lote, se supone que es el nombre de lote por defecto para el punto extremo. Si el nombre de evento no está

registrado en este lote por defecto, la pasarela devolverá un error (código 522 – No hay tal evento o señal).

El MGC puede enviar una NotificationRequest con una lista vacía de señales solicitadas. Esto tiene por efecto detener todas las señales sometidas a temporización que estén activas. Puede hacer esto, por ejemplo, cuando se debe detener la generación de un tono, por ejemplo, el tono de llamada de retorno.

QuarantineHandling es un parámetro facultativo que especifica opciones para el tratamiento de los eventos almacenados en la memoria intermedia de cuarentena (véase 7.4.3.1), es decir, los eventos detectados por la pasarela antes de la llegada de esta instrucción NotificationRequest, que aún no han sido notificados al MGC. El parámetro puede indicar distintas opciones de procesamiento:

- procesar (valor por defecto) o descartar los eventos puestos en cuarentena;
- si se espera que la pasarela genere a lo sumo una notificación (lockstep, valor por defecto), o varias notificaciones (loop), como respuesta a esta petición.

Si no se ha incluido este parámetro, HAY QUE procesar los eventos puestos en cuarentena. El sistema tiene que soportar los modos "lockstep" (por defecto) y "loop". Un punto extremo que recibe una NotificationRequest con un valor de parámetro QuarantineHandling no soportado DEBERÍA responder con un código de error 508 (QuarantineHandling no soportado).

Obsérvese que el parámetro quarantine-handling también rige el tratamiento de eventos que fueron detectados y procesados, pero que no han sido notificados aún cuando se recibe la instrucción.

DetectEvents es un parámetro facultativo que especifica una lista mínima de eventos que la pasarela deberá detectar en los estados "notification" y "lockstep". La lista persiste hasta que se notifique un nuevo valor. Explicación más completa de este parámetro en 7.4.3.1.

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y consiste en un número entero (véase 7.5), facultativamente seguido de un comentario.

7.3.2 Notificaciones

La pasarela envía notificaciones al observar eventos (depende de la instrucción Notify):

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

EndpointId es el nombre del punto extremo en la pasarela que emite la instrucción Notify, como se define en 7.1.1. El identificador DEBE ser un nombre de punto extremo completamente calificado, que incluye el nombre de dominio de la pasarela. En la parte local del nombre NO SE UTILIZARÁN comodines. Si un MGC recibe una Notificación con comodines, devolverá un código de error (que DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

NotifiedEntity es un parámetro facultativo que identifica la entidad a la que se envía la notificación. Este parámetro es idéntico al parámetro NotifiedEntity de la NotificationRequest que ocasionó esta notificación. Nótese que la MG PUEDE incluir solamente el nombre de dominio de esta entidad notificada si sólo se recibió ese nombre en la instrucción NotificationRequest originaria. En tal caso, el MGC DEBERÍA aceptar el valor. El parámetro está ausente si no había tal parámetro en la petición que ocasionó la notificación. Cualquiera que sea el valor del parámetro NotifiedEntity, la notificación DEBE enviarse a la "entidad notificada" actual para el punto extremo.

RequestIdentifier es un parámetro que repite el parámetro RequestIdentifier de la NotificationRequest que ocasionó esta notificación. Se utiliza para relacionar esta notificación con la petición de notificación que la ocasionó. A estos efectos, los eventos persistentes se consideran como si hubieran sido incluidos en la última NotificationRequest (incluye la petición de notificación insertada en primitivas de tratamiento de la conexión). Cuando no se ha recibido ninguna NotificationRequest se utilizará el RequestIdentifier cero ("0").

ObservedEvents es una lista de eventos que la pasarela ha detectado y acumulado por la acción "accumulate" o "notify". Una sola notificación puede informar una lista de eventos, que se informarán en el mismo orden en que se detectaron. La lista sólo puede contener eventos persistentes y eventos que fueron solicitados en el parámetro RequestedEvents de la NotificationRequest que ocasionó la notificación. Los eventos que fueron detectados en una conexión incluirán el nombre de esa conexión. La lista contendrá los eventos que fueron acumulados (pero no notificados) y el evento final que ocasionó la notificación.

ReturnCode es un parámetro que devuelve el MGC. Indica el resultado de la instrucción y consiste en un número entero (véase 7.5) seguido facultativamente de un comentario.

7.3.3 CreateConnection

Esta instrucción se utiliza para crear una conexión.

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

Esta función se utiliza cuando se establece una conexión entre dos puntos extremos. Una conexión se define por sus atributos y por los puntos extremos que asocia. Como mínimo, una pasarela troncal soportará una conexión por punto extremo. Los parámetros de entrada de CreateConnection proporcionan los datos necesarios para determinar la conexión desde el punto de vista de uno de los dos puntos extremos.

CallId es un parámetro que identifica la llamada (o sesión) a que pertenece una determinada conexión. Este parámetro es único, al menos dentro del grupo de MGC que controla las mismas pasarelas; las conexiones pertenecientes a una misma llamada comparten el mismo identificador. El identificador de llamada puede utilizarse para identificar llamadas con fines de informe y contabilidad.

EndpointId es el identificador del punto extremo de la pasarela en que se ejecuta la instrucción CreateConnection. El EndpointId puede especificarse completamente asignando al parámetro EndpointId un valor sin comodín en la invocación de la función o se puede especificar de una forma menos estricta utilizando el convenio de comodín "anyone". Si el punto extremo está subespecificado, el identificador de punto extremo será asignado por la pasarela y su valor completo SE DEVOLVERÁ en el parámetro SpecificEndPointId de la respuesta solamente si la instrucción tuvo éxito. En este caso, el punto extremo designado ESTARÁ en servicio y no tendrá aún ninguna conexión. NO SE UTILIZARÁ el comodín "all". Si una pasarela recibe un CreateConnection con el

comodín "all", DEBE responder con un código de error (que DEBERÍA ser 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

NotifiedEntity es un parámetro facultativo que especifica una nueva "entidad notificada" para el punto extremo.

LocalConnectionOptions es una estructura que describe las características de la conexión de datos de medios desde el punto de vista de la pasarela que ejecuta la instrucción CreateConnection. Imparte al punto extremo instrucciones sobre el envío y la recepción de características de la conexión de medios. La estructura LocalConnectionOptions comprende los siguientes campos básicos:

- **Método de codificación:** Hay que especificar, al menos con un valor, una lista de nombres literales para el algoritmo de compresión (método de codificación/decodificación) utilizado para enviar y recibir medios en la conexión. Los elementos de la lista están ordenados según su nivel de preferencia. El punto extremo elegirá al menos uno de los códecs y el códec se DEBERÍA elegir de acuerdo con el nivel de preferencia indicado. Si el punto extremo recibe por la conexión medios codificados por un método de codificación diferente, PUEDE descartarlos. El punto extremo también indicará cuál de los algoritmos de compresión restantes desea soportar como algoritmos alternativos. Para más detalles, véase 8.4.1. En la Rec. UIT-T J.161 se enumeran los métodos de codificación admisibles. Hay que utilizar los nombres literales definidos en 7.5 (cuadro 3)/J.161. No se DEBERÍAN tener en cuenta los algoritmos de compresión desconocidos que se reciban. Véase 7.7 para más detalles acerca del proceso de selección de códec.

NOTA – En "método de codificación" se incluyen las codificaciones de audio, imagen y vídeo.

- **Periodo de paquetización:** Se PUEDE especificar un solo periodo de paquetización, en milisegundos, con exactamente un valor decimal. Si se utiliza este especificador, se DEBE entonces emplear el mismo periodo de paquetización para todos los métodos de codificación permitidos por LocalConnectionOptions. Cuando no se especifica en LCO ningún campo de método de codificación, la MG NO PODRÁ escoger ningún método de codificación que tenga un periodo de paquetización diferente del especificado aquí. Si se quiere tener periodos de paquetización diferentes para diversas codificaciones, NO se DEBE utilizar este campo. Este valor atañe tanto a los medios que se envían como a los que se reciben. Obsérvese que la MG sólo ha de emplear el periodo de paquetización válido junto con el método de codificación asociado. En la especificación IPCablecom de Códecs Audio/Video (J.161) se indica una lista de periodos de paquetización aceptables. Este especificador NO SE PODRÁ incluir en la misma LCO que el campo periodo de paquetización múltiple (Multiple Packetization Period). Una MG devolverá un error (código 524 – Inconsistencia de LocalConnectionOptions) si recibe una LCO que incluya ambos campos, periodo de paquetización y periodo de paquetización múltiple.
- **Periodo de paquetización múltiple:** Se PUEDE especificar una lista de periodos de paquetización, en milisegundos, pero sólo si se incluye el campo método de codificación. De haber sido especificado, el periodo de paquetización múltiple, en milisegundos, DEBE contener un valor decimal o un guión por cada valor en el campo método de codificación incluido en LocalConnectionOptions. Esto sigue siendo válido aun si varios de los métodos de codificación tienen el mismo valor. El primer valor en la lista DEBE ser un número decimal. Cuando se utiliza un guión, el códec en cuestión UTILIZARÁ el mismo periodo de paquetización que algún otro valor de la lista que sí contiene un número decimal y NO PODRÁ utilizar más ancho de banda que ese otro valor. Se puede utilizar, por ejemplo, para códecs no vocales (por ejemplo, evento telefónico o ruido de confort) que utilicen el mismo periodo de paquetización que el códec vocal con el que se utilizan. Los valores sucesivos en la lista de periodos de paquetización DEBEN ir en el mismo orden que los métodos de codificación correspondientes. Los valores atañen tanto a los medios que se

envían como a los que se reciben. Obsérvese que la MG NO PODRÁ escoger un códec que tenga un periodo de paquetización diferente del especificado aquí. La MG sólo ha de emplear el periodo válido de paquetización junto con el método de codificación asociado. En la especificación IPCablecom de Códecs Audio/Vídeo (J.161) se indica una lista de periodos de paquetización aceptables. Este especificador NO SE PODRÁ incluir en la misma LCO que el campo periodo de paquetización (Packetization Period). Una MG DEVOLVERÁ un error (código 524 – Inconsistencia de LocalConnectionOptions) en los siguientes casos:

- Cuando recibe una LCO que incluya ambos campos, periodo de paquetización y periodo de paquetización múltiple.
- Cuando recibe una LCO en la que el número de códecs especificados en el campo método de codificación es diferente del número de elementos en el campo periodo de paquetización múltiple.
- **Compensación de eco:** Indica si habrá o no de utilizarse compensación de eco en el lado circuito troncal¹¹. El parámetro puede tener el valor "on" (cuando se solicita compensación de eco) u "off" (cuando no se solicita). Este parámetro es facultativo. Cuando se ha omitido el parámetro, la pasarela de circuitos troncales APLICARÁ compensación de eco. Luego, la pasarela de medios DEBERÍA habilitar o inhabilitar la compensación de eco conforme a la Rec. UIT-T V.8 cuando se detecte banda vocal. Para la rehabilitación de la compensación de eco véase la Rec. UIT-T G.168. Cuando termine la transmisión de datos de banda vocal HAY QUE restablecer lo que indique el valor actual del parámetro para la compensación de eco. Se RECOMIENDA que se determine la compensación de eco en la pasarela, no en el MGC.
- **Tipo de servicio:** Especifica la clase de servicio que se utilizará para el envío de medios por la conexión, codificando el valor del parámetro tipo de servicio (8 bits) en el encabezamiento IP como dos dígitos hexadecimales. El parámetro es facultativo. Cuando se omite este parámetro, se DEBE utilizar un valor por defecto de 0x00 (salvo si se ha configurado de otra manera). Cuando el parámetro está presente y es válido, el punto extremo ASIGNARÁ el valor proporcionado al parámetro DSCP en el encabezamiento IP (véase la RFC 2474 para más información acerca del DSCP). El valor del parámetro DEBE ser 0x00 o DEBE ser un múltiplo de cuatro entre 0x01 y 0xFF (se reservan los bits 6 y 7, bits ECN, y por ende debe ponerse a "00"). Un punto extremo DEVOLVERÁ un error (código 532 – Valor o valores no soportados en LocalConnectionOptions) cuando recibe un valor no válido. El "bit" más a la izquierda en el parámetro corresponde al más significativo en el encabezamiento IP.
- **Supresión de (los periodos de) silencio:** Las pasarelas telefónicas pueden detectar actividad vocal y evitar el envío de paquetes durante los periodos de silencio. No obstante, ciertos tipos de llamada (por ejemplo, llamadas con módem) requieren que no haya supresión de silencio. El parámetro puede tener el valor "on" (cuando se ha de suprimir el silencio) u "off" (cuando no). El valor es facultativo. Cuando se omite el parámetro, el valor por defecto es "off". Si el valor es "on", el punto extremo DEBERÍA suspender la función de supresión de silencio al detectar datos en banda vocal. Al terminar de transmitir datos en banda vocal HAY QUE restablecer lo que indique el valor actual del parámetro supresión de silencio.

¹¹ El sistema no soporta la compensación de eco en el lado del paquete.

Además, se utilizan los siguientes campos de LocalConnectionOptions para soportar los servicios de seguridad IPCablecom (SE PUEDEN incluir los dos conjuntos de cifrados, o sólo uno):

- **Conjunto de cifrado RTP:** Lista de conjuntos de cifrados para seguridad RTP, ordenadas según su nivel de preferencia. El primer conjunto de cifrado de la lista es el preferido. El punto extremo ELEGIRÁ uno de los conjuntos de cifrados conforme a las reglas descritas en la especificación de seguridad IPCablecom (J.170). Además, el punto extremo DEBERÍA indicar cuáles de los restantes conjuntos de cifrados está dispuesto a soportar (véase 8.4.1 para más detalles). Cada conjunto de cifrado se representa mediante cadenas ASCII compuestas por dos subcadenas (posiblemente vacías) separadas por una barra inclinada ("/"): el primero identifica el algoritmo de autenticación y el segundo el de criptación. En la Rec. UIT-T J.170 se enumeran los posibles conjuntos de cifrados válidos. El parámetro conjunto de cifrado RTP se aplica solamente a los trenes de medios RTP. Si el MGC incluye un parámetro LocalConnectionOptions que requiere la utilización exclusiva de medios no RTP (por ejemplo, transmisión de fax T.38 mediante UDPTL), no se incluirá el parámetro conjunto de cifrado RTP. Cuando el LCO acepte medios RTP y no-RTP y se incluya el parámetro conjunto de cifrado RTP, sólo se podrá aplicar a los medios RTP. En cualquier caso, si el tren de medios que resulta en la conexión no es RTP (como es el caso, por ejemplo, de la transmisión de fax T.38 que utiliza UDPTL), no se tendrá en cuenta el parámetro conjunto de cifrado RTP: no se emplea la seguridad RTP ni se incluyen los parámetros de seguridad RTP en el LocalConnectionDescriptor.
- **Conjunto de cifrado RTCP:** Lista de conjuntos de cifrado para seguridad RTCP, ordenadas según su nivel de preferencia. El primer conjunto de cifrado de la lista es el preferido. El punto extremo ELEGIRÁ uno de los conjuntos de cifrados conforme a las reglas descritas en la especificación de seguridad IPCablecom (J.170). Además, el punto extremo DEBERÍA indicar cuáles de los restantes conjuntos de cifrados está dispuesto a soportar (véase 8.4.1 para más detalles). Cada conjunto de cifrado se representa mediante cadenas ASCII compuestas por dos subcadenas (posiblemente vacías) separadas por una barra inclinada ("/"): el primero identifica el algoritmo de autenticación y el segundo el de criptación. En la Rec. UIT-T J.170 se enumeran los posibles conjuntos de cifrados válidos. El parámetro conjunto de cifrado RTCP se aplica al RTCP solamente cuando se trata de trenes de medios RTP. Si el MGC incluye un parámetro LocalConnectionOptions que requiere la utilización exclusiva de medios no-RTP (por ejemplo, transmisión de fax T.38 utilizando UDPTL), NO SE INCLUIRÁ el conjunto de cifrado RTCP. Cuando el LCO acepte medios RTP y no-RTP y se incluya el parámetro conjunto de cifrado RTCP, sólo se aplica al RTCP para los medios RTP. En cualquier caso, si el tren de medios que resulta en la conexión no es RTP (como es el caso, por ejemplo, de la transmisión de fax T.38 que utiliza UDPTL), no se tendrá en cuenta el parámetro conjunto de cifrado RTCP: no se emplea la seguridad RTCP ni se incluyen los parámetros de seguridad RTCP en LocalConnectionDescriptor.

La pasarela de circuitos troncales DEBE responder con un error (código de error 524 – Inconsistencia de LocalConnectionOptions) si se infringe cualquiera de las reglas antes mencionadas. En el proceso de configuración se pueden modificar todos los valores por defecto mencionados.

Además, el TGCP soporta la vigilancia electrónica de IPCablecom. Cuando una conexión está sometida a vigilancia electrónica, todos los paquetes de medios válidos recibidos a través de la conexión y todos los paquetes de medios enviados a través de la conexión serán reproducidos y reenviados a una función de entrega de vigilancia electrónica¹² tras la inclusión de un identificador

¹² Obsérvese que la replicación se produce en el nivel de red (para más detalles, véase PKT-SP-ESP-I01-991229).

de conexión de contenido de llamada. Las características de reproducción dependen del modo de conexión, excepto en el caso de medios generados por señales aplicadas a la conexión, que se reproducirán siempre, siendo indiferente el modo de conexión. Por ejemplo, una conexión en modo "inactivo" no generará ningún medio interceptado¹³, en tanto que una conexión en modo "sólo envío" sólo generará medios interceptados en el sentido de ida. Los paquetes reproducidos no se incluirán en las estadísticas de la conexión. Los siguientes campos de LocalConnectionOptions se utilizan para el soporte de la vigilancia electrónica de IPCablecom:

- **Identificador de conexión para el contenido de llamada:** El identificador de conexión para el contenido de llamada (CCC, *call content connection*) es un valor de 32 bits que identifica una conexión sometida a vigilancia electrónica. Se añadirá al encabezamiento de paquetes vocales interceptados.
- **Destino de contenido de llamada:** El destino de contenido de llamada especifica una dirección IPv4 seguida del carácter dos puntos (":") y un número de puerto UDP. El destino de contenido de llamada especifica la dirección IP de destino y el puerto para el contenido de llamada interceptado.

RemoteConnectionDescriptor es el descriptor del lado distante de una conexión, en el otro lado de la red IP. Incluye los mismos campos que LocalConnectionDescriptor (que no debe confundirse con LocalConnectionOptions), es decir, los campos que describen una sesión de acuerdo con la norma SDP. La cláusula 8.4 describe detalladamente el uso soportado de SDP en el perfil TGCP. Este parámetro puede tener un valor nulo cuando no se conoce la información para el extremo distante. Esto sucede porque la entidad que construye una conexión comienza enviando una instrucción CreateConnection a una de las dos pasarelas que intervienen. Para la primera CreateConnection emitida no hay información disponible sobre el otro lado de la conexión. Esta información puede proporcionarse posteriormente invocando ModifyConnection.

Cuando se modifican los códecs en el curso de una comunicación, es posible que los puntos extremos utilicen códecs diferentes durante un momento. Como se ha expresado antes, las pasarelas de circuitos troncales PUEDEN descartar cualquier medio recibido que haya sido codificado con un códec diferente del especificado en LocalConnectionOptions para una conexión.

Modo indica el modo de funcionamiento para este lado de la conexión. Las opciones son "sólo envío", "sólo recepción", "envío/recepción", "inactivo", "conexión en bucle de red", "prueba de continuidad de red". El tratamiento de estos modos se especifica al principio de 7.3. Obsérvese que las señales aplicadas a una conexión no siguen el modo de conexión. Es posible que algunos puntos extremos no soporten todos los modos (véase V.1). Si la instrucción especifica un modo no soportado por el punto extremo, hay que devolver un error (código de error 517 – Modo no soportado). Asimismo, si una conexión todavía no ha recibido un RemoteConnectionDescriptor, hay que devolver un error si se intenta efectuar la conexión en cualquiera de los modos "sólo envío" "envío/recepción", "conexión en bucle de red", "prueba de continuidad de red", o si se ha de aplicar una señal (no sólo detectar un evento) a la conexión (se RECOMIENDA el código de error 527 – Falta el RemoteConnectionDescriptor).

ConnectionId es un parámetro que devuelve la pasarela e identifica unívocamente la conexión en el contexto del punto extremo en cuestión. Se DEBE incluir el parámetro con toda respuesta provisional o de ejecución satisfactoria de una instrucción CreateConnection. El parámetro no se incluirá con las respuestas de error cuando no se haya creado la conexión.

LocalConnectionDescriptor es un parámetro que devuelve la pasarela, una descripción de sesión que contiene información sobre, por ejemplo, direcciones y puertos RTP para conexiones "IN" definidas en SDP. Es similar al RemoteConnectionDescriptor, del que sólo se diferencia en que especifica este lado de la conexión. En la cláusula 8.4 se da información detallada sobre la

¹³ Suponiendo que no se aplicó a la conexión una señal generadora de medios.

utilización soportada de SDP en el perfil de TGCP. Se DEBE incluir el parámetro LocalConnectionDescriptor con toda respuesta provisional o de ejecución satisfactoria de una instrucción CreateConnection. El parámetro LocalConnectionDescriptor no se incluirá con las respuesta de error, cuando no se haya creado la conexión.

La pasarela que recibe una instrucción "CreateConnection" que no tiene un parámetro RemoteConnectionDescriptor está en una situación ambigua en cuanto a la conexión en cuestión. Por el hecho de haber exportado un parámetro LocalConnectionDescriptor, tiene la posibilidad de recibir paquetes por esa conexión. Por el hecho de no haber recibido todavía el parámetro RemoteConnectionDescriptor de la otra pasarela, no sabe si los paquetes que recibe han sido autorizados por el MGC. Por tanto tiene que maniobrar entre dos riesgos: suprimir anuncios importantes o tratar datos que no tienen sentido. El comportamiento de la pasarela está determinado por el valor del parámetro modo (disposiciones de seguridad):

- si el modo es "sólo recepción", la pasarela ACEPTARÁ las señales vocales recibidas por la conexión y las transmitirá al punto extremo;
- si el modo es "inactivo", "conexión en bucle" o "prueba de continuidad", la pasarela (como siempre) descartará las señales vocales recibidas por la conexión;
- si el modo es "conexión en bucle de red" o "prueba de continuidad de red", la pasarela efectuará la devolución en eco o la respuesta esperadas. Los medios devueltos en eco o generados se enviarán a la fuente del medio recibido.

Si el punto extremo no tiene un RemoteConnectionDescriptor para la conexión, por definición la conexión no puede estar en los modos "sólo emisión" o "emisión/recepción".

Los parámetros **RequestedEvents**, **RequestIdentifier**, **SignalRequests**, **QuarantineHandling** y **DetectEvents** son, todos ellos, facultativos. El MGC puede utilizarlos para incluir efectivamente una petición de notificación que se ejecuta simultáneamente con la creación de la conexión. Si están presentes uno o más de estos parámetros, el RequestIdentifier DEBE ser uno de ellos. Por tanto, la inclusión de una petición de notificación puede reconocerse por la presencia de un RequestIdentifier. Los parámetros restantes pueden o no estar presentes. Si uno de los parámetros no está presente, se CONSIDERARÁ como si fuera una NotificationRequest normal en la que se ha omitido el parámetro en cuestión. Esto puede tener por efecto que se supriman señales y que se detenga la búsqueda de eventos. Obsérvese que la ausencia de los parámetros RequestedEvents y SignalRequests se interpreta como lista vacía sólo si se incluye un parámetro RequestIdentifier.

Como ejemplo de utilización, considérese el caso de un MGC que desea efectuar una llamada a un sistema de servicios de operador a través de una pasarela de circuitos troncales MF. El MGC podría:

- pedir a la pasarela de circuitos troncales que cree una conexión, para asegurarse de que la pasarela de medios tiene recursos para la llamada;
- pedir a la pasarela de circuitos troncales que tome un circuito troncal de servicios de operador MF e inicie la llamada;
- pedir a la pasarela de circuitos troncales que notifique al MGC cuándo se ha efectuado la llamada.

Todo esto puede realizarse con una sola instrucción CreateConnection, incluyendo una petición de notificación con los parámetros RequestedEvents para el evento de contestación y el parámetro SignalRequests para la señal de establecimiento.

Cuando estos parámetros están presentes, la creación de la conexión y la petición de notificación DEBEN estar sincronizadas, lo que significa que hay que aceptar las dos o rechazarlas. En este ejemplo, la instrucción CreateConnection debe ser rechazada si la pasarela no tiene suficientes recursos o no puede obtener recursos adecuados de la red de acceso local. La petición de notificación al iniciar la llamada deberá rechazarse en la condición de doble toma si el circuito ya

ha sido tomado. En este ejemplo, la llamada no podrá efectuarse si la conexión no puede establecerse y la conexión no se efectuará si el circuito ya está tomado. En tal caso, se devolvería un error (código de error 401 – Circuito ya tomado), por el que se informa al MGC la condición de doble toma.

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y consta de un número entero (véase 7.5) seguido facultativamente de un comentario.

7.3.4 ModifyConnection

Esta instrucción se utiliza para modificar las características de una conexión desde el punto de vista de una pasarela. Esta "visión" de la llamada incluye el descriptor de conexión local y el descriptor de conexión distante.

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

Los parámetros utilizados son los mismos de la instrucción CreateConnection, al que se añade un **ConnectionId** que identifica unívocamente la conexión dentro del punto extremo. La instrucción CreateConnection devuelve este parámetro junto con el descriptor de conexión local. Este identificador identifica unívocamente la conexión en el contexto del punto extremo.

EndpointId SERÁ un nombre de punto extremo totalmente calificado. En el nombre local NO SE UTILIZARÁN comodines. Si la pasarela recibe una ModifyConnection con comodines, devolverá un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

La instrucción ModifyConnection puede utilizarse para modificar parámetros de la conexión, respetando las mismas reglas y restricciones especificadas para CreateConnection:

- Proporcionar información en el otro extremo de la conexión mediante el **RemoteConnectionDescriptor**.
- Activar o desactivar la conexión cambiando el valor del parámetro **modo**. Esto puede efectuarse en cualquier momento durante la conexión, con valores arbitrarios del parámetro. Por ejemplo, una activación puede fijarse al modo "sólo recepción".
- Modificar los parámetros de la conexión mediante **LocalConnectionOptions**, por ejemplo, conmutando a un esquema de codificación diferente, modificando el periodo de paquetización o modificando el tratamiento de la compensación de eco.

La instrucción sólo hará que se devuelva un **LocalConnectionDescriptor** si se modifican parámetros de conexión locales, tales como puertos RTP. Por ejemplo, si sólo se modifica el modo de conexión, no se devolverá un LocalConnectionDescriptor. El parámetro **LocalConnectionDescriptor** NO DEBE incluirse cuando hay una respuesta de error y la conexión no ha sido modificada. Si se omite un parámetro de conexión, por ejemplo, modo o supresión de silencio, se mantendrá el anterior valor de ese parámetro, si es posible. Si una modificación de parámetro requiere la modificación de uno o más parámetros *no especificados*, la pasarela puede

escoger libremente valores adecuados para los parámetros no especificados que deban modificarse¹⁴.

La información de dirección RTP proporcionada en el RemoteConnectionDescriptor especifica la dirección RTP distante del receptor de medios para la conexión. Esta información de dirección RTP puede haber sido modificada por el MGC¹⁵. Cuando se da información de dirección RTP a una pasarela de circuitos troncales para una conexión, la pasarela de circuitos troncales DEBERÍA aceptar solamente trenes de medios (y RTCP) de la dirección RTP especificada. Los trenes de medios recibidos de cualquier otra dirección DEBERÍAN descartarse. Para otros requisitos de seguridad se debe consultar la Rec. UIT-T J.170.

Los parámetros **RequestedEvents**, **RequestIdentifier**, **SignalRequests**, **QuarantineHandling** y **DetectEvents** son facultativos. El MGC puede utilizar estos parámetros para incluir una petición de notificación que se vincula a la modificación de la conexión y se ejecuta simultáneamente con ella. Si se suministra uno o más de estos parámetros, el RequestIdentifier TIENE QUE ser uno de ellos. Obsérvese que la ausencia de los parámetros RequestedEvents y SignalRequests se interpreta como lista vacía sólo si se incluye un parámetro RequestIdentifier.

Cuando estos parámetros están presentes, la modificación de la conexión y la petición de notificación DEBEN estar sincronizadas: hay que aceptar las dos o rechazarlas.

NotifiedEntity es un parámetro facultativo que especifica una nueva "entidad notificada" para el punto extremo.

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y está constituido por un número entero (véase 7.5) seguido facultativamente de un comentario.

7.3.5 DeleteConnection (desde el controlador de pasarela de medios)

Esta instrucción se utiliza para terminar una conexión y accesoriamente registra datos estadísticos sobre la ejecución de la conexión.

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

Para este tipo de instrucción DeleteConnection, el identificador de punto extremo tiene que ser completamente calificado. NO DEBEN utilizarse comodines. Si una pasarela recibe una DeleteConnection con comodín DEBE retornar un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

En el caso general de conexión con dos extremos, esta instrucción hay que enviarla a las dos pasarelas que intervienen en la conexión. Una vez suprimida la conexión, los trenes de medios de la red de paquetes anteriormente soportados por la conexión dejan de estar disponibles. Todo paquete

¹⁴ Esto puede suceder, por ejemplo, si se especifica la modificación de un códec, y el antiguo códec utilizaba supresión de silencio pero el nuevo códec no soporta la supresión de silencio. Por ejemplo, si, además, no se ha especificado el periodo de paquetización, y el nuevo códec soportaba el antiguo periodo de paquetización, el valor de este parámetro no se modificaría, pues tal modificación sería innecesaria.

¹⁵ Por ejemplo, si los medios tienen que atravesar un corta fuego.

de medios recibido para la antigua conexión se descarta, simplemente, y no se envía ningún nuevo paquete de medios para el tren de medios. Asimismo, una vez suprimida la conexión, hay que cancelar todo bucle solicitado para la conexión (salvo si el punto extremo tiene otra conexión que está solicitando el bucle).

En respuesta a la instrucción DeleteConnection, la pasarela devuelve una lista de parámetros que describen el status de la conexión¹⁶. Los parámetros de conexión se DEVOLVERÍAN solamente si la instrucción ha tenido éxito y la conexión ha sido suprimida. Estos parámetros son:

- **Número de paquetes enviados:** Número total de paquetes de datos RTP transmitidos por el emisor a través de la conexión desde el comienzo de la transmisión. La cuenta no se reinicia si el emisor cambia su identificador de fuente de sincronización (SSRC, definido en RTP), por ejemplo como resultado de una instrucción Modify. El valor se DEBE basar en la misma información suministrada a través del mecanismo RTCP.
- **Número de octetos enviados:** Número total de octetos de cabida útil (es decir, sin contar los octetos de encabezamiento o relleno) transmitidos en paquetes de datos RTP por el emisor desde el inicio de la transmisión en la conexión. La cuenta no se reinicia si el emisor cambia su identificador SSRC, por ejemplo como resultado de una instrucción ModifyConnection. El valor se DEBER basar en la misma información suministrada a través del mecanismo RTCP.
- **Número de paquetes recibidos:** Número total de paquetes de datos RTP recibidos por el emisor desde el comienzo de la recepción en la conexión. La cuenta incluye los paquetes recibidos de distintos SSRC si el emisor utilizó varios valores. HAY que contar todos los paquetes recibidos sin importar el modo de conexión o cualquier tipo de error de procesamiento, por ejemplo, un fallo de autenticación.
- **Número de octetos recibidos:** Número total de octetos de cabida útil (es decir, sin contar los octetos de encabezamiento o relleno) transmitidos en paquetes de datos RTP por el emisor desde el comienzo de la transmisión en la conexión. La cuenta incluye los paquetes recibidos de distintos SSRC si el emisor utilizó varios valores. HAY que contar todos los paquetes recibidos sin importar el modo de conexión o cualquier tipo de error de procesamiento, por ejemplo, un fallo de autenticación.
- **Número de paquetes perdidos:** Número total de paquetes de datos RTP que se han perdido desde el comienzo de la recepción. Es el número de paquetes que se esperaba recibir menos el número de paquetes efectivamente recibidos, incluidos los paquetes recibidos tardíamente y los que están duplicados. La cuenta incluye los paquetes recibidos de distintos SSRC si el emisor utilizó varios valores. Por tanto, los paquetes recibidos tardíamente no se cuentan como perdidos y la cuenta de paquetes perdidos puede ser negativa si se reciben paquetes duplicados. La cuenta incluye paquetes recibidos de distintos SSRC si el emisor utilizó varios valores. El número de paquetes que se espera recibir viene dado por la diferencia entre el último número secuencial y el primero de la serie. La cuenta incluye paquetes recibidos de distintos SSRC, si el emisor utilizó varios valores. El valor es cero si, por ejemplo, no se recibieron paquetes en la conexión.
- **Fluctuación entre-llegadas:** Estimación de la varianza estadística del tiempo entre llegadas de paquetes de datos RTP medida en milisegundos y expresada como un entero sin signo. La fluctuación entre-llegadas "J" se define como la desviación media (valor absoluto redondeado) de la diferencia "D" en el espaciamiento de los paquetes en el receptor, en comparación con el emisor, para un par de paquetes. En IETF RFC 1889 se presentan algoritmos de cálculo detallados. La cuenta incluye paquetes recibidos de distintos SSRC si

¹⁶ Los valores calculados no incluirán paquetes creados como resultado de la vigilancia electrónica.

el emisor utilizó varios valores. El valor es cero si, por ejemplo, no se recibieron paquetes en la conexión.

- **Retardo medio de transmisión:** Estimación de la latencia de la red, expresada en milisegundos. Es el valor promedio de la diferencia entre la indicación de tiempo NTP de los emisores de mensajes RTCP y la indicación de tiempo NTP de los receptores, medida cuando se recibieron los mensajes. El promedio se obtiene dividiendo la suma de todas las estimaciones por el número de mensajes RTCP recibidos. Debe señalarse que el cálculo correcto de este parámetro presupone que los relojes están sincronizados. Los dispositivos de pasarela de circuitos troncales también PUEDEN calcular el retardo promedio de transmisión dividiendo por dos el tiempo medido de ida y retorno.

Para una definición más detallada de estas variables, véase IETF RFC 1889.

Los parámetros **RequestedEvents**, **RequestIdentifier**, **SignalRequests**, **QuarantineHandling** y **DetectEvents** son facultativos. El MGC puede utilizarlos para transmitir una petición de notificación insertada que se vincula a la supresión de la conexión y se ejecuta simultáneamente con ella. Sin embargo, si uno o más de estos parámetros están presentes, **RequestIdentifier** TIENE que ser uno de ellos. Por ejemplo, cuando se desconecta un circuito, se podría ordenar a la pasarela que suprimiera la conexión y comenzara a buscar un evento de toma. Esto también puede efectuarse mediante una sola instrucción **DeleteConnection**, transmitiendo el parámetro **RequestedEvents** para el evento de toma, y un parámetro **SignalRequests** vacío. Obsérvese que la ausencia de los parámetros **RequestedEvents** y **SignalRequests** se interpreta como lista vacía sólo si se incluye un parámetro **RequestIdentifier**.

Cuando estos parámetros están presentes, la instrucción de supresión de conexión y la petición de notificación DEBEN estar sincronizadas: hay que aceptar las dos o rechazarlas.

NotifiedEntity es un parámetro opcional que especifica una nueva "entidad notificada" para el punto extremo.

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y está constituido por un número entero (véase 7.5) seguido facultativamente de un comentario.

7.3.6 DeleteConnection (desde la pasarela de circuitos troncales)

En algunas circunstancias la pasarela tendrá que liberar una conexión, por ejemplo, por haberse perdido el recurso asociado con la conexión. La pasarela puede terminar la conexión utilizando una variante de la instrucción **DeleteConnection**:

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

En esta forma de la instrucción **DeleteConnection**, **EndpointId** DEBE estar totalmente calificado. No se utilizarán comodines. Si un MGC recibe una **DeleteConnection** con comodín, devolverá un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

Reason-code es una cadena de texto que comienza por un código de motivo numérico y va seguido facultativamente de una cadena de texto descriptiva. En 7.6 figura una lista de códigos de motivo.

Además del **CallId**, **EndpointId** y **ConnectionId**, la pasarela de circuitos troncales enviará los parámetros de la conexión, posiblemente devueltos al MGC en respuesta a una instrucción **DeleteConnection**. El código de motivo indica la causa de la **DeleteConnection**.

ReturnCode es un parámetro que devuelve el MGC. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase 7.5).

7.3.7 DeleteConnection (suprimir múltiples conexiones por instrucción del controlador de pasarela de medios)

El MGC puede utilizar una variante de la función DeleteConnection para suprimir múltiples conexiones al mismo tiempo. La instrucción puede utilizarse para suprimir todas las conexiones que se relacionan con una llamada para un punto extremo:

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

En esta forma de la instrucción DeleteConnection, **EndpointId** NO PODRÁ incluir el comodín "any of". Todas las conexiones para el punto extremo o los puntos extremos con el CallId especificado serán suprimidas. La instrucción no origina una respuesta de estadísticas ni parámetros individuales de la llamada. Si una pasarela recibe una DeleteConnection (múltiples conexiones desde el controlador de pasarela de medios) con comodín "any of", DEVOLVERÁ un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

El MGC también puede utilizar DeleteConnection para suprimir todas las conexiones que terminan en un punto extremo dado:

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

Con esta forma de la instrucción DeleteConnection, los MGC pueden servirse de la estructura de denominación jerárquica de los puntos extremos para suprimir todas las conexiones pertenecientes a un grupo de puntos extremos. En este caso, parte del componente "nombre del punto extremo local" del EndpointId puede especificarse utilizando el comodín "all" especificado en 7.1.1. NO SE UTILIZARÁ el comodín "any of". La instrucción no origina una respuesta de estadísticas ni parámetros de llamada individuales.

Una vez suprimida la conexión, los trenes de medios de la red de paquetes antes soportados por la conexión dejan de estar disponibles. Todo paquete de medios recibido para la antigua conexión será, simplemente, descartado y no se enviará ningún nuevo paquete de medios. Asimismo, una vez suprimida la conexión, SE CANCELARÁ todo bucle solicitado para la conexión (salvo si el punto extremo tiene otra conexión que está solicitando el bucle).

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase 7.5).

7.3.8 Auditoría

El MGCP se basa en una arquitectura de control de llamada centralizado en la que un MGC actúa como el controlador a distancia de dispositivos de cliente que proporcionan interfaces de comunicaciones vocales a usuarios y redes. Con el fin de alcanzar los mismos niveles de disponibilidad que la RTPC, o niveles aún más altos, algunos protocolos tienen implementados mecanismos para "tocar" periódicamente a los abonados con el fin de reducir al mínimo el tiempo necesario para detectar una interrupción individual. Con esta finalidad se proporciona un mecanismo de auditoría específico del MGCP entre las pasarelas de circuitos troncales y los MGC en un sistema IPCablecom para permitir que el MGC realice la auditoría del estado del punto extremo y de la conexión y recupere las capacidades de un punto extremo específicas del protocolo.

Para las pasarelas de circuitos troncales se definen dos instrucciones para auditoría:

AuditEndPoint: utilizada por el MGC para determinar el status de un punto extremo.

AuditConnection: utilizada por el MGC para obtener información sobre la conexión.

Es conveniente tener una gestión de red que vaya más allá de las capacidades proporcionadas por estas instrucciones, por ejemplo, información sobre el status de la pasarela de circuitos troncales y no sólo de puntos extremos individuales. Cabe esperar que esas capacidades sean soportadas mediante el empleo del protocolo de gestión de red simple (SNMP) y por la definición de una MIB para la pasarela de circuitos troncales; estos dos aspectos están fuera del ámbito de esta Recomendación.

7.3.8.1 AuditEndPoint

El MGC puede utilizar esta instrucción para averiguar el status de un determinado punto extremo.

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

EndpointId identifica el punto extremo al que se le realiza la auditoría. NO SE UTILIZARÁ el comodín "any of". Si una pasarela recibe una AuditEndPoint con el comodín "any of", DEVOLVERÁ un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

El convenio de comodín "all of" puede utilizarse para efectuar la auditoría de un grupo de puntos extremos. Si se utiliza este convenio, la pasarela devolverá la lista de identificadores de punto extremo que satisfacen el comodín en el parámetro **EndPointIdList**, que es simplemente una lista de SpecificEndpointIds (en este caso NO SE INCLUIRÁ RequestedInfo). **MaxEndPointIDs** es un valor numérico que indica el número máximo de EndpointIds a devolver. Si existen puntos extremos adicionales, el parámetro de retorno **NumEndPoints** DEBE estar presente e indicar el número total de puntos extremos que concuerdan con el EndpointID especificado. Para recuperar el siguiente bloque de EndpointIDs, el **SpecificEndPointID** se fija al valor del último punto extremo retornado en la anterior EndpointIDList y se emite la instrucción.

Cuando no se utiliza el comodín, la **RequestedInfo** (que puede estar vacía) describe la información que se solicita para el EndpointId especificado (en tal caso, los parámetros SpecificEndpointID y MaxEndpointID NO SE UTILIZARÁN). Con esta instrucción puede efectuarse la auditoría de la siguiente información específica de punto extremo:

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

Si se consulta un punto extremo acerca de un parámetro que no soporta, el punto extremo no generará un error, sino que omitirá el parámetro en la respuesta.

Si se consulta un punto extremo acerca de un parámetro que soporta, pero para el que no tiene ningún valor, el punto extremo NO GENERARÁ un error, sino que lo INCLUIRÁ en la respuesta con un valor de parámetro vacío.

Si la auditoría se realiza satisfactoriamente, la respuesta AuditEndPoint INFORMARÁ acerca de cada uno de los ítems para los que se solicitó información. Obsérvese que los parámetros marcados explícitamente como "facultativos" en la lista a continuación, y que no son soportados por el punto extremo, se omiten en la respuesta:

- **RequestedEvents:** Valor actual de RequestedEvents que el punto extremo está utilizando, incluida la acción asociada con cada evento. Los eventos persistentes se incluyen en la lista.
- **SignalRequests:** Lista de las señales sometidas a temporización que se encuentran activas en ese momento, señales On/Off que están "activadas" ("on") en ese momento para el punto extremo (con o sin parámetro) y cualquier señal breve pendiente¹⁷. No se incluirán las señales sometidas a temporización para las cuales haya expirado el periodo de temporización, ni las señales breves que estén produciendo efecto en ese momento. Las señales parametrizadas se informan junto con los parámetros con los que se aplican.
- **RequestIdentifier:** RequestIdentifier para la última NotificationRequest recibida por el punto extremo (incluye la petición de notificación insertada en primitivas de tratamiento de la conexión). Si no se ha recibido ninguna petición de notificación, se retornará el valor cero.
- **NotifiedEntity:** Actual "entidad notificada" para el punto extremo. Obsérvese que la MG PUEDE incluir solamente el nombre de dominio de su Notified Entity si el parámetro Notified Entity de un mensaje o de un acuse de recibo de mensaje TGCP sólo ha proporcionado el nombre de dominio. En este caso, el MGC DEBERÍA aceptar el valor.
- **ConnectionIdentifiers:** Lista de ConnectionIdentifiers, separados por un carácter coma (",") para todas las conexiones que existen en ese momento para el punto extremo especificado.
- **DetectEvents:** valor de DetectEvents que el punto extremo está utilizando. Los eventos persistentes se incluyen en la lista.
- **ObservedEvents:** Lista actual de eventos observados para el punto extremo.
- **EventStates:** Para eventos que tienen asociados estados auditables, el evento que corresponde al estado en que se encuentra el punto extremo, por ejemplo, estado de toma si el circuito troncal MF para el punto extremo está tomado en ese momento. La definición de cada evento individual determinará si el evento en cuestión tiene asociado un estado auditable.
- **VersionSupported:** Lista de las versiones de protocolo soportadas por el punto extremo.
- **MaxMGCPDatagram:** Tamaño máximo de un datagrama MGCP en bytes soportado por el punto extremo (véase 8.5.3). El valor excluye cualquier tara de capa inferior. El soporte de este parámetro es facultativo. Cuando no se obtenga ningún valor, se supone que es el valor máximo por defecto de datagrama MGCP.
- **Capabilities:** Capacidades del punto extremo, similares a las del parámetro LocalConnectionOptions e incluyendo lotes de eventos y modos de conexión. Si es necesario especificar que algunos parámetros, como por ejemplo, supresión de silencio, sólo son compatibles con algunos códecs, la pasarela devolverá varios conjuntos de capacidades.

¹⁷ No debería de haber señales breves pendientes.

- **Algoritmo de compresión** Lista de códecs soportados. HAY QUE UTILIZAR los nombres literales definidos en la cláusula 7.5 (cuadro 3) de la especificación IPCablecom de códecs audio/video (J.161). Si se reciben algoritmos de compresión desconocidos, NO SE DEBERÍAN tener en cuenta. Los parámetros restantes se aplicarán a todos los códecs especificados en esta lista.
- **Periodo de paquetización** Se puede especificar un solo valor o una gama de valores.
- **Ancho de banda** Se puede especificar un solo valor o una gama que corresponda a la gama del periodo de paquetización (suponiendo que no hay supresión de silencio).
- **Compensación de eco** Indica si la compensación de eco está o no soportada¹⁸.
- **Supresión de silencio** Indica si la supresión de silencio está o no soportada.
- **Tipo de servicio** Indica si el tipo de servicio está o no soportado.
- **Lotes de eventos** Lista de los lotes de eventos soportados. El primer lote de eventos en la lista será el lote por defecto.
- **Modos** Lista de los modos de conexión soportados.
- **Combinación de algoritmos RTP** Lista de algoritmos de autenticación y criptación soportados para el RTP.
- **Combinación de algoritmos RTCP** Lista de algoritmos de autenticación y criptación soportados para el RTCP.
- **Vigilancia electrónica** Indica si el sistema soporta o no la vigilancia electrónica IPCablecom.

El MGC puede optar por utilizar la instrucción AuditConnection para obtener más información sobre las conexiones.

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase 7.5).

Si no se ha solicitado información y el EndpointId hace referencia a un EndpointId totalmente especificado, la pasarela simplemente devuelve una respuesta de éxito (código de retorno 200 – Transacción ejecutada normalmente).

Debe señalarse que todas las informaciones retornadas son sólo una "instantánea". Las nuevas instrucciones recibidas, actividades locales, etc. pueden modificar la mayor parte de esa información. Por ejemplo, el estado de toma puede cambiar antes de que el MGC reciba esa información.

¹⁸ Actualmente, todos los puntos extremos TGCP deben soportar la compensación de eco.

7.3.8.2 AuditConnection

La auditoría de conexiones individuales en un punto extremo puede efectuarse utilizando la instrucción AuditConnection.

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

El **EndpointId** identifica el punto extremo objeto de la auditoría; NO SE UTILIZARÁN comodines. Si una pasarela recibe una AuditConnection con comodín, DEVOLVERÁ un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido). La **RequestedInfo** (que puede estar vacía) describe la información solicitada por el **ConnectionId** dentro del EndpointId especificado. Mediante esta instrucción, se puede efectuar la auditoría de la siguiente información de conexión:

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

Si se consulta un punto extremo acerca de un parámetro que no soporta, el punto extremo NO GENERARÁ un error, sino que omitirá el parámetro en la respuesta.

Si se consulta un punto extremo acerca de un parámetro que soporta, pero para el que no tiene ningún valor, el punto extremo NO GENERARÁ un error, sino que lo INCLUIRÁ en la respuesta con un valor de parámetro vacío.

Si la auditoría se realizó satisfactoriamente, la respuesta AuditConnection INFORMARÁ acerca de cada uno de los ítems para los que se solicitó información de auditoría. Obsérvese que los parámetros marcados explícitamente como "facultativos" en la lista a continuación, y que no son soportados por el punto extremos, se omiten en la respuesta:

- **CallId** El CallId para la llamada a que pertenece la conexión
- **NotifiedEntity** La actual "entidad notificada" para el punto extremo.
- **LocalConnectionOptions** El parámetro LocalConnectionOptions suministrado para la conexión.
- **Mode** El actual modo de conexión.
- **ConnectionParameters** Los parámetros de conexión actuales para la conexión.
- **LocalConnectionDescriptor** El LocalConnectionDescriptor suministrado por la pasarela para la conexión.
- **RemoteConnectionDescriptor** El RemoteConnectionDescriptor más reciente suministrado por la pasarela en una instrucción anterior CreateConnection o ModifyConnection para la conexión.

ReturnCode es un parámetro que devuelve la pasarela. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase 7.5).

Si no se solicitó información, y el EndpointId hace referencia a un punto extremo válido, la pasarela simplemente comprueba si la conexión solicitada existe y, en caso afirmativo, devuelve una respuesta positiva (código de retorno 200 – Transacción ejecutada).

7.3.9 Rearranque en curso

La pasarela utiliza la instrucción `RestartInProgress` para señalar que un punto extremo, o un grupo de puntos extremos, se retira del servicio o se restablece al servicio.

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

El **EndpointId** identifica los puntos extremos que son retirados del servicio o que se ponen en servicio. Puede utilizarse el comodín "all of" para aplicar la instrucción a un grupo de puntos extremos, por ejemplo, a todos los puntos extremos que están asociados a una interfaz especificada, o incluso a todos los puntos extremos que están asociados a una pasarela dada. NO SE UTILIZARÁ el comodín "any of". Si un MGC recibe una `Restart in Progress` con el comodín "any of", DEVOLVERÁ un error (cuyo código DEBERÍA ser el 500 – La transacción no pudo ejecutarse porque el punto extremo es desconocido).

El parámetro `RestartMethod` especifica el tipo de rearranque:

- Un método de rearranque "paulatino" indica que el punto extremo o puntos extremos especificados serán retirados del servicio después del "retardo de rearranque" especificado. Las conexiones establecidas no son afectadas todavía, pero el MGC debería abstenerse de establecer nuevas conexiones y debería tratar de suprimir paulatinamente toda conexión existente. Al expirar el periodo de retardo, la MG debería enviar un nuevo mensaje RSIP con un método de rearranque "forzado". De esta manera, se indicará explícitamente al MGC que los puntos extremos están ahora fuera de servicio.
- Un método de rearranque "cancelar-paulatino" indica que la pasarela está cancelando un método de rearranque "paulatino" iniciado anteriormente para los mismos puntos extremo. Los puntos extremos siguen en servicio. Al enviar esta instrucción, la pasarela empieza a autorizar inmediatamente el establecimiento de nuevas conexiones en estos puntos extremo.
- Un método de rearranque "forzado" indica que los puntos extremos especificados se retiran del servicio bruscamente. Las conexiones establecidas, si existían algunas, se pierden.
- Un método "rearranque" indica que el servicio será restablecido en los puntos extremos una vez transcurrido el "retardo de rearranque" especificado. No existen conexiones establecidas en ese momento en los puntos extremos.
- Un método "desconectado" indica que el punto extremo ha sido desconectado y en este momento está tratando de establecer la conectividad. El "retardo de rearranque" especifica el número de segundos que el punto extremo ha estado desconectado. Las conexiones establecidas no son afectadas.

El parámetro "retardo de rearranque" es facultativo y se expresa en segundos. Si se omite, debe considerarse que el valor del retardo es nulo. En el caso del método "paulatino", un retardo nulo indica que el punto extremo nunca saldrá de servicio como resultado de esta operación, y que el MGC debería simplemente esperar la terminación natural de las conexiones existentes, y no establecer nuevas conexiones. El retardo de rearranque se considera siempre nulo en el caso del método "forzado" y del método "cancelar-paulatino" y, por tanto, la pasarela NO UTILIZARÁ el parámetro "retardo de rearranque" con estos métodos de rearranque. Un retardo de rearranque nulo en el caso del método "rearranque" indica que el servicio ya ha sido restablecido. Esto suele ocurrir cuando la pasarela ha rearrancado/reinicializado. Para mitigar los efectos de un cambio de dirección IP de la pasarela, el MGC PUEDE desear resolver el nombre de dominio de la pasarela interrogando al servidor de nombres de dominio (DNS) cualquiera que sea el tiempo de vida (TTL) de un registro de recursos actual para la pasarela rearrancada.

Las pasarelas de circuitos troncales DEBERÍAN enviar un mensaje informativo RestartInProgress "paulatino" o "forzado" al MGC cuando son retiradas del servicio, por ejemplo, cuando se desconectan. El agente de llamada no puede confiar en que siempre habrá de recibir tales mensajes. Ahora bien, la pasarela de circuitos troncales sí TIENE QUE ENVIAR un mensaje RestartInProgress "paulatino" o "forzado" cuando se retira de servicio un punto extremo mediante un proceso de configuración o tras la detección de un fallo de la facilidad de transporte del punto extremo (por ejemplo pérdida de señal, recepción de una alarma amarilla, etc.). Las pasarelas de circuitos troncales ENVIARÁN un mensaje RestartInProgress "rearranque" con un retardo nulo a su MGC cuando retornan al servicio de acuerdo con el procedimiento de rearranque especificado en 7.4.3.5; los MGC siempre recibirán este mensaje. Asimismo, las pasarelas de circuitos troncales ENVIARÁN un mensaje RestartInProgress "desconectado" a su "entidad notificada" actual de acuerdo con el procedimiento "desconectado" especificado en 7.4.3.6. El parámetro "retardo de rearranque" NO SE UTILIZARÁ con el método de rearranque "forzado".

El mensaje RestartInProgress se enviará a la actual "entidad notificada" para el EndpointId en cuestión. Cabe esperar que para cada punto extremo se ha configurado un MGC por defecto, es decir, una "entidad notificada"; entonces, tras una reinicialización el MGC por defecto será la "entidad notificada" para cada punto extremo. Las pasarelas de circuitos troncales UTILIZARÁN todas las posibilidades de comodines para minimizar el número de mensajes RestartInProgress generados cuando rearrancan múltiples puntos extremos en una pasarela y los puntos extremos son gestionados por el mismo MGC.

ReturnCode es un parámetro retornado por el MGC. Indica el resultado de la instrucción y está constituido por un número entero seguido facultativamente de un comentario (véase 7.5).

Además, el MGC PUEDE devolver una **NotifiedEntity** con la respuesta a la instrucción RestartInProgress; normalmente, esto sólo se debería hacer como respuesta a "rearranque" o "desconectado" (véanse también 7.4.3.5 y 7.4.3.6). Cuando se incluya en la respuesta un parámetro NotifiedEntity, se estará especificando una nueva "entidad notificada" para los puntos extremos; sólo DEBERÍA incluirse cuando es una respuesta de error con el código 521 (punto extremo redirigido). Obsérvese que el comportamiento descrito para devolver una NotifiedEntity con la respuesta sólo se define para las respuestas RestartInProgress y NO DEBERÍA utilizarse en respuestas a otras instrucciones. Los otros comportamiento no están definidos:

- Si es una respuesta de ejecución satisfactoria (código de retorno 200 – Transacción ejecutada), el procedimiento de rearranque ha sido completado y la NotifiedEntity retornada es la nueva "entidad notificada" para el punto extremo o los puntos extremos.
- Si la respuesta del MGC indicaba un error, el procedimiento de rearranque no ha sido completado todavía. Si la respuesta es 521 (punto extremo redirigido), la respuesta tiene que incluir un parámetro NotifiedEntity que especifique la nueva "entidad notificada" para el punto o los puntos extremos, que habrá que utilizar cuando se vuelva a intentar el procedimiento de rearranque (como nueva transacción).

En el caso de "rearranque" y "desconectado", es OBLIGATORIO intentar el rearranque en cuestión siempre que el MGC retorne un código de error transitorio (4xx) y se DEBERÍA intentar para cualquier otro restartMethod. Se RECOMIENDA terminar cualquier tipo de rearranque cuando el código de error es permanente (5xx), salvo en el caso del 521, como ya se dijo.

Por último, se puede devolver un parámetro **VersionSupported** con una lista de versiones soportadas si la respuesta indicaba incompatibilidad de la versión (código de error 528).

7.4 Estados, condiciones de cambio-por-fallo y situación de competencia

Para implementar una señalización adecuada de las llamadas, el MGC debe registrar el estado del punto extremo y la pasarela debe asegurarse de que los eventos son correctamente notificados al MGC. Pueden existir condiciones especiales cuando la pasarela o el MGC son rearrancados: la

pasarela puede tener que ser redireccionada hacia un nuevo MGC en el curso de procedimientos de "cambio-por-fallo". De manera similar, el MGC puede tener que ejecutar acciones especiales cuando se pone fuera de línea o se reanuda la pasarela.

7.4.1 Resumen y aspectos importantes

Como se dijo en 7.1.4, los MGC se identifican por su nombre de dominio y en cualquier instante dado cada punto extremo tiene asociada una, y sólo una "entidad notificada". En esta cláusula se destacan los aspectos que son de especial importancia para la fiabilidad y el cambio-por-fallo en MGCP:

- Un MGC se identifica por su nombre de dominio, no por sus direcciones de red, y varias direcciones pueden estar asociadas con un nombre de dominio.
- En cualquier instante dado, un punto extremo tiene asociado un MGC, y sólo uno. El MGC asociado con un punto extremo es el valor actual de la "entidad notificada".
- La "entidad notificada" se fija inicialmente a un valor suministrado. Cuando para un punto extremo se reciben instrucciones con un parámetro NotifiedEntity que incluye nombres de punto extremo que contienen comodines, la "entidad notificada" se fija al valor especificado. Si la "entidad notificada" para un punto extremo está vacía o ha sido fijada explícitamente¹⁹, la "entidad notificada" adopta por defecto la dirección fuente de la última instrucción de tratamiento de la conexión o petición de notificación recibida para el punto extremo. En este caso, el MGC será identificado de esa forma por su dirección de red, lo que sólo DEBERÍA hacerse excepcionalmente.
- Las respuestas a instrucciones siempre se envían a la dirección fuente de la instrucción, cualquiera que sea la "entidad notificada" actual. Cuando es necesario remolcar un mensaje Notify con la respuesta, el datagrama se envía igualmente a la dirección fuente de la nueva instrucción recibida, sin tener en cuenta la NotifiedEntity para cualquiera de las instrucciones.
- Cuando la "entidad notificada" hace referencia a un nombre de dominio que se resuelve en múltiples direcciones IP, los puntos extremos pueden conmutar entre estas direcciones, pero no pueden, por sí mismos, cambiar la "entidad notificada" por otro nombre de dominio. Sin embargo, un MGC puede ordenarles que conmuten a una nueva dirección proporcionándoles una nueva "entidad notificada".
- Si un MGC quedara indisponible, los puntos extremos gestionados por ese MGC resultarán finalmente "desconectados". Estos puntos extremos sólo pueden volver a conectarse si se da cualquiera de estos dos casos: que el MGC que falló quede de nuevo disponible, o que otro MGC (un MGC de respaldo) establezca contacto con los puntos extremos afectados y les proporcione una nueva "entidad notificada".
- Cuando otro MGC (un MGC de respaldo) ha asumido el control de un grupo de puntos extremos, se supone que el MGC que ha fallado se comunicará y sincronizará con el MGC de respaldo para que éste restituya a aquél el control de los puntos extremos afectados, si así se desea. Otra posibilidad es que el MGC que falló desempeñe ahora el papel de MGC de respaldo.

Debe señalarse que no se ha proporcionado ningún procedimiento para la resolución de conflictos de traspaso entre distintos MGC; se confía en que los MGC saben lo que están haciendo y se comunicarán entre sí (no obstante, se puede utilizar AuditEndpoint para averiguar cuál es la "entidad notificada" actual).

¹⁹ Esto podría suceder, por ejemplo, cuando se especifica un parámetro NotifiedEntity vacío.

7.4.2 Retransmisión y detección de asociaciones perdidas

El protocolo MGCP está organizado como un conjunto de transacciones, cada una de las cuales se compone de una instrucción y una respuesta. Los mensajes MGCP, que se transportan mediante UDP, pueden sufrir pérdidas. Cuando no se recibe oportunamente una respuesta (véase 8.5), las instrucciones se repiten. Las pasarelas DEBEN mantener en memoria una lista de las respuestas que envían a transacciones recientes (una lista de todas las respuestas que han enviado en los últimos T_{hist} segundos) y una lista de las transacciones que están en ejecución. El valor por defecto de T_{hist} es 30 segundos.

Los identificadores de transacción de las instrucciones entrantes se comparan primeramente con los identificadores de transacción de las respuestas recientes. Si se encuentra una concordancia, la pasarela no ejecuta la transacción y se limita a repetir la respuesta antigua. Si no se encuentra una concordancia con una transacción a la que se haya respondido antes, el identificador de transacción de la instrucción entrante se compara con la lista de transacciones cuya ejecución todavía no ha terminado. Si se encuentra tal concordancia, la pasarela no realiza la transacción; el tratamiento ulterior dependerá de cuál sea la instrucción. Si se trata de CreateConnection o ModifyConnection, la pasarela ENVIARÁ una respuesta provisional, si es otra la instrucción simplemente no se tiene en cuenta. En cualquier caso, se ha de proporcionar una respuesta final cuando se complete la ejecución de la instrucción.

Este mecanismo de repetición se utiliza para prevenir cuatro tipos posibles de errores:

- errores de transmisión, cuando, por ejemplo, se pierde un paquete por causa de ruido en una línea o congestión en una cola;
- fallo de un componente, cuando, por ejemplo, una interfaz para un MGC está indisponible;
- fallo de un MGC, cuando, por ejemplo, todas las interfaces de un MGC están indisponibles;
- cambio-por-fallo, cuando, por ejemplo, un nuevo MGC está "asumiendo el control" transparentemente.

Debería ser posible que los elementos hagan a partir de los datos históricos una estimación de la tasa de pérdida de paquetes. En un sistema debidamente configurado, esta tasa debería ser muy baja, típicamente inferior a 1% en promedio. Si un MGC o una pasarela tienen que repetir un mensaje varias veces, cabe suponer que se esté en presencia de algo más que un error de transmisión. Por ejemplo, con una tasa de pérdidas uniformemente distribuida de 1%, la probabilidad de que fracasen cinco intentos de transmisión consecutivos es de 1 en 10 mil millones: sólo debería producirse una vez cada diez días en el caso de un MGC que procesa 1000 transacciones por segundo. (Es la tasa efectiva de pérdida de paquetes la que determina cuántas veces puede repetirse una instrucción sin que pueda sospecharse otro problema.) Cuando los errores no están distribuidos uniformemente, la probabilidad de fallos consecutivos puede ser algo más alta. Se debe señalar que el "umbral de sospecha" ("Max1") normalmente es más bajo que el "umbral de desconexión" ("Max2"); el valor de Max2 TIENE que ser superior.

El algoritmo de retransmisión MGCP se muestra en la figura 2 y se describe en los siguientes párrafos:

Un algoritmo de retransmisión clásico contaría simplemente el número de repeticiones sucesivas y llegaría a la conclusión de que la asociación se rompe después de retransmitir el paquete un número excesivo de veces (típicamente, entre 7 y 11 veces). Para tener en cuenta la posibilidad de un "cambio-por-fallo" no detectado o en curso, el algoritmo clásico se ha modificado como sigue:

- La pasarela siempre HARÁ una comprobación para determinar la presencia de un nuevo MGC. Esto se puede saber por:
 - la recepción de una instrucción en la que la NotifiedEntity apunta a un nuevo MGC; o
 - la recepción de una respuesta de redireccionamiento que apunta a un nuevo MGC.

- Si se detecta un nuevo MGC, la pasarela REDIRECCIONARÁ hacia ese nuevo MGC las retransmisiones de todas las instrucciones pendientes dirigidas inicialmente al punto o puntos extremos. Las respuestas a instrucciones nuevas o antiguas seguirán transmitiéndose a la dirección fuente de la instrucción.
- Antes de efectuar una retransmisión, se verifica que el tiempo transcurrido desde el envío del datagrama inicial no es mayor que $T_{s_{max}}$. Si el tiempo transcurrido es mayor que $T_{s_{max}}$, se SUSPENDERÁ la retransmisión. Si ha transcurrido más de $2 \cdot T_{thist}$, el punto extremo es desconectado.
- Si el número de retransmisiones a este MGC es igual a "Max1", la pasarela PUEDE interrogar al servidor de nombres para detectar un posible cambio de interfaces de MGC, cualquiera que sea el tiempo de actividad (TTL, *time to live*) asociado al registro del servidor DNS.
- La pasarela puede haber aprendido varias direcciones IP para el MGC. Si el número de retransmisiones para una de estas direcciones IP es mayor que "Max1" y menor que "Max2", y si hay más direcciones IP con las cuales no se han intentado retransmisiones, la pasarela DIRIGIRÁ las retransmisiones a las direcciones alternativas restantes en su lista local. Asimismo, la recepción de notificaciones explícitas de red, por ejemplo que no hay red, anfitrión, protocolo o puerto ICMP disponibles, DEBERÍA provocar que la pasarela intente utilizar otra dirección (teniendo en cuenta desde luego los posibles aspectos de seguridad).
- Si no queda ninguna interfaz por ensayar, y el número de retransmisiones es Max2, la pasarela DEBERÍA establecer contacto con el DNS una vez más para averiguar si hay otras interfaces disponibles. En caso negativo, se INTERRUMPIRÁN las retransmisiones. Si ha transcurrido más de $2 \cdot T_{thist}$, el punto extremo se desconecta.
- Cuando el punto extremo se haya desconectado, el tratamiento ulterior será diferente según el elemento que detectó la pérdida de asociación (pasarela o GC):
 - La pasarela INICIARÁ el procedimiento "desconectado" conforme a la cláusula 7.4.3.6.
 - El MGC NO TRATARÁ de utilizar el punto extremo para ninguna nueva llamada mientras no se haya restaurado la conectividad. Más aún, el MGC IMPLEMENTARÁ un algoritmo para detectar la restauración de la conectividad con el punto extremo (por ejemplo, al recibir una respuesta a una instrucción periódica Audit Endpoint). Cuando se restablezca la conectividad con el punto extremo, y de no haber ninguna otra condición que impida a éste soportar las llamadas, el MGC GARANTIZARÁ que se puede utilizar el punto extremo para nuevas llamadas sin que se requiera ninguna intervención manual.

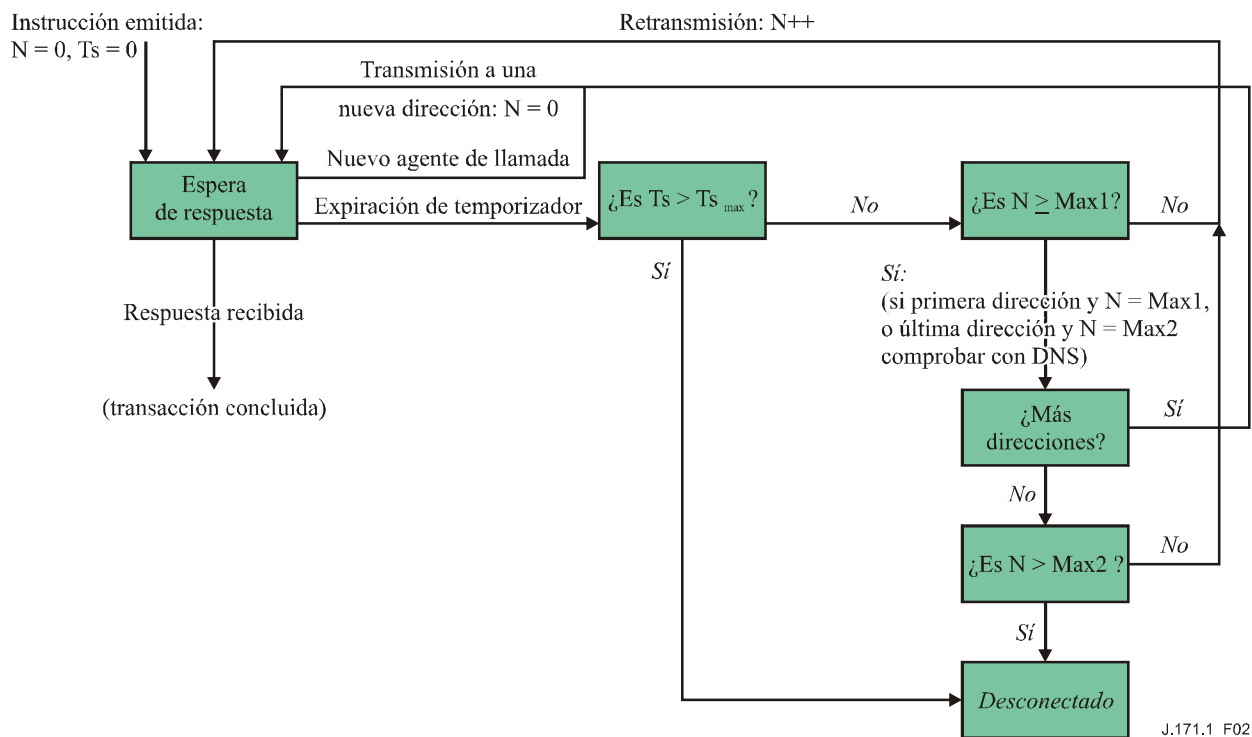


Figura 2/J.171.1 – Algoritmo de retransmisión

Para adaptarse automáticamente a la carga de la red, el protocolo MGCP especifica temporizadores que aumentan exponencialmente (véase 8.5.2). Si el periodo inicial de temporización se fija a 200 milisegundos, la pérdida de una quinta retransmisión se detectará después de unos seis segundos. Probablemente, éste es un tiempo de espera aceptable para la detección de un cambio-por-fallo. Las retransmisiones deberían continuar después de ese plazo no sólo para resolver tal vez un problema transitorio de conectividad, sino también para dar un poco más de tiempo a la ejecución de un cambio-por-fallo; un periodo de espera (en total) de 30 segundos probablemente sea aceptable.

Sin embargo, es importante limitar el tiempo máximo de las retransmisiones. Antes de cualquier retransmisión se comprueba si el tiempo transcurrido (T_s) desde el envío del datagrama inicial no es mayor que $T_{s_{max}}$. Si ha transcurrido un tiempo mayor que $T_{s_{max}}$, se INTERRUMPIRÁN las retransmisiones. Una vez expire el tiempo $T_{s_{max}}$ o se hayan enviado todas las retransmisiones a todas las direcciones IP conocidas, se hace una pausa antes de declarar desconectado el punto extremo. Esta pausa es un periodo de tiempo durante el que la única acción consiste en esperar la respuesta a alguna de las retransmisiones recientes. El periodo pasivo dura lo que resta del doble de la expectativa de vida de la transacción inicial ($2 * T_{t_{hist}}$). Este periodo de estabilización permite que todas las transacciones activas finalicen o que expiren sus periodos de temporización antes de que se desconecte el punto extremo. Con esta precaución es más probable que el punto extremo re arranque cada vez a partir de un estado despejado o inicial. El punto extremo quedará desconectado cuando hayan transcurrido más de $2 * T_{t_{hist}}$ segundos. El valor $T_{s_{max}}$ está relacionado con el valor $T_{t_{hist}}$: $T_{t_{hist}}$ tiene que ser mayor o igual a $T_{s_{max}}$ o superior, más el tiempo máximo de propagación en la red, $T_{p_{max}}$.

En otras palabras, para evitar que las instrucciones retransmitidas se ejecuten más de una vez se DEBE satisfacer la siguiente relación: $T_{t_{hist}} \geq T_{s_{max}} + T_{p_{max}}$.

El valor por defecto de $T_{s_{max}}$ es 20 segundos. Por tanto, si el máximo tiempo de propagación supuesto es diez segundos, las respuestas a las transacciones antiguas deben conservarse durante un

periodo de al menos 30 segundos. Es muy importante lograr que el emisor y el receptor convengan en estos valores.

El valor por defecto para "Max1" es cinco retransmisiones y el valor por defecto para "Max2" es siete retransmisiones. Estos dos valores pueden ser modificados por la configuración.

Además, ES OBLIGATORIO que se pueda inhabilitar la interrogación de DNS sobre "Max1" sobre "Max2" o sobre ambos valores en la configuración.

7.4.3 Condiciones de competencia

En esta cláusula se describe cómo el MGCP trata las condiciones de pugna por llegar primero (condiciones de competencia).

En primer lugar, para el tratamiento de las condiciones de competencia el MGCP utiliza la noción de una "lista de cuarentena" que aísla los eventos, y la detección explícita de la desincronización, por ejemplo, el fracaso en la obtención del estado de toma por haberse producido una doble toma para un punto extremo.

En segundo lugar, el MGCP no presupone que el mecanismo de transporte mantendrá el orden de las instrucciones y respuestas. Esto puede ocasionar condiciones de competencia, las cuales pueden ser eliminadas por un comportamiento adecuado del MGC al ordenar correctamente las instrucciones.

Por último, en algunos casos, muchas pasarelas pueden, al mismo tiempo, tomar la decisión de comenzar a funcionar. Esto puede suceder, por ejemplo, si en una zona se pierde la alimentación en energía o la capacidad de transmisión como consecuencia de un terremoto o una tempestad de hielo. Cuando se restablece la alimentación y la capacidad de transmisión, es posible que muchas pasarelas decidan, al mismo tiempo, enviar instrucciones RestartInProgress; esta situación, si no se controla cuidadosamente, puede conducir a un funcionamiento muy inestable.

7.4.3.1 Lista de cuarentena

Las pasarelas controladas por MGCP recibirán peticiones de notificación en las que se les pide que estén atentas a una lista de eventos. Los elementos del protocolo que determinan el tratamiento de estos eventos son la lista de "eventos solicitados" y la lista de "eventos a detectar".

Cuando se inicializa el punto extremo, la lista de eventos solicitados consta solamente de los eventos persistentes para el punto extremo. Tras la recepción de una instrucción, la pasarela comienza a observar el punto extremo para detectar la aparición de los eventos indicados en la lista, y la de los eventos persistentes.

Los eventos se examinan a medida que van produciéndose. Lo que se haga después depende del parámetro "acción" asociado al evento en la lista de eventos solicitados. Los eventos que están definidos como "accumulate" se acumulan en una lista de eventos observados. Se continúa así hasta que se encuentre un evento que provoca una instrucción Notify, que se enviará a la "entidad notificada".

En esta situación, la pasarela transmitirá la instrucción Notify y colocará el punto extremo en el "estado de notificación". Mientras el punto extremo está en el "estado de notificación", los eventos que se detectan en el punto extremo se almacenan en una memoria intermedia para "cuarentena", con miras a su ulterior procesamiento. Los eventos son, en cierto sentido, "puestos en cuarentena". Los eventos detectados son los especificados por la unión lógica del parámetro RequestedEvents y el parámetro DetectEvents recibido más recientemente o, en caso de no haberse recibido ningún parámetro DetectEvents, los eventos a que se hace referencia en los RequestedEvents. Los eventos persistentes también son detectados.

El punto extremo sale del "estado notificación" cuando se recibe la respuesta a la instrucción Notify²⁰. La instrucción Notify puede retransmitirse en el "estado notificación", como se especifica en 7.4.2.

Si entretanto el punto extremo está desconectado, o ha quedado desconectado (véase 7.4.2), nunca se recibirá respuesta a la instrucción Notify. Se considerará que la instrucción Notify se perdió y que por lo tanto ya no está pendiente, pero el punto extremo continúa en el "estado de notificación". Si esto llegase a ocurrir, la ejecución del procedimiento "desconectado" especificado en la cláusula 7.4.3.6 permitirá suspender el "estado de notificación" del punto extremo.

Cuando el punto extremo sale del "estado notificación" reinicia o pone a cero la lista de eventos observados del punto extremo. Después de esto, el comportamiento de la pasarela depende del valor del parámetro QuarantineHandling en la instrucción NotificationRequest:

Si el agente de llamada/controlador de pasarela de medios especificó un parámetro QuarantineHandling, "step", o si no se especificó ningún valor, la pasarela utilizará entonces el "modo lockstep", lo que implica que la pasarela recibirá una nueva instrucción NotificationRequest después de enviar una instrucción Notify. Entretanto, el punto extremo está en el "estado lockstep": los eventos que sucedan y que se tengan que detectar se almacenan simplemente en la memoria intermedia de cuarentena. Los eventos que se ponen en cuarentena en este estado y en el "estado notificación" son los mismos. Una vez recibida y ejecutada correctamente la nueva NotificationRequest, el punto extremo sale del "estado lockstep".

El procedimiento será diferente si el agente de llamada/controlador de pasarela de medios especificó un parámetro "loop" (modo "loop"). Una vez sale del "estado de notificación", la pasarela reinicia a un valor nulo la lista de eventos observados para el punto extremo y da inicio al procesamiento de la lista de eventos en cuarentena, utilizando la lista de eventos solicitados ya recibida. Al procesar estos eventos, la pasarela podría encontrarse con un evento que provoque el envío de una instrucción Notify. En este caso, la pasarela puede adoptar uno de los siguientes dos comportamientos:

- puede transmitir inmediatamente una instrucción Notify que informe todos los eventos que se acumularon en la lista de eventos observados hasta el evento que provocó el envío de la instrucción Notify inclusive, dejando los eventos no procesados en la memoria intermedia de cuarentena;
- puede intentar vaciar la memoria intermedia de cuarentena y transmitir una sola instrucción Notify que informe varios conjuntos de eventos. Hay que dejar en la memoria de cuarentena los eventos que se produzcan después del último que provocó el envío de una instrucción.

Si la pasarela transmite una instrucción Notify, el punto extremo volverá a entrar al "estado de notificación" y permanecerá así hasta que se reciba un acuse de recibo (como se describió anteriormente). Si la pasarela no encuentra ningún evento puesto en cuarentena que provoque una instrucción Notify, entonces coloca al punto extremo en el estado normal. Los eventos se procesarán al llegar, exactamente como si se acabara de recibir una instrucción NotificationRequest.

Una pasarela puede recibir en cualquier momento una nueva instrucción NotificationRequest para el punto extremo, incluso cuando el punto extremo esté desconectado, que también saca al punto extremo del "estado de notificación", si la NotificationRequest se ejecuta correctamente. La activación de una NotificationRequest insertada tiene el mismo efecto que la recepción de una nueva NotificationRequest, salvo que no se procesa de nuevo la lista actual de ObservedEvents, no se modifica.

²⁰ Debe señalarse que la acción Notify no puede combinarse con una NotificationRequest insertada.

Cuando se recibe una nueva instrucción NotificationRequest en el "estado de notificación", la pasarela DEBERÍA intentar entregar la instrucción Notify pendiente (se señala que ya no se considera pendiente una Notify perdida debido a una desconexión) antes de enviar una respuesta de ejecución satisfactoria de la nueva NotificationRequest. Con este fin se utiliza la funcionalidad de "remolque" del protocolo, ordenando los mensajes (instrucciones y respuestas), de forma que primero se envíe el más antiguo. Los mensajes se enviarán entonces en un solo paquete a quien originó la nueva NotificationRequest, cualesquiera que sean la fuente y la "entidad notificada" para las instrucciones antigua y nueva. Se siguen los siguientes pasos:

- 1) la pasarela crea un mensaje que contiene, en un solo paquete, una repetición de la antigua instrucción Notify pendiente y la respuesta a la nueva instrucción NotificationRequest;
- 2) se hace que el punto extremo salga del "estado notificación" sin esperar la respuesta a la instrucción Notify;
- 3) se conserva una copia de la instrucción Notify pendiente hasta que se reciba una respuesta. Si expira un periodo de temporización, se repite la instrucción, en un paquete que también contendrá una repetición de la respuesta a la NotificationRequest:
 - si se pierde el paquete que contiene la respuesta a la NotificationRequest, el agente de llamada/MGC retransmitirá la NotificationRequest. La pasarela reaccionará a esta repetición retransmitiendo en un solo paquete la instrucción Notify pendiente y la respuesta a la NotificationRequest; este datagrama se enviará a la fuente de la NotificationRequest;
 - se DEBEN entregar en orden las instrucciones Notify de un punto extremo dado. Si la pasarela tiene que transmitir una nueva instrucción Notify antes de que se reciba una respuesta a la anterior instrucción Notify, crea un paquete que transporta en remolque una repetición de la antigua instrucción Notify, una repetición de la respuesta a la última NotificationRequest y la nueva instrucción Notify; este datagrama se enviará a la actual "entidad notificada".

Tras la recepción de una instrucción NotificationRequest, la lista "eventos solicitados" se reemplaza por los nuevos parámetros recibidos. Si la instrucción NotificationRequest se recibió en el "estado de notificación", la lista de "eventos observados" se reinicializa al valor nulo. Lo que se hace después depende del valor del parámetro QuarantineHandling. Este parámetro puede especificar que se descarten los eventos puestos en cuarentena y los eventos observados (que en este caso es una lista vacía), en cuyo caso se descartarán todos los eventos que estén en cuarentena, así como todos los eventos observados. Si el parámetro especifica que se procesen los eventos puestos en cuarentena y los eventos observados, la pasarela comenzará a procesar la lista de eventos puestos en cuarentena y los eventos observados, utilizando la nueva lista de "eventos solicitados". Cuando procesa estos eventos, la pasarela puede encontrar un evento que provoca el envío de una instrucción Notify. En tal caso, la pasarela transmite inmediatamente una instrucción Notify que informa todos los eventos que estaban acumulados en la lista de "eventos observados" hasta ese evento que provoca un envío de instrucción, inclusive, y deja en la memoria intermedia de cuarentena los eventos que no han sido procesados. Después de esto, el punto extremo retorna al "estado notificación".

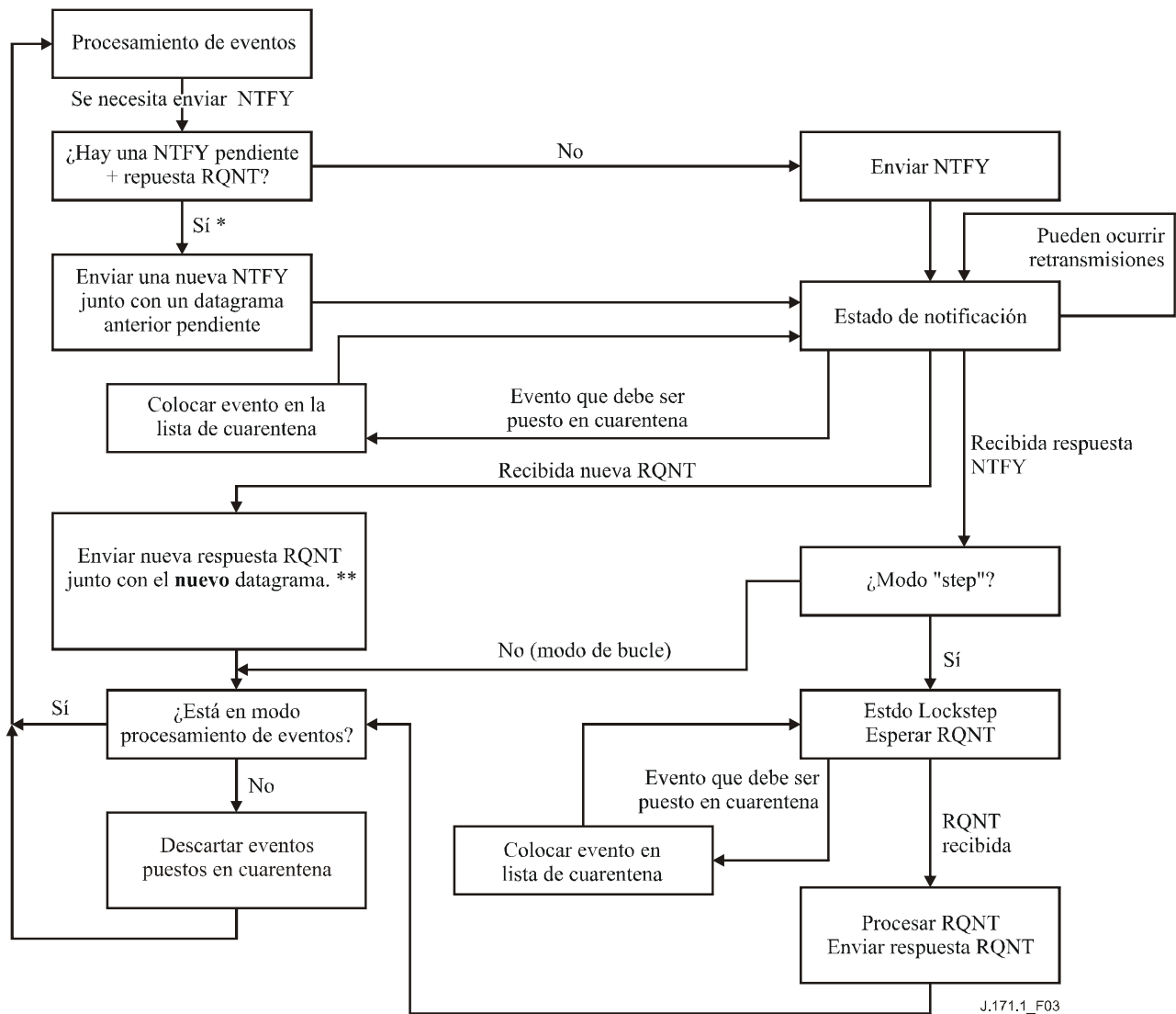
Es posible que al recibir una nueva solicitud de notificación la pasarela haya acumulado eventos conforme a las anteriores solicitudes de notificación, sin que haya detectado algún evento que provoque una notificación. Los parámetros QuarantineHandling determinan el destino de los eventos no notificados aún, de forma similar a lo que sucede con los eventos puestos en cuarentena:

- Si el parámetro QuarantineHandling especifica que no se tengan en cuenta los eventos puestos en cuarentena, sencillamente se reinicializa la lista de eventos observados.
- Si el parámetro QuarantineHandling especifica que se procesen los eventos puestos en cuarentena, se transfiere la lista de eventos observados a la lista de eventos puestos en

cuarentena. La única excepción es la activación de una petición de notificación insertada: la lista de eventos observados no se procesa nuevamente, no se modifica.

Este procedimiento es aplicable a todas las formas de peticiones de notificación, independientemente de que formen parte de una instrucción de tratamiento de la conexión o de que hayan sido proporcionadas por la instrucción NotificationRequest. Las instrucciones de tratamiento de la conexión que no incluyan una petición de notificación no son afectadas por el procedimiento antes mencionado, ni éste es afectado por aquéllas.

La figura 3 ilustra dicho procedimiento, partiendo del supuesto de que todas las transacciones se ejecutan con éxito:



* Se toma este camino de decisión si la pasarela necesita enviar una nueva Notify mientras espera, en el mismo punto extremo, una respuesta a una Notify anterior. Esto podría suceder si se recibe una nueva RQNT en el "estado de notificación", según se describe en el texto descriptivo del diagrama.

** El "nuevo datagrama pendiente" es el datagrama que contiene la instrucción Notify pendiente, posiblemente enviada junto con otra u otras instrucciones Notify y respuestas RQNT que se transmitían en el "estado de notificación" cuando se recibió la nueva RQNT. Es opcional entregar ordenadamente la respuesta RQNT con la o las Notify pendientes; la pasarela también puede enviar la respuesta RQNT en un datagrama diferente. Por otra parte, es de obligatorio cumplimiento el requisito de garantizar la entrega ordenada de instrucciones Notify.

Figura 3/J.171.1 – Algoritmo de cuarentena

Los MGC DEBERÍAN proporcionar la respuesta a un mensaje Notify tramitado satisfactoriamente y a la nueva NotificationRequest en el mismo datagrama utilizando el mecanismo de remolque²¹.

7.4.3.2 Detección explícita

Un elemento esencial del estado de varios puntos extremos es el estado de toma de un circuito. Si bien los eventos de cambio del estado de toma son persistentes en el TGCP, pueden no obstante producirse condiciones de pugna por llegar primero (condiciones de competencia) y discordancia de estados, por ejemplo cuando se toma un circuito mientras el MGC está en trámites de pedir a la pasarela que trate de detectar tomas (la condición de "doble toma" bien conocida en la telefonía; sin embargo, es principalmente un problema de circuitos troncales CAS bidireccionales, que no están soportados en esta versión de la Recomendación).

Para evitar estas condiciones de competencia, la pasarela DEBE comprobar la condición del punto extremo antes de responder a una NotificationRequest. Concretamente, RETORNARÁ un error:

- 1) si se pide a la pasarela que notifique una transición a "toma"²² en un momento en que el circuito está ya tomado (código de error 401 – circuito tomado);
- 2) si se pide a la pasarela que notifique una condición de no-toma²³ en un momento en que el circuito no está tomado (código de error 402 – circuito no tomado).

Además, definiciones de llamadas individuales pueden especificar que una señal sólo funcionará en ciertas condiciones, por ejemplo, la señal de llamada de retorno, de un operador MF, sólo puede ser posible si el circuito ya está tomado. Si tales requisitos previos existen para una señal, la pasarela DEBE retornar el error especificado en la definición de la señal si no se cumplen los requisitos previos.

Debe señalarse que la comprobación relativa a la condición debe efectuarse cuando se recibe la petición de notificación, en tanto que el evento que efectivamente causó la condición actual puede haber sido informado, o ignorado con anterioridad, o puede estar en cuarentena en ese momento.

Las demás variables de estado de la pasarela, como la lista de eventos solicitados o la lista de señales solicitadas, son totalmente reemplazadas después de cada NotificationRequest exitosa, con lo que se evita toda discrepancia de larga duración entre el MGC y la pasarela.

Cuando fracasa una instrucción NotificationRequest, esté o no incluida en una instrucción de tratamiento de la conexión, la pasarela simplemente continuará como si la instrucción no se hubiera recibido y devolverá un error. Al igual que todas las demás transacciones, la NotificationRequest DEBE funcionar como una transacción atómica; por tanto, todo cambio iniciado como resultado de la instrucción DEBE ser invertido.

Cuando el controlador de pasarela de medios recibe una respuesta de error indicando que fracasó una instrucción NotificationRequest, HARÁ lo propio para que se procesen o se descarten los eventos que el punto extremo pudiera haber puesto en cuarentena, y para que el punto extremo regrese a un modo normal de operación en el que se informen los nuevos eventos solicitados a medida que ocurran. Supóngase, por ejemplo, que el controlador de pasarela de medios recibe el código de error "401 – Circuito tomado" como respuesta a una NotificationRequest que solicita detección de la señal "MT/sup". Entonces, el controlador de pasarela de medios debería suponer que la NotificationRequest no afectó en absoluto al punto extremo, que se encuentra en el mismo estado

²¹ Los proveedores que optan por no seguir esta Recomendación deben examinar detenidamente los escenarios de fallo del controlador de pasarela de medios.

²² Por ejemplo, solicitando el evento "sup" en un circuito troncal BLV/OI de terminación MF cuando ya hay una llamada en curso.

²³ Por ejemplo, solicitando el evento "rel" en un circuito troncal de servicios de operador MF sin que haya una llamada en curso.

en que estaba antes de recibir la instrucción. Si antes de recibir la NotificationRequest el punto extremo se encontraba en el estado "lockstep", poniendo los eventos en cuarentena, seguirá en ese estado y poniendo los eventos en cuarentena tras enviar la respuesta de error "401". Para evitar que el punto extremo quede en un estado en el que esté poniendo permanentemente eventos en cuarentena, el controlador de la pasarela de medios debería enviar una nueva NotificationRequest con un conjunto diferente (posiblemente vacío) de eventos solicitados, para que el punto extremo salga del estado "lockstep" y vuelva a funcionar en modo normal, en el que puede informar de nuevos eventos.

Puede producirse otra condición de competencia cuando se emite una instrucción Notify poco tiempo antes de que la pasarela reciba una NotificationRequest. El RequestIdentifier se utiliza para correlacionar instrucciones Notify con instrucciones NotificationRequest (también las peticiones de notificación insertadas en las primitivas de tratamiento de conexión), lo que permite al MGC determinar si la instrucción Notify fue generada antes o después de que la pasarela recibiera la nueva NotificationRequest.

7.4.3.3 Semántica transaccional

A medida que aumentan los posibles tiempos de ejecución de las transacciones, por ejemplo, debido a la reserva de recursos externos, una definición precisa de la semántica transaccional cobra importancia. Concretamente, la cuestión de las condiciones de competencia, y en particular los aspectos relacionados con el estado de toma, deben ser objeto de una definición bien estudiada.

Un punto importante que debe tomarse en consideración es que el estado de toma puede en realidad cambiar entre el instante en que una transacción es iniciada y el instante en que es concluida. Dicho de una manera más general: la conclusión exitosa de una transacción depende de una o más precondiciones, una o más de las cuales pueden cambiar dinámicamente durante la ejecución de la transacción.

La semántica más simple consiste en exigir que se cumplan todas las precondiciones desde el instante en que se inicia hasta el instante en que concluye la transacción. Por tanto, si cualquiera de las precondiciones cambia durante la ejecución de la transacción, ésta DEBE fracasar. Además, a partir del instante mismo en que se inicia la transacción, todos los nuevos eventos que se produzcan se pondrán en cuarentena. Una vez que se sepa el resultado de la transacción, se procesan los eventos que están en cuarentena.

A título de ejemplo, considérese una transacción que incluye una petición del evento "toma". Cuando se inicia la transacción, el circuito está "no tomado", por lo que esta precondición está satisfecha. Si el estado de toma cambia y pasa a "tomado" antes de la conclusión de la transacción, la condición deja de estar satisfecha, lo que entraña el fracaso inmediato de la transacción. El evento "toma" se almacena entonces en la memoria intermedia "de cuarentena", cuyo contenido será procesado ulteriormente.

7.4.3.4 Ordenación de las instrucciones y su tratamiento cuando no están ordenadas

El protocolo MGCP no prescribe que el protocolo de transporte subyacente deba garantizar la secuencia de las instrucciones enviadas a la pasarela o al punto extremo. Esta propiedad contribuye a que las acciones se realicen oportunamente, pero tiene algunos inconvenientes. Por ejemplo:

- Las instrucciones Notify pueden ser demoradas y llegar al MGC después de la transmisión de una nueva instrucción de petición de notificación.
- Si se transmite una nueva instrucción NotificationRequest antes de recibirse una respuesta a una anterior, es posible que la primera instrucción se reciba en segunda posición.

Los MGC y las pasarelas que desean garantizar una operación coherente de los puntos extremos pueden seguir las reglas especificadas:

- 1) Cuando una pasarela trata varios puntos extremos, las instrucciones pertenecientes a diferentes puntos extremos pueden enviarse en paralelo, por ejemplo, siguiendo un modelo en el que cada punto extremo es controlado por su propio proceso o su propio hilo.
- 2) Cuando se crean varias conexiones en un mismo punto extremo, las instrucciones pertenecientes a diferentes conexiones pueden enviarse en paralelo.
- 3) En una conexión dada, normalmente debe haber una sola instrucción pendiente (instrucción de crear o modificar). Sin embargo, en cualquier momento se puede emitir una instrucción DeleteConnection. Por consiguiente, una pasarela puede a veces recibir una instrucción ModifyConnection que es aplicable a una conexión anterior que ha sido suprimida. Se DEBE ignorar tal instrucción y devolver un error (código de error 515 – Identificador de conexión incorrecto).
- 4) En un punto extremo dado, normalmente debe haber una sola instrucción NotificationRequest pendiente en un momento dado cualquiera. El parámetro RequestId se utiliza para correlacionar instrucciones Notify con la NotificationRequest activadora.
- 5) En algunos casos, una instrucción DeleteConnection, con uso implícito o explícito de comodín, aplicable a un grupo de puntos extremos, puede aparecer antes que una instrucción CreateConnection pendiente. El MGC debería suprimir cada una de las conexiones cuya realización esté pendiente al recibir la instrucción DeleteConnection global. Asimismo, no deberían enviarse nuevas instrucciones CreateConnection para puntos extremos denominados mediante el uso de comodines, hasta que se haya recibido una respuesta a la instrucción DeleteConnection en la que se emplearon comodines.
- 6) Cuando se insertan instrucciones unas dentro de otras, se cumplirán los requisitos de secuenciación para todas las instrucciones. Por ejemplo una instrucción CreateConnection dentro de la cual hay una petición de notificación debe cumplir, al mismo tiempo, los requisitos de secuenciación para CreateConnection y NotificationRequest.
- 7) AuditEndpoint y AuditConnection no están sometidas a secuenciación.
- 8) RestartInProgress debe ser la primera instrucción enviada por un punto extremo, de acuerdo con lo definido para el procedimiento de re arranque (véase 7.4.3.5). Toda otra instrucción o respuesta deberá entregarse después de esta instrucción RestartInProgress (está permitido el remolque).
- 9) Cuando múltiples mensajes son remolcados en un solo paquete, los mensajes siempre son procesados en su orden.

De estas reglas, las que especifican un comportamiento de pasarelas son OBLIGATORIAS para las pasarelas de circuitos troncales; en cambio, las pasarelas de circuitos troncales NO HARÁN SUPOSICIONES en cuanto a que los MGC habrán de seguir o no las reglas. En consecuencia, las pasarelas DEBEN siempre responder a las instrucciones, tanto si siguen las mencionadas reglas, como si no las siguen.

A fin de garantizar un funcionamiento coherente, si no se observa alguna o varias de las reglas enunciadas anteriormente, las pasarelas de circuitos troncales DEBERÍAN actuar como se indica a continuación:

- Si se espera una sola instrucción pendiente (ModifyConnection, NotificationRequest), pero la misma instrucción en una transacción se recibe antes de que finalice la ejecución de la anterior, la pasarela DEBERÍA suspender la instrucción anterior. Se incluye el caso en que hubiese una o varias instrucciones encapsuladas. Se RECOMIENDA que se utilice el código de error 407 (Transacción suspendida).

- Si para una instrucción CreateConnection pendiente se recibe una instrucción ModifyConnection, DEBERÍA sencillamente rechazarse la instrucción ModifyConnection. Se RECOMIENDA que se utilice el código de error 400 (Error transitorio). Obsérvese que esta situación constituye un error de programación del controlador de la pasarela de medios.

Cuando la recepción de una nueva instrucción implica que se suspenda la instrucción anterior, esa instrucción anterior DEBERÍA suspenderse siempre, que la nueva instrucción sea o no exitosa. Por ejemplo, si se suspende una instrucción ModifyConnection, como consecuencia de una instrucción DeleteConnection, que también falla como resultado de una NotificationRequest insertada, la instrucción ModifyConnection de todas formas será cancelada.

7.4.3.5 Prevención de los rearranques simultáneos de numerosas pasarelas

Supóngase que se activa simultáneamente un gran número de pasarelas. Si todas ellas iniciaran una transacción RestartInProgress, es muy probable que el MGC resulte inundado, lo que conduciría a pérdidas de mensajes y congestión de la red durante el periodo crítico de restablecimiento del servicio. Para prevenir esta situación se DEBE observar el siguiente comportamiento:

- 1) Cuando se activa una pasarela o se pone nuevamente en servicio la totalidad o un subconjunto de puntos extremos, la pasarela pone en marcha un temporizador de rearranque inicializado a un valor aleatorio, uniformemente distribuido entre 0 y el máximo tiempo de espera (MWD, *maximum waiting delay*) que puede proporcionarse, por ejemplo, 360 segundos (véase más adelante). En la generación de los números aleatorios HAY QUE evitar una sincronización no deseada entre las múltiples pasarelas que vayan a utilizar el mismo algoritmo.
- 2) La pasarela espera entonces, sea la expiración del temporizador, sea la recepción de una instrucción del MGC, sea la detección de una actividad de un circuito local, como por ejemplo una transición a toma en una pasarela de circuitos troncales. Una condición de toma preexistente tiene por consecuencia la generación de un evento de toma.
- 3) Cuando expira el temporizador de rearranque, o se recibe una instrucción, o se detecta una actividad o una condición de toma preexistente, la pasarela inicia el procedimiento de rearranque.

El procedimiento de rearranque supone simplemente que el punto extremo ENVIARÁ una instrucción RestartInProgress al MGC por la que le informe sobre el rearranque y garantice que el primer mensaje (instrucción o respuesta) que el MGC recibe de este punto extremo ES esta instrucción RestartInProgress. Durante cada inicio del procedimiento "desconectado", la instrucción SE AJUSTARÁ a los requisitos normales relacionados con los identificadores de retransmisión y de transacción (véase 7.4.2).

El punto extremo DEBE aprovechar la posibilidad que se le ofrece de transportar en remolque. Por ejemplo, si se produce una actividad de toma de circuito antes de la expiración del temporizador de rearranque, se generará un paquete que contendrá la instrucción RestartInProgress, con una instrucción Notify transportada en remolque para el evento de toma. En caso de que el temporizador de rearranque expire sin que se haya producido ninguna otra actividad, la pasarela envía simplemente un mensaje RestartInProgress.

El procedimiento de rearranque finaliza una vez se reciba una respuesta de éxito. Si se recibe una respuesta de error, el comportamiento subsiguiente depende del código de error de que se trate:

- Si el código de error indica un error transitorio (4xx), HAY QUE iniciar nuevamente (como transacción nueva) el procedimiento de rearranque.
- Si el código de error es 521, entonces se redirige el punto extremo y HAY QUE iniciar nuevamente (como transacción nueva) el procedimiento de rearranque. La respuesta 521 debería haber incluido una NotifiedEntity, que es la "entidad notificada" hacia la que se inicia el rearranque.

- Si el error es algún otro error permanente (5xx), entonces se RECOMIENDA que el punto extremo ya no inicie por su cuenta el procedimiento de re arranque (hasta que se vuelva a reinicializar), a no ser que se especifique otra cosa. Si se recibe una instrucción, el punto extremo iniciará nuevamente el procedimiento de re arranque.

Obsérvese que si RestartInProgress se envía con la respuesta (R) a una instrucción recibida durante el re arranque, entonces la retransmisión de RestartInProgress no requiere que se envíe conjuntamente la respuesta R. No obstante, si se reenvía la respuesta R durante el re arranque del punto extremo sí es necesario que se envíe conjuntamente RestartInProgress a fin de asegurar la entrega ordenada de las dos.

Si el punto extremo pasara al estado "disconnected" durante el procedimiento de re arranque, se ejecutará el procedimiento "desconectado" descrito en 7.4.3.6, y se ENVIARÁ un mensaje "desconectado" durante el procedimiento.

Se espera que a cada punto extremo en una pasarela será posible asignarle un MGC, es decir, una "entidad notificada", al cual se deberá dirigir el mensaje inicial de re arranque. Cuando el conjunto de puntos extremos en una pasarela es gestionado por más de un MGC, el procedimiento antes descrito se efectuará para cada conjunto de puntos extremos gestionados por un MGC dado. La pasarela DEBE aprovechar la posibilidad de utilizar comodines para minimizar el número de mensajes RestartInProgress generados cuando múltiples puntos extremos en una pasarela re arrancan y los puntos extremos son gestionados por el mismo MGC.

El valor de MWD es un parámetro de la configuración que depende del tipo de pasarela. El siguiente razonamiento puede utilizarse para determinar el valor de este retardo en una pasarela.

Generalmente, los MGC están dimensionados para el tratamiento de la carga de tráfico de la hora cargada, durante la cual, en promedio, el 60% de los circuitos troncales están ocupados atendiendo a llamadas que tienen una duración típica de tres minutos. El tratamiento de una llamada comprende por lo general de cinco a seis transacciones entre cada punto extremo y el MGC. Un sencillo cálculo muestra que cabe esperar que, en promedio, el MGC trate de cinco a seis transacciones para cada punto extremo cada cinco minutos o, dicho sea de otra forma, aproximadamente una transacción/punto extremo/minuto. Esto hace pensar que un valor razonable de MWD sería dos minutos/punto extremo. Cuando se fija el valor de MWD para la pasarela, dicho valor debe ser inversamente proporcional al número de puntos extremos que se re arrancan. Por ejemplo, MWD debería fijarse a cinco segundos para una pasarela que trata una línea T1, o a 180 milisegundos para una pasarela que trata una línea T3.

7.4.3.6 Puntos extremos desconectados

Además del procedimiento de re arranque, las pasarelas de circuitos troncales también disponen de un procedimiento "desconectado", que se inicia cuando un punto extremo es "desconectado", como se describe en 7.4.2. Debe señalarse a este respecto que los puntos extremos sólo pueden ser desconectados cuando tratan de comunicar con el MGC. Cuando el punto extremo queda "desconectado" se realizan las siguientes operaciones:

- 1) Se inicializa un temporizador "desconectado" a un valor aleatorio, uniformemente distribuido entre 0 y un periodo de espera inicial "desconectado" ($T_{d_{init}}$), por ejemplo, de 15 segundos. En la generación de los números aleatorios, se DEBE tomar la precaución de evitar una sincronización no deseada entre las múltiples pasarelas que vayan a utilizar el mismo algoritmo.
- 2) La pasarela espera entonces: sea la expiración del temporizador, sea la recepción de una instrucción del MGC, sea la detección de una actividad de circuito local para el punto extremo, como por ejemplo una transición a toma.
- 3) Cuando expira el temporizador "desconectado" o se recibe una instrucción o se detecta una actividad de circuito local, la pasarela INICIARÁ el procedimiento "desconectado" con un

nuevo identificador de transacción para el punto extremo. En el caso de actividad de circuito local, es necesario, además, que haya transcurrido un mínimo periodo de espera "desconectado" ($T_{d_{min}}$), que se puede configurar, desde que la pasarela quedó desconectada o desde la última vez que inició el procedimiento "desconectado" para limitar la velocidad a la que se aplica el procedimiento.

- 4) Si, pese a haberse aplicado el procedimiento "desconectado" al punto extremo, éste continúa desconectado, se duplica el valor del temporizador "desconectado", aunque respetando el periodo de espera máximo "desconectado" ($T_{d_{max}}$) fijado en la configuración, por ejemplo, de 600 segundos, y la pasarela retorna al paso 2).

El procedimiento "desconectado" es similar al procedimiento de rearranque porque simplemente establece que el punto extremo ENVIARÁ una instrucción RestartInProgress al MGC por la que de informa que el punto extremo fue desconectado y además garantiza que el primer mensaje (instrucción o respuesta) que el MGC registrará como procedente del punto extremo SERÁ esta instrucción RestartInProgress. Durante cada inicio del procedimiento "desconectado", la instrucción observará los requisitos normales relacionados con los identificadores de retransmisión y de transacción (véase 7.4.2). Para conseguir esto, el punto extremo APROVECHARÁ la posibilidad que se le ofrece de transportar en remolque.

Cuando el controlador de pasarela de medios recibe un mensaje que contiene un método de rearranque "desconectado", HARÁ lo necesario para que se procesen o se descarten los eventos que el punto extremo pudiera haber puesto en cuarentena, y para que el punto extremo regrese a un modo normal de operación en el que se informen los nuevos eventos solicitados, a medida que éstos ocurran. El controlador de pasarela de medios DEBERÍA enviar una NotificationRequest que contenga un parámetro QuarantineHandling que en este caso será "discard". El controlador de pasarela de medios también podría realizar una o varias de las siguientes acciones: auditar el punto extremo, liberar todas las conexiones del punto extremo o enviar una NotificationRequest en la que solicite al punto extremo que procese eventos puestos en cuarentena (véase 7.4.3.7).

Es posible que el punto extremo desconectado intente enviar una instrucción (además de RestartInProgress) estando desconectado. Como esta acción sólo tendrá éxito una vez que se tenga nuevamente acceso al controlador de pasarela de medios, hay que saber qué se hace con esa instrucción mientras tanto. En un caso extremo, el punto extremo podría descartar inmediatamente la instrucción, pero eso no funcionaría muy bien si el controlador de pasarela de medios estaba en realidad disponible y el punto extremo no había aún finalizado el procedimiento "desconectado" (considérese, por ejemplo, el caso en que una NotificationRequest recién recibida da como resultado la generación inmediata de una instrucción Notify). A fin de evitar esta situación, los puntos extremos desconectados NO DESCARTARÁN sin más las nuevas instrucciones que se han de enviar; esto durante un periodo de $T_{s_{max}}$ segundos posterior a la recepción de una instrucción que no sea de auditoría.

Una forma de satisfacer este requisito es almacenar temporalmente las instrucciones a enviar en una memoria intermedia, pero el punto extremo debe garantizar que:

- no creará una cola larga de instrucciones a enviar;
- no inundará el controlador de pasarela de medios reconectado enviando rápidamente demasiadas instrucciones.

Se considera una práctica segura almacenar las instrucciones durante $T_{s_{max}}$ segundos y, una vez se vuelva a conectar el punto extremo, limitar la velocidad de envío de instrucciones almacenadas a una instrucción pendiente por punto extremo. Si no se conecta el punto extremo dentro de $T_{s_{max}}$ segundos, pero se da inicio a un procedimiento "desconectado" dentro de los $T_{s_{max}}$ segundos, el punto extremo PUEDE enviar la o las instrucciones almacenadas junto con esa RestartInProgress. Cuando se ha enviado una instrucción, la retransmisión tiene que cesar $T_{s_{max}}$ segundos después del

envío inicial, conforme se describe en 7.4.2, independientemente del tratamiento anterior (almacenada o enviada remolcada).

El proceso desconectado finaliza con la recepción de una respuesta. Las respuestas de error se tratan de forma similar que el procedimiento de rearmado (cláusula 7.4.3.5). Si se ha de iniciar nuevamente el procedimiento "desconectado" tras una respuesta de error, siguen siendo válidas las consideraciones relativas al temporizador limitante de velocidad, especificadas anteriormente.

Se señala también que si se envía la RestartInProgress junto con la respuesta (R) a una instrucción recibida cuando estaba desconectado, la retransmisión de la instrucción RestartInProgress no requiere que se envíe conjuntamente la respuesta R. No obstante, si el punto extremo está desconectado, el reenvío de la respuesta R sí exige que la instrucción RestartInProgress también se envíe conjuntamente para garantizar la entrega ordenada de las dos.

Si ya se estaba ejecutando el procedimiento desconectado cuando se recibe una instrucción, SE FINALIZARÁ el procedimiento desconectado existente y se INICIARÁ un nuevo procedimiento. Esto con el fin de soportar un posible redireccionamiento del controlador de pasarela de medios.

Obsérvese también que un punto extremo desconectado no está "fuera de servicio". "Desconectado" no es un estado que indique la disponibilidad de servicio del punto extremo, sino la incapacidad de la pasarela de comunicarse con su MGC.

En esta Recomendación, intencionalmente, no se ha especificado ningún comportamiento adicional de un punto extremo desconectado. Los proveedores PUEDEN, por ejemplo, elegir entre proporcionar silencio, reproducir un tono de volver a llamar o incluso permitir que un fichero wav telecargado sea reproducido en puntos extremos afectados.

El valor por defecto para $T_{d_{init}}$ es 15 segundos, el valor por defecto para $T_{d_{min}}$ es 15 segundos y el valor por defecto para $T_{d_{max}}$ es 600 segundos.

7.4.3.7 Tratamiento por el agente de llamada de los puntos extremos desconectados

Cuando un punto extremo se encuentra en estado "desconectado" es posible que acumule muchos eventos en la memoria intermedia de cuarentena. Asimismo, puede ocurrir que un punto extremo "desconectado" suprima autónomamente conexiones establecidas (por ejemplo, cuando se reinicia la pasarela). Por ende, al restablecerse la conexión entre uno de tales puntos y su agente de llamada, este último ESTARÁ preparado para:

- La gran cantidad de mensajes Notify que podría generar el punto extremo si se procesan todos los eventos de la lista de cuarentena.
- La recepción de eventos anteriores/desactualizados, reportados por el punto extremo, que ya no son significativos. La memoria intermedia de cuarentena es del tipo cola FIFO (*first-in-first-out*), en la que primero se procesan los eventos más antiguos y se notifican (si así se hubiere solicitado) al agente de llamada. Cuando un evento ha sido reemplazado por nuevos eventos, es posible que la acción que realiza el agente de llamada al recibirlo ya no tenga sentido (por ejemplo, un evento "MT/sup" sería irrelevante si el punto extremo ya estuviera colgado).
- Una incoherencia entre estados de conexiones: por ejemplo, el agente de llamada ha registrado que el punto extremo tiene una o más conexiones, pero en realidad éste no las tiene.

Los agentes de llamada pueden emplear cualquier mecanismo soportado por el protocolo para ocuparse de los puntos mencionados. Una manera de hacerlo es la siguiente:

- 1) Se define una nueva variable booleana denominada "disconnect-event-sync", cuyo valor ha de ser atribuido por el agente de llamada para cada uno de sus puntos extremos. Cuando es verdadera, indica que se acaba de restaurar la conectividad con un punto extremo "desconectado", pero que aún no se ha logrado la sincronización de evento/señal.

(Obsérvese que esta variable se define para describir el comportamiento del agente de llamada y no implica la utilización de ninguna implementación determinada. Esta variable no es visible desde el exterior.)

a) Tan pronto el agente de llamada se entera de que un punto extremo está "desconectado", atribuye el valor verdadero a la variable "disconnect-event-sync". En el procedimiento "desconectado", el agente de llamada recibe un mensaje RestartInProgress "desconectado" que le informa que el punto extremo está desconectado. Cuando el punto extremo recibe un acuse de recepción positivo al RSIP "desconectado", termina el procedimiento "desconectado". El punto extremo puede entonces enviar una instrucción Notify que fue almacenada en memoria intermedia mientras el punto estuvo desconectado o notificar un evento Notify-triggering en la lista de cuarentena si el punto extremo estaba en estado "notification" y modo "loop" mientras estaba desconectado.

- Si el punto extremo funciona en el modo "step", la respuesta a Notify no permitirá por sí misma que se generen otros mensajes Notify (para ello haría falta una NotificationRequest).
- No obstante, si el punto extremo funciona en el modo "loop", la respuesta a Notify permitirá que se generen otros mensajes Notify. Como ya se dijo, en algunos casos esto puede resultar inconveniente, pues los eventos sobre los que se informa pueden ser anteriores y muchos pueden estar en cuarentena, y siendo así habría una gran cantidad de mensajes Notify y los mensajes NotificationRequest subsiguientes, basados en información desactualizada.

b) Mientras que el valor de "disconnect-event-sync" para un punto extremo sea verdadero, conviene que el agente de llamada haga lo necesario para descartar o procesar de manera controlada y ordenada los eventos de la lista de cuarentena (que podría ser bastante larga). Esto se puede lograr de varias formas, a saber:

- El agente de llamada puede enviar una sola NotificationRequest en la que se especifique que se han de descartar todos los eventos puestos en cuarentena. Al recibir un acuse de recibo positivo a esta instrucción, o una instrucción Notify cuyo RequestIdentifier sea el mismo, el agente de llamada debería calificar como falso la variable "disconnect-event-sync", tras lo cual se reiniciaría el procesamiento normal del evento en el punto extremo. La desventaja de este método es que con él se suprimen todos los eventos acumulados, sin importar cuántos sean, lo que en algunos casos puede provocar una interrupción innecesaria del servicio. Para obviar este inconveniente habrá que ampliar el protocolo.
- El agente de llamada puede enviar una NotificationRequest donde se especifique que se han de procesar los eventos puestos en cuarentena. Cuando el punto extremo esté funcionando en el modo "step", sólo señalará un evento causante de notificación para cada instrucción NotificationRequest recibida, mientras que en el modo "loop" puede señalar varios eventos con una sola NotificationRequest.

Debido a que es posible que la información transportada por los eventos notificados haya perdido entretanto su pertinencia, no es recomendable que el agente de llamada los procese sin más (por ejemplo, al ser notificado de un evento que origina la acción "MT/sup", el agente de llamada no debería enviar automáticamente una NotificationRequest solicitando notificación de las cifras "MT/inf"). El agente de llamada tiene que sincronizar su información de estado interno con el estado real del punto extremo. Puesto que un punto extremo procesa las señales y eventos de una manera manifiestamente vinculada al estado de toma del circuito, el agente de llamada puede conocer el estado de toma basándose en la respuesta a una NotificationRequest. Por ejemplo, una respuesta "402 – Circuito no

tomado" a una instrucción NotificationRequest que solicitaba la detección de "MT/inf" implica que el circuito troncal MT no está tomado actualmente. El agente de llamada puede omitir los eventos notificados que sean incompatibles con el estado actual colgado/descolgado (por ejemplo, no tener en cuenta el informe de cifras "MT/inf" si la troncal no está tomada).

Tan pronto se hayan procesado todos los eventos de la lista de cuarentena, el agente de llamada debería atribuir el valor falso al indicador "disconnect-event-sync". El agente de llamada podrá suponer sin mayor riesgo que todos los eventos en cuarentena han sido procesados cuando haya transcurrido un tiempo $T_{t_{hist}}$ desde la última vez que solicitó al punto extremo procesar el evento siguiente (es decir, si ha expirado un temporizador $T_{t_{hist}}$ desde que el agente de llamada envió la respuesta a la Notify anterior, en el modo "loop", o desde que el agente de llamada recibió la última respuesta positiva a una NotificationRequest, en el modo "step").

- 2) Cuando un punto extremo queda en estado "desconectado", las conexiones que se han creado en él no deberían resultar afectadas. Sin embargo, es posible que se suprima una conexión porque el punto extremo no puede mantenerla, lo que provoca el envío de una instrucción DeleteConnection al agente de llamada. Cuando el punto extremo está desconectado, esta instrucción no llegará hasta el agente de llamada y éste no se enterará de la supresión de la conexión. Por consiguiente, es recomendable que cuando un agente de llamada sepa que se ha desconectado un punto extremo, realice una auditoría de las conexiones presentes en él.

7.5 Códigos de retorno y códigos de error

En el MGCP, para toda instrucción se recibe una respuesta. La respuesta contiene un código de retorno que indica el status de la instrucción. El código de retorno es un número entero para el que se han definido cinco gamas de valores:

- el valor 000 indica un acuse de recibo de respuesta²⁴,
- los valores entre 100 y 199 indican una respuesta provisional,
- los valores entre 200 y 299 indican una ejecución satisfactoria,
- los valores entre 400 y 499 indican un error transitorio,
- los valores entre 500 y 599 indican un error permanente.

Los valores que han sido definidos se indican en el cuadro 3:

Cuadro 3/J.171.1 – Códigos de retorno

Código	Significado
000	Acuse de recibo de respuesta.
100	La transacción se está ejecutando en este momento. Después se enviará el mensaje de finalización propiamente dicho.
200	La transacción solicitada se ejecutó normalmente.
250	Las conexiones fueron suprimidas.
400	La transacción no pudo ejecutarse debido a un error transitorio.
401	El teléfono ya está descolgado o el circuito ya está tomado.
402	El teléfono ya está colgado o el circuito no está tomado.

²⁴ El acuse de recibo de respuesta se utiliza para respuestas provisionales (véase 8.8).

Cuadro 3/J.171.1 – Códigos de retorno

Código	Significado
407	Transacción suspendida. Una acción externa suspendió la transacción, por ejemplo, una instrucción ModifyConnection interrumpida por otra DeleteConnection.
500	La transacción no pudo ejecutarse porque el punto extremo es desconocido.
501	La transacción no pudo ejecutarse porque el punto extremo no está listo.
502	La transacción no pudo ejecutarse porque el punto extremo no tiene recursos suficientes.
503	No soporta completamente el comodín "All of". Aunque la pasarela no soporta completamente dichos comodines "all of", la transacción contenía uno de ellos. Obsérvese que actualmente éste sólo puede ser el caso para las NotificationRequests no vacías.
505	No soporta el RemoteConnectionDescriptor. DEBERÍA emplearse cuando el sistema no soporta uno o varios valores o parámetros obligatorios en el RemoteConnectionDescriptor.
506	No se satisfacen al mismo tiempo LocalConnectionOptions y RemoteConnectionDescriptor. DEBERÍA emplearse cuando LocalConnectionOptions y RemoteConnectionDescriptor contengan uno o varios valores o parámetros obligatorios discrepantes entre sí y/o que no pueden ser soportados al mismo tiempo (salvo para el caso del fallo de negociación de códec – véase el código de error 534).
510	La transacción no pudo ejecutarse porque se detectó un error de protocolo.
511	La transacción no pudo ejecutarse porque la instrucción contenía una extensión no reconocida.
512	La transacción no pudo ejecutarse porque la pasarela no está equipada para detectar uno de los eventos solicitados.
513	La transacción no pudo ejecutarse porque la pasarela no está equipada para generar una de las señales solicitadas.
514	La transacción no pudo ejecutarse porque la pasarela no puede enviar el anuncio especificado.
515	La transacción hace referencia a un identificador de conexión incorrecto (puede haber sido ya suprimido).
516	La transacción hace referencia a un identificador de llamada desconocido.
517	Modo no soportado o no válido.
518	Lote no soportado o desconocido.
519	El punto extremo no tiene un mapa de dígitos.
520	La transacción no pudo ejecutarse porque el punto extremo está "rearrancando".
521	Punto extremo redireccionado a otro MGC.
522	No hay tal evento o señal.
523	Acción desconocida o combinación ilícita de acciones.
524	Inconsistencia interna en LocalConnectionOptions.
525	Extensión desconocida en LocalConnectionOptions.
526	Anchura de banda insuficiente.
527	Falta RemoteConnectionDescriptor.
528	Versión de protocolo incompatible.
529	Fallo de soporte físico interno.
532	Valor o valores no soportados en LocalConnectionOptions.
533	Respuesta demasiado grande.

Cuadro 3/J.171.1 – Códigos de retorno

Código	Significado
534	Fallo de negociación de códec.
538	Error de parámetro evento/señal (por ejemplo, parámetro faltante, erróneo, no soportado, desconocido, etc.).

7.6 Códigos de motivo

Cuando la pasarela suprime una conexión, utiliza códigos de motivo para informar al MGC sobre el motivo por el que se suprime la conexión. El código de motivo es un número entero; se han definido los siguientes valores en el cuadro 4:

Cuadro 4/J.171.1 – Códigos de motivo

Código	Significado
900	Mal funcionamiento de punto extremo.
901	Punto extremo puesto fuera de servicio.
902	Pérdida de conectividad de capa inferior (por ejemplo, pérdida de la sincronización en el sentido de ida).

7.7 Utilización de los descriptores de opciones de conexión local y de conexión

La secuencia normal de establecimiento de una conexión bidireccional tiene por lo menos tres etapas, a saber:

- 1) El MGC o el CA solicitan a la primera pasarela que "cree una conexión" en un punto extremo. La pasarela atribuye los recursos para dicha conexión y responde a la instrucción mediante una "descripción de sesión" (es su LocalConnectionDescriptor). La descripción de sesión contiene la información necesaria para que otra parte envíe paquetes hacia la nueva conexión.
- 2) Luego, el MGC o el CA solicitan a la segunda pasarela que "cree una conexión" en un punto extremo. Con la instrucción se incluye la "descripción de sesión" proporcionada por la primera pasarela (que ahora se denomina RemoteConnectionDescriptor). La pasarela atribuye los recursos para dicha conexión y responde a la instrucción comunicando su propia "descripción de sesión" (LocalConnectionDescriptor).
- 3) El MGC o el CA utilizan una instrucción "modificar conexión" para suministrar esta segunda "descripción de sesión" (que ahora se denomina RemoteConnectionDescriptor) al primer punto extremo, tras lo cual queda establecida la comunicación en ambos sentidos.

Cuando el MGC o el CA envían una instrucción para crear o modificar conexión, hay tres parámetros que determinan los medios soportados por dicha conexión:

- LocalConnectionOptions: suministrado por el MGC o el CA, permite comprobar cuáles son los parámetros de medios empleados por la pasarela para la conexión. Si se incluye, la pasarela debe mantener dichos parámetros de medios hasta que se suprima la conexión o se reciba una instrucción ModifyConnection.
- RemoteConnectionDescriptor: suministrado por el MGC o el CA, permite conocer los parámetros de medios soportados por el otro extremo de la conexión. Si se incluye, la pasarela debe mantener dichos parámetros de medios hasta que se suprima la conexión o se reciba una instrucción ModifyConnection.

- LocalConnectionDescriptor: suministrado por la pasarela al MGC o al CA, sirve para informar cuáles son los parámetros de medios que soporta la pasarela para la conexión. Si se incluye, la pasarela debe mantener dichos parámetros de medios hasta que emita un nuevo LocalConnectionDescriptor o se suprima la conexión. Además de los parámetros de medios atribuidos a la conexión, la pasarela también puede anunciar en el LocalConnectionDescriptor otras capacidades que soporta. Obsérvese que dichas capacidades NO PUEDEN ir en la línea m = del SDP. La pasarela puede anunciar todas las capacidades que soporta, sin importar cuáles hayan sido los parámetros LCO o RCD que se reciben del CA, ni los parámetros de medios correspondientes a la conexión.

La selección del códec y del periodo de paquetización HA DE efectuarse solamente, como se describe en esta cláusula, si se cumple una de las dos condiciones siguientes:

- a) la pasarela recibe una CRCX; o
- b) la pasarela recibe una MDCX y está presente cualquiera de los parámetros siguientes:
 - método de codificación (a: en LocalConnectionOptions);
 - periodo de paquetización (p: en LocalConnectionOptions);
 - periodo de paquetización múltiple (mp: en LocalConnectionOptions);
 - RemoteConnectionDescriptor.

Además, este proceso de selección del códec y del periodo de paquetización sólo UTILIZARÁ información presente en la petición de conexión, sin tener en cuenta ninguno de los valores posiblemente comunicados en peticiones anteriores de conexión. Así, por ejemplo, si una pasarela recibió una MDCX con todos los parámetros LCO necesarios, pero sin un RemoteConnectionDescriptor, negociará como si no hubiera recibido un RemoteConnectionDescriptor para dicha conexión. Asimismo, si se omiten todos los parámetros arriba mencionados en una instrucción MDCX, no se modificarán los códecs ni los periodos actuales de paquetización negociados.

Al establecer cuál(es) códec(s) y periodo(s) de paquetización se deben proporcionar en LocalConnectionDescriptor, hay tres listas de códecs y periodos de paquetización que la pasarela debe tener en cuenta:

- Una lista de códecs y periodos de paquetización suministrada en LocalConnectionOptions. Un códec es permitido por LocalConnectionOptions si cumple las condiciones especificadas por los campos método de codificación, periodo de paquetización y periodos de paquetización múltiple. La omisión de uno de estos campos no incide de ninguna manera en la definición de códecs válidos.
- Una lista de códecs y periodos de paquetización en RemoteConnectionDescriptor.
- Una lista interna de códecs y periodos de paquetización que la pasarela puede soportar para la conexión. Es posible que una pasarela soporte uno o varios códecs y periodos de paquetización para determinada conexión.

Descripción del proceso de selección de códec (incluidos todos los parámetros de medios que vengan al caso):

- 1) Se forma una lista aprobada de periodos de paquetización/códecs compuesta por aquellos periodos de paquetización/códecs de la lista interna que también sean válidos conforme a LocalConnectionOptions. De no haber LocalConnectionOptions, la lista aprobada contiene la lista interna. Si existe el LocalConnectionOptions mas no el parámetro códecs, LocalConnectionOptions permite implícitamente todos los códecs de la lista interna, siempre y cuando sean compatibles con los periodos de paquetización especificados. De igual manera, si se proporciona LocalConnectionOptions pero se omite el o los periodos de paquetización, LocalConnectionOptions contiene implícitamente el conjunto de periodos de paquetización soportado por la lista interna.

- 2) Si la lista aprobada de periodos de paquetización/códecs está vacía, ha ocurrido un fallo de negociación de códec y se genera una respuesta de error (se recomienda el código de error 534 – Fallo de negociación de códec).
- 3) En otros casos, se crea una lista negociada de periodos de paquetización/códecs compuesta por aquellos periodos de paquetización/códecs de la lista aprobada que también sean válidos conforme a RemoteConnectionDescriptor. Si no se suministró RemoteConnectionDescriptor, entonces la lista negociada contiene la lista aprobada. Si el RemoteConnectionDescriptor no contiene líneas de tren de medios, ha ocurrido un fallo de negociación de códec y se genera una respuesta de error (se recomienda el código de error 534 – Fallo de negociación de códec). Si el RemoteConnectionDescriptor contiene varios trenes de medios, la MG DEBERÍA aceptar solamente uno de ellos y rechazar los otros atribuyéndole el valor cero al puerto correspondiente en el LocalConnectionDescriptor. De haber un RemoteConnectionDescriptor sin ningún periodo de paquetización, la lista negociada de periodos de paquetización contiene el conjunto de dichos periodos que figura en la lista aprobada. La MG TENDRÁ que escoger valores por defecto dentro de límites razonables (conforme a la RFC 2327) si se omite explícitamente el periodo de paquetización tanto en LocalConnectionOptions como en RemoteConnectionDescriptor.
- 4) Si la lista negociada de periodos de paquetización/códecs está vacía, ha ocurrido un fallo de negociación de códec y se genera una respuesta de error (se recomienda el código de error 534 – Fallo de negociación de códec).
- 5) De lo contrario, ha tenido éxito la negociación de códec y se devuelve en el LocalConnectionDescriptor la lista negociada de periodos de paquetización/códecs.

Obsérvese que el intervalo de paquetización para T.38 se escoge empleando el mismo procedimiento que se utiliza para los códecs de audio, como se describió *supra*.

Cuando una pasarela soporta más de un códec por punto extremo, se presentan dos casos de selección del número de códecs para la conexión en cuestión:

- 1) La pasarela soporta varios códecs y puede pasar de uno a otro en tiempo real. La pasarela devuelve todos los códecs negociados en la línea de trenes de medios del SDP. Varios códecs en la línea m= implican que el dispositivo debe estar listo para recibir paquetes de medios de cualquiera de los códecs negociados. Asimismo, es posible que la pasarela envíe paquetes de medios de cualquiera de los códecs negociados y cambie de uno a otro cuando haga falta.
- 2) La pasarela soporta uno o varios códecs pero no puede pasar de uno a otro en tiempo real. En tal caso, la pasarela negocia y devuelve solamente un códec en la línea de trenes de medios del SDP, (como alternativa, la pasarela también puede poner otros códecs soportados en el atributo 'X-pc-codecs' del SDP). Con este método, el MGC puede iniciar un cambio de códec.

7.7.1 Negociación conforme a RFC 2833

La lista interna de códecs soportados TIENE que incluir el códec evento telefónico (*telephone-event*) con eventos del 0 al 15, garantizándose así que se empleará la retransmisión con DTMF conforme a la RFC 2833 en esa conexión, cuando lo autoricen los parámetros LCO (mediante la inclusión de un parámetro a: vacío) y lo permita el RCD.

Ejemplo de parámetros LCO que autorizan la transmisión con DTMF conforme a la RFC 2833:

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

El códec de evento telefónico NO SERÁ el único suministrado en LCO por el MGC. Si el punto extremo recibe parámetros LCO que contienen solamente dicho códec, tiene que responder con el

código de error 524 – Inconsistencia interna en LocalConnectionOptions. Si la lista aprobada de códecs, descrita en 7.7, contiene únicamente el códec telephone-event, el punto extremo tiene que responder con el código de error 534 – Fallo de negociación de códec. De igual manera, si la lista negociada de códecs, descrita en 7.7, contiene únicamente el códec de evento telefónico, el punto extremo tiene que responder con el código de error 534.

Si se utiliza el parámetro LCO periodo de paquetización para la conexión, el punto extremo TIENE que emplear dicha velocidad de paquetización para los paquetes transmitidos con DTMF. Si se utiliza el parámetro LCO periodo de paquetización múltiple, el MGC TIENE que señalar con un guión la velocidad de paquetización del códec de evento telefónico. Si el punto extremo recibe parámetros LCO con un parámetro periodo de paquetización múltiple cuya velocidad para ese códec no se indica con un guión, tiene que devolver el código de error 524 – Inconsistencia de LCO. Cuando un punto extremo devuelve un LCD que incluye la capacidad de recibir el códec de evento telefónico, TIENE que señalar con un guión la velocidad de paquetización en el atributo mptime del SDP.

Ejemplo de LCD que anuncia el soporte de cifras DTMF conforme a la RFC 2833:

```
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=rtptime:105 telephone-event/8000/1
```

Véase la especificación de códecs audio-vídeo IPCablecom, J.161 que describe la utilización de la DTMF conforme a RFC 2833.

7.7.2 Negociación de IP y puerto del lado distante de la conexión

El IP y el puerto se indican en el descriptor de conexión distante (Remote Connection Descriptor). Una vez obtenidos, en una instrucción de manejo de conexión tramitada satisfactoriamente (por ejemplo, una instrucción Modify Connection), el punto extremo tiene que seguir utilizándolos hasta que se suministre un nuevo Remote Connection Descriptor que especifique un nuevo IP o puerto distante, o se suprima la conexión. Obsérvese que la recepción de una instrucción Modify Connection sin un Remote Connection Descriptor que pueda empezar la negociación de códec no invalida los actuales IP y puerto distantes, aun en el caso en que se modifique el medio de audio a imagen o viceversa.

8 Protocolo de control de pasarela de medios

El MGCP implementa la interfaz de control de pasarela de medios como un conjunto de transacciones. Las transacciones se componen de una instrucción y una respuesta obligatoria. Hay ocho tipos de instrucciones:

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

Las instrucciones de los cuatro primeros tipos las envía el MGC a una pasarela. La instrucción Notify la envía la pasarela al MGC. La pasarela puede también enviar una instrucción DeleteConnection como se indica en 7.3.6. El MGC puede enviar cualquiera de las dos instrucciones de auditoría a la pasarela y, por último, la pasarela puede enviar una instrucción RestartInProgress al MGC.

8.1 Descripción general

Todas las instrucciones constan de un encabezamiento de instrucción, que en algunas instrucciones puede ir seguido de una descripción de sesión.

Todas las respuestas constan de un encabezamiento de respuesta, que en algunas respuestas puede ir seguido de una descripción de sesión.

Los encabezamientos y las descripciones de sesión se codifican como un conjunto de líneas de texto, separados por un carácter de retorno del carro y un carácter de nueva línea (o, facultativamente, un solo carácter de nueva línea). Entre el encabezamiento y la descripción de sesión hay una línea vacía.

El MGCP utiliza un identificador de transacción con un valor comprendido entre 1 y 999999999 para correlacionar instrucciones y respuestas. El identificador de transacción se codifica como un componente del encabezamiento de instrucción y se repite como un componente del encabezamiento de respuesta.

8.2 Encabezamiento de instrucción

El encabezamiento de instrucción se compone de:

- una línea de instrucción que identifica la acción o verbo solicitados, el identificador de transacción, el punto extremo al que se dirige la acción solicitada y la versión de protocolo MGCP;
- un conjunto de líneas de parámetro constituidas por un nombre de parámetro seguido de un valor de parámetro.

A menos que se indique otra cosa, o que en otras normas a que se hace referencia se tomen disposiciones diferentes, los componentes del encabezamiento de instrucción son insensibles a la escritura en mayúscula o minúscula. Esto es aplicable a los verbos, así como a los parámetros y valores, y en todas las comparaciones se considerará que son iguales las cadenas todo en mayúsculas o en minúsculas, o que combinan mayúsculas y minúsculas.

8.2.1 Línea de instrucción

La línea de instrucción se compone de lo siguiente:

- el nombre del verbo solicitado;
- la identificación de la transacción;
- el nombre del punto extremo o puntos extremos que deben ejecutar la instrucción (en notificaciones o rearranques, el nombre del punto extremo o puntos extremos que expiden la instrucción);
- la versión de protocolo.

Estos cuatro elementos se codifican como cadenas de caracteres ASCII imprimibles separados por espacios, es decir, por los caracteres ASCII espacio (0x20) o tabulación (0x09). Las pasarelas de circuitos troncales DEBERÍAN utilizar sólo un separador espacio ASCII, pero podrán efectuar el análisis gramatical de mensajes con caracteres de espacio blanco adicionales.

8.2.1.1 Codificación del verbo solicitado

Los verbos solicitados se codifican como códigos ASCII de cuatro letras mayúsculas y/o minúsculas (las comparaciones deben ser insensibles a la escritura en mayúscula o minúscula), como se define en el siguiente cuadro 5:

Cuadro 5/J.171.1 – Códigos del verbo solicitado

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

En futuras versiones del protocolo podrán definirse nuevos verbos. Puede ser necesario, para fines experimentales, utilizar nuevos verbos antes de que sean aprobados e incluidos en una versión publicada de este protocolo. Los verbos experimentales deberían identificarse por un código de cuatro letras que comience por la letra X (por ejemplo, XPER).

Si una pasarela recibe una instrucción con un verbo experimental que no soporta, devolverá un código de error 511 – Extensión no reconocida).

8.2.1.2 Identificadores de transacción

Se utilizan identificadores de transacción para correlacionar instrucciones y respuestas.

Una pasarela de circuitos troncales soporta dos espacios de nombre de identificador de transacción distintos:

- un espacio de nombre de identificador de transacción para enviar transacciones, y
- un espacio de nombre de identificador de transacción para recibir transacciones.

Como mínimo, los identificadores de transacción para instrucciones enviadas a una determinada pasarela de circuitos troncales SERÁN únicos para el tiempo de actividad máximo de las transacciones dentro de la serie de MGC que controlan esa pasarela de circuitos troncales (véase 8.5). Por tanto, cualquiera que sea el MGC emisor, las pasarelas de circuitos troncales siempre pueden detectar transacciones duplicadas, simplemente examinando el identificador de transacción. Sin embargo, la coordinación de estos identificadores de transacción entre los MGC está fuera del ámbito de esta Recomendación.

Los identificadores de transacción para todas las instrucciones enviadas desde una determinada pasarela de circuitos troncales serán únicos durante el tiempo de actividad máximo de las transacciones (véase 8.5) cualquiera que sea el MGC a que se envía la instrucción. Por tanto, un MGC siempre puede detectar una transacción duplicada procedente de una pasarela de circuitos troncales por la combinación del nombre de dominio del punto extremo y el identificador de transacción. La pasarela a su vez siempre puede detectar un acuse de recibo de respuesta duplicado examinando el identificador o los identificadores de transacción.

El identificador de transacción se codifica como una cadena de nueve dígitos decimales como máximo. En las líneas de instrucción, este identificador sigue inmediatamente a la codificación del verbo.

Los identificadores de transacción tienen valores comprendidos entre 1 y 999999999. Los identificadores de transacción NO DEBERÍAN utilizar ceros iniciales. La igualdad la determina el valor numérico y se ignoran los ceros iniciales. Una entidad MGCP NO REUTILIZARÁ un identificador de transacción dentro de los tres minutos que siguen a la finalización de la instrucción precedente en la que se utilizó el identificador.

8.2.1.3 Codificación de los nombre de punto extremo, controlador de pasarela de medios y entidad notificada

Los nombres de punto extremo y los nombres de MGC se codifican como direcciones de correo electrónico, según se define en IETF RFC 2821. En estas direcciones, el nombre de dominio identifica el sistema en el que el punto extremo está vinculado, mientras que el lado izquierdo identifica un punto extremo concreto de ese sistema. Ambos componentes DEBEN ser insensibles a la escritura en mayúscula o minúscula.

Son ejemplos de tales nombres:

ds/ds1-3/2@TGCP2.whatever.net	Segundo circuito en el tercer DS1 de la pasarela de circuitos troncales TGCP2 en la red "whatever".
MGC@mgc.whatever.net	Controlador de pasarela de medios para la red "whatever".

El nombre de entidades notificadas se expresa en la misma sintaxis, con la posible adición de un número de puerto, como en:

MGC@mgc.whatever.net:5234

Si se omite el número de puerto, se utiliza el puerto MGCP por defecto (puerto 2727, salvo que se haya dispuesto otra cosa). En 7.1.1, se da una información más detallada sobre los nombres de punto extremo.

8.2.1.4 Codificación de la versión de protocolo

El código de la versión de protocolo es la palabra clave "MGCP" seguida de un espacio blanco y el número de versión, que a su vez va seguido del nombre de perfil "TGCP" y de un número de versión de perfil. Los números de versión se componen de un número de versión principal, un punto y un número de versión complementario. El número de versión principal y el número de versión complementario se codifican como números decimales. El número de versión de perfil definido por esta Recomendación es 1.0.

La versión de protocolo para esta Recomendación DEBE codificarse como:

MGCP 1.0 TGCP 1.0

La porción "TGCP 1.0" indica que este es el perfil TGCP 1.0 de MGCP 1.0.

Si una entidad recibe una instrucción con una versión de protocolo que no soporta, responderá con un error (código de error 528 – Versión de protocolo incompatible).

8.2.2 Líneas de parámetro

Las líneas de parámetro se componen de un nombre de parámetro, que en la mayoría de los casos está formado por un solo carácter letra mayúscula, seguido de un carácter dos puntos (":"), un espacio blanco y el valor de parámetro. Los nombres y los valores de parámetros son insensibles a la escritura en mayúsculas o minúsculas. En el cuadro 6 se definen los parámetros que pueden estar presentes en instrucciones:

Cuadro 6/J.171.1 – Parámetros de instrucción

Nombre del parámetro	Código	Valor del parámetro
ResponseAck (nota)	K	Véase la descripción.
CallId	C	Cadena hexadecimal, NO EXCEDERÁ de 32 caracteres.
ConnectionId	I	Cadena hexadecimal, NO EXCEDERÁ de 32 caracteres.
NotifiedEntity	N	Un identificador, en formato IETF RFC 821, compuesto de una cadena cualquiera y del nombre de dominio de la entidad solicitante, posiblemente completado por un número de puerto, como en: Call-agent@ca.whatever.net:5234
RequestIdentifier	X	La cadena hexadecimal RequestIdentifier NO EXCEDERÁ de 32 caracteres.
LocalConnectionOptions	L	Véase la descripción.
ConnectionMode	M	Véase la descripción.
RequestedEvents	R	Véase la descripción.
SignalRequests	S	Véase la descripción.
ObservedEvents	O	Véase la descripción.
ConnectionParameters	P	Véase la descripción.
ReasonCode	E	Véase la descripción.
SpecificEndPointId	Z	Un identificador, en formato IETF RFC 821, compuesto de una cadena cualquiera, seguido facultativamente de un carácter "@" y el nombre de dominio de la pasarela de circuitos troncales a la que está vinculado este punto extremo.
MaxEndPointIds	ZM	Cadena decimal, NO EXCEDERÁ de 16 caracteres.
NumEndpoints	ZN	Cadena decimal, NO EXCEDERÁ de 16 caracteres.
RequestedInfo	F	Véase la descripción.
QuarantineHandling	Q	Véase la descripción.
DetectEvents	T	Véase la descripción.
EventStates	ES	Véase la descripción.
RestartMethod	RM	Véase la descripción.
RestartDelay	RD	Un número de segundos codificado como un número decimal.
Capabilities	A	Véase la descripción.
VersionSupported	VS	Véase la descripción.
MaxMGCPDatagram	MD	Véase la descripción.
NOTA – El parámetro ResponseAck no se mostró en 7.3 porque los identificadores de transacción no son visibles en el ejemplo de API. Los implementadores pueden elegir un método diferente.		

Los parámetros no están necesariamente presentes en todas las instrucciones. El cuadro 7 presenta la asociación entre parámetros e instrucciones. La letra M (*mandatory*) significa obligatorio, O (*optional*) significa facultativo, y F (*forbidden*) significa prohibido:

Cuadro 7/J.171.111 – Asociación de parámetros con instrucciones

Nombre del parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck (nota)	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
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)} Los parámetros RequestedEvents y SignalRequests son facultativos en la instrucción NotificationRequest. Si se omiten estos parámetros, se considerará que las listas correspondientes están vacías. En el caso de instrucciones de tratamiento de conexión, la omisión de estos dos parámetros cuando la instrucción incluya un RequestIdentifier, implica que las listas correspondientes están vacías.

NOTA – El parámetro ResponseAck no se mostró en 7.3 porque los identificadores de transacción no son visibles en el ejemplo de API. Los implementadores pueden elegir un método diferente.

Las pasarelas de circuitos troncales y los MGC DEBERÍAN proporcionar siempre parámetros obligatorios antes que parámetros facultativos; no obstante, las pasarelas de circuitos troncales NO FALLARÁN por el hecho de no se haya seguido esta Recomendación.

Si los implementadores necesitan experimentar con nuevos parámetros, por ejemplo en el desarrollo de una nueva aplicación MGCP, deberían identificar estos parámetros por nombres que comiencen por la cadena "X-" o "X+", como por ejemplo:

X-FlowerOfTheDay: Daisy

Los nombres de parámetros que comienzan por "X+" son extensiones de parámetros obligatorios. Si una pasarela recibe una extensión de parámetro obligatorio que no entiende, RESPONDERÁ con un error (código de error 511 – Extensión no reconocida).

Los nombres de parámetro que comienzan por "X-" son extensiones de parámetro no críticos. Si una pasarela que recibe una extensión de parámetro no crítico, puede ignorar ese parámetro sin que ello afecte a la seguridad de funcionamiento.

Debe señalarse que los verbos experimentales tienen la forma *XABC*, mientras que los parámetros experimentales tienen la forma *X-ABC*.

Si se recibe una línea de parámetro con un parámetro prohibido, o cualquier otro error de formato, la entidad receptora debe responder con el código de error más específico para el error en cuestión. El código de error menos específico es 510 – Error de protocolo. Siempre puede proporcionarse un comentario en forma de texto.

8.2.2.1 Acuse de recibo de respuesta

El parámetro acuse de recibo de respuesta se utiliza para el soporte de la triple toma de contacto descrita en 8.7. Contiene una lista de "identificadores de transacciones confirmadas" separados por comas.

Cada "serie de identificadores de transacciones confirmadas" se compone de un número decimal (una sola transacción), o de dos números decimales separados por un solo guión, que indican, respectivamente, el identificador de transacción más bajo y el más alto de esa serie.

Un ejemplo de acuse de recibo de respuesta es:

K: 6234-6255, 6257, 19030-19044

8.2.2.2 Identificador de petición

El identificador de petición correlaciona una instrucción Notify con la NotificationRequest que la ocasionó (que incluye la petición de notificación incorporada en primitivas de tratamiento de conexión). Un RequestIdentifier es una cadena hexadecimal; la longitud NO EXCEDERÁ de 32 caracteres. Se comparan los RequestIdentifiers como cadenas en lugar de como valores numéricos. La cadena "0" está reservada para el informe de eventos persistentes cuando todavía no se ha recibido ninguna NotificationRequest (véase 7.3.2).

8.2.2.3 Opciones de conexión local

Las opciones de conexión local son los parámetros operacionales especificados por el MGC a la pasarela para una conexión. Estos parámetros son:

- El periodo de paquetización en milisegundos, codificado como la palabra clave "p" seguida de un carácter dos puntos (":") y un número decimal.
- El periodo de paquetización múltiple en milisegundos para cada códec en los parámetros LCO del método de codificación, codificado como la palabra clave "mp" seguida por un carácter dos puntos (":") y una lista de números decimales o de guiones, con un elemento por cada entrada del campo método de codificación. Los valores de los periodos de paquetización se separan por un solo punto y coma. El primer elemento de la lista SERÁ un número decimal y los siguientes SERÁN números decimales o guiones.
- El nombre literal del algoritmo de compresión según se indica en la Rec. UIT-T J.161, codificado como la palabra clave "a" seguida de un carácter dos puntos (":") y una cadena de caracteres. Si el controlador de pasarela de medios (MGC) especifica una lista de valores, éstos se separarán con un punto y coma. En el caso del RTP, los códec de audio se ESPECIFICARÁN mediante nombres de codificación definidos en el AV Profile de RTP, RCF 1890, y registrados ante la IANA, o mediante nombres de codificación indicados o definidos en la especificación de códec de audio y vídeo de IPCablecom. Los medios que no son de audio y que están registrados como del tipo MIME, UTILIZARÁN el formato "<MIME type>/<MIME subtype>" como en "image/t38". Se RECOMIENDA que también se soporten otras variantes reconocidas de los nombres literales de códec.

- El parámetro de compensación de eco, codificado como la palabra clave "e" seguida de un carácter dos puntos (":") y el valor "on" u "off".
- El parámetro tipo de servicio, codificado como la palabra clave "t" seguida de un carácter dos puntos (":") y el valor codificado como dos dígitos hexadecimales.
- El parámetro supresión de silencio, codificado como la palabra clave "s" seguida de un carácter dos puntos (":") y el valor "on" u "off".

Los parámetros LocalConnectionOptions utilizados para seguridad se codifican como sigue:

- El código de conjunto de cifrado RTP es la palabra clave "sc-rtp" seguida de un carácter dos puntos (":") y una cadena de algoritmos RTP como se define más adelante. Se puede especificar una lista de valores, que habrá que separar por un carácter punto y coma (";").
- El código de conjunto de cifrado RTCP es la palabra clave "sc-rtcp" seguida de un carácter dos puntos (":") y una cadena de algoritmos RTCP como se define más adelante. Puede especificarse una lista de valores, que habrá que separar por un carácter punto y coma (";").

Las cadenas del conjunto de cifrado RTP y RTCP observan la gramática:

```

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

```

donde ALPHA y DIGIT se definen en IETF RFC 2234. NO SE ENVIARÁN espacios blancos dentro de un conjunto de cifrado ni entre conjuntos de cifrado adyacentes (si hay varias). El siguiente ejemplo ilustra el formato de un conjunto de cifrado y de una lista de conjunto de cifrado:

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

La lista de los conjuntos cifrados de cifrado efectivamente soportados por IPCablecom se presenta en la Rec. UIT-T J.170.

Cuando están presentes varios parámetros, los valores van separados por una coma. La inclusión de un parámetro sin un valor se considerará un error (código de error 524 – Inconsistencia de LocalConnectionOptions).

Son ejemplos de opciones de conexión local:

```

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

```

El valor hexadecimal de tipo de servicio "20" implica una precedencia IP de 1, y el valor hexadecimal de tipo de servicio "A0" implica una precedencia IP de 5.

Este conjunto de atributos puede extenderse mediante atributos de extensión. Los atributos de extensión se componen de un nombre de atributo, seguido de un carácter dos puntos (":"), y una lista de valores de atributo separados por un carácter punto y coma (";"). El nombre de atributo comenzará por los dos caracteres "x+" en el caso de una extensión obligatoria, o "x-" en el caso de una extensión no obligatoria. Si una pasarela recibe un atributo con una extensión obligatoria que no reconoce, RECHAZARÁ la instrucción con un error (código de error 525 – Extensión desconocida en LocalConnectionOptions).

8.2.2.4 Capacidades

El parámetro Capabilities (capacidades) permite al MGC conocer las capacidades cuando se hace una auditoría. La codificación de Capabilities se basa en la codificación de las opciones de conexión local para los parámetros que son comunes a las dos entidades. Capabilities puede contener, además, una lista de paquetes soportados y una lista de modos soportados.

Se utilizan los siguientes parámetros:

- El periodo de paquetización en milisegundos, codificado como la palabra clave "p" seguida de un carácter dos puntos (":") y un número decimal. Una gama puede especificarse como dos números decimales separados por un guión.
- El nombre literal del algoritmo de compresión, codificado como la palabra clave "a" seguida de un carácter dos puntos (":") y una cadena de caracteres. HAY QUE utilizar los nombres literales definidos en el cuadro 3 de la especificación de códecs de audio y vídeo de IPCablecom (J.161). Puede especificarse una lista de valores, en cuyo caso los valores irán separados por un carácter punto y coma (",").
- La anchura de banda en kilobits/segundo (1000 bits/segundo), codificada como la palabra clave "b" seguida de un carácter dos puntos (":") y un número decimal. Una gama puede especificarse como dos números decimales separados por un guión.
- El parámetro compensación de eco, codificado como la palabra clave "e" seguida de un carácter dos puntos (":") y el valor "on" si la compensación de eco está soportada, o el valor "off" si no está soportada.
- El parámetro tipo de servicio, codificado como la palabra clave "t" seguida de un carácter dos puntos (":") y el valor "0" si el tipo de servicio no está soportado; cualquier otro valor indica que el sistema soporta ese tipo de servicio.
- El parámetro supresión de silencio, codificado como la palabra clave "s" seguida de un carácter dos puntos (":") y el valor "on" si la supresión de silencio está soportada, u "off" si no está soportada.
- Los lotes de eventos soportados por este punto extremo, codificados como la palabra clave "v" seguida de un carácter dos puntos (":") y una lista de los nombres de lotes soportados separados por un carácter punto y coma (","). El primer valor especificado será el lote por defecto para el punto extremo.
- Los modos de conexión soportados por este punto extremo, codificados como la palabra clave "m" seguida de un carácter dos puntos (":") y una lista de modos de conexión definidos en 8.2.2.7, separados por un carácter punto y coma (",").
- La palabra clave "sc-rtp" seguida de un carácter dos puntos (":") y una lista de conjuntos de cifrados RTP separados por un carácter punto y coma (","), con la misma codificación de LocalConnectionOptions.
- La palabra clave "sc-rtcp" seguida de un carácter dos puntos (":") y una lista de conjuntos de cifrados RTCP separados por un carácter punto y coma (","), con la misma codificación de LocalConnectionOptions.

Cuando están presentes varios parámetros, los valores se separan por una coma.

Son ejemplos de capacidades:

```
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
```

Obsérvese que los códecs y algoritmos de seguridad no son más que ejemplos; en distintas Recomendaciones sobre IPCablecom se proporciona información detallada sobre los códecs y algoritmos reales soportados, así como la codificación utilizada (véanse las Recs. J.170, J.162 y J.161). Obsérvese que los códecs y algoritmos de seguridad no son más que ejemplos; en distintas Recomendaciones sobre IPCablecom se proporciona información detallada sobre los códec y algoritmos reales soportados, así como la codificación utilizada (véanse RFC 1827 y RFC 1034).

Obsérvese también que cada conjunto de capacidades se indica en una sola línea. En los ejemplos que se dan anteriormente, cada conjunto se muestra en varias líneas debido únicamente a las restricciones de formato de esta Recomendación.

8.2.2.5 Parámetros de conexión

Los parámetros de conexión se codifican como una cadena de pares de tipo y valor, donde el tipo es uno de los códigos indicados en la tabla a continuación, y el valor es un entero decimal. Los tipos se separan de los valores por un signo "=". Los parámetros se separan unos de otros por una coma.

Los tipos de parámetros de conexión se especifican en el cuadro 8:

Cuadro 8/J.171.1 – Tipos de parámetros de conexión

Nombre del parámetro de conexión	Código	Valor del parámetro de conexión
Paquetes enviados	PS	Número de paquetes que fueron enviados por la conexión.
Octetos enviados	OS	Número de octetos que fueron enviados por la conexión.
Paquetes recibidos	PR	Número de paquetes que fueron recibidos por la conexión.
Octetos recibidos	OR	Número de octetos que fueron recibidos por la conexión.
Paquetes perdidos	PL	Número de paquetes que no fueron recibidos por la conexión; se deduce de los saltos en el número secuencial.
Fluctuación	JI	Fluctuación promedio de los tiempos de llegada entre los paquetes, en milisegundos, expresada como un número entero.
Latencia	LA	Latencia promedio, en milisegundos, expresada como un número entero.
Paquetes enviados del punto distante	PC/RPS	Número de paquetes enviados por la conexión, desde el punto de vista del punto extremo distante.
Octetos enviados del punto distante	PC/ROS	Número de octetos que fueron enviados por la conexión, desde el punto de vista del punto extremo distante.
Paquetes perdidos del punto distante	PC/RPL	Número de paquetes que no fueron recibidos por la conexión, desde el punto de vista del punto extremo distante; se deduce de los saltos en el número secuencial.
Fluctuación del punto distante	PC/RJI	Fluctuación media en milisegundos, desde el punto de vista del punto extremo distante, expresada como un número entero.

Los nombres de los parámetros de conexión de extensión se componen de la cadena "X-" seguida de un nombre de parámetro de extensión formado por dos o tres letras. Si los MGC reciben extensiones no reconocidas, no las tendrán en cuenta y no harán nada. Si un punto extremo recibe paquetes RTCP con estas estadísticas, DEBERÁ retornar los parámetros distantes (Rxx mencionados) en la respuesta a las instrucciones Suprimir Conexión y Auditar Conexión.

Ejemplo de codificación de un parámetro de conexión:

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 Códigos de motivo

Los códigos de motivo son valores numéricos constituidos por tres dígitos. El código de motivo va seguido facultativamente por un espacio blanco y un comentario, por ejemplo:

E: 900 Endpoint malfunctioning

En 7.6 se presenta una lista de códigos de motivo.

8.2.2.7 Modo de conexión

El modo de conexión describe el modo de funcionamiento de la conexión. Los posibles valores están indicados en el cuadro 9:

Cuadro 9/J.171.1 – Valores del modo de conexión

Modo	Significado
M: sendonly	La pasarela debería enviar paquetes solamente.
M: recvonly	La pasarela debería recibir paquetes solamente.
M: sendrecv	La pasarela debería enviar y recibir paquetes.
M: inactive	La pasarela no debería enviar ni recibir paquetes.
M: loopback	La pasarela debería poner el punto extremo en el modo conexión en bucle.
M: conttest	La pasarela debería poner el punto extremo en el modo prueba de continuidad.
M: netwloop	La pasarela debería poner el punto extremo en el modo conexión en bucle de red.
M: netwtest	La pasarela debería poner el punto extremo en el modo prueba de continuidad de red.

8.2.2.8 Codificación de nombre de evento/señal

Los nombres de evento/señal se componen de un nombre de lote facultativo, separado por una barra oblicua (/) del nombre del evento propiamente dicho. El nombre de evento puede ir seguido facultativamente de un signo (@) y el identificador de una conexión en la que se debe observar el evento. Se utilizan nombres de evento en los parámetros RequestedEvents, SignalRequests, DetectEvents, ObservedEvents y EventStates. Cada evento se identifica por un código de evento. Estas codificaciones ASCII son insensibles a la escritura en mayúscula o minúscula. Por tanto, los valores "co", "Co", "CO" y "cO" deben considerarse iguales.

Ejemplos de nombres de evento válidos:

IT/co1	Iniciación de prueba de continuidad en el lote de circuitos troncales de PU-RDSI.
MT/oc	Operación finalizada en el lote protocolo de terminación MF.
co1	Iniciación de prueba de continuidad en el lote de circuitos troncales de PU-RDSI, suponiendo que el lote de circuitos troncales de PU-RDSI es el lote por defecto para el punto extremo.
IT/rt@0A3F58	Tono de llamada de retorno en conexión "0A3F58".

Además, la notación de eventos con comodines, en lugar de nombres individuales, puede utilizarse en las instrucciones RequestedEvents y DetectEvents (pero no en las instrucciones SignalRequests, ObservedEvents ni EventStates):

IT/all	Todos los eventos en el lote de circuitos troncales de PU-RDSI.
--------	---

Por último, el asterisco puede utilizarse para designar "todas las conexiones", y el signo de dólar puede utilizarse para designar la conexión "actual". Los siguientes son ejemplos válidos de estas notaciones:

IT/ma@*	El evento comienzo de medios RTP en todas las conexiones para el punto extremo.
IT/rt@\$	Tono de llamada de retorno en la conexión actual.

En el anexo A figura un lote de eventos para pasarelas de circuitos troncales.

8.2.2.9 RequestedEvents

El parámetro RequestedEvents proporciona la lista de eventos que se ha solicitado. Los códigos de evento actualmente definidos se describen en el anexo A. Cada evento puede ser calificado por una acción solicitada, o por una lista de acciones. No es posible combinar de cualquier forma las acciones; véanse las combinaciones válidas en 7.3.1. Las acciones, cuando se especifican, se codifican como una lista de palabras clave encerradas entre paréntesis y separadas por comas. Los códigos para las diversas acciones se indican en el cuadro 10:

Cuadro 10/J.171 – Códigos de acciones

Acción	Código
Notificar inmediatamente	N
Acumular	A
Ignorar	I
Mantener señales activas	K
NotificationRequest insertada	E
ModifyConnection insertada	C

Cuando no se especifica ninguna acción, la acción por defecto es notificar el evento. Esto significa que, por ejemplo, "ft" y "ft(N)" son equivalentes. Los eventos que no están incluidos en la lista se descartan, salvo si son eventos persistentes.

La lista de eventos solicitados se codifica en una sola línea, con los grupos evento/acción separados por comas. Un ejemplo de codificación de RequestedEvents es:

R: oc(N), of(N) Notificar operación completada, notificar fallo de la operación.

La acción NotificationRequest insertada tiene el siguiente formato:

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

donde cada R y cada S son facultativos y pueden suministrarse en otro orden.

La acción ModifyConnection insertada tiene el siguiente formato:

C (M (<ConnectionMode₁> (<ConnectionID₁>)) , . . . ,
M (<ConnectionMode_n> (<ConnectionID_n>)))

El siguiente ejemplo ilustra la utilización de ModifyConnection insertada:

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

Al empezar a transmitir medios en la conexión "23B34D", cambiar el modo de conexión de la "conexión actual" a "enviar y recibir". Notificar eventos en caso de "operación completa" y "fallo de operación".

8.2.2.10 SignalRequests

El parámetro SignalRequests proporciona el nombre de las señales que han sido solicitadas. Las señales actualmente definidas se indican en el anexo A. Una determinada señal sólo puede aparecer una vez en la lista y, por definición, todas las señales se aplican al mismo tiempo. La MG SOPORTARÁ al menos una señal en cada punto extremo y soportará simultáneamente la generación de una señal en cada conexión para un punto extremo dado. Es posible que un lote defina otros requisitos además de estas capacidades mínimas. Si la MG no soporta combinaciones de señales que exceden este requisito mínimo, DEBERÍA retornar el código de error 502.

Algunas señales pueden ser calificadas por parámetros de señal. Cuando una señal es calificada por múltiples parámetros de señal, los distintos parámetros de señal se separan por comas. Cada parámetro de señal TENDRÁ el formato especificado a continuación (se permiten espacios blancos):

```
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 )
```

donde signal-parameter-value puede ser una cadena, o una cadena entre comillas. Si aparecen dos veces las comillas consecutivamente dentro de una cadena encerrada entre comillas, se interpretará como una vez el carácter comillas en esa cadena. Por ejemplo, la cadena entre comillas "ab" "c" producirá la cadena ab"c.

Cada señal tiene asociado uno de los siguientes tipos de señal (véase 7.3.1):

- On/Off (OO);
- Time-out (TO);
- Brief (BR).

Las señales On/Off (señales activado/desactivado) pueden parametrizarse con un carácter "+" para activar la señal, o con un carácter "-" para desactivar la señal. Si una señal on/off no está parametrizada, la señal está activada. En el ejemplo que sigue, las dos cadenas activan la señal "mysignal":

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

Las señales sometidas a periodo de temporización pueden parametrizarse con el parámetro de señal "TO" y un valor de temporización que prevalece sobre el valor de temporización por defecto. Si la señal sometida a periodo de temporización no está parametrizada con un valor de temporización, se utilizará el valor de temporización por defecto. En el ejemplo siguiente, ambas cadenas aplican el tono de llamada de retorno durante seis segundos:

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

Algunas señales pueden definir parámetros de señal adicionales.

Los parámetros de señal se encierran entre paréntesis, como se muestra en el siguiente ejemplo:

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

Cuando se solicitan varias señales, sus códigos se separan por una coma, como en el siguiente ejemplo:

```
S: signal1, signal2
```

8.2.2.11 ObservedEvents

Los parámetros "eventos observados" proporcionan la lista de eventos que han sido observados. Los códigos de evento son los mismos utilizados en la NotificationRequest. Cuando se detecta un evento en la conexión, el evento observado identificará la conexión en la que se detectó el evento utilizando la sintaxis "@<connection>". Son ejemplos de eventos observados:

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

8.2.2.12 RequestedInfo

El parámetro RequestedInfo contiene una lista de códigos de parámetro separados por una coma, como se define en la sección "Líneas de parámetro"; en 7.3.8 se indican los parámetros que pueden conocerse por auditoría. Los siguientes valores indicados en el cuadro 11 también son soportados:

Cuadro 11/J.171.1 – Valores de RequestedInfo

Parámetro RequestedInfo	Código
LocalConnectionDescriptor	LC
RemoteConnectionDescriptor	RC

Por ejemplo, si se desea conocer por auditoría el valor de los parámetros NotifiedEntity, RequestIdentifier, RequestedEvents, SignalRequests, DetectEvents, EventStates, LocalConnectionDescriptor y RemoteConnectionDescriptor, los valores del parámetro RequestedInfo serán:

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

La petición de capacidades, para la instrucción AuditEndPoint, se codifica por el código de parámetro "A", como en:

F: A

8.2.2.13 QuarantineHandling

El parámetro tratamiento de cuarentena contiene una lista de palabras clave separadas por comas:

- La palabra clave "process" o "discard" para indicar el tratamiento de eventos observados puestos en cuarentena. Cuando no se indique nada, se supone que se hace el tratamiento.
- La palabra clave "step" o "loop" para indicar si se espera a lo sumo una notificación o si se permiten múltiples notificaciones. Cuando no se indique nada se supone la palabra clave "step". Es obligatorio soportar estas dos palabras clave.

Los siguientes son ejemplos válidos:

Q: loop
Q: process
Q: loop discard

8.2.2.14 DetectEvents

El parámetro DetectEvents se codifica como una lista de eventos separados por una coma, por ejemplo:

T: ft, mt

Debe señalarse que no se pueden asociar acciones a estos eventos.

8.2.2.15 EventStates

El parámetro EventStates se codifica como una lista de eventos separados por una coma, por ejemplo:

ES: MO/rlc

Debe señalarse que no se pueden asociar acciones a estos eventos.

8.2.2.16 RestartMethod

El parámetro RestartMethod se codifica con una de las palabras clave "graceful" (paulatino), "cancel-graceful" (anular paulatino), "forced" (forzado), "restart" (rearranque), o "disconnected" (desconectado), por ejemplo:

RM: restart

8.2.2.17 VersionSupported

El parámetro VersionSupported se codifica como una lista de versiones soportadas, separadas por una coma, por ejemplo:

VS: MGCP 1.0, MGCP 1.0 TGCP 1.0

8.2.2.18 Identificador de llamada

El identificador de llamada se codifica como una cadena hexadecimal de hasta 32 caracteres. En la comparación los identificadores de llamada se tratan como cadenas y no como valores numéricos.

8.2.2.19 Identificador de conexión

El identificador de conexión se codifica como una cadena hexadecimal de hasta 32 caracteres. En la comparación los identificadores de conexión se tratan como cadenas y no como valores numéricos.

8.2.2.20 MaxMGCPDatagram

El parámetro MaxMGCPDatagram se codifica como una lista de hasta nueve cifras digitales en la que no se admiten ceros iniciales. El siguiente ejemplo ilustra el uso de este parámetro:

MD: 8100

8.3 Formatos del encabezamiento

El encabezamiento de respuesta se compone de una línea de respuesta seguida facultativamente de encabezamientos que codifican los parámetros de respuesta.

La línea de respuesta comienza por el código de respuesta, que es un valor numérico de tres dígitos. El código va seguido de un espacio blanco, el identificador de transacción, y un comentario facultativo precedido por un espacio blanco, por ejemplo:

200 1201 OK

En el cuadro 12 se recapitulan los parámetros de respuesta cuya presencia es obligatoria o facultativa en un encabezamiento de respuesta, en función de la instrucción que ocasionó la respuesta. Se invita al lector a examinar distintas definiciones de instrucciones, ya que en este cuadro sólo se ofrece una breve información. La letra M (*mandatory*) significa obligatorio, O (*optional*) significa facultativo, y F (*forbidden*) significa prohibido.

Cuadro 12/J.171.1 – Asociación de parámetros con respuesta de instrucción

Nombre del parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
ResponseAck (nota 5)	O (nota 1)	O (nota 1)	O (nota 1)	O (nota 1)	O (nota 1)	O (nota 1)	O (nota 1)	O (nota 1)
CallId	F	F	F	F	F	F	O	F
ConnectionId	O (nota 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

Cuadro 12/J.171.1 – Asociación de parámetros con respuesta de instrucción

Nombre del parámetro	CRCX	MDCX	DLCX	RQNT	NTFY	AUEP	AUCX	RSIP
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 (nota 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 (nota 4)	O (nota 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

NOTA 1 – El parámetro ResponseAck NO SE UTILIZARÁ con otra respuesta que no sea la respuesta final emitida después de una respuesta provisional para la transacción en cuestión. En ese caso, la presencia del parámetro ResponseAck provocará un mensaje de acuse de recibo de respuesta; los valores que se comuniquen con ResponseAck no se tendrán en cuenta.

NOTA 2 – En el caso de un mensaje CreateConnection, la línea de respuesta va seguida de un parámetro identificador de conexión y de un LocalConnectionDescriptor. También puede ir seguida de un parámetro Specific-Endpoint-Id, si la solicitud de creación fue enviada a un identificador de punto extremo con comodín. Aunque en el cuadro se señalan como facultativos los parámetros identificador de conexión y LocalConnectionDescriptor, éstos son en realidad obligatorios con todas las respuestas positivas si se ha creado una conexión, y prohibidos si la respuesta es negativa y no se ha creado una conexión.

NOTA 3 – Los parámetros de conexión son válidos únicamente con una respuesta de ejecución satisfactoria para una instrucción DeleteConnection sin comodín enviada por el agente de llamada.

NOTA 4 – Hay que transmitir un LocalConnectionDescriptor con la respuesta positiva (código 200) a la instrucción CreateConnection. También hay que transmitirlo en respuesta a una instrucción ModifyConnection si se han modificado los parámetros de la sesión. El LocalConnectionDescriptor se codifica como una "descripción de sesión", según se define en 8.4. Está separado del encabezamiento de respuesta por una línea vacía.

NOTA 5 – El parámetro ResponseAck no se mencionó en 7.3 porque los identificadores de transacción no son visibles en este ejemplo de API. Los implementadores pueden optar por otra solución.

Los parámetros de respuesta para cada una de las instrucciones se describen a continuación.

8.3.1 CreateConnection

En el caso de un mensaje CreateConnection, la línea de respuesta va seguida de un parámetro identificador de conexión con una respuesta exitosa (código 200). Se transmite además un LocalConnectionDescriptor con una respuesta positiva. El LocalConnectionDescriptor se codifica como una "descripción de sesión", como se define en 8.4. Está separado del encabezamiento de respuesta por una línea vacía, por ejemplo:

```
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
```

Si ya se había emitido una respuesta provisional, la respuesta final puede contener, además, el parámetro acuse de recibo de respuesta, como en el ejemplo siguiente:

```
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
```

Se acusa recibo de la respuesta final por medio de un acuse de recibo de respuesta:

```
000 1204
```

8.3.2 ModifyConnection

En el caso de un mensaje ModifyConnection exitoso, la línea de respuesta va seguida de un LocalConnectionDescriptor si la modificación ocasionó una modificación de los parámetros de la sesión (por ejemplo, el solo cambio del modo de conexión no entraña una modificación de los parámetros de la sesión). El LocalConnectionDescriptor se codifica como una "descripción de sesión", como se define en 8.4. Está separado del encabezamiento de respuesta por una línea vacía.

```
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
```

Si ya se ha emitido una respuesta provisional, la respuesta final puede contener además el parámetro de acuse de recibo de respuesta, como en:

```
526 1207 No bandwidth
K:
```

Se acusa recibo de la respuesta final mediante un acuse de recibo de respuesta:

```
000 1207 OK
```

8.3.3 DeleteConnection

En dependencia de la variante del mensaje DeleteConnection, la línea de respuesta puede ir seguida de una línea de parámetro de ConnectionParameters, como se define en 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 NotificationRequest

Una respuesta NotificationRequest no incluye parámetros de respuesta adicionales.

8.3.5 Notify

Una respuesta Notify no incluye parámetros de respuesta adicionales.

8.3.6 AuditEndpoint

En el caso de una instrucción AuditEndPoint, la línea de respuesta puede ir seguida de información para cada uno de los parámetros solicitados; cada parámetro aparecerá en una línea distinta. Los parámetros para los cuales no exista ningún valor en ese momento, se proporcionarán de todas formas. Cada nombre de punto extremo local "expandido" por un carácter comodín aparecerá en una línea individual utilizando el código de parámetro "SpecificEndPointId", por ejemplo:

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

A continuación se muestra el ejemplo de una respuesta a un mensaje AuditEndPoint con un nombre de punto extremo sin comodín. Obsérvese que en este caso no se suministra SpecificEndPointId. Obsérvese también que se proporciona cada conjunto de capacidades en una sola línea. En el ejemplo a continuación los conjuntos aparecen en varias líneas debido únicamente a las restricciones de formato de esta Recomendación.

```
200 1200 OK
A: a:PCMU; p:10, e:on, s:off, t:1, v:IT,
  m:sendonly;recvonly;sendrecv;inactive
A: a:G728, p:20, e:on, s:off, t:1, v:IT,
  m:sendonly;recvonly;sendrecv;inactive
A: a:G729A; p:30-90, e:on, s:on, t:1, v:L,
  m:sendonly;recvonly;sendrecv;inactive;confrnce
```

8.3.7 AuditConnection

En el caso de una instrucción AuditConnection, la línea de respuesta puede ir seguida de información para cada uno de los parámetros solicitados. Los parámetros para los cuales no exista ningún valor en ese momento se proporcionarán de todas formas. Los descriptors de conexión aparecerán siempre en último lugar y cada uno de ellos irá precedido de una línea vacía, como por ejemplo:

```
200 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 se proporciona un descriptor de conexión local y un descriptor de conexión distante, el descriptor de conexión local será el primero de los dos. Si se solicita un descriptor de conexión pero no existe ninguno para la conexión autorizada, dicho descriptor de conexión aparecerá con el campo versión de protocolo SDP solamente.

8.3.8 RestartInProgress

La respuesta a un RestartInProgress puede incluir el nombre de otro MGC, para establecer contacto con el mismo, por ejemplo cuando el MGC redirecciona el punto extremo hacia otro MGC, como en el siguiente ejemplo:

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

8.4 Codificación de descripción de sesión

La descripción de sesión se codifica de acuerdo con el protocolo de descripción de sesión (SDP); sin embargo, las pasarelas de circuitos troncales pueden partir de ciertos supuestos simplificadores sobre la descripción de sesión como se especifica más adelante. Conforme a IETF RFC 2327, las descripciones de sesión son sensibles a la escritura en mayúscula o minúscula.

La utilización del SDP depende del tipo de sesión, como se especifica en el parámetro "medios".

- Si el valor de los medios es "audio", la descripción de la sesión corresponde a un servicio de audio.
- Si el valor de los medios es "image", la descripción de la sesión es para un servicio de imagen, como el FAX.

8.4.1 Utilización del servicio de audio en el protocolo SDP

En una pasarela de circuitos troncales sólo hay que describir sesiones que utilizan exactamente un solo tipo de medios MIME en un momento dado, que puede ser "audio" (para voz o datos en la banda vocal) o "image" (Para llamadas de Fax T.38). En 8.4.2 se especifican los parámetros de SDP pertinentes tanto para los tipos de medios basados en "audio" como para los basados en "image". En 8.4.3 se especifican los parámetros particulares de "audio", mientras que en 8.4.4 se especifican los parámetros particulares de "image" cuando se utilizan para T.38. La pasarela de circuitos SOPORTARÁ descripciones de sesión que cumplan estas reglas, y en el orden siguiente:

- 1) el perfil SDP presentado más adelante;

2) IETF RFC 2327 (*SDP: session description protocol*).

Habría que tomar precauciones en los casos en que el MGC tenga que modificar el SDP que recibe de un punto extremo. El SDP proporciona los medios para comunicar las capacidades de un punto extremo a otro punto extremo. Si el MGC modifica el SDP, NO PODRÁ infringir ninguna de las reglas definidas en la presente cláusula.

El perfil SDP proporcionado describe la utilización del protocolo de descripción de sesión en el TGCP. La descripción general y la explicación de los distintos parámetros, así como la mayoría de los parámetros usados únicamente para audio, pueden encontrarse en IETF RFC 2327; mientras que en T.38 figuran los parámetros particulares de T.38 para codificación de imagen. A continuación se indican detalladamente los valores que los puntos extremos TGCP deben proporcionar para estos campos (send (enviar)), y las acciones correspondientes a los valores suministrados o no suministrados para estos campos (receive (recibir)). Obsérvese que este perfil de SDP no es conforme al modelo de oferta/respuesta (offer/answer) definido en RFC 3264. Por lo tanto, para interactuar con otra entidad que utiliza el modelo de oferta/respuesta, será posiblemente necesario que el MGC modifique el SDP que recibe del punto extremo.

8.4.2 Parámetros del SDP comunes para el uso de servicios de imagen y de audio

8.4.2.1 Protocol Version (v=) (versión de protocolo)

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

Send: DEBE proporcionarse de acuerdo con IETF RFC 2327 (es decir, v=0).

Receive: DEBE proporcionarse de acuerdo con IETF RFC 2327.

8.4.2.2 Origin (o=) (origen)

El campo origin (o=) consta de 6 subcampos según IETF RFC 2327:

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

Username (nombre de usuario):

Send: SE UTILIZARÁ el carácter guión como nombre de usuario cuando se pide confidencialidad. El carácter guión DEBERÍA utilizarse en todo otro caso.²⁵

Receive: Este campo no se DEBERÍA tener en cuenta.

Session-ID (identificador de sesión):

Send: DEBE ser conforme con IETF RFC 2327 para facilitar la interoperabilidad con clientes que no son IPCablecom.

Receive: Este campo no se DEBERÍA tener en cuenta.

Version (versión):

Send: De conformidad con IETF RFC 2327.

Receive: Este campo no se DEBERÍA tener en cuenta.

Network Type (tipo de red):

²⁵ Como los puntos extremos TGCP no saben cuándo se ha pedido confidencialidad, siempre se DEBERÍA utilizar un guión.

Send: DEBE utilizarse el tipo "IN".

Receive: Este campo no se DEBERÍA tener en cuenta.

Address Type (tipo de dirección):

Send: DEBE utilizarse el tipo "IP4"

Receive: Este campo no se DEBERÍA tener en cuenta.

Address (dirección):

Send: Será conforme con IETF RFC 2327 para facilitar la interoperabilidad con clientes que no son IP4com.

Receive: Este campo no se tendrá en cuenta.

8.4.2.3 Session name (s=) (nombre de sesión)

s=<session-name>

s=-

Send: El carácter guión SE UTILIZARÁ como nombre de sesión.

Receive: Este campo NO SE TENDRÁ en cuenta.

8.4.2.4 Session and media Information (i=) (información de sesión y medios)

i=<session-description>

Send: Para TGCP, el campo NO SE UTILIZARÁ.

Receive: Este campo NO SE TENDRÁ en cuenta.

8.4.2.5 URI (u=) (identificador de recurso universal)

u= <URI>

Send: Para TGCP, el campo NO SE UTILIZARÁ.

Receive: Este campo NO SE TENDRÁ en cuenta.

8.4.2.6 E-Mail address and phone number (e=, p=) (dirección de correo electrónico y número de teléfono)

e=<e-mail-address>

p=<phone-number>

Send: Para TGCP, el campo NO SE UTILIZARÁ.

Receive: Este campo NO SE TENDRÁ en cuenta.

8.4.2.7 Connection data (c=) (datos de conexión)

El campo datos de conexión consta de 3 subcampos:

c=<network-type> <address-type> <connection-address>

c=IN IP4 10.10.111.11

Network Type (tipo de red):

Send: SE UTILIZARÁ el tipo "IN".

Receive: HAY QUE INCLUIR el tipo "IN".

Address Type (tipo de dirección):

Send: SE UTILIZARÁ el tipo "IP4".

Receive: HAY QUE INCLUIR el tipo "IP4".

Connection Address (dirección de conexión)

Send: En este campo HABRÁ una dirección IP unicast (unidifusión) en la que la aplicación recibirá el tren de medios. Por tanto NO se incluirán valores TTL ni "number of addresses" (número de direcciones). Este campo NO SE INDICARÁ con un nombre de dominio plenamente calificado en lugar de una dirección IP. Una dirección diferente de cero especifica la dirección de envío y la dirección de recepción para los trenes de medios correspondientes.

Receive: HAY QUE indicar una dirección IP unicast (unidifusión) o un nombre de dominio plenamente calificado. Una dirección diferente de cero especifica la dirección de envío y la dirección de recepción para los trenes de medios correspondientes.

8.4.2.8 Bandwidth (b=) (anchura de banda)

b=<modifier> : <bandwidth-value>

b=AS : 64

Send: La información de anchura de banda es facultativa en SDP pero siempre DEBERÍA incluirse²⁶. Cuando se utiliza un rtpmap o un códec que no sea muy conocido²⁷, SE UTILIZARÁ la información de anchura de banda.

Receive: DEBERÍA incluirse la información de anchura de banda. Si no se incluye un modificador de anchura de banda, el receptor SUPONDRÁ valores de anchura de banda por defecto razonables para los códecs bien conocidos.

Modifier (modificador):

Send: SE UTILIZARÁ el tipo "AS".

Receive: HAY QUE incluir el tipo "AS".

Bandwidth Value (valor de anchura de banda)

Send: En este campo se indicará el valor que corresponda a la máxima anchura de banda requerida por el tren de medios, en kilobits/segundo. La cláusula 7.5 de la especificación de códecs de IPCablecom (J.161) contiene información detallada para el cálculo del valor de la anchura de banda.

Receive: SE INDICARÁ la máxima anchura de banda requerida por el tren de medios, en kilobits/segundo.

²⁶ Si no se utiliza este campo, es posible que el controlador de puerta no autorice la anchura de banda adecuada.

²⁷ Por códec no muy conocido ha de entenderse un códec que no esté definido en la Rec. UIT-T J.161 sobre códecs IPCablecom.

8.4.2.9 Time, repeat times and time zones (t=, r=, z=) (hora, hora de repetición y uso horario)

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

Send: Hay que indicar la hora; "start time" PUEDE ser cero, pero DEBERÍA ser la hora actual, y "stop time" DEBERÍA ser cero. Repeat Times, y Time Zones NO DEBERÍAN utilizarse; si se utilizan deben ser conformes con IETF RFC 2327.

Receive: Si cualesquiera de estos campos estuviera presente, no se DEBERÍA tener en cuenta.

8.4.2.10 Atributes (a=) (atributos)

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

Send: Se PUEDE incluir una o varias de las líneas de atributo "a" especificadas más adelante.

Receive: Se PUEDE incluir una o varias de las líneas de atributo "a" especificadas a continuación y se DEBE actuar en consecuencia.

Obsérvese que el SDP exige que no se tengan en cuenta los atributos desconocidos.

mptime: Este atributo define una lista de valores del periodo de paquetización que el punto extremo está en capacidad de utilizar (enviando y recibiendo) para esta conexión.

Send: Hay que incluir este atributo. En la lista DEBE haber exactamente un elemento por cada elemento de <format> provisto en la línea "m=". El elemento número j de esta lista define el periodo de paquetización del elemento número j de la línea "m=". El primer elemento de la lista será un número decimal, mientras que los elementos subsiguientes serán números decimales o guiones. En los formatos de medios en los que no es aplicable una sola velocidad de paquetización (por ejemplo, los códecs no vocales, como el evento telefónico y el ruido de confort), DEBE codificarse un guión ("-") en la ubicación correspondiente de la lista de periodos de paquetización.

Receive: Comunica la lista de periodos de paquetización que el punto extremo distante está en capacidad de utilizar para esta conexión; uno por formato de medios de la línea "m=". Para los formatos de medios cuyo periodo de paquetización se especifique con un guión ("-"), el punto extremo UTILIZARÁ uno de los periodos de paquetización especificados en la lista. Si el atributo "mptime" no está presente, entonces DEBE interpretarse que el valor del atributo "ptime", si está presente, indica el periodo de paquetización de todos los códec que figuran en la línea "m=". Si no están presentes los atributos "mptime" ni "ptime", entonces la MG debe usar el valor por defecto de códecs bien conocidos (según se define en RFC 1890).

X-pc-bridge:

Send: Los puntos extremos TGCP NO UTILIZARÁN este atributo.

Receive: Los puntos extremos TGCP NO TENDRÁN en cuenta este atributo, si se recibe.

recvonly:

Send: Ningún punto extremo TGCP debería proporcionar este campo.

Receive: NO SE TENDRÁ en cuenta este campo.

sendrecv:

Send: Ningún punto extremo TGCP debería proporcionar este campo.

Receive: NO SE TENDRÁ en cuenta este campo.

sendonly:

Send: Ningún punto extremo TGCP debería proporcionar este campo.

Receive: NO SE TENDRÁ en cuenta este campo.

ptime:

Send: DEBERÍA enviarse el ptime si éste se recibió en un RemoteConnectionDescriptor o si el MGC utilizó el valor de LocalConnectionOption para el periodo de paquetización ('p:').

Receive: NO SE TENDRÁ en cuenta este campo si el SDP incluye el atributo "mptime" (como lo exigen los dispositivos conformes con PacketCable). Si el atributo "mptime" no se encuentra presente, entonces se utiliza este campo para definir el intervalo de paquetización de todos los codificadores presentes en la descripción del SDP. Si no está presente "mptime" ni "ptime", la MG UTILIZARÁ valores por defecto razonables para los códecs bien conocidos, según se define en la Rec. UIT-T J.161.

8.4.3 Utilización del servicio de audio del SDP

Los siguientes parámetros del SDP se aplican al nivel de los medios y atañen a la utilización del servicio de audio. Los puntos extremos de PacketCable NO ENVIARÁN ninguno de estos parámetros junto con un descriptor de medios de imagen (véase 8.4.4). Sin embargo, si en un descriptor de medios de audio, el punto extremo recibe un SDP con parámetros de atributo exclusivos de imagen, los parámetros DEBERÍAN ignorarse. Si se han de proporcionar parámetros de capacidad de medios, todos los descriptores de capacidad de medios (que incluye la línea de descripción de capacidades, a=cpsc, junto con 0 o varias líneas de atributos de capacidades, por ejemplo, a=cpar) aparecerán tras el último atributo de medios y cada descriptor de capacidades de medios se indicará por separado.

8.4.3.1 Encryption Keys (claves de criptación)

k=<method>

k=<method> : <encryption-keys>

Los servicios de seguridad para IPCablecom se definen en la Rec. UIT-T J.170. Los servicios de seguridad especificados para RTP y RTCP no son conformes con los de IETF RFC 1889, IETF RFC 1890, y IETF RFC 2327. Por razones de interoperabilidad con dispositivos no IPCablecom, el parámetro "k" no se utilizará para transportar parámetros de seguridad.

Send: NO SE UTILIZARÁ.

Receive: Este campo no se DEBERÍA tener en cuenta.

8.4.3.2 Attributes (a=) (atributos)

```
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> ...
```

Send: PUEDE incluirse una o más de las líneas de atributo "a" especificadas más adelante.

Receive: PUEDE incluirse una o más líneas de atributo "a" especificadas más adelante y hay que actuar en consecuencia.

rtpmap:

Send: Si se utiliza este campo, su utilización DEBE ser conforme con IETF RFC 2327. PUEDE utilizarse para códecs bien conocidos y no muy conocidos. Los nombres de codificación utilizados se indican en otras dos Recomendaciones sobre IPCablecom (véanse las Recs. UIT-T J.161 y J.170). En una conexión dada, el tipo de cabida útil para un método de codificación en particular, será el mismo para las direcciones de envío y de recepción. Además, una vez definido el método de codificación, para un tipo de cabida útil en una conexión, NO SE PODRÁ definir otro método de codificación.

Receive: Si se utiliza será conforme a IETF RFC 2327. Para una determinada conexión, las implementaciones NO DEBERÍAN fallar si se define un método de codificación dado para varios tipos de cabida útil diferentes en las direcciones de envío y de recepción, y tampoco si hay cambio para un determinado tipo de cabida útil.

fmtp:

Send: Se PUEDE usar este campo para proporcionar parámetros que son específicos de un formato en particular. Por ejemplo, se podría utilizar para describir eventos telefónicos soportados para un formato RFC 2833. Si se utiliza, será uno de los formatos especificados para los medios. Los formatos especificados se presentan por separado en una especificación, en la que se describe su forma de uso.

Receive: Si se utiliza, será conforme a IETF RFC 2327.

X-pc-codecs:

Send: El campo contiene una lista de códecs alternativos que el punto extremo puede utilizar para esta conexión. Los códecs alternativos incluidos en la lista están ordenados por nivel decreciente de preferencia, es decir, el códec alternativo de nivel de preferencia más elevado es el primero en la lista. Un códec se codifica de manera similar a "encoding name" en rtpmap.

Receive: El campo contiene una lista de códecs que el punto extremo distante puede utilizar para esta conexión. Sólo SE UTILIZARÁN los códecs señalados por una línea de medios (m=).

X-pc-secret:

Send: El campo contiene un secreto de extremo a extremo (y posiblemente) el PAD (relleno) que habrán de utilizarse para la seguridad de RTP y RTCP. El secreto y el PAD se codifican de manera similar al parámetro clave de encriptación (k=) de IETF RFC 2327, con las siguientes restricciones:

La clave de criptación NO CONTENDRÁ una combinación de algoritmos, sino sólo una frase secreta.

El <method> que especifica la codificación de la frase secreta SERÁ "clear" o "base64", como se define en IETF RFC 2045, excepto que la longitud máxima de línea no se especifica en la presente Recomendación. El método "clear" NO SE UTILIZARÁ si el secreto contiene cualquiera de los caracteres que están prohibidos en SDP.

En la Recomendación sobre seguridad J.170 se describen los requisitos que indican cuándo transmitir PAD. Si está presente, hay que separarlo del secreto con al menos un espacio blanco. Para el relleno y el secreto SE UTILIZARÁ el mismo método de codificación.

Receive: El campo contiene un secreto de extremo a extremo que habrá de utilizarse para la seguridad de RTP y RTCP. Si está presente, se debe usar de conformidad con la Recomendación sobre seguridad (J.170) y hay que separarlo del secreto con al menos un espacio blanco. Para el relleno y el secreto SE UTILIZARÁ el mismo método de codificación.

X-pc-suites-rtp
X-pc-suites-rtcp

Send: El campo contiene una lista de combinación de algoritmos que el punto extremo puede utilizar para esta conexión (respectivamente RTP y RTCP). La primera combinación de algoritmos que figura en la lista es la que el punto extremo en ese momento tiene previsto utilizar. Las otras combinaciones de algoritmos en la lista son alternativas ordenadas por nivel decreciente de preferencia, es decir, la alternativa con nivel de preferencia más elevado es la segunda en la lista. Una combinación de algoritmos se codifica como se especifica a continuación:

ciphersuite = [AuthenticationAlgorithm] "/" [EncryptionAlgorithm]

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

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

donde ALPHA, y DIGIT se definen en IETF RFC 2234. En un conjunto de cifrado no se permiten espacios blancos. El siguiente ejemplo ilustra la utilización de un conjunto de cifrado:

62/51

La verdadera lista de conjuntos de cifrado se proporciona en la Rec. UIT-T J.170.

Receive: Contiene una lista de conjuntos de cifrado que el punto extremo distante puede utilizar para esta conexión. Para poder utilizar un conjunto de cifrado que no sea la primera en la lista es necesario que esté señalada en una nueva línea de conjunto de cifrado en la que ahora aparece en primera posición.

8.4.3.3 Media Announcements (m=) (anuncios de medios)

Los anuncios de medios (m=) constan de tres subcampos:

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

Media (Medios):

Send: HAY QUE utilizar el tipo de medios "audio".

Receive: El tipo recibido SERÁ "audio".

Port (Puerto)

Send: SERÁ conforme a IETF RFC 2327. El puerto especificado es el puerto de recepción, independientemente de que el tren sea unidireccional o bidireccional. El puerto de envío puede ser diferente.

Receive: SERÁ conforme a IETF RFC 2327. El puerto especificado es el puerto de recepción. El puerto de envío puede ser diferente.

Transport (Transporte)

Send: HAY QUE utilizar el protocolo de transporte "RTP/AVP".

Receive: El protocolo de transporte SERÁ "RTP/AVP".

Media Formats (Formato de los medios):

Send: Hay que utilizar un tipo de medios adecuado, según lo definido en IETF RFC 2327. Concretamente, este campo contiene una lista con uno o varios tipos de cabida útil RTP que este punto extremo estaría preparado a recibir por la conexión y que preferiría utilizar para la emisión. Cada tipo de cabida útil corresponde, estática o dinámicamente, exclusivamente a un códec. Se DEBERÍA utilizar la correspondencia estática, en su caso (por ejemplo, 0 para PCMU, 8 para PCMA). Si se utiliza una correspondencia dinámica, también es condición incluir un atributo RTPMAP y seguir las orientaciones de 8.4.3.2.

Receive: Conforme a IETF RFC 2327. Indica el tipo o los tipos de cabida útil que el otro lado de esta conexión está preparado para recibir.

8.4.4 Utilización del servicio de imagen del SDP para T.38

Los siguientes parámetros de SDP tienen que ver con los medios y son particulares del servicio de imagen para T.38. Los puntos extremos de IPCablecom NO ENVIARÁN ninguno de estos parámetros dentro de un descriptor de medios de audio (véase 8.4.3). Sin embargo, si el punto extremo recibe un SDP con parámetros de atributo exclusivos de audio en un descriptor de medios, de imagen, NO SE DEBERÍAN tener en cuenta. De otra parte, si se han de proporcionar parámetros de capacidad de medios, todos los descriptores de capacidad de medios (que incluyen la línea de descripción de capacidades, a=cdsc, junto con 0 o varias líneas de atributos de capacidades, por ejemplo, a=cpar) aparecerán tras el último atributo de medios y cada descriptor de capacidades de medios SE INDICARÁ por separado.

8.4.4.1 Encryption keys (claves de criptación)

k= <method>

k= <method> : <encryption-keys>

No se han definido servicios de seguridad para el tipo de medios "image/t38".

Send: NO SE UTILIZARÁ.

Receive: No se DEBERÍA tener en cuenta este campo.

8.4.4.2 Attributes (a=) (atributos)

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

Send: PUEDE incluirse una o varias de las líneas de atributos "a" especificadas más adelante.

Receive: PUEDE incluirse una o varias de las líneas de atributo "a" especificadas más adelante y se actuará en consecuencia. Los valores de atributos son insensibles a mayúsculas y minúsculas. Las implementaciones aceptarán que todas las codificaciones de atributos en minúsculas, en mayúsculas o combinaciones de mayúsculas y minúsculas.

Obsérvese que el SDP exige que se haga caso omiso de los atributos desconocidos.

T38FaxVersion:

Conforme a lo definido en T.38: Al transmitir una respuesta a la oferta inicial, el receptor de la oferta DEBE aceptar esa versión o modificar el atributo de versión de forma que corresponda a una versión igual o menor. El receptor de la oferta NO podrá indicar en su respuesta una versión más alta que la ofrecida.

Como también se define en T.38: Es posible que las implementaciones iniciales de T.38 no proporcionen un número de versión de T.38. Si recibe un SDP sin el atributo de versión, el punto extremo supondrá que es la versión 0. Esto se tiene en cuenta en las siguientes consideraciones sobre el envío y recepción de este atributo:

Send: El punto extremo INDICARÁ la versión que pretende utilizar con el atributo T38FaxVersion. Sin embargo, NO INDICARÁ una versión más alta que la recibida en un RemoteConnectionDescriptor.

Receive: Si se recibe un RemoteConnectionDescriptor sin el atributo T38FaxVersion, el punto extremo UTILIZARÁ la versión 0 de la especificación T.38. Si se ha incluido el atributo, el punto extremo UTILIZARÁ una versión de la especificación igual o inferior a la indicada.

T38MaxBitRate:

Send: NO SE INCLUIRÁ el atributo T38MaxBitRate.

Receive: No se DEBERÍA tener en cuenta el atributo T38MaxBitRate.

T38FaxRateManagement:

Send: Se INCLUIRÁ el atributo T38FaxRateManagement y tendrá el valor "transferredTCF" si se utiliza UDPTL. Si el valor es "transferredTCF", la secuencia de comprobación de acondicionamiento (TCF) se transmite extremo a extremo, mientras que si el valor del atributo es "localTCF", la TCF se genera localmente. Obsérvese que "localTCF" es adecuado únicamente si se utiliza un transporte fiable como TCP.

Receive: Si se utiliza UDPTL, se incluirá el atributo T38FaxRateManagement con el valor "transferredTCF", o no se incluirá, en cuyo caso se supone que la opción es "transferredTCF". Se RECHAZARÁN todos los demás valores del atributo (código de error 505 – RemoteConnectionDescriptor no soportado).

T38FaxMaxBuffer:

Send: NO SE INCLUIRÁ el atributo T38FaxMaxBuffer.

Receive: No se DEBERÍA tener en cuenta el atributo.

T38FaxMaxDatagram:

Send: Se INCLUIRÁ el atributo T38FaxMaxDatagram. El valor indicado NO SERÁ inferior a 160 bytes. Esto se fundamenta en un periodo de paquetización de 40 ms y una velocidad de datos de 14 400 bits/s. Incluye un datagrama UDPTL sin los encabezamientos IP ni UDP.

Receive: Los puntos extremos NO ENVIARÁN datagramas mayores que lo especificado en el atributo T38FaxMaxDatagram. Antes de enviar un datagrama T.38, el punto extremo DEBERÁ asegurarse de que éste se encuentra dentro de los límites definidos por dicho atributo. Si el valor especificado de T38FaxMaxDatagram es demasiado bajo para soportar redundancia para un determinado datagrama, pero sí permite soportar T.38 sin redundancia, entonces el punto extremo DEBERÁ enviar ese datagrama T.38 sin redundancia. Si el valor es demasiado bajo, tanto que no permite enviar el datagrama sin redundancia, el punto extremo NO ENVIARÁ el datagrama T.38 y generará una indicación de fallo.

T38FaxUdpEC:

Es obligatorio soportar la redundancia, y es facultativo soportar la corrección de errores en recepción (FEC).

Send: Se INCLUIRÁ el atributo T38FaxUdpEC. Se PODRÍA enviar el valor "t38UDPFEC" si el sistema soporta FEC y, se da una de estas situaciones: en la instrucción no se proporcionó un RCD o el valor de este atributo recibido en el RCD para la instrucción es "t38UDPFEC". De lo contrario, se ENVIARÁ "t38UDPRedundancy".

Receive: Si el valor del atributo "T38FaxUdpEC" es "t38UDPRedundancy", hay que usar redundancia. Si el valor de atributo T38FaxUdpEC es "t38UDPFEC" y el punto extremo soporta FEC, entonces DEBERÍA usarse FEC. Si el valor de atributo T38FaxUdpEC es "t38UDPFEC" y no se soporta FEC, hay que usar redundancia. Si no se incluye este atributo, entonces el punto extremo NO USARÁ redundancia ni FEC.

T38FaxFillBitRemoval:

Es opcional soportar la supresión de los bits de relleno y su uso debe ser negociado.

Send: Si el sistema soporta las funciones de inserción y supresión de bits de relleno, y se van a utilizar, y la instrucción no incluyó un RCD o lo incluyó con el atributo T38FaxFillBitRemoval presente, entonces se incluirá T38FaxFillBitRemoval y se utilizará la función de inserción y supresión de bits de relleno. En todos los demás casos, NO SE INCLUIRÁ el atributo T38FaxFillBitRemoval y NO SE USARÁ la función de inserción y supresión de bits de relleno.

Receive: Si el atributo T38FaxFillBitRemoval no está presente, NO SE USARÁ la inserción y supresión de bits de relleno.

T38FaxTranscodingMMR:

La transcodificación MMR no aplica a T.38 basado en UDPTL.

Send: Si se usa UDPTL para T.38, NO SE INCLUIRÁ el atributo T38FaxTranscodingMMR.

Receive: Si el atributo T38FaxTranscodingMMR está presente para T.38 basada en UDPTL, se RECHAZARÁ esta instrucción (código de error 505 – RemoteConnectionDescriptor no soportado).

T38FaxTranscodingJBIG:

La transcodificación JBIG no aplica a T.38 basado en UDPTL.

Send: Si se usa UDPTL para T.38, NO SE INCLUIRÁ el atributo T38FaxTranscodingJBIG.

Receive: Si el atributo T38FaxTranscodingJBIG está presente para T.38 basada en UDPTL, se RECHAZARÁ esta instrucción (código de error 505 – RemoteConnectionDescriptor no soportado).

8.4.4.3 Media Announcements (m=) (anuncios de medios)

Los anuncios de medios (m=) constan de 4 subcampos:

```
m= <media> <port> <transport> <fmt list>  
"m= image 3456 udptl t38"
```

Media (Medios):

Send: Se UTILIZARÁ el tipo de medios 'image' para T.38 basado en UDPTL.

Receive: El tipo recibido SERÁ 'image' para T.38 basado en UDPTL.

Port (Puerto):

Send: Será conforme a RFC 2327. El puerto especificado es el puerto de recepción. El puerto de envío puede ser diferente.

Receive: Será conforme a RFC 2327. El puerto especificado es el puerto de recepción. El puerto de envío puede ser diferente.

Transport (Transporte):

Send: Se UTILIZARÁ el protocolo de transporte 'udptl' para T.38 basado en UDPTL.

Receive: Para T.38 basado en UDPTL, el protocolo de transporte será 'udptl'. Las implementaciones también DEBERÍAN admitir la forma en mayúsculas "UDPTL", así como la mezcla de mayúsculas y minúsculas en la cadena "udptl".

Media Formats (Formatos de medios):

Send: El formato de medios SERÁ "t38".

Receive: El formato de medios SERÁ "t38".

8.5 Transmisión utilizando el protocolo UDP

8.5.1 Entrega fiable de mensajes

Los mensajes MGCP se transmiten por el protocolo UDP. Las instrucciones se envían a una de las direcciones IP definidas en el sistema de nombres de dominio (DNS) para el punto extremo o MGC especificado. Las respuestas se envían en retorno a la dirección fuente de la instrucción. Ahora bien, la respuesta puede proceder de otra dirección IP diferente de aquella a la que se envió la instrucción.

Cuando no se proporciona un puerto para el punto extremo²⁸, las instrucciones se enviarán al puerto MGCP por defecto, que es 2427 para las instrucciones enviadas a pasarelas, y 2727 para las enviadas a los controladores de medios. Para evitar los inconvenientes de compatibilidad con versiones anteriores, SE RECOMIENDA que el controlador de pasarela de medios indique explícitamente el puerto MGCP que usará en mensajes TGCP (evitan la opción del valor por defecto).

Los mensajes MGCP, transportados por UDP, pueden sufrir pérdidas. En ausencia de una respuesta tempestiva, se repiten las instrucciones. Se prevé que las entidades MGCP memoricen una lista de las respuestas enviadas a transacciones recientes, es decir, una lista de todas las respuestas enviadas en los últimos T_{hist} segundos, así como una lista de las transacciones que se están ejecutando en ese momento. Los identificadores de transacción de instrucciones entrantes se comparan con los identificadores de transacción de las respuestas recientes. Si en una de las comparaciones los identificadores concuerdan, la entidad MGCP no ejecuta la transacción, y se limita a repetir la respuesta. Si no se encuentra ningún caso de concordancia de los identificadores, la entidad MGCP examina la lista de las transacciones que están en ejecución en ese momento. Si se encuentra un caso de concordancia, la entidad MGCP no ejecuta la transacción, y simplemente la ignora.

Incumbe a la entidad solicitante proporcionar periodos de temporización adecuados para todas las instrucciones pendientes, e intentar de nuevo las instrucciones cuando hayan expirado los periodos de temporización. En 8.5.2 se especifica una estrategia de retransmisión.

Además, si después de repetidas las instrucciones no se ha obtenido respuesta, se supone que la entidad de destino no está disponible. Incumbe a la entidad solicitante buscar servicios redundantes y/o liberar conexiones existentes o pendientes, como se especifica en 7.4.

8.5.2 Estrategia de retransmisión

En esta Recomendación se ha evitado especificar valores estáticos para los temporizadores de retransmisión, pues estos valores generalmente dependen de la red. Normalmente, los temporizadores de retransmisión deben determinar el periodo de temporización midiendo el tiempo que transcurre entre el instante en que se envía una instrucción y el instante en que se devuelve una respuesta. Las pasarelas de circuitos troncales implementarán una estrategia de retransmisión con variación exponencial del tiempo de espera, configurándose los valores inicial y máximo.

Las pasarelas de circuitos troncales DEBERÍAN utilizar el algoritmo implementado en TCP-IP, que emplea dos variables:

- El retardo de acuse de recibo medio (AAD, *average acknowledgement delay*), calculado mediante un promedio alisado exponencialmente de los retardos de respuesta observados.
- La desviación media, (ADEV, *average deviation*) calculada mediante un promedio alisado exponencialmente del valor absoluto de la diferencia entre el retardo de respuesta observado y el promedio actual.

El valor del temporizador de retransmisión (RTO) en TCP será la suma del retardo promedio más N veces la desviación promedio, siendo N una constante.

Después de cualquier retransmisión, la entidad MGCP debería hacer lo siguiente:

- multiplicar por dos el valor estimado del retardo promedio, AAD;
- calcular un valor aleatorio, uniformemente distribuido entre 0,5 AAD y AAD;

²⁸ Es posible configurar una dirección MGC y un puerto MGC para cada punto extremo.

- fijar el temporizador de retransmisión (RTO) al mínimo de estos dos valores:
 - la suma del valor aleatorio y N veces la desviación promedio;
 - $RTO_{m\acute{a}x}$, siendo el valor por defecto para $RTO_{m\acute{a}x}$ 4 segundos.

Este procedimiento produce un doble efecto. Por el hecho de incluir un componente que aumenta exponencialmente, reduce automáticamente la velocidad del tren de mensajes en caso de congestión, de acuerdo con las necesidades de la comunicación en tiempo real. Por el hecho de incluir un componente aleatorio, rompe la sincronización potencial entre notificaciones ocasionadas por el mismo evento externo.

El valor inicial utilizado para el temporizador de retransmisión es 200 milisegundos, por defecto, y el valor máximo para el temporizador de retransmisión es 4 segundos, por defecto. Estos valores por defecto pueden ser modificados por el proceso de configuración.

8.5.3 Tamaño máximo del datagrama, fragmentación y recomposición

Los mensajes TGCP transmitidos por UDP dependen de IP para la fragmentación y recomposición de datagramas grandes. El tamaño máximo teórico de un datagrama IP es 65 535 bytes. Si el encabezamiento IP tiene 20 bytes y el encabezamiento UDP 8 bytes, se tiene un tamaño máximo teórico de los mensajes TGCP de 65 507 bytes, si se usa UDP.

Sin embargo, IP no exige que los equipos reciban datagramas de más de 576 bytes (RFC 1122), lo que resultaría en un tamaño de mensaje TGCP inaceptablemente pequeño. En consecuencia, con el protocolo TGCP las implementaciones tienen que soportar datagramas TGCP de 4000 bytes o más, lo que implica que se deba soportar la correspondiente fragmentación y reensamble IP. Obsérvese que el límite de 4000 bytes es válido para al nivel de TGCP. Las taras en una capa inferior exigirán que se soporten datagramas más grandes: la tara de UDP e IP tiene 28 bytes o más, y puede haber más tara, por ejemplo cuando utiliza IPsec.

Se señala que esto se aplica tanto a los agentes de llamada como a los puntos extremos. Los agentes de llamada pueden auditar los puntos extremos para determinar si soportan datagramas TGCP de tamaño mayor que el aquí especificado. Actualmente los puntos extremos no pueden determinar si los agentes de llamada soportan tamaños mayores de datagrama TGCP.

8.6 Remolque o envío de instrucciones adjuntas

En algunos casos es conveniente que el agente de llamada envíe varios mensajes simultáneamente a uno o más puntos extremos en una pasarela, y viceversa. Cuando hay que enviar varios mensajes en los mismos paquetes UDP, los mensajes se separan mediante una línea de texto en la que sólo hay un punto, por ejemplo:

```
200 2005 OK
.
DLCX 1210 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
```

Los mensajes remolcados se PROCESARÁN como si se hubieran recibido en varios datagramas diferentes. Hay que procesar en orden cada uno de los mensajes del datagrama hasta su término, empezando por el primero y respondiendo a cada instrucción. Los errores que aparezcan en un mensaje remolcado NO AFECTARÁN a ninguno de los otros mensajes recibidos en ese paquete. Es decir que cada mensaje se procesa independientemente.

Se puede utilizar el remolque con dos fines:

- Garantizar que los mensajes se entregan y procesan en orden.
- Asimilar los resultados de los mensajes.

Si el procedimiento de remolque se utiliza para garantizar la entrega ordenada de mensajes, las entidades mantendrán el principio de entrega ordenada en la retransmisión de mensajes individuales. Un ejemplo de esto es el envío de varias instrucciones Notify utilizando remolque (conforme a lo descrito en 7.4.3.1).

La asimilación de resultados de mensajes garantiza que, o bien se entregan todos los mensajes, o no se entrega ninguno. Si el remolque se utiliza con este fin, las entidades también MANTENDRÁN este principio en la retransmisión. Por ejemplo, tras recibir una instrucción Notify de un punto extremo que funciona en modo lockstep, puede ser conveniente que el agente de llamada envíe la respuesta junto con una nueva instrucción NotificationRequest en un mismo datagrama, para garantizar que las dos tienen el mismo resultado.

8.7 Identificadores de transacción y triple toma de contacto

Los identificadores de transacción son números enteros comprendidos en la gama de 1 a 999 999 999. Los MGC pueden optar por utilizar un espacio de número específico para cada una de las pasarelas que gestionan, o por utilizar el mismo espacio de número para todas las pasarelas que pertenecen a cierto grupo de pasarelas. Los MGC pueden optar por repartir la carga de la gestión de una pasarela de gran tamaño entre varios procesos independientes. Estos procesos compartirán el mismo espacio de número de transacción. Esta compartición puede realizarse de varias maneras, por ejemplo previendo una ubicación centralizada de los identificadores de transacción, o preasignando a diferentes procesos gamas de identificadores que no se superpongan unas a otras. Las implementaciones garantizarán que se asignan identificadores de transacción únicos a todas las transacciones procedentes de cualquier MGC, y que hayan sido enviadas a cualquier pasarela dentro de un periodo de T_{hist} segundos. Las pasarelas pueden detectar fácilmente las transacciones duplicadas, examinado solamente el identificador de transacción.

El parámetro acuse de recibo de respuesta puede encontrarse en cualquier instrucción. Este parámetro transporta un conjunto de "gamas de identificadores de transacción confirmada" para respuestas finales recibidas; las respuestas provisionales NO SERÁN confirmadas.

Las pasarelas MGCP pueden optar por suprimir las copias de las respuestas a transacciones cuyo identificador esté incluido en "gamas de identificadores de transacción confirmada" recibidas en un mensaje, pero HAY QUE MANTENER un registro de transacción ejecutada durante T_{hist} segundos. Asimismo, cuando se recibe un mensaje de acuse de recibo de respuesta²⁹, la respuesta de la cual se acusa recibo puede suprimirse. Las pasarelas deberán descartar las ulteriores instrucciones que reciban de ese MGC, sin ejecutar ninguna otra acción, cuando el identificador de transacción esté comprendido en estas gamas y hayan transcurrido menos de T_{hist} segundos desde la emisión de la respuesta.

Sean $term_{new}$ y $term_{old}$ el nombre de punto extremo en una nueva instrucción (cmd_{new}) y en una instrucción anterior (cmd_{old}) respectivamente. Los identificadores de transacciones por confirmar en cmd_{new} DEBERÍAN determinarse conforme a los siguientes principios:

- 1) Si $term_{new}$ no contiene comodines:
 - a) Respuestas no confirmadas a instrucciones anteriores, cuando $term_{old}$ es igual a $term_{new}$.
 - b) Facultativamente, una o más respuestas no confirmadas, cuando $term_{old}$ contenía el comodín "any-of", y el nombre de punto extremo devuelto en la respuesta era $term_{new}$.
 - c) Facultativamente, una o más respuestas no confirmadas, cuando $term_{old}$ contenía el comodín "all", y $term_{new}$ concuerda con la sintaxis de comodín de $term_{old}$.

²⁹ Por oposición a una instrucción con un parámetro acuse de recibo de respuesta.

- d) Facultativamente, una o más respuestas no confirmadas, cuando $term_{old}$ contenía el comodín "any-of", no se ha devuelto ningún nombre de punto extremo, y $term_{new}$ concuerda con la sintaxis de comodín de $term_{old}$.
- 2) Si $term_{new}$ contiene el comodín "all":
 - a) Facultativamente, una o más respuestas no confirmadas, cuando $term_{old}$ contenía el comodín "all", y $term_{new}$ concuerda con la sintaxis de comodín de $term_{old}$.
 - 3) Si $term_{new}$ contiene el comodín "any of":
 - a) Facultativamente, una o más respuestas no confirmadas, cuando $term_{old}$ contenía el comodín "all", y $term_{new}$ concordará con la sintaxis de comodín de $term_{old}$ si se reemplazara el comodín "any of" en $term_{new}$ por el comodín "all".

Una determinada respuesta NO DEBERÍA confirmarse en dos mensajes distintos.

Los siguientes ejemplos ilustran la aplicación de estas reglas:

- Si $term_{new}$ es "ds/ds1-2/1" y $term_{old}$ es "ds/ds1-2/1", la respuesta anterior puede ser confirmada de acuerdo con la regla 1a.
- Si $term_{new}$ es "ds/ds1-1/3" y $term_{old}$ es "*", la respuesta anterior puede ser confirmada de acuerdo con la regla 1c.
- Si $term_{new}$ es "ds/ds1-2/*" y $term_{old}$ es "*", la respuesta anterior puede ser confirmada de acuerdo con la regla 2a.
- Si $term_{new}$ es "ds/ds1-2/\$" y $term_{old}$ es "ds/ds1-2/*", la respuesta anterior puede ser confirmada de acuerdo con la regla 3a.

Los valores de "gammas de identificadores de transacciones confirmadas" NO DEBERÍAN utilizarse si han transcurrido más de T_{hist} segundos desde que la pasarela envió su última respuesta a ese MGC, o cuando una pasarela entra nuevamente en actividad. En esta situación, las instrucciones deberían aceptarse y procesarse sin efectuar ninguna prueba sobre el identificador de transacción.

Asimismo, una respuesta NO DEBERÍA confirmarse si han transcurrido más de T_{hist} segundos desde la recepción.

Los mensajes que confirman respuestas pueden transmitirse y recibirse en un orden diferente. La pasarela mantendrá la vinculación lógica de los identificadores de transacción confirmada recibidos en instrucciones recientes.

8.8 Respuestas provisionales

En algunos casos se otorga considerablemente el tiempo de ejecución. El TGCP utiliza UDP como el protocolo de transporte, y la fiabilidad se consigue mediante retransmisiones selectivas basadas en la expiración de un periodo de temporización basado en una estimación de la suma del tiempo de propagación de ida y retorno de la red y el tiempo de finalización de las transacciones. Por tanto, una varianza significativa del tiempo de ejecución de las transacciones plantea problemas cuando se desea una rápida detección de la pérdida de mensajes sin una tara excesiva.

En consecuencia, para resolver este problema se emitirá una respuesta provisional, pero únicamente si el tiempo de ejecución de las transacciones es mayor que un periodo de tiempo breve (se RECOMIENDA 200 ms). La respuesta provisional acusa recibo de la instrucción aunque todavía no se sepa el resultado de ésta, por ejemplo porque la reserva de recursos está pendiente. Por regla general, se emitirán respuestas provisionales para transacciones que no se puedan finalizar sin una comunicación externa, por ejemplo, la reserva de un recurso de red. Además, si se recibe una instrucción CreateConnection o ModifyConnection duplicada, y todavía no ha terminado la ejecución de la transacción, se DEBE enviar una respuesta provisional.

Una semántica transaccional pura implicaría que las respuestas provisionales debieran informar únicamente que la transacción se encuentra en ejecución en ese momento; sin embargo, un enfoque optimista, que incluya otras informaciones, permite reducir el retardo del sistema.

Las respuestas provisionales SE ENVIARÁN solamente en respuesta a una instrucción CreateConnection o ModifyConnection. Para reducir el retardo del sistema, en la respuesta provisional a la instrucción CreateConnection SE INCLUIRÁ un identificador de conexión y una descripción de sesión. Si la instrucción ModifyConnection hace que se comunique una descripción de sesión, la descripción de sesión también SE INCLUIRÁ en la respuesta provisional. Si la transacción concluye con éxito, la información retornada en la respuesta provisional SE REPETIRÁ en la respuesta final. Se considera un error de protocolo no repetir esta información o cambiar en una respuesta de ejecución satisfactoria cualquiera de las informaciones previamente suministradas. Si la transacción fracasa, se retorna un código de error; la información retornada anteriormente deja de ser válida.

Una transacción CreateConnection o ModifyConnection que esté en ejecución DEBE anularse si se recibe una instrucción DeleteConnection para el punto extremo. En ese caso, de todas formas, se DEBERÍA devolver automáticamente una respuesta para la transacción anulada, y hay QUE DEVOLVER una respuesta para la transacción anulada si se detecta una retransmisión de la transacción anulada (DEBERÍA utilizarse código de error 407).

Cuando se recibe una respuesta provisional, HAY QUE ESTABLECER un periodo de temporización para la transacción en cuestión significativamente mayor ($T_{longtran}$). Este temporizador tiene por finalidad principal detectar el fallo de un punto extremo. El valor por defecto de $T_{longtran}$ es 5 segundos, pero puede ser modificado en la configuración.

Cuando termina la ejecución de la transacción, se envía la respuesta final, y se suprime la respuesta provisional, la cual ha perdido su vigencia. Para asegurar una rápida detección de una respuesta final perdida se DEBE acusar recibo de las respuestas finales emitidas después de respuestas provisionales. Por tanto, el punto extremo incluirá un parámetro "ResponseAck" vacío en esas respuestas finales, y sólo en esas. La presencia del parámetro "ResponseAck" en la respuesta final ocasionará el envío de una respuesta "acuse de recibo de respuesta" en retorno, al punto extremo. La respuesta "acuse de recibo de respuesta" incluirá, en el encabezamiento de respuesta, el identificador de transacción de la respuesta de la que se acusa recibo. La recepción de esta respuesta "acuse de recibo de respuesta" está sometida al mismo periodo de temporización y a las mismas estrategias y procedimientos de retransmisión que las respuestas a las instrucciones (véase 7.4), es decir, el emisor de la respuesta final la retransmitirá si el "acuse de recibo de respuesta" no se recibe en tiempo oportuno. No se acusa recibo de la respuesta "acuse de recibo de respuesta".

9 Seguridad

Si las entidades no autorizadas pudieran utilizar el MGCP, podrían establecer comunicaciones no autorizadas o interferir en llamadas autorizadas. Debe señalarse que no se proporciona seguridad como parte integrante de MGCP, sino que el MGCP presupone la existencia de una capa inferior que proporciona la seguridad efectiva.

En la Rec. UIT-T J.170 se especifican requisitos de seguridad y se presentan soluciones aplicables en el TGCP; para una información más detallada debe consultarse dicha Recomendación.

Anexo A

Lotes de eventos

En este anexo se define un conjunto inicial de lotes de eventos para los diversos tipos de puntos extremos definidos actualmente en IPCablecom para las pasarelas de circuitos troncales.

Cada lote tiene un nombre, y tiene un código y una definición para cada uno de los eventos en el lote. Los cuadros de eventos/señales para cada lote tienen cinco columnas:

- **Código** El código de evento, único en el lote, utilizado para el evento/señal.
- **Descripción** Una breve descripción del evento/señal.
- **Evento** En esta columna aparece una marca de comprobación si el evento puede ser solicitado por el MGC. Como alternativa, puede aparecer uno o más de los siguientes símbolos:
 - **"P"** indica que el evento es persistente;
 - **"S"** indica que el evento es un estado de evento que se puede auditar;
 - **"C"** indica que el evento/señal puede ser detectado/aplicado en una conexión.
- **Señal** Si la casilla en esta columna para un evento está en blanco, el MGC no puede señalar el evento en una instrucción. En otro caso, los siguientes símbolos identifican el tipo de evento:
 - **"OO"** Señal On/Off. La señal está activa hasta que el MGC ordene que sea desactivada, y viceversa.
 - **"TO"** Señal sometida a un periodo de temporización. La señal está activa durante cierto periodo de tiempo, a menos que sea reemplazada por una nueva señal. Se suministran valores de periodo de temporización por defecto. Un valor cero indica que el periodo de temporización es infinito. Estos valores por defecto pueden modificarse en la configuración.
 - **"BR"** Señal breve. El evento tiene una corta duración, conocida.
- **Información adicional** Proporciona información adicional sobre el evento/señal, por ejemplo, la duración por defecto de señales TO.

A menos que se indique otra cosa, todos los eventos/señales son detectados/aplicados en puntos extremos, y el audio generado por esos eventos/señales no se reenvía por ninguna conexión del punto extremo. Por el contrario, el audio generado por eventos/señales que son detectados/aplicados en una conexión se reenviará por la conexión asociada, cualquiera que sea el modo de conexión.

A.1 Lote de circuitos troncales ISUP

Nombre del lote: IT.

Se utilizan los siguientes códigos para identificar eventos y señales del lote "IT". Las MG que soporten el lote "IT" TIENEN QUE SOPORTAR todos estos eventos y señales.

Cuadro A.1/J.171.1 – Eventos y señales del lote de circuitos troncales ISUP

Código	Descripción	Evento	Señal	Información adicional
co1	Tono de continuidad 1	√	TO	Temporización = 3 segundos
co2	Tono de continuidad 2	√	TO	Temporización = 3 segundos
ft	Tono de facsímil	√	–	
ld	Conexión de larga duración	C	–	
ma	Principio de medios	C	–	
mt	Tono de módem	√	–	
oc	Operación finalizada	√	–	
of	Operación infructuosa	√	–	
ro	Tono de circuito ocupado	–	TO	Temporización = 30 segundos
rt	Tono de llamada de retorno	–	C, TO	Temporización = 180 segundos
TDD	Tonos de dispositivos de telecomunicaciones para sordos (TDD, <i>telecommunications devices for the deaf</i>)	√		

Definiciones de los distintos eventos y señales:

Tono de continuidad 1 (co1): Un tono de 2010 Hz de acuerdo con la Rec. UIT-T Q.724. Para ajustarse a las actuales prácticas en materia de pruebas de continuidad, el evento NO DEBERÍA generarse antes de suprimir el tono. El tono es de tipo TO; el tono de continuidad sólo se aplicará durante el periodo de tiempo especificado. El valor por defecto se puede modificar en la configuración.

Tono de continuidad 2 (co2): Un tono de 1780 Hz de acuerdo con la Rec. UIT-T Q.724. Para ajustarse a las actuales prácticas en materia de pruebas de continuidad, el evento NO DEBERÍA generarse antes de suprimir el tono. El tono es de tipo TO; el tono de continuidad sólo se aplicará durante el periodo de tiempo especificado. El valor por defecto se puede modificar en la configuración.

Los tonos de continuidad se utilizan cuando el MGC inicia una prueba de continuidad. Hay dos tipos de pruebas de continuidad: con un solo tono y con dos tonos. La parte que inicia la prueba de continuidad comprueba las señales y detecta los tonos adecuados para el circuito troncal en cuestión. Por ejemplo, para una prueba de continuidad de un circuito a cuatro hilos a un circuito a dos hilos podrían utilizarse los siguientes mensajes:

Pasarela de origen

```
RQNT 1234 ds/ds3-1/ds1-6/17@tgw1.example.net
X: AB123FE0
S: co2
R: co1
```

Pasarela de terminación

```
CRCX 1234 ds/ds1-4/7@tgw2.example.net
C: A3C47F21456789F0
L: p:10, a:PCMU
M: conttest
```

La pasarela de origen envía la señal solicitada, y espera el tono adecuado para el circuito troncal en cuestión. Cuando detecta ese tono y si considera que la prueba de continuidad ha tenido éxito,

genera el evento "co1" que en el ejemplo se notifica al MGC. Si la prueba no tiene éxito antes de la expiración del periodo de temporización, se genera un evento de operación finalizada y se envía en este caso al MGC. De manera similar, si se produce un error antes de la expiración del periodo de temporización, se genera un evento "fallo de operación". Los eventos "oc" y "of" se parametrizarán con el nombre del evento/señal que ellos informan, esto es, "co1" en este caso.

Tono de facsímil (ft): El evento tono de facsímil se genera cuando se detecta una comunicación de facsímil (véase, por ejemplo, la Rec UIT-T T.30, o la Rec UIT-T V.21).

Conexión de larga duración (ld): Se detecta la "conexión de larga duración" cuando se determina que una conexión ha estado establecida durante un periodo de tiempo mayor que un determinado periodo de tiempo. El valor por defecto es una hora, pero este valor por defecto se puede modificar en la configuración.

El evento puede detectarse en una conexión. Cuando no se especifica una conexión, el evento es aplicable a todas las conexiones para el punto extremo, cualquiera que sea el momento en que se crearon las conexiones.

Principio de medios (ma): El evento principio de medios se produce en una conexión cuando se recibe por ésta el primer paquete válido de medios RTP³⁰. Este evento puede utilizarse para sincronizar una señal local, por ejemplo, la señal de llamada de retorno, con la llegada de medios procedentes de la otra parte.

El evento puede detectarse en una conexión. Cuando no se especifica una conexión, el evento es aplicable a todas las conexiones para el punto extremo, cualquiera que sea el momento en que se crearon las conexiones.

Tonos de módem (mt): El evento tonos de módem se genera cuando se detecta una comunicación en la que interviene un módem (véase por ejemplo, la Rec UIT-T V.8).

Operación finalizada (oc): El evento operación finalizada se genera cuando se pide a la pasarela que aplique una o más señales de tipo TO en el punto extremo, y una o más de esas señales han concluido sin que hayan sido detenidas por la detección de un evento solicitado, como por ejemplo un "tono de continuidad 1". El informe de finalización puede contener como parámetro el nombre de la señal que llegó al final de su tiempo de actividad, como en:

O: IT/oc(IT/co1)

Cuando la señal informada se aplica en una conexión, el parámetro suministrado incluirá también el nombre de la conexión, como por ejemplo:

O: IT/oc(IT/rt@0A3F58)

Cuando se solicita el evento operación finalizada, no se le puede añadir ningún parámetro de evento. Si se omite el nombre de lote, se supone que tiene el nombre de lote por defecto.

El evento operación finalizada se puede generar también como se define en el protocolo de base, por ejemplo, cuando una instrucción ModifyConnection insertada concluye con éxito, como en³¹:

O: IT/oc(B/C)

³⁰ Cuando se utilizan servicios de seguridad de autenticación e integridad, un paquete RTP no se considerará válido hasta que haya pasado las verificaciones de seguridad.

³¹ Obsérvese que en este ejemplo se utiliza "B" como el prefijo para el parámetro informado.

Operación infructuosa (of): En general, el evento operación infructuosa puede generarse cuando se pide al punto extremo que aplique una o varias señales de tipo TO en el punto extremo, y una o más de esas señales fallan antes de la expiración del periodo de temporización. El informe de finalización puede contener como parámetro el nombre de la señal que falló, como en:

O: IT/of(IT/co2)

Cuando la señal informada se aplica en una conexión, el parámetro suministrado incluirá también el nombre de la conexión, como por ejemplo en:

O: IT/of(IT/rt@0A3F58)

Cuando se solicita el evento operación infructuosa, no se puede especificar parámetros de evento. Si se omite el nombre de lote, se supone que tiene el nombre de lote por defecto.

El evento operación infructuosa se puede generar también como se define en el protocolo de base, por ejemplo, cuando una instrucción ModifyConnection insertada fracasa, como en:

O: IT/of(B/C(M(sendrecv(AB2354))))

Obsérvese que en esta expresión se utiliza "B" como prefijo para el parámetro informado.

Tono de circuito ocupado (ro): El tono de circuito ocupado, o tono de congestión, se especifica en la Rec. UIT-T E.180/Q.35.

Tono de llamada de retorno (rt): El tono de llamada audible se especifica en la Rec. UIT-T E.180/Q.35. La definición del tono viene determinada por las características nacionales del tono de llamada de retorno y PUEDE establecerse en el proceso de provisión. La señal de llamada de retorno puede aplicarse a un punto extremo y a una conexión.

Tonos de dispositivos de telecomunicaciones para sordos (TDD): El evento TDD se genera cuando se detecta una comunicación TDD (véase por ejemplo, la Rec. UIT-T V.18).

Apéndice I

Interacciones de modos

Una conexión MGCP puede establecer uno o más trenes de medios. Estos trenes de medios son entrantes (procedentes de un punto extremo distante) o salientes (generados en el punto extremo del circuito). El parámetro "modo de conexión" determina el sentido de transmisión y la generación de estos trenes. Cuando un punto extremo tiene establecida una sola conexión, la correspondencia de estos trenes es directa; el punto extremo del circuito aplica el tren entrante al circuito y genera el tren saliente a partir de la señal del circuito, de acuerdo con el modo indicado por el parámetro modo.

En cambio, cuando un punto extremo tiene establecidas varias conexiones, puede haber muchos trenes entrantes y salientes. Según el modo de conexión utilizado, estos trenes pueden interactuar diferentemente unos con otros, y con los trenes que van hacia el punto extremo o que vienen del punto extremo.

Cuadro I.1/J.171.1 – Reagrupamiento de diferentes conexiones cuando una o más conexiones están simultáneamente activas

		Modo de la conexión A					
		sendonly	recvonly	sendrecv	loopback/ conttest	inactive	netwloop/ netwttest
Modo de la conexión 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/ netwttest						$A_{out}=A_{in}$ $B_{out}=B_{in}$ $H_{out}=NA$

El cuadro I.1 describe la forma en que deben combinarse diferentes conexiones cuando una o más conexiones están concurrentemente "activas". En este contexto, por conexión activa ha de entenderse una conexión que está en uno de los modos siguientes:

- "envío/recepción";
- "sólo envío";
- "sólo recepción".

Las conexiones que están en los modos "conexión en bucle de red", "prueba de continuidad de red", o "inactivo" no son afectadas por conexiones que se encuentran en el modo "activo". En el cuadro I.1 se han seguido los siguientes convenios:

- A_{in} es el tren de medios entrante procedente de la conexión A.
- B_{in} es el tren de medios entrante procedente de la conexión B.
- H_{in} es el tren de medios entrante procedente del circuito troncal.
- A_{out} es el tren de medios saliente que va a la conexión A.
- B_{out} es el tren de medios saliente que va a la conexión B.
- H_{out} es el tren de medios saliente que va al punto extremo, donde "cot" indica prueba de continuidad, sea en el modo "prueba de continuidad", sea en el modo "conexión en bucle".
- NA indica que no hay ningún tren.

Apéndice II

Ejemplos de codificación de instrucciones

En este apéndice se presentan ejemplos de instrucciones y respuestas, con las formas de codificación realmente utilizadas. Se presentan ejemplos para cada instrucción. Todos los comentarios que aparecen en las instrucciones y respuestas son facultativos.

II.1 NotificationRequest

El primer ejemplo muestra una NotificationRequest que inicia una prueba de continuidad y busca la verificación de la prueba. La "entidad notificada" para el punto extremo se fija a "ca@ca1.whatever.net:5678" y el RequestIdentifier se repite en la correspondiente instrucción Notify:

```
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: col, oc(N), of(N)
S: col
```

La respuesta indica que la transacción tuvo éxito:

```
200 1201 OK
```

II.2 Notify

El ejemplo que sigue ilustra un mensaje Notify que notifica una prueba de continuidad exitosa como se indica por los eventos observados. Como en la NotificationRequest que ocasionó la activación se especificó una "entidad notificada", ésta se repite aquí. Además, se incluye el RequestIdentifier para correlacionar esta instrucción Notify con la instrucción NotificationRequest que ocasionó la activación:

```
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: col
```

La respuesta Notify indica que la transacción tuvo éxito:

```
200 2002 OK
```

II.3 CreateConnection

El primer ejemplo ilustra una instrucción CreateConnection para crear una conexión en el punto extremo especificado. La conexión será parte del CallId especificado. Los parámetros de LocalConnectionOptions especifican un códec ley μ G.711, y que el periodo de paquetización será de 10 ms. El modo de conexión será "receive only":

```
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 respuesta indica que la transacción tuvo éxito, por lo que se incluye un identificador de conexión para la conexión recientemente creada. También se incluye una descripción de sesión para la nueva conexión (obsérvese que va precedida de una línea vacía):

```
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
```

El segundo ejemplo ilustra una instrucción CreateConnection que contiene una petición de notificación y un 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 respuesta indica que la transacción fracasó porque el circuito troncal ya había sido tomado. En consecuencia, no se retorna un identificador de conexión ni una descripción de sesión:

```
401 2005 Circuit already seized
```

En el tercer ejemplo se ilustra la utilización de la respuesta provisional y la triple toma de contacto:

```
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
```

Inicialmente se retorna una respuesta provisional:

```
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
```

Obsérvese que el punto extremo eligió soportar únicamente el códec PCMU, es decir, la cabida útil número 0.

Poco después se recibe la respuesta final:

```
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
```

El MGC acusa recibo de la respuesta final, como se solicitó:

```
000 1206
```

y se ha finalizado la transacción.

II.4 ModifyConnection

El primer ejemplo muestra una instrucción ModifyConnection que simplemente pone la conexión en modo "send/receive"; también se fija la "entidad notificada":

```
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 respuesta indica que la transacción tuvo éxito:

```
200 1209 OK
```

En el segundo ejemplo, se transfiere una descripción de sesión y se incluye una petición de notificación con la instrucción ModifyConnection. El punto extremo comenzará a aplicar tonos de llamada de retorno a la red telefónica pública conmutada (RTPC) y los aplicará hasta que detecte audio en la conexión especificada para la señal de llamada de retorno:

```
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 respuesta indica que la transacción tuvo éxito:

```
200 1206 OK
```

II.5 DeleteConnection (procedente del controlador de pasarela de medios (MGC))

En este ejemplo, el MGC ordena simplemente a la pasarela de circuitos troncales que suprima la conexión FDE234C8 en el punto extremo especificado:

```
DLCX 1210 ds/ds1-1/1@tgw.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
I: FDE234C8
```

La respuesta indica que la transacción tuvo éxito y que se suprimió la conexión. Por tanto, también se incluyen parámetros (de conexión) para la conexión:

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

II.6 DeleteConnection (procedente de la pasarela de circuitos troncales)

En este ejemplo, la pasarela de circuitos troncales envía una instrucción DeleteConnection al MGC para comunicarle que se ha suprimido una conexión en el punto extremo especificado. El ReasonCode especifica el motivo para la supresión, y también se proporciona parámetros (de conexión) para la conexión:

```
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
```

El MGC envía una respuesta de éxito a la pasarela:

```
200 1210 OK
```

II.7 DeleteConnection (procedente del controlador de pasarela de medios, en caso de múltiples conexiones)

En el primer ejemplo, el MGC ordena a la pasarela de circuitos troncales que suprima todas las conexiones relacionadas con la llamada "A3C47F21456789F0" en el punto extremo especificado:

```
DLCX 1210 ds/ds1-1/6@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
C: A3C47F21456789F0
```

La respuesta indica éxito y que la conexión o conexiones fueron suprimidas:

```
250 1210 OK
```

En el segundo ejemplo, el MGC ordena a la pasarela de circuitos troncales que suprima todas las conexiones relacionadas con todos los puntos extremos especificados:

```
DLCX 1210 ds/ds1-1/*@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
```

La respuesta indica éxito:

```
250 1210 OK
```

II.8 AuditEndpoint

En el primer ejemplo, el MGC desea averiguar qué puntos extremos están presentes en la pasarela de circuitos troncales especificada, por lo que utiliza el comodín "all of" para la porción local del nombre de punto extremo. El MGC sólo busca dos nombres de punto extremo:

```
AUEP 1200 *@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
ZM: 2
```

La pasarela de circuitos troncales indica éxito e incluye una lista de hasta dos nombres de punto extremo. Un total de 24 nombres de punto extremo concordaron con el comodín especificado:

```
200 1200 OK
Z: ds/ds1-1/1@tgw-2567.whatever.net
Z: ds/ds1-1/2@tgw-2567.whatever.net
ZN: 24
```

En el segundo ejemplo, se solicitan las capacidades de uno de los puntos extremos:

```
AUEP 1201 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
F: A
```

La respuesta indica éxito, y también las capacidades. El sistema soporta dos códecs que tienen capacidades diferentes. En consecuencia, la respuesta devuelve dos conjuntos distintos de capacidades:

```
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
```

En el tercer ejemplo, el MGC hace una auditoría para todas las informaciones posibles relacionadas con el punto extremo:

```
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 respuesta indica éxito:

```
200 2002 OK
R: IT/ft,mt (N)
S:
X: 0123456789B1
N: [128.96.41.12]
I: 32F345E2
T: ft
O:
ES:
```

La lista de eventos solicitados contiene dos eventos. Cuando no se especifica ningún nombre de lote, se supone el lote por defecto. La misma regla es aplicable a las acciones, por lo que se debe suponer la acción por defecto (Notify) para el evento "IT/ft". El hecho de que no se haya dado ningún valor para las "SignalRequests" significa que no hay señales activas en ese momento. La "entidad notificada" actual hace referencia a una dirección IP y existe una sola conexión para el punto extremo. El valor actual de DetectEvents es "ft", y la lista de ObservedEvents está vacía, como también lo está la de EventStates.

II.9 AuditConnection

El primer ejemplo muestra una instrucción AuditConnection para auditoría de los parámetros CallId, NotifiedEntity, LocalConnectionOptions, ConnectionMode, LocalConnectionDescriptor y los parámetros de la conexión:

```
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 respuesta indica éxito e incluye información para la 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
```

En el segundo ejemplo se solicita una auditoría sobre RemoteConnectionDescriptor y LocalConnectionDescriptor:

```
AUCX 1203 ds/ds1-1/2@tgw.whatever.net MGCP 1.0 TGCP 1.0
I: FDE234C8
F: RC,LC
```

La respuesta indica éxito e incluye información para la RequestedInfo. En este caso no existe RemoteConnectionDescriptor, por lo que sólo se incluye el campo versión de protocolo para el RemoteConnectionDescriptor:

```
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 RestartInProgress

El primer ejemplo ilustra un mensaje RestartInProgress enviado por una pasarela de circuitos troncales para informar al MGC que el punto extremo será retirado del servicio en 300 segundos:

```
RSIP 1200 ds/ds1-1/1@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
RM: graceful
RD: 300
```

La respuesta del MGC indica que la transacción tuvo éxito:

```
200 1200 OK
```

En el segundo ejemplo, el mensaje RestartInProgress enviado por la pasarela de circuitos troncales informa al MGC que todos los puntos extremos de la pasarela de circuitos troncales se ponen en servicio en 0 segundos, esto es, que están restablecidos al servicio. También se hubiera podido omitir el plazo:

```
RSIP 1204 *@tgw-2567.whatever.net MGCP 1.0 TGCP 1.0
RM: restart
RD: 0
```

La respuesta del MGC indica éxito y, además, proporciona a los puntos extremos en cuestión una nueva "entidad notificada":

```
200 1204 OK
N: MGC-1@whatever.net
```

Como otra posibilidad, la respuesta podría haber indicado que la transacción ha fracasado, y proporcionar una nueva "entidad notificada", como en:

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

En tal caso, hubiera sido necesario reintentar la instrucción para cumplimentar el "procedimiento de reenganche" (véase 7.4.3.5), esta vez yendo a MGC "MGC-1@whatever.net".

Apéndice III

Ejemplo de flujo de llamada

En este apéndice se presenta un ejemplo de flujo de llamada entre un usuario en la red (on-net) que utiliza un MTA y un protocolo de señalización³² no especificados, y un usuario fuera de la red (off-net) conectado a través de una pasarela de circuitos troncales que utiliza el protocolo TGCP y una pasarela de señalización que soporta la señalización de PU de la RDSI del SS7. Debe observarse que este flujo de llamada, aunque es válido, no representa más que un ejemplo, que puede o no seguirse en la práctica.

En el flujo de llamada que se muestra en la figura III.1, CMS (*call management server*) designa el servidor de gestión de llamadas, MGC designa el controlador de pasarela de medios, TGW designa la pasarela de circuitos troncales, y SG designa la pasarela de señalización:

³² Este protocolo podría ser NCS o DCS.

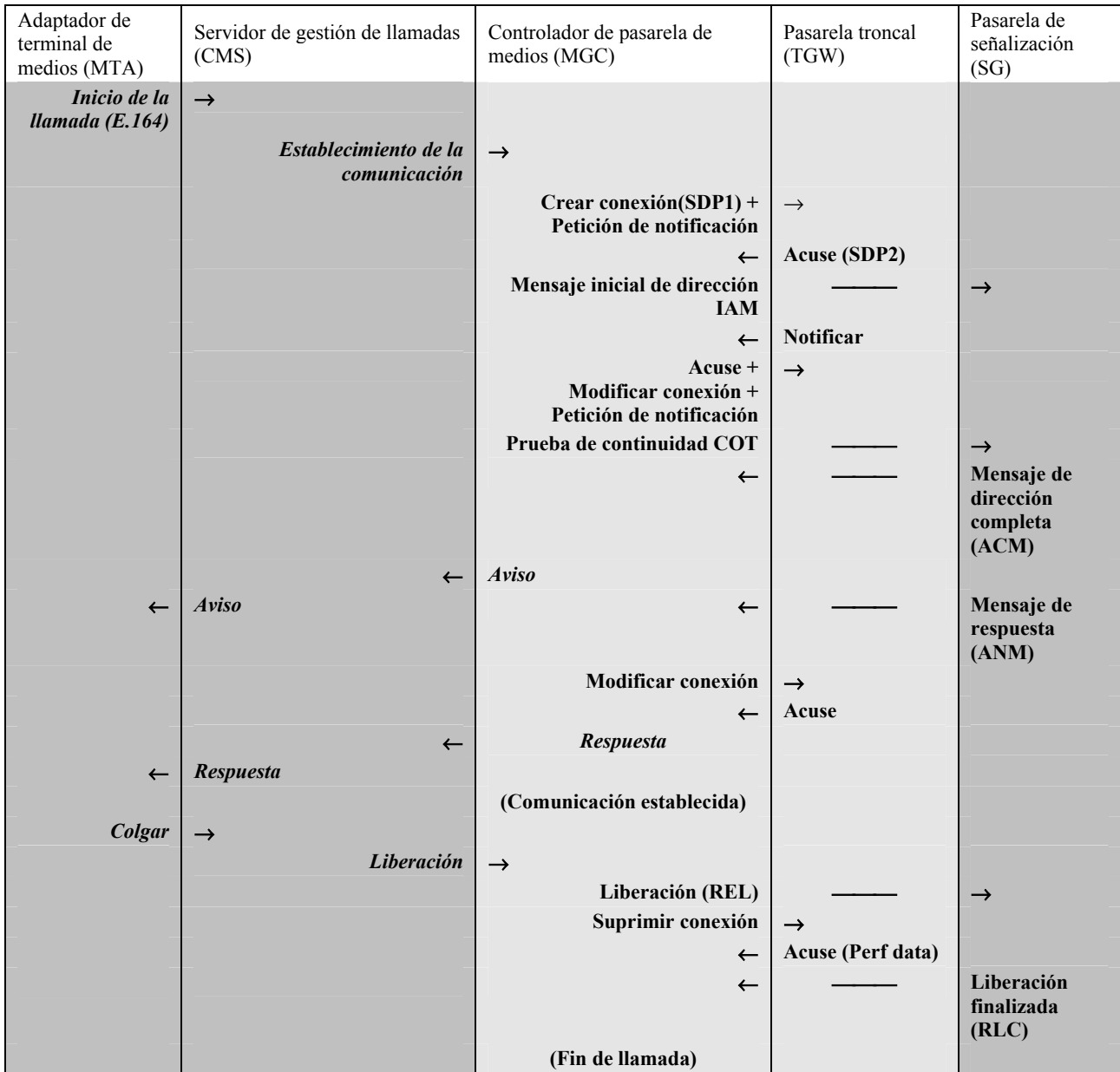


Figura III.1/J.171.1 – Ejemplo de flujo de llamada

Durante estos intercambios, el MGC utiliza el perfil TGCP del MGCP para controlar la pasarela de circuitos troncales. Se supone que entre el MTA, el CMS y el MGC se aplica un protocolo no identificado.

Se ha partido del supuesto de que el MTA envía (directa o indirectamente) al MGC un mensaje para establecer una comunicación vocal con un número de teléfono E.164 y que ha añadido una descripción de sesión a su petición. El CMS busca el número E.164 solicitado y determina que necesita hacer una llamada fuera de la red (off-net), por lo que establece contacto con el MGC apropiado. El MGC determina que debe efectuar la llamada a través de la pasarela de circuitos troncales `tgw.whatever.net`. Además, el MGC determina que se debe realizar una prueba de continuidad para esta llamada.

La primera instrucción es una instrucción combinada CreateConnection y NotificationRequest enviada a la pasarela de circuitos troncales:

```
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: col
```

```
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
```

Entonces se ordena a la pasarela de circuitos troncales que inicie la prueba de continuidad, que busque el resultado de la prueba, y que lo informe. La generación de la señal de prueba de continuidad y la detección de su éxito (o fracaso) mediante el mecanismo de eventos están sincronizadas, de manera que cuando se produce el evento "co2", se detiene la prueba "col". Obsérvese que el punto extremo eligió soportar únicamente el códec PCMU, es decir, la cabida útil 0. La parte (de la instrucción) relativa a la creación de conexión ordena crear una conexión inactiva en el punto extremo especificado, de acuerdo con G.711, con un periodo de paquetización de 10 ms. Además, la instrucción incluye la descripción de sesión recibida del MTA de origen.

La pasarela de circuitos troncales de egreso acusará recibo de la instrucción y enviará en la descripción de sesión sus propios parámetros, como la dirección, puertos y perfil RTP, así como el identificador de conexión para la nueva conexión:

```
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
```

El MGC envía un mensaje IAM SS7 a través de la pasarela de señalización al conmutador conectado al circuito troncal por el que se efectúa la llamada. En este mensaje se incluye una indicación de que se ha realizado la prueba de continuidad.

Se ha supuesto también que la prueba de continuidad tiene éxito, por lo que se genera el evento "co2" y se notifica al MGC:

```
NTFY 3001 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0
X: 0123456789B0
O: co2
```

El MGC envía un mensaje COT SS7 que indica "éxito de la prueba de continuidad" al conmutador distante y acusa recibo de la instrucción Notify recibida. Asimismo, envía en remolque una instrucción combinada ModifyConnection y NotificationRequest por la que ordena a la pasarela que efectúe la conexión en modo "sólo recepción" y comience la búsqueda tonos de facsímil y de módem:

```
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
```

En esta fase, el MGC ha establecido un trayecto de transmisión semidúplex. El teléfono asociado al MTA de ingreso podrá recibir las señales, tales como tonos y anuncios, que podrían generarse en caso de error, así como el material hablado que muy probablemente se generará cuando el usuario de egreso conteste al teléfono.

El MGC recibe entonces un mensaje ACM SS7 que indica que se está avisando a la parte llamada, y seguidamente un mensaje ANM SS7 que indica que la parte llamada ha contestado. El MGC efectúa la conexión en modo dúplex enviando la siguiente instrucción ModifyConnection a la pasarela de circuitos troncales:

```
MDCX 2007 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0  
C: A3C47F21456789F0  
I: 32F345E2  
M: sendrecv
```

La pasarela de circuitos troncales responde inmediatamente a la instrucción con:

```
200 2007 OK
```

Al mismo tiempo el MGC informa al MTA de origen sobre el evento contestación a la llamada y registra la hora de la contestación a la llamada.

En este momento, la comunicación está totalmente establecida.

Ulteriormente se cuelga el teléfono asociado al MTA de origen en este ejemplo, y se transmite al MGC (directamente, o a través del CMS como se muestra en este ejemplo) un evento de colgar, con lo que se indica al MGC que se debe terminar la llamada.

El MGC verifica que la llamada debe realmente desconectarse, por ejemplo por el hecho de que no existe un dispositivo de retención de llamada, y envía entonces un mensaje REL SS7 al conmutador distante, y también una instrucción DeleteConnection a la pasarela de circuitos troncales:

```
DLCX 2009 ds/ds1-1/6@tgw.whatever.net MGCP 1.0 TGCP 1.0  
C: A3C47F21456789F0  
I: 32F345E2
```

La pasarela de circuitos troncales responderá con un acuse de recibo que incluirá los parámetros de conexión para la conexión:

```
250 2009 OK  
P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

El MGC recibe también una confirmación de la desconexión de la llamada en forma de un mensaje RLC SS7 y, por último, registra la terminación de la llamada.

Apéndice IV

Requisitos de los puntos extremos

En este apéndice se define un conjunto de requisitos específicos de los puntos extremos TGCP.

IV.1 Modos de conexión soportados

En el cuadro IV.1 se indican los modos de conexión que un determinado punto extremo TGCP TIENE QUE soportar:

Cuadro IV.1/J.171.1 – Lista de modos de conexión que deben estar soportados por un punto extremo TGCP

Tipo de punto extremo	Información adicional de punto extremo	sendonly	recvonly	sendrecv	inactive	loopback	conftest	netwloop	netwtest
DS-0	Circuito troncal PU-RDSI	√	√	√	√	√	√	√	√
DS-0	Circuito troncal MF	√	√	√	√	–	–	√	√

Apéndice V

Información sobre compatibilidad

Este apéndice proporciona información de compatibilidad relativa al protocolo TGCP.

V.1 Compatibilidad con NCS

Esta versión de TGCP se basa en la Rec. UIT-T J.162, con la que se ha alineado lo más posible. Puesto que TGCP y NCS están destinados a dos tipos diferentes de pasarelas, presentan ciertas diferencias, que se recapitulan a continuación:

- **Modos de conexión:** NCS y TGCP tienen un conjunto de modos de conexión que son comunes a ambos protocolos, pero cada uno de ellos tiene un conjunto de modos de conexión que no están soportados por el otro:
 - NCS soporta los modos de conexión "conferencia" y "replicación", que no están soportados por TGCP.
 - TGCP soporta los modos de conexión "prueba de continuidad" y "conexión en bucle", que no están soportados por NCS.
- **Mapas de dígitos:** TGCP no soporta mapas de dígitos, pero sí NCS. Esto tiene algunas implicaciones, por ejemplo:
 - Ninguna instrucción TGCP puede aceptar un mapa de dígitos como parámetro.
 - TGCP no soporta la acción "acumular de acuerdo con mapa de dígitos".
 - No es posible hacer auditoría del "mapa de dígitos".
- **Calidad de servicio dinámica:** NCS soporta la señalización de calidad de servicio dinámica IPCablecom, la cual no está soportada en TGCP.

Además de las diferencias antes mencionadas relativas a los modos de conexión de los protocolos, entre NCS y TGCP existen también las siguientes diferencias que no se relacionan con los protocolos.

- **Lotes de eventos:** Los lotes de eventos iniciales en TGCP NCS son diferentes.
- **Esquema de denominación de puntos extremos:** Los esquemas de denominación de los puntos extremos TGCP y de los puntos extremos NCS son algo diferentes.

V.2 Compatibilidad con MGCP

TGCP (y NCS) es además un perfil del MGCP 1.0 de IETF RFC 2705. Sin embargo, TGCP también ha introducido algunas adiciones. A continuación se indican las adiciones introducidas TGCP que no están incluidas actualmente en MGCP:

- **Esquema de denominación de puntos extremos:** Se ha introducido un esquema de denominación de puntos extremos específico para puntos extremos DS-0. Las reglas para uso de comodín son más estrictas que en MGCP; también se ha introducido el concepto de "gama" para puntos extremos DS-0.
- **ModifyConnection insertada:** Se ha introducido una nueva acción ModifyConnection insertada.
- **Seguridad:** Los servicios de seguridad IPCablecom están soportados en TGCP. Esto influye en las LocalConnectionOptions, Capabilities, y SDP.

- **Consulta de nombres de punto extremo:** La instrucción AuditEndpoint se ha ampliado con una capacidad para retornar el número de puntos extremos que son abarcados por un comodín, y con un mecanismo para la consulta, bloque por bloque, de estos nombres de punto extremo. Además de ampliar la instrucción AuditEndpoint, esto implica la introducción de dos nuevos nombres de parámetros: MaxEndPointIds y NumEndpoints
- **Versiones soportadas:** La respuesta RestartInProgress y la instrucción AuditEndpoint se han ampliado con un parámetro VersionSupported para permitir que los MGC y pasarelas determinen las versiones de protocolo que cada uno soporta.
- **Códigos de error:** Se han introducido dos nuevos códigos de error: 532 y 533.
- **Utilización de SDP:** Se ha incluido en TGCP un nuevo perfil de utilización de SDP. Un aspecto digno de mención es que el perfil y todos los ejemplos de utilización requieren específicamente una estricta observancia de SDP, independientemente de la utilidad que ofrezcan los campos incluidos. Además, se han añadido a SDP extensiones específicas de IPCablecom.
- **Respuesta provisional:** Se ha incluido en TGCP detalles adicionales y la recomendación de un mecanismo de respuesta provisional. Se ha introducido una respuesta Acuse de recibo de respuesta (000), se ha permitido un parámetro ResponseAck vacío en respuestas las finales que sigan a respuestas provisionales, y se ha especificado un procedimiento para el mecanismo.
- **Parámetros de señal:** La sintaxis de parámetro de señal se ha ampliado para que permita la utilización de paréntesis equilibrados dentro de parámetros de señal. El valor del periodo de temporización de todas las señales de temporización puede ser modificado por un parámetro de señal.
- **Lotes de eventos:** TGCP introduce un conjunto de nuevos lotes de eventos.

Por último, debe señalarse que TGCP proporciona interpretaciones del comportamiento del protocolo MGCP de base, y en algunos casos una recomendación y una aclaración adicionales de dicho comportamiento, que pueden o no reflejar el comportamiento deseado del protocolo MGCP.

Apéndice VI

Gramática ABNF para TGCP

En la RFC 3435 se presenta una descripción formal de la sintaxis del protocolo MGCP con arreglo a la "Forma aumentada de Backus-Naur (ABNF, *augmented Backus-Naur form*) para especificaciones de sintaxis". Los programadores se remiten a esta descripción formal durante la concepción de dispositivos interfuncionales. En el presente apéndice se presenta una copia comentada y modificada de la sintaxis del protocolo MGCP en la que se indica su aplicabilidad a las Recomendaciones IPCablecom.

Las aplicaciones DEBERÍAN cumplir con las partes de esta gramática ABNF pertinente a sus especificaciones respectivas, es decir, NCS, TGCP. Obsérvese también que hay algunas pocas codificaciones de parámetros (por ejemplo, solicitud insertada, mapa de dígitos, nombres extendidos de fabricantes) en los que la gramática NCS y/o la gramática TGCP difieren de la gramática MGCP.

Se utilizan cinco comentarios para distinguir los siguientes cinco casos diferentes:

- 1) Se ha modificado el lenguaje de la RFC a fin de incluir los requisitos de NCS y TGCP.
- 2) El lenguaje de la RFC se puede aplicar a NCS y posiblemente a MGCP, pero no a TGCP.
- 3) El lenguaje de la RFC se puede aplicar a TGCP y posiblemente a MGCP, pero no a NCS.
- 4) El lenguaje de la RFC se puede aplicar únicamente a NCS y a MGCP.
- 5) El lenguaje de la RFC se puede aplicar únicamente a MGCP.

En cada caso se indica el lenguaje mediante el uso de un tipo de letra particular según se especifica a continuación:

;Se modificó la gramática de RFC 3435 a fin de incluir NCS y TGCP

```
; El texto en negrilla indica NCS únicamente (y posiblemente MGCP)
; El texto en itálica indica TGCP únicamente (y posiblemente TGCP)
; El texto en negrilla y en itálica indica NCS y TGCP únicamente
; El texto gris indica MGCP únicamente
```

```
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) ; experimental starts with X
transaction-id = 1*9(DIGIT)
endpointName   = LocalEndpointName "@" DomainName
LocalEndpointName = LocalNamePart 0*("/") LocalNamePart
LocalNamePart  = AnyName / AllName / NameString
AnyName        = "$"
AllName        = "*"
NameString     = 1*(range-of-allowed-characters)
; VCHAR except "$", "*", "/", "@"
range-of-allowed-characters = %x21-23 / %x25-29 / %x2B-2E
                             / %x30-3F / %x41-7E
DomainName     = 1*255(ALPHA / DIGIT / "." / "-") ; as defined
               / "#" number ; in RFC 821
               / "[" IPv4address / IPv6address "]" ; see RFC 2373
; Rewritten to ABNF from RFC 821
number = 1*DIGIT
;From RFC 2373
IPv6address = hexpart [ ":" IPv4address ]
```

```

IPv4address = 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT
; this production, while occurring in RFC 2373, is not referenced
; 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" ; For NCS
              / "TGCP 1.0" ; For TGCP
              / VCHAR *( WSP / VCHAR)
MGCPPParameter = ParameterValue EOL
; Check infoCode if more parameter values defined
; Most optional values can only be omitted when auditing
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]) ; For NCS (and 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]) ; Auditing only
                / ("MD" ":" 0*(WSP) MaxMGCPDatagram) ; Auditing only
                / (extensionParameter ":" 0*(WSP) [parameterString])
                / VersionSupported ; NCS and TGCP - response only
                / MaxEndpointIds ; NCS and TGCP
                / NumEndpoints ; NCS and TGCP - response only
; <extensionParameter> ":" parameterString defined by NCS and TGCP
VersionSupported = "VS" ":" MGCPversion *( "," 0*(WSP) MGCPversion)
MaxEndpointIds = "ZM" ":" 0*(WSP) 1*16(DIGIT)
NumEndpoints = "ZN" ":" 0*(WSP) 1*16(DIGIT) ; Responses only
; A final response may include an empty ResponseAck
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)
; The audit request response may include a list of identifiers
ConnectionId = 1*32(HEXDIG) 0*( "," 0*(WSP) 1*32(HEXDIG))
SecondConnectionID = ConnectionId
NotifiedEntity = [LocalName "@"] DomainName [":" portNumber]
LocalName = LocalEndpointName ; No internal structure

```

```

portNumber = 1*5(DIGIT)
RequestIdentifier = 1*32(HEXDIG)
LocalConnectionOptions = LocalOptionValue 0*(WSP)
                        0*("," 0*(WSP) LocalOptionValue 0*(WSP))
LocalOptionValue = ("p" ":" packetizationPeriod)
                  / ("a" ":" compressionAlgorithm)
                  / ("b" ":" bandwidth) ; Only for capabilities in
                  ; NCS and TGCP
                  / ("e" ":" echoCancellation)
                  / ("gc" ":" gainControl)
                  / ("s" ":" silenceSuppression)
                  / ("t" ":" typeOfService)
                  / ("r" ":" resourceReservation)
                  / ("k" ":" encryptiondata)
                  / ("nt" ":" ( typeOfNetwork /
                                supportedTypeOfNetwork) )
                  / (LocalOptionExtensionName
                     [":" LocalOptionExtensionValue])
                  / MPacketizationPeriod ; NCS and TGCP only
                  / RTPciphersuite ; NCS and TGCP only
                  / RTCPciphersuite ; NCS and TGCP only
                  / DQoSGateID ; NCS only
                  / DQoSReservation ; NCS only
                  / DQoSResourceID ; NCS only
                  / DQoSReserveDestination ; NCS only
                  / CallContentId ; TGCP only
                  / CallContentDestination ; TGCP only

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)
; For NCS and TGCP, range format is only allowed for capabilities
; and not for LocalConnectionOptions.
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 only for capabilities
resourceReservation = "g" / "cl" / "be"
;encryption parameters are coded as in SDP (RFC 2327)
;NOTE: encryption key may contain an algorithm as specified in RFC 1890
encryptiondata = ( "clear" ":" encryptionKey )
                 / ( "base64" ":" encodedEncryptionKey )
                 / ( "uri" ":" URIToObtainKey )
                 / ( "prompt" ) ; defined in SDP, not usable in MGCP!
encryptionKey = 1*(SuitableLCOCharacter) / quotedString
; See RFC 2045
encodedEncryptionKey = 1*(ALPHA / DIGIT / "+" / "/" / "=")
URIToObtainKey = 1*(SuitableLCOCharacter) / quotedString
typeOfNetwork = "IN" / "ATM" / "LOCAL" / OtherTypeOfNetwork
; Registered with IANA - see RFC 2327
OtherTypeOfNetwork = 1*(SuitableLCOCharacter)
supportedTypeOfNetwork = typeOfNetwork *("; typeOfNetwork)
supportedModes = ConnectionMode 0*("; ConnectionMode)
supportedPackages = packageName 0*("; packageName)

```

```

packageName = 1*(ALPHA / DIGIT / HYPHEN) ; Hyphen neither first or last
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>
; defined by 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>
; defined by NCS only
DQoSGateID           = "dq-gi" [":" 1*8(HEXDIG)] ; Only empty for
; capabilities
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>
; defined by TGCP only
CallContentId        = "es-cci" ":" 1*8(HEXDIG)
CallContentDestination = "es-ccd" ":" IPv4address ":" portNumber

LocalOptionExtensionValue = (1*(SuitableExtLCOValChar)
                             / quotedString)
                          *( ";" (1*(SuitableExtLCOValChar)
                                  / quotedString))

;Note: No "data" mode.
ConnectionMode = "sendonly" / "recvonly" / "sendrecv"
               / "confrnce" / "inactive"
               / "loopback" / "conttest" ; TGCP (and MGCP) only
               / "replcate" ; NCS only
               / "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
              / "*" / "#") ; for DTMF
            ["@" (ConnectionId / "$" / "*")]
eventId = 1*(ALPHA / DIGIT / HYPHEN) ; Hyphen neither first nor last
eventRange = [" 1*(DigitMapLetter / (DIGIT "-" DIGIT) /
              (DTMFLetter "-" DTMFLetter)) "]"
DTMFLetter = "A" / "B" / "C" / "D"
requestedActions = requestedAction 0*("," 0*(WSP) requestedAction)
requestedAction  = "N" / "A"
                  / "D" ; For NCS (and MGCP)
                  / "S" / "I" / "K"
                  / "E" ["(" EmbeddedRequest ")"]
                  / ExtensionAction

```

```

        / "C" "(" EmbeddedModeChange ; For NCS and TGCP
          0*("," 0*WSP EmbeddedModeChange) ")" ; only
; NCS and TGCP define the Embedded ModifyConnection action.
; MGCP grammar does not allow for the format used in NCS and TGCP:
EmbeddedModeChange = "M" "(" ConnectionMode (" EmConnectionId ")" )" "
EmConnectionId     = ConnectionId / "$"
ExtensionAction    = PackageExtAction
PackageExtAction   = packageName "/" Action [(" ActionParameters ")"]
Action              = 1*ALPHA
ActionParameters   = eventParameters ; May contain actions
;NOTE: Should tolerate different order when receiving, e.g. for NCS
EmbeddedRequest = (
    "R" "(" EmbeddedRequestList ")"
    [ "," 0*(WSP) "S" "(" EmbeddedSignalRequest ")" ]
    [ "," 0*(WSP) "D" "(" EmbeddedDigitMap ")" ]
    / (
        "S" "(" EmbeddedSignalRequest ")"
        [ "," 0*(WSP) "D" "(" EmbeddedDigitMap ")" ]
    )
    / (
        "D" "(" EmbeddedDigitMap ")"
    )
) / NCSTGCPEmbeddedRequest
;Text below is for NCS and TGCP only. The difference compared to MGCP
;is simply that the order of the items is not fixed. Also for TGCP Digit Maps
;are not used
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
; For NCS (and MGCP)
DigitMap = DigitString / "(" DigitStringList ")"
DigitStringList = DigitString 0*( "|" DigitString )
DigitString = 1*(DigitStringElement)
DigitStringElement = DigitPosition [ "." ]
DigitPosition = DigitMapLetter / DigitMapRange
; NOTE "X" is now included
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 "[x]" is now allowed in MGCP.
; In NCS, only the "x" form is allowed
DigitMapRange = "[" 1*DigitLetter "]"
                / "X" ; Added for NCS only
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
; NCS and TGCP define the following four connection parameter extension
; names:
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 = VendorCPEExtensionName
                                / PackageCPEExtensionName
VendorCPEExtensionName = "X" "-" 2*ALPHA
                        / NCSTGCPVendorCPEExtensionName
;Text below is for NCS and TGCP only. The difference compared to MGCP
;is simply that MGCP requires 2 alpha characters whereas NCS and TGCP
;allow 2 or 3 alpha characters for VendorCPEExtensionName
NCSTGCPVendorCPEExtensionName = "X" "-" 2*3ALPHA
PackageCPEExtensionName = packageName "/" CPName
CPName = 1*(ALPHA / DIGIT / HYPHEN)
ConnectionParameterExtensionValue = 1*9(DIGIT)
MaxMGCPDatagram = 1*9(DIGIT)
ReasonCode = 3DIGIT
                [1*(WSP) "/" packageName] ; Only for 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" ; For NCS (and MGCP) only
           / "O" / "P" / "E" / "Z" / "Q" / "T" / "RC" / "LC"
           / "A" / "ES" / "RM" / "RD" / "PL" / "MD" / extensionParameter
           / "VS" / "ZM" / "ZN" ; NCS and TGCP define these
                               ; three extensionParameters
;NCS and TGCP allows for process and loop control in either order
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)

```

```

; must not start with "x-" or x+"
OtherExtensionParameter = 1*32(ALPHA / DIGIT / HYPHEN)

;If first character is a double-quote, then it is a quoted-string
parameterString = (%x21 / %x23-7F) *(%x20-7F) ; first and last must not
; be white space
/ quotedString
MGCPResponse = MGCPResponseLine 0*(MGCPPParameter)
*2(EOL *SDPinformation)
MGCPResponseLine = responseCode 1*(WSP) transaction-id
[1*(WSP) "/" packageName] ; Only for 8xx
[WSP responseString] EOL

responseCode = 3DIGIT
responseString = *(%x20-7E)
SuitablePkgExtLCOCharacter = SuitableLCOCharacter
SuitableExtLCOCharacter = DIGIT / ALPHA / "+" / "-" / "_" / "&"
/ "!" / "'" / "|" / "=" / "#" / "?"
/ "." / "$" / "*" / "@" / "[" / "]"
/ "^" / "`" / "{" / "}" / "~"
SuitableLCOCharacter = SuitableExtLCOCharacter / "/"
SuitableExtLCOValChar = SuitableLCOCharacter / ":"
; VCHAR except "", "(", ")", ",", and "="
SuitableEventParamCharacter = %x21 / %x23-27 / %x2A-2B
/ %x2D-3C / %x3E-7E

; NOTE: UTF8 encoded
quotedString = DQUOTE 0*(quoteEscape / quoteChar) DQUOTE
quoteEscape = DQUOTE DQUOTE
quoteChar = (%x00-21 / %x23-FF)
EOL = CRLF / LF
HYPHEN = "-"
; See RFC 2327 for proper SDP grammar instead.
SDPinformation = SDPLine CRLF *(SDPLine CRLF) ; see RFC 2327
SDPLine = 1*(%x01-09 / %x0B / %x0C / %x0E-FF) ; for proper def.

```


Apéndice VII

Vigilancia electrónica

VII.1 MGC

El siguiente es el formato de los parámetros para vigilancia electrónica en los parámetros LocalConnectionOptions de una instrucción CRCX o MDCX enviadas a la MG:

- El identificador de conexión de contenido de llamada codificado como la palabra clave "es-cci" seguida de un carácter dos puntos (":") y una cadena de hasta ocho caracteres hexadecimales que corresponde a un identificador con una longitud de 32 bits para el identificador de conexión de contenido de llamada.
- El destino de contenido de llamada codificado como la palabra clave "es-ccd" seguida de un carácter dos puntos (":") y una dirección IP codificada en forma similar a la de una dirección IP para la porción nombre de dominio de un nombre de punto extremo. La dirección IP va seguida de un carácter dos puntos (":") y hasta cinco caracteres decimales para un número de puerto UDP que habrá de utilizarse.

La MGC incluirá los parámetros "es-cci" y "es-ccd" en LocalConnectionOptions de las instrucciones CRCX y MDCX, al notificar una MG los parámetros de vigilancia electrónica.

El siguiente ejemplo ilustra una instrucción CRCX con parámetros de vigilancia electrónica:

```
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
```

El siguiente ejemplo ilustra una instrucción MDCX con parámetros de vigilancia electrónica:

```
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 La MG

Si una MG recibe una instrucción CRCX en la que "es-cci" y "es-ccd" de los parámetros LCO no están vacías, INICIARÁ la duplicación, reenvío y encapsulamiento de todos los paquetes recibidos y transmitidos por la conexión. El proceso de duplicación, reenvío y encapsulamiento de todos los paquetes de una conexión es la vigilancia del contenido de las llamadas. La MG que realice vigilancia del contenido de las llamadas duplicará, reenviará y encapsulará todos los paquetes que genera para la conexión. Los paquetes encapsulados serán iguales a los paquetes de la conexión. La MG encapsulará todos los paquetes duplicados y reenviados cuidando de incluir el identificador de conexión de contenido de la llamada en el campo "es-cci" de los parámetros LCO. La MG REENVIARÁ todos los paquetes duplicados y encapsulados a la dirección IP y puerto UDP indicados en el campo "es-cci" de LCO.

La MG que realiza vigilancia del contenido de llamadas FINALIZARÁ la vigilancia del contenido de llamadas cuando suceda alguno de los siguientes eventos:

- 1) Se finaliza la conexión debido a un DLCX del MGC.
- 2) Se finaliza la conexión debido a un DLCX de la MG.

- 3) Se finaliza la conexión debido a un error interno, por ejemplo:
 - si hay fallos en los componentes de la MG;
 - si el DS1 en el que se encuentre el punto extremo quede fuera de servicio.
- 4) La MG recibe una MDCX con el campo "es-cci" vacío ("es-cci:") o con el campo "es-ccd" vacío ("es-ccd:").

Si la MG que realiza vigilancia del contenido de llamadas recibe una instrucción MDCX con parámetros LCO que contienen parámetros "es-cci" o "es-ccd" válidos y nuevos, UTILIZARÁ los nuevos parámetros "es-cci" o "es-ccd" al realizar vigilancia del contenido de llamadas.

Si una MG recibe una instrucción CRCX o una MDCX con los campos "es-cci" y "es-ccd" y no soporta vigilancia electrónica, pero está en capacidad de ejecutar la instrucción salvo por la parte relativa a la vigilancia electrónica, la MG DEVOLVERÁ el código de respuesta 210 (la transacción solicitada se ejecutó normalmente, pero la pasarela no soporta vigilancia electrónica).

Si una MG recibe una instrucción CRCX o una MDCX con los campos "es-cci" y "es-ccd" y soporta vigilancia electrónica, y está en capacidad de ejecutar la instrucción salvo por la parte relativa a la vigilancia electrónica por tener recursos limitados, la MG DEVOLVERÁ el código de respuesta 211 – La transacción solicitada se ejecutó normalmente, pero la pasarela no pudo efectuar vigilancia electrónica por falta de recursos.

Si una MG recibe una instrucción CRCX o una MDCX que sólo tiene uno de los campos "es-cci" o "es-ccd", no los dos, y está en capacidad de la instrucción salvo por la parte relativa a la vigilancia electrónica, la MG DEVOLVERÁ el código de respuesta 212 – La transacción solicitada se ejecutó normalmente, pero los parámetros LCO no contenían todos los parámetros necesarios para vigilancia electrónica.

Si una MG que no está realizando vigilancia del contenido de llamadas en una conexión recibe una instrucción MDX con los campos "es-cci" o "es-ccd" para dicha conexión, DEVOLVERÁ el código de retorno 213 – La transacción solicitada se ejecutó normalmente, pero no se puede dar inicio a vigilancia electrónica en el transcurso de las llamadas.

Si una MG recibe una instrucción CRCX o una MDCX con un campo "es-cci" o "es-ccd" inutilizable, y está en capacidad de ejecutar la instrucción salvo por la parte relativa a la vigilancia electrónica, la MG DEVOLVERÁ el código de retorno 214 – La transacción solicitada se ejecutó normalmente, pero no se reconocen los parámetros de vigilancia electrónica. Si la MG que recibe la instrucción MDCX con los parámetros "es-cci" o "es-ccd" inutilizables ya estaba realizando vigilancia del contenido de llamadas, la MG SEGUIRÁ realizando vigilancia del contenido de llamadas sin tener en cuenta los parámetros "es-cci" y "es-ccd" de la MDCX.

Cuando una MG reciba una instrucción AuditEndpoint con el parámetro de capacidades, DEVOLVERÁ la palabra clave "es" si soporta vigilancia electrónica.

Si una MG que realiza vigilancia del contenido de llamadas recibe una instrucción AuditEndpoint para esa conexión, y LocalConnectionsOptions es uno de los parámetros auditados, la MG DEVOLVERÁ en los LCO los parámetros "es-cci" y "es-ccd" que la MG está utilizando para efectuar la vigilancia del contenido de llamadas.

Apéndice VIII

Ejemplos de lotes de eventos

VIII.1 Lote de servicios de operador MF FGD

Nombre de lote: MO

Los códigos del cuadro VIII.1 se utilizan para identificar eventos y señales del lote "MO" para la "señalización de servicios de operador" en el caso de circuitos troncales de servicios de operador MF unidireccionales salientes. La señalización de servicios de operador MF FGC también está soportada. Este lote se utilizará para circuitos troncales de servicios de operador generales, y también para circuitos troncales de emergencia especializados:

Cuadro VIII.1/J.171.1 – Códigos utilizados para identificar eventos y señales de lote MO

Código	Descripción	Evento	Señal	Información adicional
ans	Contestación de llamada	P	–	
ft	Tono de facsímil	√	–	
ld	Conexión de larga duración	C	–	
mt	Tono de módem	√	–	
orbk	Señal de llamada de retorno de operador	√	–	
rbz	Marca de ocupación	P	–	
rcl	Rellamada de operador	–	BR	
rel	Liberar llamada	P	BR	
res	Reanudar llamada	–	BR	
rlc	Liberación completa	P, S	BR	
sup(<addr>, <id>)	Establecimiento de comunicación	–	TO	Temporización variable
sus	Suspender llamada	–	BR	
swk	Start wink	√	–	
TDD	Tonos de dispositivos de telecomunicaciones para sordos (TDD)	√		
oc	Operación finalizada	√		
of	Operación fracasada	√		

Definiciones de los distintos eventos y señales:

Contestación de llamada (ans): La contestación de llamada se produce cuando se hace la petición ANI del sistema de soporte de operaciones (OSS, *operations support system*), esto es, la llamada puede no haber sido pasada necesariamente a un operador. Después de producirse la contestación de llamada, se establecerá el dispositivo de retención de llamada, es decir, a partir de este momento sólo el OSS puede liberar el circuito troncal.

Tono de facsímil (ft): Se genera el evento tono de facsímil cuando se detecta una llamada de facsímil (véase por ejemplo, la Rec. UIT-T T.30, o V.21).

Conexión de larga duración (ld): Se detecta "conexión de larga duración" cuando una conexión ha estado establecida durante un periodo de tiempo mayor que cierto valor dado. El valor por defecto es una hora, pero puede modificarse en la configuración.

Este evento puede detectarse en una conexión. Cuando no se especifica ninguna conexión, el evento es aplicable a todas las conexiones del punto extremo, cualquiera que sea el momento en que se crearon.

Tonos de módem (mt): Se genera el evento tono de módem cuando se detecta una llamada de módem (véase por ejemplo, la Rec. UIT-T V.8).

Señal de llamada de retorno de operador (orbk): Se genera este evento cuando el OSS pide que se avise a la parte llamante³³.

Marca de ocupación (rbz): Se produce este evento cuando el OSS marca el circuito troncal. Se generará un evento liberación cuando el circuito troncal deja de estar ocupado.

Rellamada de operador (rcl): Se puede aplicar esta señal para invocar una rellamada de operador, por ejemplo, debida a una señal de gancho conmutador hecha por el cliente para obtener el retorno del operador.

Liberar llamada (rel): Esta señal puede enviarse a la pasarela de medios; sin embargo, si está establecido un dispositivo de retención de llamada, la llamada no será desconectada hasta que el OSS la libere. La pasarela de medios genera un evento "liberar llamada" cuando se considera que el OSS ha liberado el circuito troncal. En este caso, el evento puede parametrizarse con uno de los códigos de causa del cuadro VIII.2, que indican el motivo de la liberación:

Cuadro VIII.2/J.171.1 – Códigos de motivos de liberación de llamada

Código de causa	Motivo
0	Liberación normal
3	No hay ruta al destino
8	Preapropiación
19	Ausencia de respuesta
21	Llamada rechazada
27	Destino fuera de servicio
28	Formato de número no válido (por ejemplo, dirección incompleta)
38	Red fuera de servicio
111	Error de protocolo/señalización, no especificado (por ejemplo, expiración de periodo de temporización)

Reanudar llamada (res): Esta señal indica que la otra parte reanudó la llamada, es decir, descolgó.

Liberación completa (rlc): El punto extremo y el MGC utilizan este evento/señal para confirmar que la llamada ha sido liberada y que el circuito troncal está disponible para una nueva llamada.

Establecimiento de comunicación (sup(<addr>, <id>)): Establece una comunicación al sistema de servicios de operador utilizando la información de dirección y de identificación proporcionada. La información de dirección será de la forma siguiente:

addr (MF₁, MF₂, ..., MF_n)

³³ Si el teléfono de la parte llamante está colgado, normalmente se aplicará una corriente de timbre; en cambio, si está descolgado, normalmente se aplicará el tono de circuito ocupado.

y la información de identificación será de la forma siguiente:

$id(MF_1, MF_2, \dots, MF_n)$

donde cada MF_i será uno de los siguientes símbolos de dígito MF en el cuadro VIII.3:

Cuadro VIII.3/J.171.1 – Símbolos de dígito MF

Símbolo	Dígito MF	Símbolo	Dígito MF
0	MF 0	K0	MF K0 or KP
1	MF 1	K1	MF K1
2	MF 2	K2	MF K2
3	MF 3	S0	MF S0 or 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 or KP
8	MF 8		
9	MF 9		

Por tanto, un ejemplo de señal de establecimiento de comunicación podría ser:

$sup(addr(K0, 5,5,5,1,2,1,2, SO), id(K0, 5,5,5,1,2,3,4, SO))$

Suspender llamada (sus): Esta señal indica que la otra parte ha suspendido la llamada, es decir, colgó.

Start Wink (swk): Un controlador de pasarela de medios puede pedir a la pasarela una notificación cuando se produce la señal de activación instantánea.

Tonos de dispositivos de telecomunicaciones para sordos (TDD): El evento TDD se genera cuando se detecta una llamada TDD (véase por ejemplo, la Rec. UIT-T V.18).

Operación finalizada (oc): El evento de operación finalizada se genera cuando se ha pedido a la pasarela que aplique una o más señales de tipo TO en el punto de extremo, y una o más de estas señales fueron completadas sin que hubieran sido detenidas por la detección de un evento solicitado, como la transición de descolgado o la marcación de una cifra. En el informe de finalización se puede indicar como un parámetro el nombre de la señal que llegó hasta el final de su ciclo, por ejemplo:

O: MO/oc (MO/sup)

Si se solicita el evento operación finalizada, no puede parametrarse con ningún parámetro de evento. Cuando no se incluye el nombre del lote, se supone que es el nombre implícito.

Operación fracasada (of): En general, el evento operación fracasada puede ser generado cuando se ha solicitado al punto de extremo que aplique una o más señales de tipo TO en el punto de extremo, y una o más de estas señales fueron interrumpidas antes de la temporización. En el informe de finalización puede incluirse como un parámetro el nombre de la señal no realizada, por ejemplo:

O: MO/of (MO/sup)

Cuando se solicita el evento operación fracasada, no pueden especificarse parámetros de evento. Cuando no se incluye el nombre del lote, se supone que es el nombre implícito.

VIII.2 Lote de protocolo de terminación MF

Nombre de lote: MT

En esta versión de la Recomendación sobre TGCP, el lote sólo puede utilizarse para verificación de línea ocupada (BLV, *busy-line verification*) e interrupción de operador (OI, *operator interrupt*) en circuitos troncales de terminación MF unidireccionales entrantes dedicados a BLV y OI³⁴.

Los códigos del cuadro VIII.4 se utilizan para identificar eventos y señales del lote "MT" en el caso de circuitos troncales de terminación MF unidireccionales entrantes utilizados para BLV y OI:

Cuadro VIII.4/J.171.1 – Códigos utilizados para identificar eventos y señales del lote MT

Código	Descripción	Evento	Señal	Información adicional
ans	Contestación de llamada	–	BR	
bz	Tono de ocupado	–	TO	Periodo de temporización = 30 segundos
hf	Señal de gancho conmutador	–	BR	
inf	Dígitos de información	√		
oc	Operación finalizada	√	–	
of	Operación fracasada	√	–	
oi	Interrupción de operación	√	–	
pst	Tono de señal permanente	–	TO	Periodo de temporización = infinito
rel	Liberar llamada	P	BR	
res	Reanudar llamada	–	BR	
rlc	Liberación completa	P, S	BR	
ro	Tono de circuito ocupado	–	TO	Periodo de temporización = 30 segundos
sup	Establecimiento de comunicación	P	–	
sus	Suspender llamada	–	BR	

Definiciones de los distintos eventos y señales:

NOTA – Para los detalles técnicos concretos de los tonos utilizados, véase la Rec. UIT-T E.180/Q.35.

Contestación de llamada (ans): La señal de contestación de llamada informa al punto extremo que la parte verificada ha contestado. Incluye el caso en que la parte verificada estaba ya descolgada. Cabe esperar que el punto extremo pase al OSS la supervisión de la contestación de llamada.

Tono de ocupado (bz): Estación ocupada.

Señal de gancho conmutador (hf): Esta señal indica que la parte verificada produjo una señal de gancho conmutador.

Dígitos de información (inf (<inf-digits>): Se utilizan en un circuito troncal MF entrante para indicar los dígitos recibidos. El valor del parámetro <inf-digits> comprende todos los dígitos acumulados hasta el delimitador de dígitos inclusive, es decir, ST, ST', ST'', o ST'''.

El valor de <inf-digits> es una lista de dígitos MF separados por una coma:

MF₁, MF₂, ..., MF_n

³⁴ Obsérvese que cuando los servicios de operador son proporcionados por un proveedor off-net, es posible que el OSS no tenga acceso a bases de datos de abonados para determinar si se debe o no permitir BLV y OI.

donde cada MF_i será uno de los siguientes símbolos de dígito MF en el cuadro VIII.5:

Cuadro VIII.5/J.171.1 – Símbolos de dígito MF

Símbolo	Dígito MF	Símbolo	Dígito MF
0	MF 0	K0	MF K0 or KP
1	MF 1	K1	MF K1
2	MF 2	K2	MF K2
3	MF 3	S0	MF S0 or 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 or KP
8	MF 8		
9	MF 9		

Por tanto, éste puede ser un ejemplo de señal o evento:

`inf(k0, 5,5,5,1,2,3,4, s0)`

El siguiente es un ejemplo con expiración del temporizador entre dígitos después de 5,5,5:

`inf(k0, 5,5,5)`

Operación finalizada (oc): Véase la definición de "operación finalizada" en el lote circuito troncal de PU-RDSI.

Operación fracasada (of): Véase la definición de "operación fracasada" en el lote circuito troncal de PU-RDSI.

Interrupción de operador (oi): El evento interrupción de operador se produce cuando el operador intenta interrumpir la llamada y genera el tono "interrupción de operador". Como no hay un tono normalizado definido para este evento, en el presente apéndice se define que el evento se produce cuando en el circuito troncal se detecta un nivel de energía que corresponde a una transición de ruido de línea a voz o tonos. Debe señalarse que, según esto, no es posible detectar una transición inversa, es decir, de voz o tonos a ruido de línea.

Tono de señal permanente (pst): Liberar llamada (rel): El MGC puede utilizar la señal de liberación para liberar la llamada³⁵. En este caso, la señal de liberación no puede estar parametrizada.

³⁵ Obsérvese que el operador que efectúa la verificación normalmente controla la liberación de conexión completadas, no de conexiones para pruebas, y que en este caso lo usual es que se utilice la señal de suspensión.

A su vez, el punto extremo puede utilizar el evento para informar al MGC que ha liberado la llamada; en este caso, el evento puede estar parametrizado con uno de los códigos de causa del cuadro VIII.6, que indican el motivo de la liberación:

Cuadro VIII.6/J.171.1 – Códigos de causas de liberación de llamada

Código de causa	Motivo
0	Liberación normal
3	No hay ruta al destino
8	Preapropiación
19	Ausencia de respuesta
21	Llamada rechazada
27	Destino fuera de servicio
28	Formato de número no válido (por ejemplo, dirección incompleta)
38	Red fuera de servicio
111	Error de protocolo/señalización, no especificado (por ejemplo, expiración de periodo de temporización)

Reanudar llamada (res): Esta señal indica que la parte verificada reanudó la llamada, es decir, descolgó.

Liberación completa (rlc): El punto extremo y el MGC utilizan el evento/señal liberación completa para confirmar que la llamada ha sido liberada y que el circuito troncal está disponible para una nueva llamada.

Establecimiento de comunicación (sup): Se utiliza un evento "sup" para indicar cuándo llega una llamada entrante (lo que corresponde al evento descolgar entrante). El evento se proporciona sin parámetros.

Suspender llamada (sus): Esta señal indica que la parte verificada suspendió la llamada, es decir, colgó.

BIBLIOGRAFÍA

- *Bellcore Notes on the Networks*, Bellcore, SR-2275.
- *Compatibility Information for Feature Group D Switched Access Service*, Bellcore, TR-NPL-000258, Issue 1, octubre de 1985.
- *Interoffice LATA Switching Systems Generic Requirements (LSSGR): Verification Connections (25-05-0903)*, Bellcore, TR-TSY-000531, Issue 2, julio de 1987.
- *Signalling for Analog Interfaces*, Bellcore, LSSGR GR-506-CORE, Issue 1, junio de 1996.
- *Switching System Generic Requirements for Call Control Using the Integrated Services Digital Network User Part (ISDNUP)*, Bellcore, LSSGR GR-317-CORE, Issue 2, diciembre de 1997.
- *Custom Call-Handling Features (FSD 80 Series)*, Bellcore, OSSGR GR-1176-CORE, Issue 1, marzo de 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>.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedios
Serie I	Red digital de servicios integrados
Serie J	Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedios
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	Gestión de las telecomunicaciones, incluida la RGT y el mantenimiento de redes
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos, comunicaciones de sistemas abiertos y seguridad
Serie Y	Infraestructura mundial de la información, aspectos del protocolo Internet y Redes de la próxima generación
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación